$AVRS ®

SYSOUT ACCUMULATION VIEWING and RETRIEVAL SYSTEM®

USER'S MANUAL

Release 5.1B

SEA
Software Engineering of America
1230 Hempstead Turnpike
Franklin Square, NY 11010
E Mail: SEATECH@SEASOFT.COM
Web Address: WWW.SEASOFT.COM
Telephone: (516) 328-7000
Fax: (516) 354-4015

$AVRS is marketed and licensed exclusively by **Software Engineering of America, Inc.**

Copyright

Copyright© 1982, 2002 by **Confident Software, Inc.** ALL RIGHTS RESERVED.

Trademarks

$AVRS® and **S**ysout **A**ccumulation **V**iewing and **R**etrieval **S**ystem® are registered trademarks of **Confident Software, Inc.**

The following statements acknowledge the ownership of trademarks for products referenced in this manual:

ISPF, SPF, PDF, RACF, TSO, and VTAM are registered trademarks of **I**nternational **B**usiness **M**achines Corporation.

Adobe Acrobat is a registered trademark of **A**dobe **S**ystems, Inc.

TOP SECRET, ACF2, ROSCOE, and ETSO are registered trademarks of **C**omputer **A**ssociates International, Inc.

QUICKREF is a registered trademark of **C**hicago **S**oft, LTD.

Copy and Use Restrictions

The **$AVRS** product contains copyright material in machine readable format in order to assist in the installation and use of the licensed programs. Copyrights are on the supplied documentation and all of the material on the **$AVRS** distribution tape.

Technical Support

Software Engineering of America, Inc. has an experienced technical staff available 24 hours a day, every day. Direct contact between users and the **$AVRS** product development staff is readily available.

This manual applies to Release 5.1B of the **$AVRS** product. It completely replaces all previous releases of **$AVRS** documentation.

$AVRS OVERVIEW

Welcome to $AVRS!

The chapters in this section present an overview of the technical, operational and management capabilities of $AVRS. Information about related $AVRS functions and programs are detailed in subsequent sections of this manual and the Installation Manual. It is recommended that you read the chapters in this section to familiarize yourself with the workings and capabilities of $AVRS before proceeding to subsequent sections.

Introducing $AVRS

Welcome to the large family of **$AVRS** users. You will find that **$AVRS** provides a complete approach to the management and archival of Sysout and Syslog datasets. It automates the management and archival of these critical JES resources and saves up to 90% of the human and machine resources presently required.

If you have used an earlier version of **$AVRS**, turn to the section "What's New With $AVRS". It covers what's new, enhancements, and other details about the latest version of **$AVRS**. The Installation Manual gives step by step instructions on the installation and customization of a new release of **$AVRS**.

Learning $AVRS

$AVRS is relatively easy to learn and use. With its' on-line help and menu-driven flow, **$AVRS** is flexible and user friendly. After **$AVRS** has been installed use this manual to become familiar with the Online capabilities and User Batch Functions of **$AVRS**.

What's New With $AVRS

Release 5.1B

This is an overview of the major enhancements provided in **Release 5.1B of $AVRS.**

- The **$AVRS manual** has been totally revised, updated and expanded into two (2) manuals. The Installation and Administrator's Manual is to be used during the installation or maintenance upgrade of **$AVRS** and The User's Manual is to be used during everyday use of the **$AVRS** system and for user training. **$AVRS Release 5.1B manuals** are available on CD. The manuals are in PDF format that can be printed using ADOBE ACROBAT®. A more complete discussion of the contents and use of the two manuals is included under the topic "About This Manual ".

- **$AVRS 5.1B** will handle the new **seven (7) digit JES number** format. The old five (5) digit JES numbers will be converted to the new seven (7) digit format the first time your 5.1B database is initialized, see '7 Digit JES Numbers' in Section A, Chapter 4 of the Installation and Administrator's Manual for details and planning considerations. The display options available are leading zeros or leading blanks. The default is leading zeros. If you prefer leading blanks, apply $AVRC207 BEFORE starting your 5.1B accumulator. Note - After the $AVRS database has been conditioned to display seven (7) digit JES number with 5.1B, it is still suitable for use with $AVRS 5.0A

- **$AVRS** Primary Control Panel (**$AVRSP**) has been updated, reorganized, and several new filter criteria have been added. In addition, it is now two alternating panels, **$AVRSP** and **$AVRPR2.** These two panels contain the same information except for the bottom right corner. On one panel ($AVRSP), the active and defined RRDS percentage is shown. Also, there is a new field showing the **$AVRS** Environment Area that is in effect. The other panel contains the meaning of the **$AVRS** acronym.

- The **SYSLOG ID** display found on the bottom of the Primary Panel has been expanded to two lines and will display up to fourteen (14) four character SYSLOG IDs. The **LOGCLS#** IDs from the Initialization Parameter will display first and then any IDs from the assembled **$AVRLOGI** table (if $AVRC178 has been applied).

- The method to **implement a Customization** and display which Customizations have been applied has been changed to a character(s) in the **$AVRCUST** module. To implement an Integrated Customization (one created in a prior release) you will now need to apply a zap in the $AVRCUST module and not the individual modules. The maintenance upgrade process does not replace the $AVRCUST module so integrated customizations do not have to be re-applied after a maintenance upgrade.

- The **MSGCLASS**= value from the job card is now being moved to KYCLASS in the job Base Key. When the JESJCL datasets are not accumulated, KYCLASS is based on the class of the first file accumulated.

  Also, **MSGCLASS**:

  1. Has been added to the Primary Control Panel. Work selection may be filtered by the job **MSGCLASS=** parameter (or accumulation class for work accumulated without a JESJCL dataset).
  2. Has been added to the Maxlines Table Parameters. Work may now be qualified using both Name *and* MSGCLASS.

- **SELECTion RULE** has been added to the Primary Control Panel. Work may be selected based *only* on the job STARTING DATE/TIME, *only* the job ENDING DATE/TIME, or *both* the job STARTING AND ENDING DATE/TIME. The default value allows the normal start date/time and end date/time criteria to apply. SELECTion RULE is also supported in the batch jobs: $AVRPULL, $AVRTAPE, $AVRDBRP, $AVRSELR, $AVRTPRP, and the new $AVRMRGF.

- The **$AVRS "INITIALIZATION PARAMETERS AND ERRORS REPORT"**, has been changed to provide quicker access to warning indicators (*** WARNING ***), error indicators (*** ERROR ***), and **$AVRS** error messages that may prevent initialization. These indicators and messages have been left justified. The parameters and tables have been indented. This new format will make the errors and warnings stand out more quickly when viewing the **$AVRERR** file. A copy of this report is included in the **Installation and Administrator's Manual**, Section A - Chapter 5 - **Preparing for the Accumulation Task** in the topic **$AVRERR**.

- **$AVRS** now captures the four character **JES SYSTEM ID** during Accumulation and **SYSTEM ID (CPU ID)** has been added to the Primary Control Panel and to the JOBS Directory. With the Online interface, the SYSTEM ID may be used to filter the work selection based on a particular JES SYSTEM.

- **The KSDS Key File Cluster and Index percentages** have been removed from the **$AVRS** primary panel. These KSDS database usage values were based on actual **$AVRS** data within the database. However, VSAM CI/CA splits make this information misleading. Since a meaningful KSDS usage percentage cannot be presented, it has been removed from the panel. It is available on alternative panel $AVRSPK.

- Several new filter values have been added to the **RETURN CODE==>** field on the primary panel.

  1. **NZRSPN**: Selects jobs that have a non-zero completion code and were respun.
  2. **ZERO**: Selects jobs that had a zero return code.
  3. **"+n"**: Selects jobs that had a return code equal to or higher than the numeric value entered. Example: "+4" or "+0004".
  4. **"-n"**: Selects jobs that had a return code equal to or less than the numeric value entered. Example: "-16" or "-0016".

- **$AVRMRGF** (Merge Forward) has been created to provide greater flexibility for the forward merge process of the backup files. New selection features of **$AVRMRGF** include the **INCLUDE** and **EXCLUDE** parameters as well as **NAME**, **SDATE** and **EDATE**. **$AVRFMRG** (the old Forward Merge) will continue to be included in source code but will not be enhanced with new features.

- **$AVRCLEN** has been created to provide greater database maintenance functionally. It replaces **$AVRECOV** which has been discontinued. **$AVRCLEN** runs much faster than $AVRECOV. The new program has the added capability to detect and remove broken chains in the RRDS data files and orphan KSDS records. Most important, it eliminates Dead Blocks in the RRDS Database, should they exist.

- A new **User Profile** (**UPFL**) **System** is now available. This is an ISPF-like panel variable management system that provides ISPF-like services for all **$AVRS** online interfaces: Native TSO, ROSCOE, and **VCSI** as well as ISPF. It is available as a no cost optional add-on product for **$AVRS**. **UPFL** provides for persistent variables across multiple platforms. For example - **NAME** and **TYPE** on the primary panel.

- A new **Validation Record** is being created for most **Key Updates** which will insure that **$AVRS** generated keys transactions will be accumulated to the correct KSDS file. The key updates will be accepted only on the **$AVRS** database that matches the LOCK File in use at the time the key updates were created. Mismatches will result in a one or more message(s) and the key update file is deleted. The data, already placed in the RRDS data files, will become "dead blocks".

- The **$AVRSECU** module for Release 5.1B is coded to automatically check security for OUTPUT DATASET protection. A new switch indicating dataset disposition has also been provided. *Alter* authority is required to create a NEW output dataset. This prevents S913 abends.

- **$AVRS** now supports a **Quick Path entry** to view work. By entering a discrete Job Name or Job Number on the command line from within any TSO/ISPF application, **$AVRS** has the ability to present the actual work, if a single entry is selected, or a directory list if multiple entries are selected. See **Quick Path** in the **$AVRS** manual. A **Quick Exit** is also supported.

- A new **OPC/ESA® User Exit**, **EQQUX010**, has been included in the installation library. This exit may be used under **OPC/ESA®** to quickly access a job's SYSOUT, residing in a **$AVRS** database, for the purposes of restarting the job.

- **$AVRPULL** has been enhanced as follows:

  1. A new feature to select the **LAST** job accumulated, without having to enter a date or time. The JOBNAME= parameter must be used with **LAST=** parameter. Also the SDATE and EDATE may be added to limit the selection to a specific date.
  2. The **ONEMATCH** parameter is now supported. **ONE=Y**
  3. The five (5) special **DDNAME=** values have been changed to be consistent with those in **$AVRTAPE** and the Scan Table Control Statements.
  4. The Control Statements may now be specified in a member in a PDS when the new **$AVRPCTL DD** Statement is used.

- **ARCHIVE HISTORY (AH)** has been enhanced as follows:

  1. **SORT** capabilities have been added for selected fields on the Archive History directory.
  2. **ARCHIVE PRINT** capabilities have been added under the Archive History (AH) panel.
  3. **CPU TIME, ELAPSE TIME** and Accumulation **CLASS** will display for work accumulated and then archived by release **5.1B**.
  4. **RETURN CODE=RESPUN** has been added as a selection filter value.
  5. The **UNIT** and **VOLSER** JCL parameters have been added to the Restore Submit Panel so that work that is no longer cataloged may be referenced for restores. This is also true for the new **Archive Print** function.
  6. The online restore has been enhanced to support the ability to restore from a **Duplex** instead of the Primary media using the new restore selection character **"X"**.

- Various panels and reports have been enhanced to display line counts up to eight (8) digits or greater than 10-million.

- A **customization** has been provided that supports the displaying of **EXCP** counts on both the JOBS directory and the AH directory. The user must provide the required IEFACTRT exit as well as coding the **$AVRXCAP** macro. See customization $AVRC184 and $AVRXCAP in the installation library.

- Support for **FORMS ID** has been added to the ACCMTBL Initialization Parameter.

- **$AVRS** will now recognize the SMS (System Manage Storage) "Not Catalog" message (IGD1710I) and indicate **NOTCTLG** in the CCODE field of the **$AVRS** Directory.

- The **$AVRMLST** module that determines what messages are displayed on the **$AVRS INFO PANEL** has been enhanced to accept messages up to 9 characters in length.

- A **language parameter** has been added to the panel header macros. The default is **LNG=ENGLISH**. When set to **GERMAN**, the panel headers will be in GERMAN.

- The **$AVRS Print Attributes Panel (Demand Print)** has been enhanced to support an eight (8) character **FORMS ID**, **PAGEDEF** and **FORMDEF** values, and a new **P**ost **P**rint **C**ontrol **PDS M**ember.

- The **FIND LIMIT** value has been added to the ISPF profile and the optional **UPFL** product, (USER PROFILE), so that the last value entered will be retained. The **$AVRS** command, **LIMIT**, accepts from 1 to 6 numerics, e.g., LIMIT 40000. The default LIMIT is 1000.

- An additional **NEWTAPE** value has been added to **$AVRS**. The **NEWTAPE=FILES** parameter works with the **FILES=** backup parameter to force a new backup tape to be mounted when the number of files declared in the **FILES=** backup parameter is reached.

- After an Operator has replied "SKIP" to the missed **Backup message**, **$AVRS** will now determine if the **Next Backup** is later *on the same day* and not automatically reset the next possible Backup to the next day.

- The **"Reduced Online Retentions"** from emergency archives is automatically reset by an non-emergency backup/archive.

- Many of the **Enqueue** Major and Minor names have been changed to allow all components of **$AVRS** to run concurrently when multiple databases exist. All Minor names are now based on the LOCK DSNAME. See GRS in the index.

- **$AVRS** with **VCSI** is no longer dependent upon the VTAM Common Module VCSIVPRB.

- The **$AVRS** modify command ***"F the $AVRS started task name, STATUS"*** has been enchaned to additionally display the current JES Classes (ACCMCLS) and Destinations (ACCDEST).

- When a **SORT Command** is issued from the command line within a **$AVRS** job directory panel, the field that was sorted will be placed in the "SORT FIELD" on the primary panel and will be automatically invoked from every job level directory panel until it is changed or removed.

- An **ESTAE exit** has been implemented in over 90% of the modules, supporting better diagnostic messages and an orderly termination.

- A new field for job accounting has been added to the **KSDS** base record and the **SMF** record. **$AVRS** will now place the first twenty characters of raw job accounting data in the new field **KYACCNTG**. This same data will also be placed in the new field **SMFACCNT** in the SMF record. As a result, the SMF record has been expanded by 20 bytes to accommodate the new job accounting field.

- Several **$AVRS INSTLIB JCL members** have been changed and renamed. See **$$$INDEX** member in the **$AVRS** installation library for a quick reference to all the library's members.

1. The complete maintenance upgrade procedure is explained in the information member **$MAINTI.** The new procedure includes making backup copies of all the **$AVRS** libraries before overlaying them. This provides an historical copy for reference.
2. The **$MAINT** member is a procedure which includes the use of the following members: **$MAINTP $MAINTP1 $MAINTP2 $MAP2CHK**
3. The former members VSAMDEF, VSAMULTI, KEYREPRO, EXAMINE, and ENLARGE now contain only references to the new member names: **DEFINEDB**, **DEF$DB**, **DEF$KSDS**, **DEF$RRDS**, **DEF$VOLS**, **KSDSEXAM** and **KSDSMNT**.
    a) These members decrease the number of 'instream IDCAMS control cards' — The DEFINEs for the individual database files are now in separate members which can be used and maintained from each member. They can also be used as sysin datasets in other **$AVRS** JCL.
    b) The new names also allow them to be grouped together alphabetically in the **$AVRS** library. The VSAM define member prefix is DEF. The KSDS maintenance member prefix is KSDS.

- The **Backup/Archive Processing** has been enhanced as follows:

1. **$AVRS** will now invoke an external sort, based on the number of items to be sorted, supporting sorts exceeding 32,768 entries.
2. Support for SMS Management Class has been added to the **TYPE 1 BACKUP CONTROL** parameters – **MGMTCLAS.**
3. A new INITIALIZATION parm that supports SMS Storage Class for files dynamically allocated by the **$AVRS** accumulator has been added – **SRTWKSTC**.
4. Two new INITIALIZATON parameters –**BUMXKD#** and **BUMXRD#** – have been created to limit the number of emergency/abnormal backups per initialization, before the **$AVRS** accumulator terminates. Once **$AVRS** has stopped, you can manipulate the $AVRS database file(s) to avoid the problem(s). When **$AVRS** is restarted, the counters are reset. A normal backup also resets the counters. **BUMXKD#** limits the number of emergency KSDS file full backups. **BUMXRD#** limits the number of emergency RRDS file(s) full backups.

- Several new **SMS** (System Managed Storage) parameters have been added to **$AVRS**. **SRTWKSTC** - an initialization parameter that requests SMS storage class control of sort files dynamically allocated by the **$AVRS** accumulator. **MGMTCLAS** and **STORCLAS** - backup/archive parameters that use SMS management and SMS storage class definitions respectively to control the creation of the backup/archive files.

- **$AVRS** now supports a post-processing member for Demand Prints.

- The **FORM-id** field has been changed to allow 8 character form-id's.

- All **Dynamic Allocations** have been upgraded to support the **Request Block Extensions**. This will enhance diagnostic messages for SMS allocation failures.

- The user exit **$AVRUX9** now contains a pointer to a copy of the base key if a base key is available when the message is issued. This will allow for decisions based on specific job names, types of work, return codes, etc. contained in the base key record. Specific job information could be added to the modified messages created by the exit.

- Additional 24-bit storage constraints have been relieved by moving more **TABLES** into 31-bit storage.

- The explicit block sizes have been removed from $AVRS print files. This allows for system determined block sizes or JCL overrides.

- The **BLKSIZE** on the **$AVRS** LOADLIB has been changed from 6144 to **6233**.

- Due to a naming convention conflict, module **CSIDATE** has been changed to **$AVRY2KD**. All User written modules that referenced CSIDATE must change to $AVRY2KD.

- All known problems from prior releases have been corrected.

DESCRIPTION OF $AVRS

Introduction

$AVRS**®, the **S**ysout/Syslog **A**ccumulation **V**iewing and **R**etrieval **S**ystem®, is a powerful operations management package designed to significantly reduce the personnel and system resources required to handle Sysout, Syslog and JES datasets. **$AVRS accumulates Sysout and Syslog to a *dynamically reusable VSAM data base.*

$AVRS is a collection of programs that allows you to view, manipulate, and report on sysout accumulated work. What follows briefly describes the features and functions of **$AVRS**. Detailed descriptions of each feature or function and associated programs are offered in subsequent sections of the **$AVRS** manuals.

Online Interfaces

The **$AVRS** Online Interfaces provide a comprehensive approach to the reviewing of archived Sysout and Syslog. Immediate access to data is available through batch or fullscreen reentrant interfaces, including native **TSO**, **TSO SPF/ISPF**, or **ROSCOE ETSO**. Online access to **$AVRS** is also available through **VCSI**, the VTAM Common System Interface. (For more details about VCSI, refer to the VCSI User & Installation Guide provided with the **VCSI** product.) Each interface presents several similar display and command panels which are used to review and display selected elements and files in the **$AVRS** data base. The Online Interfaces can be used to perform the following functions:

- Display all work which meets the selection criteria.
- Review the completion status of the selected work.
- Select a desired file for a more detailed review.
- Browse through any selected file.
- Browse through an entire JOB's output.
- Find an argument within a file.
- Print any work with custom banner pages.
- Output work to a DASD dataset.
- Delete work from the **$AVRS** data base.
- Display the archive or microfiche history.

- Display the Archive Status Report.
- Access the Microfiche Online Index.
- Modify the original attributes before printing the file.
- Modify the PF keys for individual preference.
- Review accumulated Syslogs.
- Display a comprehensive HELP facility.
- Selectively request the restore of work from **$AVRS** archive files.

$AVRS provides an extensive **Help and Tutorial Facility** in association with the Online Interfaces. The Tutorial offers comprehensive explanations of the screens, while specific information relevant to the area of current use is provided through Help.

The $AVRS 5.1B error, warning, and informational messages are included in MVS/QUICKREF by Chicago-Soft, LTD., if this product is available at your site. Also included in the $AVRS installation library is the member QUIKHELP which contains all the $AVRS messages and suggested resolution actions if any action is required. NOTE: The QUIKHELP member may contain new and amended messages that may not be included in your site's release of QUIKHELP.

The **Information Feature** associated with the Terminal Interface provides formatted summary information about executed jobs, JCL errors and system or user messages are presented concisely. This provides the user with a fast, convenient and easy way of reviewing problem jobs, with their errors and associated messages.

The **Browse Feature** associated with the Terminal Interface provides a view of all output lines from work currently in the **$AVRS** data base. Data manipulation within **browse** can be performed through the utilization of a variety of commands.

The **Annotation and Acknowledgment Feature** allows users to add commentary and other information to be associated with a job's output, and to specify an acknowledgment code which is listed as part of each jobs directory entry.

Direct VSAM browsing is utilized, eliminating the intermediate sequential file for browse functions. This results in faster and more efficient browsing.

Browse, along with its related commands, is fully described in Section A, Online Browse.

Other Features

$AVRS provides the following features and functionality fully detailed in the Installation Manual:

The **Archive Function or Backup** can be initiated as:

- Automatic backup at user-specified daily backup time.
- Automatic (abnormal) backup if KSDS or RRDS data base files should fill.
- Operator-initiated backup via the MODIFY command.
- A batch job.

The archive facility includes multiple archive levels. This allows an installation to designate the work be placed on a particular archive file. Duplexing of archive levels is also available.

The **Microfiche Facility** allows the user to produce a formatted file, suitable for creating microfiche output. Previously produced microfiche output can be re-created based on the date and time of its original run.

Batch Facilities exist to print work from both the online data base and from **$AVRS** archive files. These batch facilities include the following features:

- Use the batch interface to extract work from archive files.
- Extract work from the **$AVRS** data base to be printed or sent to a DASD file.
- Report contents of backup tape file or the **$AVRS** data base.
- Perform a Batch Archive or Microfiche Job.
- Unload the data base to a sequential file for use as input to recreate the data base on different device types or different volumes.
- Creation of a file to be used as input to a microfiche processor.
- Selectively restore work from **$AVRS** archive files.
- Incremental archive and data base maintenance.

$AVRS Security provides various levels of restricted access to any work in the **$AVRS** data base. In addition, **$AVRS** furnishes built-in interfaces to any existing third party security system.

Reports

The reporting facilities of **$AVRS** provide a comprehensive error reporting system. **$AVRS** provides useful reports which give information on the **$AVRS** data base and on jobs processed by **$AVRS**.

Reports offering information on the **$AVRS** data base:

- Recap by Jobname Report
- File Contents Report
- Miscellaneous Error Messages and Abend Messages
- Unload/Format Report

Reports offering information on jobs processed by **$AVRS**:

- Recap by Jobname Report
- Error Summary by Name Within Type Report
- Error Summary by Name Within Date/Time Report

THE $AVRS SYSTEM

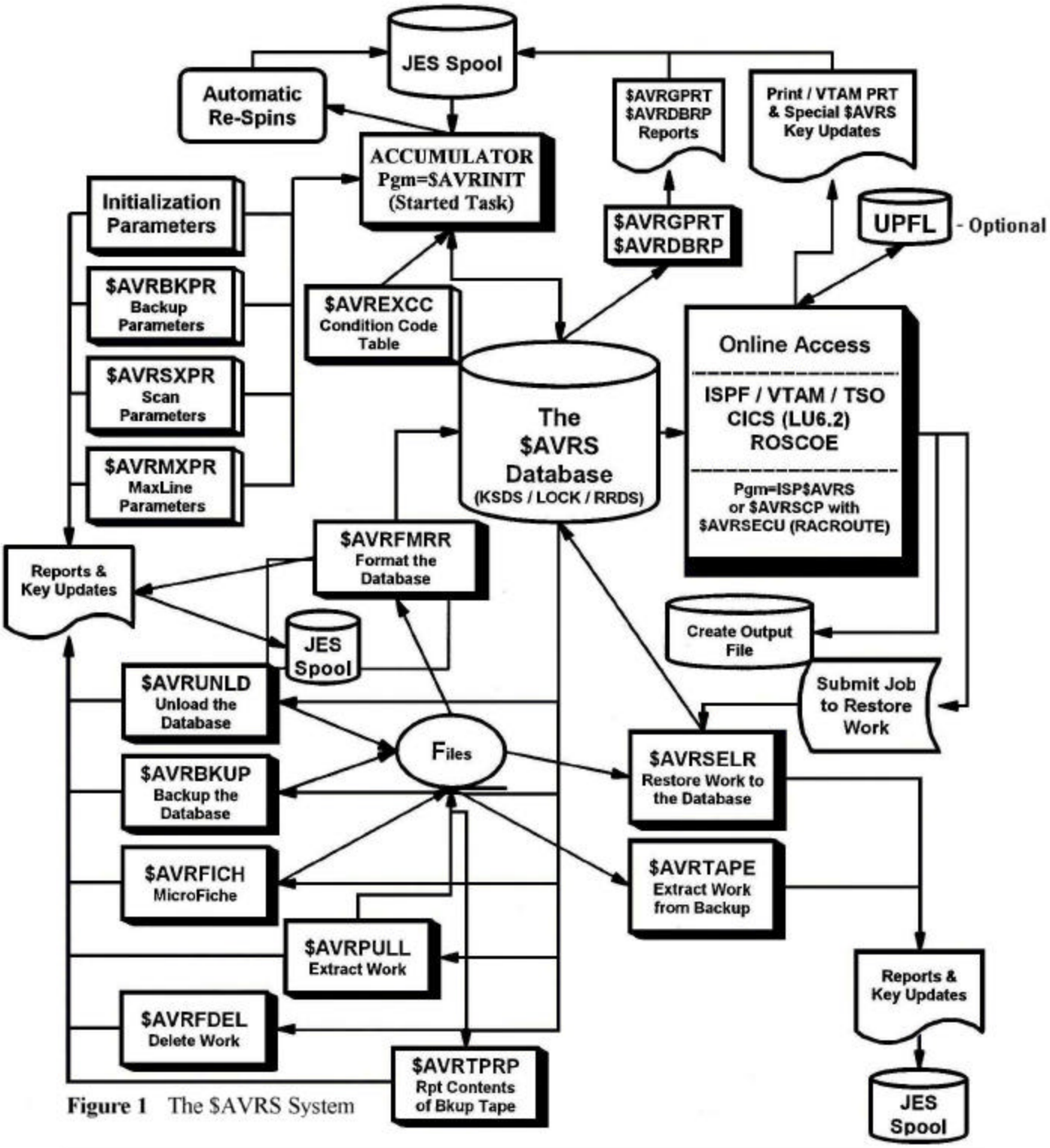


Figure 1 The $AVRS System

System Chart

The system chart describes, in general, the relationships between the major batch jobs in the **$AVRS** system.

Output from the MVS system, collected by the JES Spool, is processed by the program *$AVRINIT*, the **$AVRS** Started Task procedure.

The data is stored in the **$AVRS** VSAM data base and is made up of a KSDS and two or more RRDS files.

$AVRLOCK controls the integrity of the data base. Online users can access the data base through ISPF, VCSI, ROSCOE and TSO. They may then print the data or send it to an output dataset.

$AVRFMRR "formats" a new **$AVRS** VSAM data base as well as "adds" another RRDS Data file to an existing **$AVRS** data base.

$AVRSELR restores work from the backup data files to the **$AVRS** data base.

$AVRBKUP controls which work is to remain in the data base, which work is archived and which work is purged.

$AVRTAPE extracts work from backup data files.

$AVRTPRP reports on the contents of an archive file.

$AVRFICH selects files from the data base and writes them to tape or disk in a format suitable for microfiche.

$AVRUNLD is used to make a portable copy of the data base. Nothing is removed or changed on the $AVRS database.

$AVRDBRP reports on the current contents of the data base.

$AVRGPRT provides a detailed report of the initialization options in effect.

$AVRPULL is used to extract specific work from the data base without going online.

UPFL is the Optional User Profile system used to provide persistent variable values across all online interfaces ($AVRSCP, CICS, ROSCOE, and VCSI).

NOTE: Figure 1 describes only the most important **$AVRS** programs and their interactions.

Definition of Terms

It is important to understand the following terms used throughout this manual. These definitions are generalized and not intended to represent specific definitions.

ACCUMULATE

The function of reading the designated JES output classes and saving the data in the **$AVRS** files. The original JES datasets are normally either deleted or respun to another class and/or destination as a file under the $AVRS task name.

ARCHIVE/BACKUP

The function of backing up work to the backup media, typically tape or cartridge, *with or without deleting* the work from the online **$AVRS** files. Remember that the Backup is not complete until the $AVRLOST (ST report) and $AVRKUPD (maintenance transaction keys) writer name files are accumulated.

BASE KEY

Each job accumulated by **$AVRS** will have one BASE KEY created and saved in the KSDS VSAM dataset. This base key contains all the characteristics of the job.

RESPIN

Placing sysout back in JES in the same form that was originally obtained via the accumulation function *but as a file(s) under the **$AVRS** task name.* Sysout is read from the **$AVRS** files, either online or backup, and written to a designated output class and destination.

HOW **$AVRS** WORKS WITH YOUR SYSTEM

General Information

$AVRS normally operates as a continuous running started task under MVS and may be run at the installation's normal batch processing priority. JES treats **$AVRS** as an external writer and gives work to **$AVRS** according to selection criteria based on Sysout classes and, optionally, a destination and form id.

$AVRS runs only when there is output in the specified user-defined accumulation classes. When the accumulation classes are empty, **$AVRS** enters a 60 second wait state, remaining in this state unless posted by JES that work is available. At the end of this 60 second interval, another request for work is made, and if none is found, the wait is repeated.

$AVRS captures and accumulates up to 35 Sysout classes. It is possible to accumulated more than 35 classes by having additional $AVRS started tasks and databases. **$AVRS** captures data lines up to 256 characters in length, including a carriage control character.

An internal locking mechanism is used within **$AVRS** to establish the presence of the accumulation function for the data base. A message requiring operator response is issued if an attempt is made to initiate multiple accumulation tasks for the same data base. This protects the integrity of the data base, allowing access by only one started task at a time.

Sysout from **JOB**s, **STC**s, **TSU**s and the **SYSLOG**s from as many as seven systems may be accumulated and maintained in the **$AVRS** data base. Only work with the name of SYSLOG is processed as SYSLOG by **$AVRS**.

Work retention by number of days or number of cycles allows flexibility in the retention of work in the data base. This feature utilizes specific naming of work, generic naming or name masking to define the retention of work.

$AVRS can *automatically print a job's accumulated output* locally, or route the output to the job's original remote destination in the event that the job:

- Abends.
- Terminates with a JCL ERROR.
- Contains an unacceptable condition code.
- Contains a pre-defined specific MESSAGE ID.
- Encountered a NOT CATALOGED indicator.
- Requested an automatic reprint (respin/respool).
- Contains a pre-defined text string.

Hardware/Software Requirements

$AVRS:

- Runs on all IBM supported levels of OS/390 and Z/OS using JES2 or JES3.
- Has no explicit CSA requirements.
- Runs with any current level of ISPF that supports 31 Bit addressing.
- May be added as an option to the ISPF Primary Menu.
- Runs under ACF/VTAM Version 4.1 and higher.
- Supports SNA and non-SNA 3270-type terminals, including IBM 3278 Model 2, Model 4, Model 5, IBM 3279, and the IBM 3290 in native mode plus all the other sizes supported by 3290s.
- Runs with microcomputers that have 3270 terminal emulation.
- Runs with ROSCOE level 6.0 or higher using Version 2 EPL parameters
- Supports VCSI Release 2.1A maintenance #10 or higher.
- Supports UPFL Release 2.0B and higher.

Output Media Considerations

- **$AVRS** supports all JES printers.
- **$AVRS** supports microfiche processors.
- **$AVRS** supports VTAM Print facility release 3.0B maintenance #15 or higher.

TECHNICAL AND OPERATIONAL CONSIDERATIONS

General Information

The online **$AVRS** component runs in upper memory above the 16-meg line. The VSAM I/O uses upper memory also. **$AVRS** acquires input from up to *35 JES output classes* which are user-defined exclusively as accumulation and/or SYSLOG class. When the accumulation classes are empty, **$AVRS** enters a 60 second wait state, remaining in this state unless posted by JES that work is available. At the end of this 60 second interval, the accumulation classes are inspected for work, and if none is found, the wait is repeated.

Requeue/respin classes are used to designate non-**$AVRS** sysout class(es) to respin (respin/respool) output which contained ABENDS or some other error condition. Requeue/respin classes may not be defined that correspond to accumulation classes. The requeue/respin class list may select the same respin class for all accumulation classes or may suppress respins for selected accumulation classes.

To direct work to **$AVRS**, specify in the JCL a message class or Sysout class equal to a user-defined accumulation class within **$AVRS,** (include optional destination and form id as necessary). No modifications are required to the operating system, JES, or applications programs. The **$AVRS** accumulation process can be activated or deactivated via user commands.

The user communicates with **$AVRS** via the *MODIFY command* to request **$AVRS** to perform an orderly termination, perform an unscheduled "**ARCHIVE**" (BACKUP) and terminate, or a "**BACKUP,NOSTOP**" without terminating. Backups can also be automatically scheduled or done as batch jobs.

The creator of a job may place a **$AVRS** comment in the JCL that will be displayed on the **$AVRS** Secondary Selection Panel when the work is selected. This comment can provide a means of identification for a job that, otherwise, would not be determined unless the text of the job was browsed.

$AVRS destination cards may be placed in JCL to cause auto respun work assigned to the local destination to be respun to an alternate destination. The alternate destination must be an valid destination for the JES complex on which the **$AVRS** accumulation task is executing.

The **Selective Restore Facility** permits work to be restored to the **$AVRS** data base. The user has the ability to request that work be restored when using the online Archive History function. The JCL and SYSIN data to accomplish the restore is automatically built by **$AVRS**. The job will be submitted when the user exits a **$AVRS** session; or 20 requests for work to be restored is reached; or by the user entering the "**SUBREST**" command. The Selective Restore Facility can be executed from manually constructed JCL and SYSIN data in batch if necessary.

The Selective Restore Facility also permits archived work to be printed to a file instead of restored to the $AVRS database.

A *text string can be located* in a job and the presence can cause an automatic print of the work or produce a message with a user defined ID. Each job that is accumulated not only reports the above conditions, but also reports many other desirable statistics pertaining to a job's execution. These statistics are described in subsequent chapters of the $AVRS User's Manual.

Security

$AVRS furnishes built-in interfaces to any existing third party security system. **$AVRS** also provides various levels of restricted access to any work in the **$AVRS** database. The five methods of security provided for work that is managed by **$AVRS** are:

1. User Exits
2. Resource Access by RACF, ACF2, and TOP SECRET
3. Standard VSAM Password Protection
4. Operator Passwords
5. Normal Dataset Security.

A full discussion and explanation of the five methods is included in the Installation and Administrator's Manual.

About this Manual

The $AVRS Release 5.1B manual has been dramatically changed from previous releases. It has been totally revised, updated, and expanded. The $AVRS manuals are available on CD. The manuals are in PDF format that can be printed using ADOBE ACROBAT.

The manual has been divided into two smaller manuals:

- The Installation and Administrator's Manual
- The Users Manual

Both manuals contain an introductory section with a general discussion about:

- New $AVRS features for this release
- $AVRS and the features that will be available for use after $AVRS is installed.
- A simplified $AVRS System Chart
- Frequently used $AVRS terms
- How $AVRS works with your system
- General and Technical Considerations

THE INSTALLATION AND ADMINISTRATOR'S MANUAL has additional dividers provided for the most frequently referenced topics in the manual, (Initialization Parameters, Accumulation Tables, Archive/Backup, Security and the Browse Environment Area $AVRFSEA). It has been divided into three Sections and three Appendices:

- **Section A** - the information necessary to install the accumulation portion of the $AVRS database. Included is a new installation log that has the installation divided into task units for ease of completion and verification. Separate installation logs are included for "New Release" installation and "Maintenance Upgrade" installation.

- **Section B** - the information necessary to install the online browse interface to the $AVRS database including TSO, ISPF/SPF and Native TSO and ROCSOE ETSO environments.

- **Section C** - information about $AVRS batch and utility programs that were not described in previous sections.

These programs have been divided into three groups and listed alphabetically within the groups -

Group one programs would commonly be used only by the $AVRS Administrator and staff. These programs alter the $AVRS database, its contents, or its functioning.

Group two programs provide data base informational type reports. These programs will not interest the normal $AVRS user.

Group three programs could be valuable for Users or the $AVRS Administrator and staff.

- **Appendix A** - contains User Exits and $AVRS record layouts/dsects.
- **Appendix B** - contains $AVRS Access Method return codes, feed back codes, and macros.
- **Appendix C** - tells how to contact SEA support and has guidelines for sending information for problem resolution. Also a form for requesting enhancements and customizations is in this appendix.

THE USERS MANUAL is divided into two Sections and three Appendices:

- **Section A** - describes the use and functions of the $AVRS Online Browse Interface. This portion of the manual could be used for training new users.
- **Section B** - contains the group three programs included in the Installation and Administrator's Manual. These batch programs might be of value and interest to $AVRS users.
- **Appendix A** - contains all the $AVRS USER ABEND codes. The explanations contain information about additional $AVRS messages that may have been issued and possible problem resolutions or actions.
- **Appendix B** - contains all the $AVRS error, warning, and informational messages arranged alpha numerically. The messages contain actions and possible resolutions, if any is required.
- **Appendix C** - contains $AVRS Access Method return codes and feed back codes.

TABLE OF CONTENTS

$AVRS Overview

Welcome to $AVRS! v
Introducing $AVRS vi
Learning $AVRS vi

What's New With $AVRS vii

Description of $AVRS
Introduction xv
Online Interfaces xv
TSO SPF/ISPF xv
ROSCOE ETSO xv
Help and Tutorial Facility xvi
Information Feature xvi
Browse Feature xvi
Annotation and Acknowledgment Feature xvi
Other Features xvii
Archive Function or Backup xvii
Microfiche Facility xvii
Batch Facilities xvii
Security xvii
Reports xviii

System Chart xx
Definition of Terms xxi

How $AVRS Works With Yout System
General Information xxiii
Hardware/Software Requirements xxiv
Output Media Considerations xxiv

Technical And Operational Considerations
General Information xxv
Security xxvi
About this Manual xxvii
The INSTALLATION And ADMINISTRATOR'S Manual xxvii
The USERS Manual xxviii

Figures xxxv

Section A

$AVRS Online BROWSE

Chapter 1
Overview 3

Chapter 2
Accessing $AVRS
ISP$AVRS - The ISPF Interface 5
SPF/ISPF Interface 5
$AVRSCP - The $AVRS Command Processor 6
The VTAM Common System Interface - VCSI 6
User Profile (UPFL) 6

Chapter 3
Using The HELP/TUTORIAL Facility
Tutorial 7
Help 8

Chapter 4
Selecting Work & Utilizing Work Selection Directories
$AVRS Signon 11
$AVRS Error Panels 11
Selecting Work via the Primary Control Panel 13
Alternating Primary Control Panel 14
Primary Control Panel Fields 15
Work Selection Guidelines and Scenarios 35
Work Selection Directories 36
Directory Header Lines 38

Directory Fields 39
JOB, STC, and TSU Directories - (JN, SN, TN) 40
Syslog Directory, (SL) 42
Archive History Directory, (AH) 44
Fiche History Directory, (FH) 45
Status Report Directory, (ST) 46
Special Control Statements 46
Scrolling Directories 47
Locate Command 51
Sort Command 51
Refresh Command 52

Chapter 5
Displaying & Processing Accumulated Work
$AVRS Browse Features 56
Locate Command 59
Lock Command 59
HEX Command 62
Column Command 64
CC Command 64
Find Command 65
Find Subparameters 66
Entering FIND Strings 68
Altering FIND Search Criteria Parameters 68
CAPS Command 69
Limit Command 69
Limiting a Search by Columns 70
Finding HEX Characters 71
Finding Text Strings 71
Finding Non-Display Characters 72
Information Panel 72
Acknowledgment Code and Annotations 75
Displaying JOBs, STCs and TSUs 77
Shortcuts For Screen Navigation 78
(BR)OWSE Facility 78
(FI)LE Selection Directory 82
File Selection Directory Fields 83
Displaying Syslog (SL) 87
Displaying Status Reports (ST) 89
Displaying Archive History (AH) 92
Restoring from ARCHIVE HISTORY 93

Printing From Archive History 96
Displaying Fiche Index Reports (FI) 99
Displaying Fiche History (FH) 101
Selecting Work VIA Quick Path 103
Selection Parameters 104
Examples 106

Chapter 6
Printing, VTAM Printing, Outputting & Deleting Accumulated Work
Printing 107
Print Panel Fields 108
Closing the Print File 110
VTAM Printing 111
VTAM Print Panel Fields 112
Screen Printing - (Non ISPF) 114
Outputting Work 115
Closing the Output Dataset 117
Deleting Work 118

Chapter 7
Displaying Status Of Zaps And Customizations
Maintenance Zap Status Display Panel 121
Maintenance Customization Display Panel 122

Section B

Batch Programs

Overview - Chart of Programs 123
Group #1 127
Group #2 127
Group #3 127

Chapter 1
$AVRPULL - Online Database Extract 129
PARM Value of WTOR 129
Sample JCL - WTO Parm 131
$AVRPULL — Batch 132
Sample JCL - Procedure 132
Sample JCL - Procedure 132

TSO Foreground 135
$AVRPULL — TSO 135
Sample JCL - TSO Foreground 136
Selection Criteria 137
Selection Examples 149
Sample Work Selection List and Error Report 150

Chapter 2
$AVRSELR - Restore Archived Work For Viewing 151
$AVRS ENQs 152
Sample Batch JCL 153
Sample JCL Procedure 154
Selection Criteria 155
Selection Examples 162
Sample Selective Restore Status Report 163

Chapter 3
$AVRTAPE - Backup/Unload File Extract 165
Started Task or a Batch JOB with a PARM of WTOR 166
Sample Procedure - WTOR Parm 167
Batch Procedure with Input from SYSIN 167
Batch JOB with Input from SYSIN 169
Sample JCL for Batch 169
Selection Criteria 170
Selection Examples 180
Sample Work Selection List and Error Report 181

Chapter 4
$AVRTPRP
Archive Tape Report (Contents of a $AVRS Backup or Unload File) 183
Report On The Contents Of A $AVRS Backup Or Unload File 183
Sample JCL 184
Selection Criteria 184
Selection Examples 190
Sample $AVRTPRP Parse Report 192
Sample Archive Tape Index Report 192
Archive Tape Index Report Description 193
Field Names and Descriptions 193
Vertical Titles 194

Appendix A

$AVRS Abends 197

Appendix B

$AVRS Messages 209
Format #1 $AVR*nnnx* 211
Format #2 $AVRFM*nn* 212
Format #3 $CRDnnnX 212
Format #4 UPFLnnnX 213
Substitutional Variables 215

Appendix C

$AVRS ACCESS Method 391
IBM Return Codes 392
$AVRS Return Codes 393
$AVRS Feedback Codes 394

Index

Index 3 9 9

FIGURES

Figure 1 The $AVRS System xix
Figure 2 Sample Help Panel 9
Figure 3 $AVRS Signon Panel 11
Figure 4 Primary Recovery Panel 12
Figure 5 Problem Diagnostic Panel 12
Figure 6 $AVRS Primary Control Panel #1 13
Figure 7 $AVRS Primary Control Panel #2 (Alternating) 14
Figure 8 Partial $AVRS JOB Directory 26
Figure 9 Work Selection Directory (portion) 37
Figure 10 Work Selection Directory-(left portion) 49
Figure 11 Work Selection Directory-(right portion) 50
Figure 12 Browse Detail Display 58
Figure 13 Lock Command Example: Panel 1 60
Figure 14 Lock Command Example: Panel 2 61
Figure 15 Hexadecimal Browse Display 63
Figure 16 Info Panel, Example 1 73
Figure 17 Info Panel, Example 2 74
Figure 18 Info Panel, Example 3. 75
Figure 19 Annotation/Acknowledgment Panel 76
Figure 20 Concatenated Browse: Example 1 80
Figure 21 Concatenated Browse: Example 2 81
Figure 22 File Selection Directory 82
Figure 23 Browsing Single Files: Example 1 85
Figure 24 Browsing Single Files: Example 2 86
Figure 25 Syslog Selection Panel (Portion) 87
Figure 26 Syslog Segment Under $AVRS Browse 89
Figure 27 Status Report Directory 90
Figure 28 Status Report 91
Figure 29 Archive History Status Panel 92
Figure 30 Restore Job Confirmation and Submit Panel 94
Figure 31 Restore Dataset Not Catalogued 94
Figure 32 Archive Print Job Submission Panel 96
Figure 33 Print Dataset Not Catalogued 97
Figure 34 Fiche Index Directory 99
Figure 35 Fiche Index Report 100
Figure 36 Fiche History Selection Panel 101
Figure 37 Fiche History Detail Panel 102
Figure 38 Job Print Attributes Panel 108

Figure 39 File Print Attributes Panel 111
Figure 40 Job VTAM Print Attributes Panel 112
Figure 41 File Print Attributes Panel 113
Figure 42 $AVRS Screen Print Panel 115
Figure 43 $AVRS Dataset Allocation Panel 115
Figure 44 Delete Confirmation Panel (Job Level) 118
Figure 45 Delete Confirmation Panel (File Level) 119
Figure 46 Maintenance Zap Status Display Panel (DZ) 121
Figure 47 Maintenance Customization Zap Status Display Panel 122
Figure 48 $AVRPULL WTOR PARM Example JCL (OPERPULL) 131
Figure 49 $AVRPULL Batch Procedure Example 133
Figure 50 $AVRPULL Procedure Example (BATCPULL) 134
Figure 51 $AVRPULL Foreground Program CLIST Example 136
Figure 52 $AVRSELR Example JCL Batch 153
Figure 53 $AVRREST Example Procedure 154
Figure 54 $AVRTAPE WTOR PARM (INSTLIB member OPERTAPE) 167
Figure 55 $AVRTAPE Procedure (INSTLIB member BATCTAPE) 168
Figure 56 $AVRTAPE Procedure (INSTLIB member $AVRTAPE) 169
Figure 57 $AVRTPRP Example JCL 184
Figure 58 IBM Access Method Return Codes 392
Figure 59 $AVRS Access Method Return Codes 393
Figure 60 $AVRS Access Method Feedback Codes 394
Figure 61 $AVRS Access Method Feedback Codes (continued) 395
Figure 62 $AVRS Access Method Feedback Codes (continued) 396
Figure 63 $AVRS Access Method Feedback Codes (continued) 397
Figure 64 $AVRS Access Method Feedback Codes 3 9 8

Section A

$AVRS ONLINE BROWSE

$AVRS functionality provides for the accumulation of user-specified data and the ability to manipulate and process that accumulated data.

After installation of $AVRS and accumulation of data, the chapters in this section provide the basics you need to:

- Understand the different types of Interfaces.
- Access $AVRS.
- Select Work for Processing.
- Understand and Utilize Directories.
- Browse and Manipulate Data.
- Print, Output, and Delete Work.
- Utilize the Help/Tutorial Facility.

Chapter 1

OVERVIEW

The **$AVRS** online interfaces allow the user to view and process data accumulated by **$AVRS** and saved in the **$AVRS** data base.

$AVRS can be accessed through TSO, SPF/ISPF, NATIVE TSO, or ROSCOE ETSO. Direct VTAM logon and access from CICS are provided in conjunction with the VTAM Common System Interface (**VCSI**) product. The interface is reentrant and has the same format and command format for all applications. Since panels, commands, display formats and functions are consistent across interfaces, maximum flexibility is maintained for those sites that may use multiple interfaces. This also insures that each user has access to the complete range of functions provided by **$AVRS**.

Designed to be easy to understand and use, **$AVRS** interfaces provide extensive flexibility in the functions provided. Specific criteria used for the selection of accumulated work allows the user to easily locate exactly the work they need to review.

Once selection criteria have been entered, **directories** of selected work are presented and can be sorted by any of several fields. This provides additional ease in manipulating the data. Directories are scrollable left and right and provide up to twenty different informational fields for each entry, depending on the type of work selected. Directories can also be tailored to an installation's particular needs; fields can be re-ordered or even deleted entirely from each directory type.

From a *Primary Selection Directory*, Sysout for individual JOBs, Started Tasks or TSO users can be selected for viewing in three different ways:

- **!** all of a job's datasets may be concatenated for browsing as a single file
- **!** a file selection panel for the job may be presented, allowing for selection of files individually
- **!** an information panel may be viewed, offering a summary of information related to the work along with step completion and/or error information.

In addition, users can switch back and forth between the different viewing options by using simple commands.

Free form commentary and/or an acknowledgment code may be appended to the output of any work through the Annotation/Acknowledgment facility which may be selected from the directory panel.

Instead of invoking ISPF Browse for viewing data from work datasets, $AVRS uses its own internal browse function for the presentation of data. The $AVRS *BROWSE FEATURE* provides identical formats and functions under all interfaces. It is a fully functional interface which provides high-speed positioning in even the largest files without requiring extensive in-core storage.

The $AVRS browse feature provides:

- Complete data scrolling and 'find' functions.
- The ability to display data in hexadecimal format.
- The capability to have a variable number of columns of data 'locked' on the screen to prevent the 'locked' data from shifting right and left during scrolling.
- Right and left scrolling of data that is wider than the terminal screen.

Accumulated work can be printed in part by lines, pages, or in its entirety at a LOCAL or REMOTE designation, transferred to a sequential dataset, or deleted from the $AVRS data base. An optional VTAM Print facility is also available.

An extensive function-sensitive online *HELP FACILITY* is provided. When help is requested, the user is first presented with a panel or series of panels providing information related to the current function. The user may then, if desired, view a directory of other help panels related to the current function, which may then be selected for further information. If an error or informational message is displayed, additional specific information can usually be obtained by entering the 'help' key or command.

Chapter 2

ACCESSING $AVRS

In order to view and process data accumulated and saved in the $AVRS data base, the user must first be able to access it.

Each of $AVRS' interfaces provide an almost identical access to the functions of $AVRS. This is particularly efficient because, in installations utilizing multiple interfaces, users can shift easily from one interface to the other with no loss of orientation and no re-learning requirements. The same Panel Library is used to define the panels displayed, and functions and commands are identical. Therefore, the ISPF, Command Processor, and VCSI interfaces are basically the same. The differences are noted below.

ISP$AVRS - The ISPF Interface

The ISPF Interface is the "base" online interface for $AVRS. It is a full function interface utilizing the functions of ISPF. By using the ISPF profile dataset, input field values are retained between sessions. ISPF also provides the ability for users to change their own PF Key definitions.

The **ISP$AVRS** program employs the **SPF/ISPF Dialog Manager** Interface. In order to complete a successful initialization of the **ISP $AVRS SPF/ISPF Interface**, the $AVRS accumulation task must have completed its initialization at least once. The **$AVRFSEA** macro must be assembled and link edited into a LOAD LIBRARY available to the TSO session. The **SPF/ISPF Interface** reads directly from the online data base and displays the requested work on the terminal.

Help is presented to the user by pressing the HELP PFKey. A narrative description is shown if the HELP PF key is pressed while an error message is present on line three of a **ISP$AVRS** panel. A full tutorial is provided online by pressing the HELP PF key (normally PF1 and/or PF13) during a $AVRS session. The tutorial topics may be selected by number or read sequentially by selecting a starting topic and then pressing enter to continue forward.

$AVRSCP - The $AVRS Command Processor

The $AVRS Command Processor, $AVRSCP, is accessed through native TSO or through ROSCOE ETSO. All the functions available to $AVRS users through the ISPF interface are available to Command Processor users in an identical format, with the following exceptions:

- There is no utilization of a profile dataset, without the optional **USER PROFILE (UPFL)** product.
- No capability exists to change PF Key definitions dynamically. (Alternate environment areas provide alternate PF Key assignment, see $AVRFSEA in the INSTALLATION MANUAL.)

ROSCOE users must have Version 2 EPL (Eligible Program List) Parameters and be ROSCOE 6.0 or higher.

Users who utilize both interfaces (ISPF and TSO) will often find they are unable to distinguish one from the other.

The VTAM Common System Interface - VCSI

The optional VTAM Common System Interface (**VCSI**) is a VTAM application program which allows users to directly sign on to the online portions of SEA products. $AVRS is one of the products supported under VCSI Release 2.1A or later. VCSI provides support for both direct VTAM logons and CICS access to the $AVRS online interface.

The interface supports split screen processing, allowing each user to access two concurrent product sessions, or a combination session of two or more products if more than one product is installed. Currently, VCSI also supports the SEA product **TRMS, the Total Report Management Solution**. Refer to the VCSI User & Installation Guide shipped with the VCSI product for further information about the installation and use of VCSI.

The optional **USER PROFILE (UPFL)** product is also available under **VCSI** to provide a profile dataset.

Chapter 3

USING THE HELP/TUTORIAL FACILITY

$AVRS provides an extensive online HELP and TUTORIAL facility. Although integrally related, the HELP/TUTORIAL facility is most easily described as two separate entities.

Tutorial

The $AVRS Tutorial Facility can be thought of as a manual or book. The user has the ability to start at the beginning with the Primary Tutorial Directory and page through the sections one by one.

$AVRS Tutorial panels are organized into the Primary Tutorial Directory and four sub-directories, as follows:

1 - Primary Panel Tutorials
2 - Work Selection Tutorials
3 - File Selection Tutorials
4 - Browse Facility Tutorials.

Each sub-directory lists several topics related to the current function. Some topics are repeated on more than one directory. Selection of a particular topic by number results in access to a series of panels on the topic. The user may scroll between panels within a topic by using the RIGHT and LEFT PF KEYS or commands, to move forward and backward respectively through the panel list. The ENTER key may also be used to move forward through panels. All panels within a sub-directory may be viewed by pressing the ENTER key, the RIGHT PF KEY or the DOWN PF KEY to move from one panel and topic to the next. The sub-directory can be accessed from any tutorial panel by entering the UP PF KEY or Command.

Help

The $AVRS HELP Facility, on the other hand, is best described as 'on-line' help relative to the function being utilized at that moment.

When a $AVRS user enters HELP (by typing the word HELP at the Command Line or by utilizing the HELP PF Key, PF3 by default), they are immediately presented with information relative to their current situation. In other words, $AVRS HELP is function sensitive. The initial presentation is either the sub-directory relating to the current function, or a particular set of topic panels more specifically related to the current function. If an error or informational message is displayed while using $AVRS, entering HELP results in:

- display of an expanded explanation of the error message, or
- display of a set of topic panels related to the function in error.

For example, if a user is browsing a file and enters the following command:
FIND 4 PREV

The error message PUT STRING IN QUOTES is displayed on the header line and the command in err remains on the command line. If the user then enters the HELP PF KEY, the following Help panel is displayed:

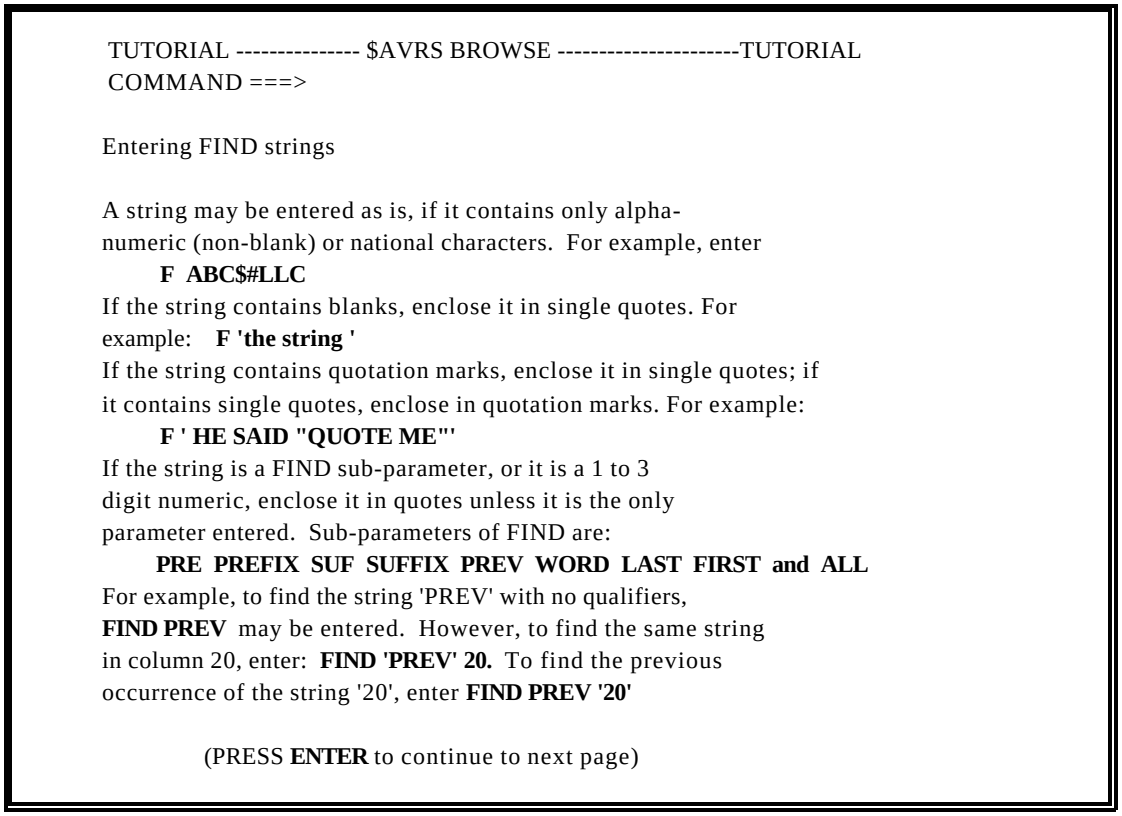


Figure 2 Sample Help Panel

The user may press ENTER to view additional panels on the subject, enter the UP PF Key to see the Browse Tutorial Directory, or enter the END PF Key to return to the original panel.

The following Chapters in this Section explain the $AVRS online functions. The best way to learn these functions is to try them as you read the explanations! As you try each feature go to tutorial panels and read them. If you receive an error message, use the help function to resolve the error.

Chapter 4

SELECTING WORK & UTILIZING WORK SELECTION DIRECTORIES

$AVRS Signon

Certain access methods will require you to sign on to the $AVRS database with your userid and password.

Below is a copy of the $AVRS Signon Panel.

```
---------------------- $AVRS SIGNON PANEL ----------
COMMAND ===>
                     Specify $AVRS SIGNON ID and PASSWORD


                  SIGNON ID   ==>      Enter $AVRS User ID

                  PASSWORD    ==>      Enter Current Password

              NEW PASSWORD    ==>      (External security only)

     VALIDATE NEW PASSWORD    ==>      Repeat New Password
                                           (to validate spelling)
```

Figure 3 $AVRS Signon Panel

$AVRS Error Panels

If errors occur before you receive the $AVRS Primary Control panel, one of the $AVRS PRIMARY ERROR/RECOVERY PANELs may be displayed with an error message. Below is an example of the recovery panel showing the $AVR065C message indicating that the RRDS file(s) for this database could not be allocated. Note this is a 'C' type message, a critical error. The panel prompts the user to correct the error and re-enter $AVRS. Press enter to exit $AVRS as indicated on the panel.

```
------------------------ $AVRS PRIMARY ERROR PANEL -------- RELEASE 5.1B-00
COMMAND ===>
$AVR065C ($AVRAKRO) NO RRDS ALLOCATED - UNABLE TO CONTINUE.



                  $AVRS HAS TERMINATED WITH THE ABOVE CONDITION.

                  CORRECT THE ERROR CONDITION AND RE-ENTER $AVRS.


                         PRESS ENTER TO EXIT $AVRS.
```

Figure 4 Primary Recovery Panel

Another important error type panel is the $AVRS PROBLEM DIAGNOSTIC PANEL. This panel is displayed if an online abend occurs. The panel should be printed and given to your $AVRS Administrator or staff to aid the resolution of the abend. An example of this panel is shown below.

```
 $AVRERPO ---------- $AVRS PROBLEM DIAGNOSTIC PANEL -------- RELEASE 5.1B-00
COMMAND ===>
– ->    PRINT THIS SCREEN AND SEND TO SEA <- - PRESS ENTER TO EXIT $AVRS

$AVR972C TASK ABEND - CODE=SOC4 REASON=00000004 OS=SP6.0.8 /HBB6608
$AVR974C ON 2000-10-24@10:10:54 ASID=0035 C203P (TSU24981)
$AVR839C PSW=078D2000 9719C8A0 TRN=00000000 ILC=06 ICD=04 SERR=40040001
$AVR840C CSECT=$AVRBROD OFFSET=00001B32 ADDR=1719AD68 PERCOLATE=NO
$AVR84IC LOAD/MODULE=ISP$AVRS OFFSET=0001BE8A ADDR=17180A10
$AVR842C INSTR ADDR=17204E7E STORAGE=00FE94DF BBC847F0 C498947F
$AVR901C REGS (00-03) 00000000 1719BB44 1719B8EC 800339C0
$AVR901C REGS (04-07) 172069F8 000000C5 171A756C 00042000
$AVR901C REGS (08-11) 17135D80 172059F8 9713B190 9719B338
$AVR901C REGS (12-15) 972049F8 00033110 97204E64 1720F2A8
$AVR925C DIAG INFO1: 00000000 00000000 00000000 00000000
$AVR925C DIAG INFO2: 00000000 00000000 00000000 00000000
$AVR925C DIAG INFO3: 600000E6 00000000 00000000 00000000
$AVR926C (5.1B-00 ISP$AVRS 01900008-080CC000) TCB=008ADD40 RB=008AFA68
$AVR927C GBLS=9719B338 00042000 $AM=0008B148 $CB=1719BC18
$AVR976C LOCK FILE FOUND   OPEN: DDN=SYS00014 ACB=0008BCF0 FLGS=C0001200
$AVRR977C LOCK FILE: DDN=SYS00014 ACTION=CLOSED FLGS=0200
$AVR976C KSDS FILE FOUND   OPEN: DDN=SYS00015 ACB=0008BAE0 FLGS=C400D200
$AVR977C KSDS FILE: DDN=SYS00015 ACTION=CLOSED FLGS=C200
```

Figure 5 Problem Diagnostic Panel

Selecting Work via the Primary Control Panel

When $AVRS is initially invoked, the user is presented with the following selection panel:

```
-- $AVRS© 1982, 2002 CSI  ---- PRIMARY CONTROL PANEL ---- REL 5.1B-00
COMMAND ===>
<<< $AVRS - USER TITLE LINE >>>

  Primary Filters
   NAME        ==> PROD567 THRU: ==>           SYSLOG ID  ==>
   TYPE        ==> JN                          SYSTEM ID  ==>
   NUMBER      ==>
  Date Filters:___DATE_____EXTENDED DATE_____________________TIME_____HH:MM_____
   SYSTEM                  THU 08 FEB 2001.039                       10:54
   START       ==>
   END         ==>
  Other Filters:_____________________________________________________
   SELECT RULE ==>                             MSGCLASS   ==>
   RETURN CODE ==>                             ACK CODE   ==>
   PGMER NAME  ==>
  Other Controls:____________________________________________________
   SORT FIELD  ==> SDATE                       RANGE      ==>    (Y/N)
   SORT A/D    ==>                             CONF DELS  ==> Y  (Y/N)
  Site Information:_____________________________UPFL ACTV:_____________
  SYSLOG ID(S): MVS1,MVS2,MVS3,MVS4,MVS5,MVS6, DATABASE FILES FOR $AVRCIEA
   MVS7,CPU1,CPU2,CPU3,CPU4,PCU5,CPU6,CPU7    ACTIVE: 22.2% DEFINED:22.2%
  LICENSED BY SEA INC., 1230 HEMPSTEAD TURNPIKE,  FRANKLIN SQUARE, N.Y. 11010
```

Figure 6 $AVRS Primary Control Panel #1

NOTE: The Site Information line on the panel contains an indicator when the optional **USER PROFILE (UPFL)** is being used.

ALTERNATING PRIMARY CONTROL PANEL

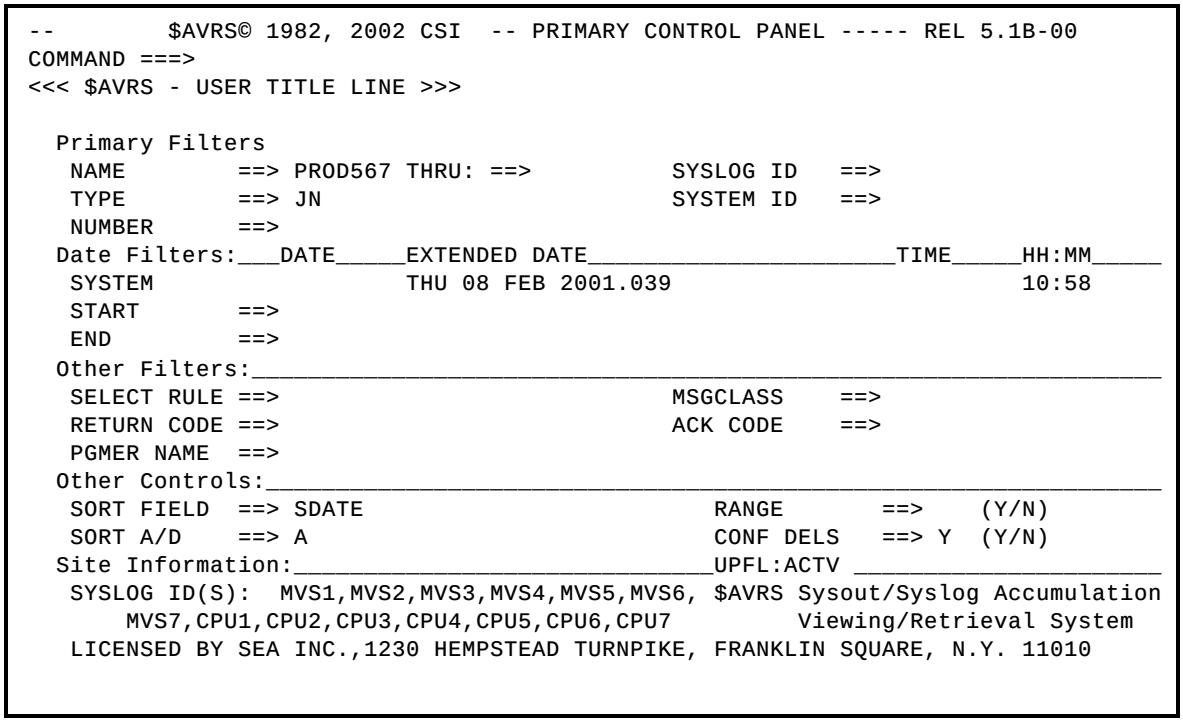


Figure 7 $AVRS Primary Control Panel #2 (Alternating)

This is called the Primary Control Panel because input to this panel may invoke other selection panels (which will be explained in subsequent paragraphs). There are two alternating Primary Control Panels as shown in the two figures above. The only difference on the two alternating panels is in the information on the bottom right side of the panel. The standard Primary Control Panel shows the database file percentages and the Full Screen Environment Area ($AVRFSEA) accessed. The standard Alternating Primary Panel explains the acronym for the $AVRS product name.

The various fields on the panel are used to define a request to view work from the $AVRS data base. Two fields are usually sufficient to define a request. The **TYPE** field is always required on the primary panel. The most commonly used secondary field is **NAME**. A more exact request may be defined by using additional fields on this panel.

Primary Control Panel Fields

A brief description of each of the fields on the Primary Control Panel is presented, in order, below. The use of each field in conjunction with other fields and appropriate values for each field is also described. Further paragraphs in this section discuss in more detail the use of the various selection fields and give an explanation of their use in requesting different types of work.

The Primary control Panel has been subdivided into logical selection groups:
Primary Filters, Date Filters, Other Filters, Other Controls, and Site Information. The most frequently used field in each group are first and on the right side of the panel. Lesser infrequently used fields are on the left side of the panel. There are also several display only fields: the SYSTEMs current date (day of the week, date and time), SYSLOG ID(s), environment area - $AVR??EA, the current percentage of RRDS data files ACTIVE and DEFINED.

NAME and THRU The **NAME** field is the most commonly used selection field. This field will accept 1-8 alphanumeric and national (@#$) characters with the first character being alphabetic or a national character.

Appropriate input for this field is a specific JES name for an **exact name specification.** Alternatively, the entry may be a **masked** name. Name masking is accomplished by replacing one or more characters of the name entered with an asterisk ("*"). Positions filled with asterisks become "wildcard" characters and match any character in a name.

EXAMPLE:	*RESULT:*
A*1***	Any work with JES names such as "A0011", "ABCD1" or "AA0B1" would qualify.

Unless otherwise specified, names of less than eight characters are considered to be **generic**. That is, they are processed as though the remaining blank characters were filled with asterisks.

EXAMPLE:	*RESULT:*
A100B	Work with names of "A100B", "A100B1" or "A100B10C" would all qualify.

Limit selections to an exact match on names shorter than eight characters by entering a *period* (.) at the end of the desired name.

EXAMPLE:	*RESULT:*
A100B.	Only "A100B" would qualify.

NAME selection may also be totally generic.
NOTE: This could produce a very large directory of entries. This is NOT a very efficient way to use $AVRS.

EXAMPLE:	*RESULT:*
*****	All the work of the **TYPE** selected would qualify.

NOTE: Mask selections can be used with a **NAME** only selection criteria.

Alternatively, the **NAME** and **THRU** fields may be used in conjunction to specify a RANGE of JES names to be selected. This will be referred to as **name range selection**. For this type of specification, use the **NAME** field to specify the beginning of the range, and the **THRU** field to specify the end of the range.

EXAMPLE:
NAME ==> **U101** THRU ==> **U200**

RESULT:
The selection of any work with JES names beginning with four characters falling between or including the names specified. Thus "U10199" and "U18" would both qualify, while "U100A" and "U201" would not.

NOTE: Neither name *masking (use of asterisks) nor exact name specification (ending with period)* are valid for name range specification.

The **NAME** field IS NOT required and is ignored for the following selection **TYPEs**:

ST	Online Status Reports
FI	Fiche Index Reports
SL	Syslog

J#	Job Number (**NAME** is used if present and required if the $AVRFSEA MINLENG requirement is specified)
DC	Display Customizations Applied
DZ	Display Maintenance ZAPs Applied.

The **NAME** field is considered for the following selection **TYPEs** as indicated. A 'yes' or 'no' in the appropriate column indicates the options available with that **TYPE** regarding exact name specification, name masking, and name range specification. Use of the NAME field is recommended whenever possible.

TYPE	**Exact Name**	**Masking**	**Name Range**	**Minleng**
AH	yes	yes	no	no
FH	yes	yes	no	no
JN	yes	yes	yes	yes
SN	yes	yes	yes	yes
TN	yes	yes	yes	yes
JO	n/a	n/a	no	yes

NOTE: The selection **TYPE** JO is used to specify a value to be matched against the ***NOTIFY*** *or* ***USER*** fields of the work requested, rather than the JES name (as explained in subsequent paragraphs).

SYSLOG ID This field is valid only with Syslog selection **TYPE SL.** It is not a required field, but may be used to specify the four character Syslog system ID used by $AVRS to distinguish Syslog segments accumulated from multiple systems. The valid entries for this field appear at the bottom of the Primary Control Panel under the heading, SYSLOG ID(S). These entries are from the LOGCLS# initialization parameters and the optional $AVRLOGI table ($AVRC178).

TYPE The **TYPE** field is always required when selecting work from the Primary Control Panel. This is a two character field which identifies the type of work to be selected and is the primary criteria for selection. Valid entries for **TYPE**, along with a brief description of each are as follows:

JN, **SN**, **TN** — *Select JOBs, Started Tasks or TSO users by JES NAME*. The **NAME** field is required. All other selection fields are valid except **SYSLOG ID.**

JO — *Select JOBs by JOB OWNER.* When this type is indicated, the **NAME** field is required and specifies a value to be matched against the NOTIFY and USER values for the JOB. If either field matches the **NAME** value specified, a match results. All other criteria except **SYSLOG ID** are valid.

J# — *Select JOBs by JOB NUMBER.* When this type is indicated, the **NUMBER** field is required and specifies a value to be matched against the JES number for the work. Only JOBs with JOB numbers exactly matching the specified number are qualified. The **NAME** field is valid, if specified, and required if minleng is set in the $AVRFSEA. All other selection criteria except **SYSLOG ID** are valid for this type.

SL — *Select SYSLOG segments.* The **NAME**, **RETURN CODE** and **PGMER NAME** fields are not valid and are ignored if specified. If syslog segments from multiple systems are kept in this $AVRS data base, the **SYSLOG ID** field may be used to specify the ID of the syslog desired. **SORT** is not valid for all fields. All other selection criteria are valid.

AH — *Select Archive History*. An entry in the **NAME** field is required, however the name may be masked or generic. A Name Range specification is not allowed. **SORT** is valid. **START DATE** may be required if AHDATE=YES is specified in the $AVRFSEA.

FH — *Select Fiche History*. The **NAME** field is required but name range specification is not valid. Like **AH**, the name may be masked or generic. No other criteria are valid except **RETURN CODE** and **RANGE. SORT** is not valid.

ST *Select Archive History Status Reports*. The **NAME** field is not valid and is ignored if specified. **START** and **END DATE**, **START** and **END TIME** and **RANGE** are the only other valid criteria. **SORT** is not valid for all fields.

FI *Select Fiche Index Reports*. The **NAME** field is not valid and is ignored if specified. **START** and **END DATE**, **START** and **END TIME** and **RANGE** are the only other valid criteria. **SORT** is not valid.

DZ *Display Maintenance Zaps*. This requests the display of the maintenance zaps that have been applied to $AVRS.

DC *Display Customization Zaps*. This requests the display of customization zaps that have been applied to $AVRS.

SYSTEM ID This field may be used to limit the selection to a specific JES System ID. Beginning with Release 5.1B, $AVRS captures the 4-character JES System ID from the JESLOG and places it in the Base Key. Valid entries are: (1) Any 4-character system id; (2) Blank; (3) **NOID.**
This field is ignored for types *SL*, *AH*, *FH*, *ST*, and *FI*

Desired system id	Work selection will be based on matching this ID
Blank	The System ID will be ignored for selection.
NOID	Work that does not have a system ID has an ID of **NOID**.
Value for Types:	JN, SN, and TN.

NUMBER The **NUMBER** field is used to specify the JES number of the work as a selection criteria. When specified, only work with numbers exactly matching the specified number qualifies. The **NUMBER** field is required with selection **TYPE J#**, and becomes the primary selection criteria. The **NUMBER** field is valid with selection **TYPE**s **JN**, **SN**, **TN** and **JO**.

START DATE The **START DATE** field is valid for all selection types. It may be specified in any of the following four formats:

mm/dd/yy (month/ day / year - in standard format)
or (month/day - year defaults to current year)
yyddd, yyyyddd
or
yy.ddd, yyyy.ddd (Julian date)
* (Today's date)
***-nnn**
or
-nnn (nnn days before today's date)

When this field is specified, the **START DATE** of otherwise qualifying work must fall on or after the date specified in this field. The date considered to be the **START DATE** varies according to the **TYPE** of work selected. For selection of JOBs, STCs, TSUs, Archive History or Fiche History, the start date is the date the work started execution. For SYSLOG, the date is the beginning date of the time period covered by the individual Syslog segment. For Online Status Reports and Fiche Index Reports, the specified **START DATE** must fall within the recap period covered by the Report.

START TIME The **START TIME** field is optional for all types of selection. It is not valid for Status Reports and Fiche Index report selections. When used, it may be specified in any of the following formats:

hhmm	when four (4) digits are entered $AVRS assumes the hours and minutes format in the 24 hour style clock
*****	Current time
***-hhmm**	hhmm before current time to a minimum of
or	00:00 when less than four (4) digits are entered
-hhmm	$AVRS assynes “minutes” and converts the
mmm	minutes to hhmm. Example - If 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31
h:mm, h:m,	when a delimiter (“.”, “.” or “/”) is included
hh:mm	$AVRS assumes the hhmm format

The time considered to be the *starting time* varies according to the type of work selected. JOBs, STCs, and TSUs selections consider the start time to be the time the work started execution. SYSLOG selections qualify if the start time specified falls within the time period of the individual SYSLOG segment. When **RANGE=Y**, SYSLOG selections ignore this field if **START DATE** is not specified.

END DATE Valid input types for **END DATE** are the same as for **START DATE** as listed above. When this field is specified, the **END DATE** of otherwise qualifying work must fall on or before the date of this field. For SYSLOG, the last date in the individual Syslog segment must be on or before the specified date. For Online Status Reports and Fiche Index Reports, the **END DATE** must fall within the recap period covered by the report.

When both **START DATE** and **END DATE** are specified, they define a range of dates to be used as selection criteria. In this case, **END DATE** may be specified in relation to **START DATE** as follows:

+nnn (nnn days after the specified START DATE)

Regardless of the format in which **END DATE** and **END DATE** are specified, $AVRS will expand the requested dates for listing to the right of the input field on the Primary Panel for ease of reference. For example, assume it is desired to see all work for the first 100 days of 2000. The dates could be entered in the input fields as follows:

```
START DATE ===> 2000001
END DATE   ===> +100
```

When next viewed, these lines on the Primary Panel would appear as follows:

```
START DATE ===> 2000001           SATURDAY, 01 JAN 2000.001
END DATE   ===> +100              MONDAY, 10 APR 2000.101
```

NOTE: It is possible to enter a negative number of days at the **END DATE** prompt. This will result in the number of days entered prior to today's date being selected.

END TIME The **END TIME** field is optional for all types of selection. It is not valid for Status Reports and Fiche Index report selections. When used, it may be specified in any of the formats valid for **START TIME** above. Additionally, a "**+**" prefix may be used to add additional hours and/or minutes to the start time. With **RANGE=N**, work ending on or before this time as well as work starting on or before this time qualifies for further selection. The time considered to be the *ending time* varies according to the type of work selected. JOBs, STCs, and TSUs selections consider the end time to be the time the work ended execution. SYSLOG selections qualify if the end time specified falls within the time period of the individual SYSLOG segment. When **RANGE=Y**, SYSLOG selections ignore this field if **END DATE** is not specified.

END TIME may be specified in relation to **START TIME** as follows:

+hhmm (hhmm time after the specified **START TIME**)

If you use the above format without a **START TIME**, the **END TIME** uses the current time as its starting point. Regardless of the format in which **START TIME** and **END TIME** are specified, $AVRS will expand the requested times for listing to the right of the input field on the Primary Panel for ease of reference. For example, assume it is desired to see all batch jobs that ran from 8:00 a.m. until 5:00 p.m. The times could be entered in the input fields as follows:

```
START TIME ===> 0800
END TIME   ===> +0900
```

When next viewed, these lines on the Primary Panel would appear as follows:

```
START TIME ===> 0800              08:00
END TIME   ===> +0900             17:00
```

NOTE: It is not possible to span a date with a time field calculation. The lowest possible time for any date is 00:00 and the highest is 23:59.

SELECT RULE The **SELECT RULE** limits work selection based on START/END DATE/TIME. **START DATE**, **START TIME**, **END DATE**, **END TIME** are required to use **SELECT RULE**. Date and Time selection may now be based on the job end time only, the start time only, both the start and end times together or any start or end time (default).

START	Only jobs that *started* between the start date/time and the end date/time from above will be selected.
END	Only jobs that *ended* between the start date/time and the end date/time from above will be selected.
BOTH	Only jobs that both *started* and *ended* between the start date/time and the end date/time from above will be selected.
blank	Ignore **SELECT RULE**.

Valid for Types: JN, JO, SN, TN, SL, AH
Default: Any entry other than **START**, **END** or **BOTH** Will be ignored.

MSGCLASS The **MSGCLASS**, often referred to as the Accumulation Class, may be used to limit work selection to a particular **MSGCLASS**. $AVRS captures the **MSGCLASS**=value from the JESJCL dataset or the initial Sysout Class accumulated when there is no JESJCL dataset. Jobs that are accumulated from several classes have only one **MSGCLASS** recorded, as described. This field is compared to the character entered on the Primary Control Panel for **MSGCLASS** and only those jobs matching the **MSGCLASS** will be selected for further filtering.

Example: **MSGCLASS**=5
The work selected will be limited to work accumulated with the **MSGCLASS** (or Accumulation Class) of 5
MSGCLASS=*blank*
MSGCLASS is ignored for the selection process

Valid for Types: JN, JO, SN, TN, and AH
Valid Entries: A thru Z and 1 thru 9
Default: None

RETURN CODE Selection of work for viewing may be further limited by specifying a value in the **RETURN CODE** field. The value entered is matched against the *Completion or Condition Code* values (**CCODE** field) displayed by $AVRS for each type of work and ***must be matched according to the methodology employed by $AVRS for displaying that field***. **RETURN CODE** specification is valid for all selection types ***except*** SYSLOG (**SL**), Status Reports (**ST**), Fiche Index (**FI**), and Fiche History (**FH**).

The value entered may be *masked, generic or entered as an exact name* according to the same rules listed for the **NAME** field above. Additionally, if the first byte of the field is numeric, it is treated as if entered with three asterisks preceding it.

EXAMPLE:	*RESULT:*
S0C4	Lists work which completed with a system abend of 0C4.

EXAMPLE:	*RESULT:*
S*.**	Lists entries which have a CCODE starting with an 'S' and are only four characters in length.

EXAMPLE:	*RESULT:*
0000 or 0	Lists entries which have a **RETURN CODE** of zero, regardless of what 3 character prefix it has. The Completion Code is ignored.

EXAMPLE:	*RESULT:*
4 or 0004	Lists entries which have a **RETURN CODE** of four, regardless of what 3 character prefix it has. The Completion Code is ignored.

Remember that unless ended with a period, all values entered are considered to be generic. That is, only the characters entered must match. Therefore it is often necessary only to enter one or two characters in the **RETURN CODE** field to indicate the desired criteria.

Six special **RETURN CODE** field entries are available as follows:

- To select all work that ended with other than a zero completion codes enter **NZERO** in the **RETURN CODE** field.

New

- To select all work that has both a non-zero completion code *and* was respun, enter **NZRSPN** in the **RETURN CODE** field.

- To select all work that has been **RESPUN** by $AVRS, enter **RESPUN** in the **RETURN CODE** field.

New

- To select all work with a zero **RETURN CODE**, enter **ZERO** in the **RETURN CODE** field. **Note** Completion Codes will be ignored.

New

- To select all work that is equal to or greater than a specific value, precede the number with a "+" (plus sign). For example, to select all work with a **RETURN CODE** of '4' or higher, enter "+4" in the return code field. **Note**: Completion Codes are ignored.

New

- To select all work that is equal to or less than a specific value, precede the number with a "-" (minus sign). For example, to select all work with a return code of '16' or less (**RETURN CODES** of **ZERO** are ignored), enter "-16" in the return code field. **Note**: Completion Codes are ignored.

Valid for Types: JN, JO, SN, TN, and AH
Valid Entries: NZERO, NZRSPN, RESPUN, ZERO, +nnnn, -nnnn, or nnnn, and the standard $AVRS CCODE values

The standard JOB directory below is truncated on the right.

```
-THU 27 APR 2000.118 ------------------------------ $AVRS JOB SELECTION -----
 COMMAND ===>
  B=BRS,S=FIL,P=PRT,I=INF,D=DEL,O=ODS,V=VPRT,A=ANN
    SEL# ACT  JOBNAME    JOB # C ACK    CCODE       START DATE      TIME  FILES
_      1      C203BKPR 0017044 N        RC=0008 WED 29 MAR 2000.089 16:53     3
_      2      C203BKPR 0017048 N        RC=0004 WED 29 MAR 2000.089 16:55     3
_      3      C203BKPR 0017150 N        RC=0000 THU 30 MAR 2000.090 11:09     3
_      4      C203BKPR 0017152 N        RC=0004 THU 30 MAR 2000.090 11:11     3
_      5      C203BKPR 0017154 N        TX=0000 THU 30 MAR 2000.090 11:12     3
_      6      C203BKPR 0017155 N        RC=0002 THU 30 MAR 2000.090 11:13     3
_      7      C203BK01 0021326 N        NOTCTLG THU 27 APR 2000.118 09:11     4
_      8      C203BK01 0021328 N        $HASP37 THU 27 APR 2000.118 09:11     3
_      9    R C203BK03 0021322 N        HC=0016 THU 27 APR 2000.118 09:11     3
_     10    R C203BK03 0021324 N        S806    THU 27 APR 2000.118 09:11     3
_     11    R C203BK03 0021325 N        U1234   THU 27 APR 2000.118 09:11     3
_     12    R C203BK99 0021320 N        HC=0016 THU 27 APR 2000.118 09:11     3
_     13      C203BK99 0021321 N        $HASP37 THU 27 APR 2000.118 09:11     3
_     14      C203BK99 0021329 N        $HASP37 THU 27 APR 2000.118 09:11     5
_     15      C203B05A 0017677 N        PARTIAL TUE 04 APR 2000.095 08:39     1
********************************************************* END OF DATA *********
```

Figure 8 Partial $AVRS JOB Directory

If $AVRS contained the work shown in the directory above it would yield the following results in the **RETURN CODE** examples below:

NZERO will display 10 entries - SEL# 1, 2, 4, 6, 7, 9, 10, 11, 12, and 15.

These entries are selected based on the CCODE values displayed.

Only standard $AVRS **CCODE** values/keywords will be used for selection.
NOTCTLG - entry #7, a standard $AVRS **CCODE** keyword was selected as **NZERO**.
Entry # 8, 13, and 14 - $HASP37 was not selected as **NZERO** because it is not a standard **CCODE** value/keyword. (The **CCODE** explanation in this manual shows the standard $AVRS values/keywords.

0000 will display 6 entries - SEL# 3, 5, 7, 8, 13, and 14.

These entries are selected based on the RETURN CODE values. Usually the **CCODE** value and the non displayed RETURN CODE are the same. The **RETURN CODE** value is stored by $AVRS in a non displayed field. **NOTCTLG** #7 and $HASP37 #8, 13, and 14

are all RC=0000 in this field. **NOTCTLG** is a standard $AVRS **CCODE**, it is information that $AVRS considers more valuable than the RC=0. Notice that when a standard $AVRS **CCODE** keyword and a RC=0 are both found in the same job, it is possible for that entry to be selected for both a ZERO and a NON ZERO directory. The $HASP37 **CCODE** was forced into this job by the ABRVMSG initialization parameter request. **CCODE** values can also be forced by the SCAN ARGUMENT TABLE, the EXCEPTIONAL CONDITION CODE TABLE, and the user exit $AVRUX7, installed by your $AVRS Administrator and staff.

-8 will display 4 entries - SEL# 1, 2, 4, and 6.

These entries are based on the non displayed RETURN CODE values of 0008 thru 0001. Usually the **CCODE** value and the non displayed RETURN CODE value are the same. RETURN CODE '0000' entries are excluded from -nnnn selections. System and User ABEND codes are excluded. Completion code - **CCODE** is ignored, since it may have been forced as described above.

+4 will display 5 entries - SEL# 1, 2, 4, 9, and 12.

These entries are based on the non displayed RETURN CODE values of 0004 and higher. Usually the **CCODE** value and the non displayed RETURN CODE value are the same. System and User ABEND codes are excluded. Completion code - **CCODE** is ignored, since it may have been forced as described above.

End of Return Code Examples

ACK CODE Selection of work for viewing may be further limited by specifying a value in the **ACK CODE** input field. The value entered is matched against any *Acknowledgment code* value (**ACK** field) which has been entered by a user for work in the $AVRS data base through the ANNOTATION/ACKNOWLEDGMENT feature of the online interface. Acknowledgment codes are valid for JOBs (JN), STCs (SN), TSUs (TN) and SYSLOG (SL).

The value entered may be *masked, generic or entered as an exact code* according to the same rules listed for the **NAME** field above.

Three special rules apply for selecting by Acknowledgment Code:

- When the characters **@ALL@** are entered, then selection will be for any otherwise matching work for which an Annotation/Acknowledgment record has been created, whether or not any Acknowledgment Code is present. (Users may add free form commentary to an entry in the data base without entering an acknowledgment code; see the section **ACKNOWLEDGMENT CODE and ANNOTATIONS** for further information.)

- When the characters **N/A** are entered, selection will be for all qualifying work for which **NO** Annotation/Acknowledgment record has been created.

- When the character ***** (asterisk) is entered, all qualifying work with any Acknowledgment Code value will be selected. Work with no Acknowledgment Code will not be selected.

Valid for Types: JN, SN, TN, and SL
Valid Entries: @ALL@, N/A, *, a masked, generic or exact code name
Default: None

PGMER NAME The **PGMER NAME** field may be used to further limit the selection of work. This twenty (20) byte field **does not support masking**. Data entered in this field (trailing blanks are ignored) is matched against the value in the **PGMER NAME** field in a rolling fashion until either a match is found, three consecutive blanks are found or the end of the field is reached.

Example: If "JAN M/E" is entered in the **PGMER NAME** field and

$AVRS contains:	The job would be:
"ACCOUNTING, JAN M/E"	selected.
"FAS - JAN M/E"	selected
" JAN M/E, FAS "	selected
" JAN-M/E, FAS "	NOT selected Dash does not match
"FAS JAN M/E"	NOT selected More than 3 consecutive blanks stopped the search

Valid for Types: JN, SN, TN and AH

SORT FIELD The **SORT FIELD** is not a selection criteria, but is a way of ordering the display of a directory. Specifying a value in the **SORT FIELD** is a way to **initially** order the presentation of selections in a directory. Directories may be re-ordered after selection through use of the SORT command which is explained in the section ***Work Selection Directories***. The last valid command line sort parameters will be retained in the $AVRS primary sort field and a/d fields.

The name, or a shortened version of the name, of any of the fields which appear on the work selection directory may be entered as a value in the **SORT FIELD** . The selected work will then be sorted according to the values in the field entered. If no sort criteria is entered, $AVRS will sort the work by JES name by default. If sort criteria **is** entered, the work is sorted by JES name as a secondary sort field. If sort criteria is entered for **AH**, the work is sorted by JES name within type as a secondary SORT FIELD.

The **SORT A/D** field to the right of the **SORT FIELD** may be used to indicate whether the work should be sorted in ascending or descending order. The default is A - ascending.

Valid **SORT** entries are listed below. A shortened version of any field name is valid, so long as it is long enough to be distinguished from any other field name (For example: **JO** is not sufficient because it could refer to **JOBNAME** or **JOB#**, but **CC** is sufficient because it can only refer to **CCODE**.) In cases where significant overlap in the first characters of field names occurs, an alternate short version is allowed for convenience. (For example, **JOBNAME** can be specified as **JN** and **JOB#** can be specified as **J#**.)

NOTE: $AVRS allows each installation to customize the display of work selection directories, including suppressing display of some fields. Because of this, not all fields listed below may actually appear on the directory selected. However, the sorting of selected work is independent of the display of the directory, so that work may be sorted on a field even if that field is not displayed.

Valid for Types: JN, SN, JO, TN, or AH.
Not Valid for Types: SL, FH, ST or FI. If a sort field value is entered that is not valid for the specified type, it is ignored.
Default: Only by JES name in ascending order.

VALID SORT FIELDS

Short Forms	Sort Entry	Explanation
J	**JOBNAME, JOBN or JN**	The JES Name of the work.
J#	**JOB# or J#**	The JES NUMBER of the entry.
C	**CLASS**	The initial accumulation CLASS for the work. **Not applicable: FH or AH**
SD	**STARTDATE or SDATE**	The START DATE for the entry.
E	**ENDDATE or EDATE**	The END DATE for the entry.
U	**USER**	User value from the JOB card of the JCL. **Not applicable: AH, FH**
F	**FILES**	Total number of job files for the work. **Not applicable: AH ,FH**
L	**LINES**	Total number of lines in all files combined.
P	**PAGES**	Total number of pages in all files combined. **Not applicable: AH, FH**
PR	**PROGNAME**	Value from Programmer Name field on the JCL.
N	**NOTIFY**	Value from the Notify field on the JCL. **Not applicable: AH, FH**
CO	**COMMENT**	Comment assigned to the //$AVRS C= card in the JCL of the work being accumulated. **Not applicable: AH, FH**
A	**ARCHIVEDATE**	The Date and Time the work was archived.

VALID SORT FIELDS

Short Forms	Sort Entry	Explanation
AC	**ACCUMDATE**	The Date and Time the work was accumulated. **Not applicable: AH, FH**
EL	**ELAPS**	The Elapsed (Clock) time used by the Job.
RO	**ROOM**	The Room assigned on a ROUTE card in the JCL. **Not applicable: AH, FH**
D	**DESTID**	The Destination assigned on a ROUTE card. **Not applicable: AH, FH**
CC	**CCODE**	The completion code. **Not applicable: AH, FH**
CP	**CPUTIME**	The CPU time used by the creating Job.
M	**MAXRECL**	The Maximum record length for any record in the Job. **Not applicable: AH, FH**
DB	**DBBLKS**	The number of data base blocks used by $AVRS to accumulate the Job. **Not applicable: AH, FH**
ACK	**ACK**	The last entered acknowledgment code for the job. **Not applicable: AH, FH**

RANGE A **Yes** or **No** value that governs how the starting and ending date and time fields are used.

Yes means all work within the starting date and time and the ending date and time is selected.
No means all work within the start date and end date range that fell within the start and end time of each day. A slice across a time period for one or more days.

EXAMPLE: START DATE = 09/15/1996 START TIME = 1500
END DATE = 09/18/1996 END TIME = 2230

RANGE=Y Will select all work beginning or ending from 09/15/1996 at 3:00 p.m. through 09/18/1996 at 10:30 p.m.

RANGE=N Will select all work which ran from 09/15/1996 through 09/18/1996 which either started or ended between 3:00 p.m. and 10:30 p.m. Work that started or ended before 3: p.m. and after 10:30 p.m. will be ignored.

When selecting history records TYPE= ST, AH, FI, FH it is suggested that RANGE=Y be used. Dates and times for history records are based on the RECAP PERIODs for the ST or FI reports. If the recap period does not qualify, none of the work inside the report is considered for the AH or FH directory.

Valid for Types: JN, JO, TN, SN, SL, ST, FI, AH and FH
Valid Entries: Yes, No and Blank — (Blank equals No)
Default: NO

CONF DELS

The **CONFIRM DELETIONS** field is a one character field with **Y** (yes) or **N** (no) being the valid entries. The response entered in this field indicates whether $AVRS should present a confirmation panel whenever work is selected for deletion from the $AVRS database. In order to avoid inadvertent deletions of work, specifying **Y** is recommended.

Valid for Types: All
Default: Y

UPFL

If this indicator is displayed on the Site Information line, the User Profile services have been requested.
Values: ACTV - active
ERRS - errors occurred at access time and UPFL is not active.

SYSLOG (ID)S This field is the SYSLOG ID(S) that have been selected for accumulation. The **SYSLOG ID(S)** field may contain up to fourteen Ids. The first Ids will be entries from the LOGCLS# initialization parameters. The next Ids will be from the optional $AVRLOGI table ($AVRC178).

DATA BASE FILES **ACTIVE** is the percent of utilization for all active RRDS Data files. **DEFINED** is the percent of utilization for all *defined* data files, some of which may be in a draining status and, thus, not active. These are display only fields. This field appears on the bottom right side of one of the alternating Primary Control Panels.

$AVR??EA The $AVR??EA field appears on the bottom right side of one of the alternating Primary Control Panels above the DATABASE FILES percentages. It is a display only field that shows which $AVRS full screen environment area ($AVRFSEA) is currently in use.

Work Selection Guidelines and Scenarios

Now that you are familiar with the available selection criteria and options, keep in mind that all criteria for the Primary Control Panel are "AND" conditions. In other words, for work to be selected it must meet ALL of the criteria specified.

Example: Assume a Primary Control Panel is filled in with the **NAME** field indicating "U102", a **TYPE** field of JN, a **START DATE** of "-20" and a **RETURN CODE** value of "S0C4". The only work selected would be Jobs with Jobnames whose first four characters were "U102" which started within the last twenty days and finished with a system abend 0C4.

Scenario 1: You are a programmer who wishes to examine the output from a started task. You do not recall the name of the started task, but know that it ran 3 or 4 days ago, finished with some kind of user abend and had a large amount of output. To find the job, enter "*" in the **NAME** field, to request all names, "SN" in the **TYPE** field to request started tasks, "-4" in the **START DATE** field and "-3" in the **END DATE** field, and "U" in the **RETURN CODE** field to get all user abends. Further, you could specify "LINES" in the **SORT** field and "D" in the **A/D** field to request that the directory of selected work be sorted by lines in descending order. This will place the work with the largest number of lines at the top of the directory.

Scenario 2: You are a supervisor who needs to occasionally review the jobs submitted by personnel in your groups to check for potential problems or irregularities. You can select jobs using the JOB OWNER (**TYPE** JO) selection criteria and specify a generic value in the **NAME** field which will select all the users in your group. You can check a particular time period by using the **START and END DATES.** A combination of use of the **RETURN CODE** selection criteria and the ability of $AVRS to sort and re-sort directories then provides several easy ways to check the work in a variety of ways. For example, to check the number of JCL errors or Abended jobs, you could select all work with the relevant **RETURN CODE** and sort by the USER field. Or you could select all work and sort by CCODE and then use the LOCATE command (as explained later in this chapter) to quickly position the display to the desired location. Other resource utilization could be checked by sorting on LINES, CPUTIME, ELAPSED time or even DBBLKS ($AVRS data base blocks utilized).

Work Selection Directories

After selection criteria have been entered from the Primary Control Panel, if any work is found to match the request, a directory of selected work is presented. The actual directory presentation is dependent upon the entry in the **TYPE** field as detailed below:

TYPE	**Directory Presented**
JN, SN, TN	JOB, STC, TSU Directory
SL	Syslog Directory
AH	Archive History Directory
FH	Fiche History Directory
ST, FI	Status Report Directory

From these directories, individual work entries may be selected for viewing or processed for printing, output or deletion as applicable. Information about each work entry can also be obtained from the directory line itself.

The following is a sample work selection directory.

```
- SAT 03 MAY 2002.123 ------ $AVRS JOB SELECTION ----------------- ROW 1 OF 35
 COMMAND ===>                                                    SCROLL ===> CUR
  B=BRS,S=FIL,P=PRT,I=INF,D=DEL,O=ODS,V=VPRT,A=ANN       #LINES TOTAL     10253
    SEL# ACT  JOBNAME   JOB # C ACK    CCODE     START DATE           TIME FILES
_      1      ADBPRT   0027878 7        RC=0000 FRI 03  MAY 2002.123  14:15    6
_      2      ADBPRT   0027902 7 ACKCDE RC=0000 FRI 03  MAY 2002.123  14:36    6
_      3    R BIGDB    0027879 7        RC=0000 FRI 03  MAY 2002.123  14:15    6
_      4      C112DBRP 0027880 7        RC=0000 FRI 03  MAY 2002.123  14:15    6
_      5      C112DBRP 0027904 7        RC=0000 FRI 03  MAY 2002.123  14:37    6
_      6      DBRPRPT  0027881 7        RC=0000 FRI 03  MAY 2002.123  14:15    6
_      7      DBRPRPT  0027905 7        RC=0000 FRI 03  MAY 2002.123  14:37    6
_      8      ENTRDBRP 0027906 7 *      RC=0000 FRI 03  MAY 2002.123  14:37    5
_      9      FIGRDBRP 0027883 7        RC=0000 FRI 03  MAY 2002.123  14:15    6
_     10      FIGRDBRP 0027907 7        RC=0000 FRI 03  MAY 2002.123  14:37    6
_     11      GEETAPEP 0027908 7        RC=0000 FRI 03  MAY 2002.123  14:37    5
_     12    * HIACC    0027850 7        HC=0016 FRI 03  MAY 2002.123  13:30    3
_     13    R HTAPEPRT 0027909 7        RC=0000 FRI 03  MAY 2002.123  14:37    5
_     14      INFOTAPE 0027919 7        RC=0000 FRI 03  MAY 2002.123  14:41    5
_     15    * JCLERR   0027852 7        JCL ERR FRI 03  MAY 2002.123  13:30    3
_     16      MAXLNIC  0027855 7        IC=0000 FRI 03  MAY 2002.123  13:31    5
_     17  P   MAXLNIN  0027856 7        INCMPLT FRI 03  MAY 2002.123  13:32    1
_     18      NOEXECC  0027857 7        NO-EXEC FRI 03  MAY 2002.123  13:32    2
_     19  A   NOSNTAPE 0027911 7 CODE01 RC=0000 FRI 03  MAY 2002.123  14:38    5
_     20    * NOTCATLG 0027849 7        NOTCTLG FRI 03  MAY 2002.123  13:30    3
_     21      OUTTAPE  0027914 7        RC=0000 FRI 03  MAY 2002.123  14:38    5
_     22  O   PARTIAL  0027854 7        PARTIAL FRI 03  MAY 2002.123  13:31    1
_     23      PRTTAPE  0027912 7        RC=0000 FRI 03  MAY 2002.123  14:38    5
_     24      PRTTAPE  0027913 7        RC=0000 FRI 03  MAY 2002.123  14:39    5
_     25      RPTTAPE  0027915 7        RC=0000 FRI 03  MAY 2002.123  14:39    5
_     26      SCAN     0027846 7        SCAN    FRI 03  MAY 2002.123  13:31    2
_     27    T SCANERR  0017840 7        SCANERR FRI 03  MAY 2002.123  13:30    3
_     28      SCAN4CD  0027848 7        GOTSCAN TUE 03  MAY 1999.124  13:30    5
_     29      SCAN4TX  0027847 7        TX=0000 FRI 03  MAY 2002.123  13:30    5
_     30 S  X SEETAPE1   00812 7        RC=0000 MON 03  MAY 1998.124  14:39    5
_     31 S    SEETAPE2 0027917 7        RC=0000 FRI 03  MAY 2002.123  14:39    5
_     32    * SYSABEND 0027851 7        S806    FRI 03  MAY 2002.123  13:30    3
_     33      TAPERPT  0027918 7        RC=0000 FRI 03  MAY 2002.123  14:39    5
_     34    * UNCONRS  0027859 7        RQ=0004 FRI 03  MAY 2002.123  13:33    4
_     35   I* USRABEND 0027853 7        U0016   FRI 03  MAY 2002.123  13:31    3
 ******************************** END OF DATA *********************************
```

Figure 9 Work Selection Directory (portion)

Directory Header Lines

Depending on the type of work selected and, therefore, on the directory presented, fields will vary. The header lines for all the directories, however, are consistent and are explained below.

```
- SAT 03 MAY 2002.123 ------ $AVRS JOB SELECTION -------------- ROW 1 OF 202
```

The **top line** of the directory screen will indicate the type of work presented, for example: **$AVRS JOB SELECTION**. The left-most portion of the top line will display today's date. The right-most portion of the top line will indicate the total number of entries in the current directory and the number of the entry which is currently at the top of the display.

```
COMMAND ===>                                                SCROLL ===> CUR
```

The **second line** of each directory contains the command entry area at the left, and the scroll variable area at the right. (Valid scroll variable values are detailed in subsequent paragraphs of this chapter.)

```
B=BRS,S=FIL,P=PRT,I=INF,D=DEL,O=ODS,V=VPRT,A=ANN    #LINES TOTAL       25774
```

At the left of the **third line** is listed the valid selection functions for the current directory, along with the character used for each function. The valid selection characters are determined by the $AVRFSEA At the right of the third line is listed the total number of lines of output contained in all the entries in the directory, if applicable.

```
SEL# ACT    JOBNAME  JOB # C CCODE      START DATE   TIME  FILES  LINES
```

The **fourth line** on the screen is the heading line for individual entries. The actual number and types of fields listed for each selection type will vary. See the descriptions of each directory type for a list of valid fields which may appear on the directory heading. The example above is only a portion of the available entries.

Following the fourth line are the individual entries for the current selection. To the left of each entry is a selection byte, marked by an underscore (_) character. Enter the letter corresponding to the function requested to the left of the desired entry to perform the function. The function letter followed by the selection number may also be entered on the command line. Following the selection character underscore is an entry number. Entries on selection directories are numbered consecutively for ease in processing and viewing. Following the number is the **ACT**ion field, which indicates what selections have been done for the particular

entry during the current selection. Following this are the values for each entry corresponding to the field headers.

The **ACT**ion field contains four columns. The default values placed in these columns are as follows, however, they may be altered during installation by the $AVRFSEA.

Col 1	Col 2	Col 3	Col 4
S	P	I	R
	V	A	*
	O	D	T
		R	X
		X	

Column 1: An "S" denotes this item was 'Selected' or 'Browsed'.

Column 2: A "P" denotes the item was 'Printed'; A "V" denotes the item was printed via VTAM PRINT; An "O" denotes the item had been 'Output' to a dataset.

Column 3: An "I" denotes the item was selected for 'Info"; An "A" represents 'Annotation or Acknowledgment'; A "D" denotes that a delete has been requested; An "R" denotes that a 'Restore' has been requested; An “X” denotes that a ‘Duplex Restore’ has been requested.

Column 4: An "R" denotes that the work was re-spun. An "*" denotes that a re-spin was suppressed; A "T" denotes that this item was restored. An “X” denotes the restore or print was from a duplex.

Directory Fields

The following are the valid directory fields for each type of Work Selection Directory. Since directories can be customized during the installation process, the fields listed below will not necessarily be present, or in the same order, on the directory at your installation.

JOB, STC, and TSU Directories - (JN, SN, TN)

Valid headings are:

JOBNAME/ STCNAME/ TSUNAME — The NAME of the entry being displayed.

JOB#/STC#/ TSU# — The JES NUMBER of the entry being displayed.

C — The initial accumulation CLASS for the work.

ACK — Last entered acknowledgment code, if any, for this work. An asterisk (*) in this field indicates that an annotation entry has been added, though no acknowledgment code is present.

CCODE — This field may show $AVRS or User defined codes. User defined codes may be entered through the $AVRUX7 or $AVRS scan table facilities. Appropriate $AVRS default values for this field include:

RC=cccc	highest Condition Code encountered.
AR=cccc	work respun due to the AUTORSP initialization parameter.
IC=cccc	max lines reached and the complete SYSMSG dataset was accumulated.
HC=cccc	work was respun due to a high Condition Code.
RQ=cccc	work was respun due to $AVREXCC exception table, $AVRUX7, or the $AVRS synad was entered.
TX=cccc	work was respun due to scan argument found.
Snnn	a system abend occurred during execution.
Unnnn	a user abend occurred during execution.
JCL ERR	a JCL error occurred during execution.
SI/O	$AVRS SYNAD exit entered or msg IEF376I was not found while processing the SYSMSG dataset.
INCMPLT	max lines reached and no SYSMSG dataset existed.
NOTCTLG	at least 1 NOTCATLG 2 condition was found.
NO-EXEC	job deleted by JES or canceled by operator before execution.

PARTIAL	the job's SYSMSG dataset was unavailable to $AVRS at the time of accumulation.
SCAN	presented when TYPRUN=SCAN was on the JOBCARD and a JCL ERROR did not occur for the scan.
SCANERR	presented when TYPRUN=SCAN was on the JOBCARD and a JCL ERROR did occur for the scan.
ABRVMSG	if the message number specified in the initialization parameter ABRVMSG was found in this job, this number will be reported here instead of the job completion code.

START DATE The START DATE for the entry or the date of the accumulation when the entry's actual START DATE is unavailable.

START TIME The START TIME for the entry or the time of the accumulation when the entry's actual START TIME is unavailable.

FILES Total number of job files for the work.

LINES Total number of lines in all files for the work.

PAGES Total number of pages in all files for the work.

USER JES2: User value from PDBCRUID or JOB JCL statement.
JES3: User value from the Job JCL statement.

END DATE The END DATE for the entry or the date of the accumulation when the entry's actual END DATE is unavailable.

END TIME The END TIME for the entry or the time of the accumulation when the entry's actual END TIME is unavailable.

PROGRAMMER NAME Value from the Programmer Name field on the JCL or the job name is there is no programmer name information available.

COMMENT FIELD Comment assigned to the //$AVRS C= card in the JCL of the work being accumulated.

NOTIFY Value from the Notify field on the JCL.

ARCHIVE DATE and TIME The Date and Time the work was archived by $AVRS or blank if the work has not yet been archived.

CPU TIME The CPU time used by the creating Job.

NOTE: The CPU time will be absent if the SYSMSG dataset was unavailable or incomplete.

ELAPS The Elapsed (Clock) time used by the Job.

ROOM The Room assigned on a ROUTE card in the JCL.

DESTID The Destination assigned on a ROUTE card.

ACCUMULATION DATE and TIME The Date and Time the work was accumulated by $AVRS.

S -ID The MVS system where the work was executed.

MREC The Maximum record length for any record in the Job.

DBBLKS The number of data base blocks used by $AVRS to accumulate the Job.

Syslog Directory, (SL)

Default headings are:

SLID The identification that has been assigned to $AVRS for a system's SYSLOG.

JOB# The JES NUMBER of producing JOB.

C The accumulation CLASS for this segment.

ACK	Last entered acknowledgment code, if any, for this work. An asterisk (*) in this field indicates that an annotation entry has been added, though no acknowledgment code is present.
START DATE	The first valid DATE STAMP encountered in this segment or the date of the accumulation when the entry's actual START DATE is unavailable.
START TIME	The first valid TIME STAMP encountered in this segment or the time of accumulation when the entries actual START TIME is unavailable.
LINES	Total number of lines in this segment.
END DATE	The last or latest valid DATE STAMP encountered in this segment or the date of the accumulation when the entry's actual END DATE is unavailable.
END TIME	The last or latest valid TIME STAMP encountered in this segment or the time of accumulation when the entry's actual END TIME is unavailable.
PAGES	Total number of pages in the segment.
FILES	Total number of job files for the work.
ARCHIVE DATE and TIME	The Date and Time the work was archived by $AVRS or blank if the work has not yet been archived.
ACCUMULATION DATE and TIME	The Date and Time the work was accumulated by $AVRS.
DESTID	The JES Destination.
DBBLKS	The number of data base blocks used by $AVRS to accumulate the Job.

MREC The Maximum record length in the segment.

Archive History Directory, (AH)

Default headings are:

JOBNAME The JES NAME of the original job.

JOB# The JES NUMBER of the original job.

TYP JOB, STC or TSU.

RESPUN Yes or No. Original job was respun.

START DATE The START DATE for the entry or the date of the accumulation when the entry's actual START DATE is unavailable.

START TIME The START TIME for the entry or the time of the accumulation when the entry's actual START TIME is unavailable.

LINES Total number of lines in all files combined.

END DATE The END DATE for the entry or the date of the accumulation when the entry's actual END DATE is unavailable.

END TIME The END TIME for the entry or the time of the accumulation when the entry's actual END TIME is unavailable.

ARCHIVE RUN DATE and TIME The Date and Time of the archive run.

CCODE Completion Code. See **CCODE** on page 42 for a complete description of values for this field.

PROGRAMMER NAME Value from the Programmer Name field on the JCL or on some directories the job name is there is no programmer name information available.

C The sysout class the work was originally accumulated from, if this work was archived by release 5.1B or higher and the original work contained the information.

CPUTIME The CPU time the work used (MMMM.SS.HH format), if this work was archived by release 5.1B or higher and the original work contained the information.

ELAPS The elapsed (wall clock) time the work was executing (MMMMM format) if this work was archived by release 5.1B or higher and the original work contained the information

Fiche History Directory, (FH)

Default headings are:

JOBNAME The JES NAME of the original job.

JOB# The JES NUMBER of the original job.

TYP JOB, STC or TSU.

START DATE The START DATE for the entry or the date of the accumulation when the entry's actual START DATE is unavailable.

START TIME The START TIME for the entry or the time of the accumulation when the entry's actual START TIME is unavailable.

END DATE The END DATE for the entry or the date of the accumulation when the entry's actual END DATE is unavailable.

END TIME The END TIME for the entry or the time of the accumulation when the entry's actual END TIME is unavailable.

CCODE	Completion Code. See **CCODE** on page 42 for a complete description of values for this field.
CREATION DATE and TIME	The Date and Time the Fiche was created.
FI#	The Fiche number on which the work appears.
R/C	The Row and Column where the work begins.

Status Report Directory, (ST)

Default headings are:

LINES	Total number of lines in the report.
RECAP PERIOD START	The earliest START DATE and TIME for any archived work contained in the report.
RECAP PERIOD END	The latest END DATE and TIME for any archived work contained in the report.
CREATION/RUN DATE and TIME	The Date and Time the archive job was run.

Special Control Statements

$AVRS provides two special control statements to be placed in the JCL of any job. When the JCL for the job is accumulated, it is inspected for the presence of these $AVRS special control statements. The special control statements must be placed between the JCL JOBCARD and the first EXECUTE statement in order to be recognized by $AVRS. The special control statements are actually JCL comment statements which are defined as follows:

`//*$AVRS C=`	up to 20 characters of comments
`//*$AVRS COMMENT=`	to be shown on the online interfaces (specifically on the Job Selection Panel in the Comment Field and on the Information Panel)
`//*$AVRS D=`	Respin Destination for Local
`//*$AVRS DEST=`	Respins

The `//*$AVRS C=` statement is used to display any information the user desires to the online interface. Twenty characters following the C= parameter of the control statement will be saved for the job in the key file.

The `//*$AVRS D=` statement is used to assign an alternate destination for respin of work which would normally be respun to the LOCAL destination on the RSLOCAL DD from the $AVRS accumulation task. The designation may be a remote, a node, a node and remote or any valid destination assigned to the JES complex running the accumulation task. Up to 8 characters may be used to define the destination.

EXAMPLES	*RESPIN DESTINATION*
1) `//*$AVRS D=RMT5`	Remote 5
2) `//*$AVRS D=N002R005`	Node 2, Remote 5
3) `//*$AVRS D=N10`	Node 10
4) `//*$AVRS D=NEWYORK`	The DEST assigned to NEWYORK

NOTE: These cards must appear between the JOBCARD and the first execute card in the JCL to be recognized by $AVRS during accumulation. Only the last card of each type is retained for use by $AVRS. When used, these special statements will display on the (IN) Information Panel.

Scrolling Directories

The amount of information contained on a given Work Selection Directory is, in most cases, too vast to present on one screen. Certainly, $AVRS directory lines contain more information than will fit on one terminal line. Therefore, directories may be scrolled up and down or left and right. The top three heading lines will not participate in the scroll.

Using the *UP and DOWN PF Keys or commands,* scrolls the selected directory up or down. Additional fields may be viewed by scrolling the directory left and right, using the *RIGHT and LEFT PF Keys or commands*. When scrolling left and right, the left most portion of each entry remains in place. The unmoved portion includes the selection byte, the entry number, the action field and the first variable field for the entry. By default, the first field is usually the JES name for the work. This may be altered by installation specification, however. The top three heading lines do not scroll left and right, but the field heading line participates in the scroll to leave the field headings above the related field. $AVRS will not "split" fields when scrolling left and right. That is, individual fields are displayed on the screen only when the entire field can fit, rather than a portion of it. Refer to **Figure 7** and **Figure 11** for an example of Right/Left scrolling.

The degree of the scroll is governed by the value specified in the **SCROLL** variable on the COMMAND line. The scroll variable is changed by placing the cursor in the **SCROLL** field and entering a valid scroll variable as listed below.

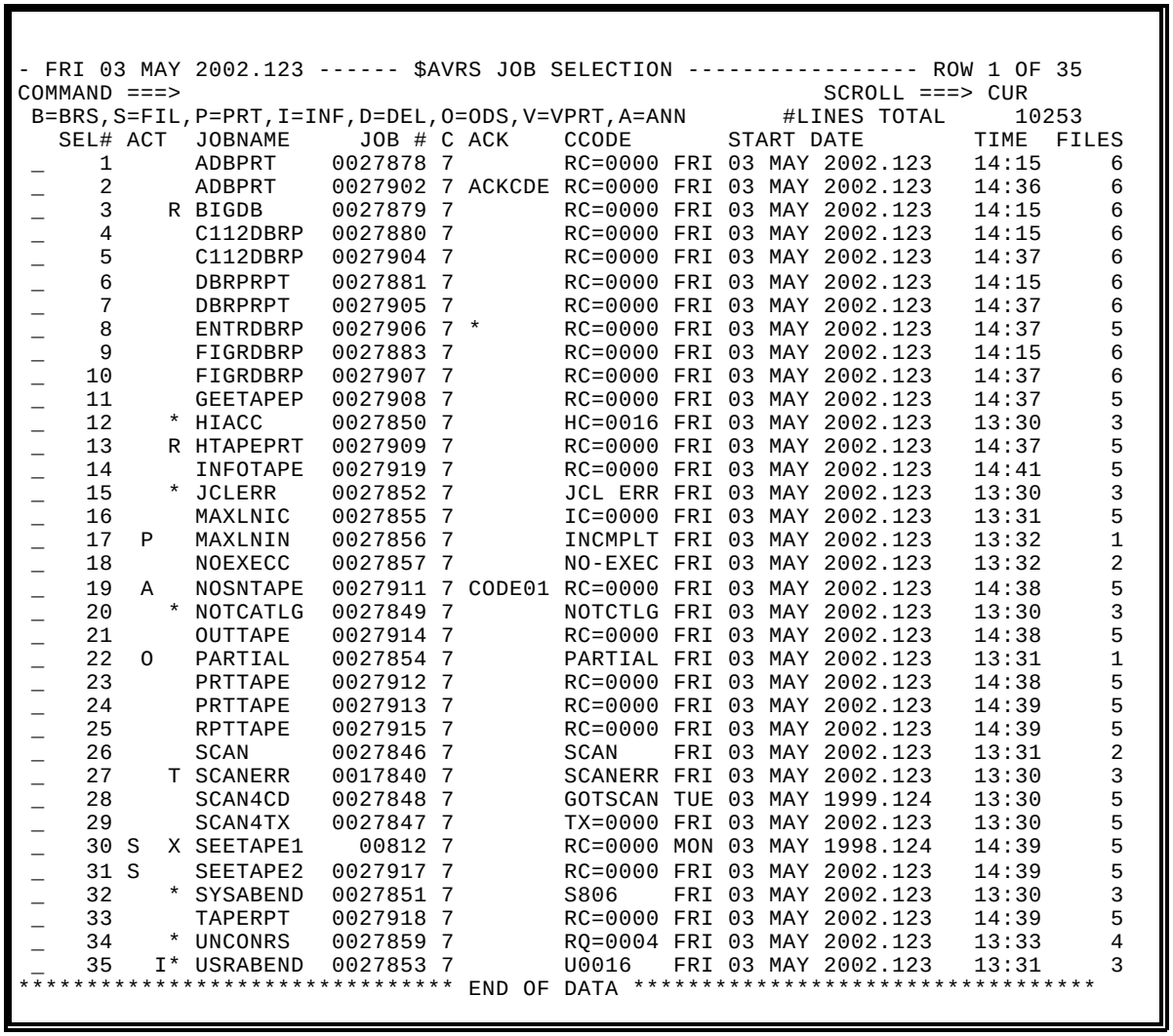
- **PAGE** - *Scroll one full screen's worth of data.* If the screen is in split screen mode, the number of current lines is calculated according to the amount actually viewable under the split.

- **HALF** - *Scroll half the number of lines on the current screen.* For split screen mode, lines are calculated as above.

- **CUR** - *Cursor mode. If the cursor is within the scrollable data, position the data so that the cursor is moved to the top, the bottom, the left or the right of the scrollable area, depending on the scroll command issued.* If the cursor is not within the scrollable area, or is already positioned at the relevant "edge" of the scrollable area, any scroll command is processed as if "PAGE" mode were in effect.

- **nnnn** - *Number from 1 to 9999.* Scroll the number of lines or columns specified.

The value in the SCROLL variable can be overridden by entering either a number from *1 to 999999 or the letter M (MAX) in the command area* in conjunction with a scroll PF Key. If M (MAX) is entered, the data is scrolled the maximum amount in the direction specified. For example, entering M (MAX) in the command line and then pressing the UP PF Key would result in the directory display being positioned to show entry number 1 at the top of the screen, as shown below.



```
- FRI 03 MAY 2002.123 ------ $AVRS JOB SELECTION ---------------- ROW 1 OF 35
COMMAND ===>                                                SCROLL ===> CUR
 B=BRS,S=FIL,P=PRT,I=INF,D=DEL,O=ODS,V=VPRT,A=ANN       #LINES TOTAL     10253
   SEL# ACT  JOBNAME     JOB # C ACK    CCODE       START DATE        TIME  FILES
 _    1      ADBPRT    0027878 7        RC=0000 FRI 03 MAY 2002.123   14:15     6
 _    2      ADBPRT    0027902 7 ACKCDE RC=0000 FRI 03 MAY 2002.123   14:36     6
 _    3    R BIGDB     0027879 7        RC=0000 FRI 03 MAY 2002.123   14:15     6
 _    4      C112DBRP  0027880 7        RC=0000 FRI 03 MAY 2002.123   14:15     6
 _    5      C112DBRP  0027904 7        RC=0000 FRI 03 MAY 2002.123   14:37     6
 _    6      DBRPRPT   0027881 7        RC=0000 FRI 03 MAY 2002.123   14:15     6
 _    7      DBRPRPT   0027905 7        RC=0000 FRI 03 MAY 2002.123   14:37     6
 _    8      ENTRDBRP  0027906 7 *      RC=0000 FRI 03 MAY 2002.123   14:37     5
 _    9      FIGRDBRP  0027883 7        RC=0000 FRI 03 MAY 2002.123   14:15     6
 _   10      FIGRDBRP  0027907 7        RC=0000 FRI 03 MAY 2002.123   14:37     6
 _   11      GEETAPEP  0027908 7        RC=0000 FRI 03 MAY 2002.123   14:37     5
 _   12    * HIACC     0027850 7        HC=0016 FRI 03 MAY 2002.123   13:30     3
 _   13    R HTAPEPRT  0027909 7        RC=0000 FRI 03 MAY 2002.123   14:37     5
 _   14      INFOTAPE  0027919 7        RC=0000 FRI 03 MAY 2002.123   14:41     5
 _   15    * JCLERR    0027852 7        JCL ERR FRI 03 MAY 2002.123   13:30     3
 _   16      MAXLNIC   0027855 7        IC=0000 FRI 03 MAY 2002.123   13:31     5
 _   17  P   MAXLNIN   0027856 7        INCMPLT FRI 03 MAY 2002.123   13:32     1
 _   18      NOEXECC   0027857 7        NO-EXEC FRI 03 MAY 2002.123   13:32     2
 _   19  A   NOSNTAPE  0027911 7 CODE01 RC=0000 FRI 03 MAY 2002.123   14:38     5
 _   20    * NOTCATLG  0027849 7        NOTCTLG FRI 03 MAY 2002.123   13:30     3
 _   21      OUTTAPE   0027914 7        RC=0000 FRI 03 MAY 2002.123   14:38     5
 _   22  O   PARTIAL   0027854 7        PARTIAL FRI 03 MAY 2002.123   13:31     1
 _   23      PRTTAPE   0027912 7        RC=0000 FRI 03 MAY 2002.123   14:38     5
 _   24      PRTTAPE   0027913 7        RC=0000 FRI 03 MAY 2002.123   14:39     5
 _   25      RPTTAPE   0027915 7        RC=0000 FRI 03 MAY 2002.123   14:39     5
 _   26      SCAN      0027846 7        SCAN    FRI 03 MAY 2002.123   13:31     2
 _   27    T SCANERR   0017840 7        SCANERR FRI 03 MAY 2002.123   13:30     3
 _   28      SCAN4CD   0027848 7        GOTSCAN TUE 03 MAY 1999.124   13:30     5
 _   29      SCAN4TX   0027847 7        TX=0000 FRI 03 MAY 2002.123   13:30     5
 _   30 S  X SEETAPE1    00812 7        RC=0000 MON 03 MAY 1998.124   14:39     5
 _   31 S    SEETAPE2  0027917 7        RC=0000 FRI 03 MAY 2002.123   14:39     5
 _   32    * SYSABEND  0027851 7        S806    FRI 03 MAY 2002.123   13:30     3
 _   33      TAPERPT   0027918 7        RC=0000 FRI 03 MAY 2002.123   14:39     5
 _   34    * UNCONRS   0027859 7        RQ=0004 FRI 03 MAY 2002.123   13:33     4
 _   35   I* USRABEND  0027853 7        U0016   FRI 03 MAY 2002.123   13:31     3
******************************** END OF DATA *********************************
```

Figure 10 Work Selection Directory-(left portion)

Scrolling right the maximum amount would produce a screen as follows:

```
- FRI 03 MAY 2002.123 ------ $AVRS JOB SELECTION ----------------- ROW 1 OF 35
COMMAND ===>                                                    SCROLL ===> CUR
 B=BRS,S=FIL,P=PRT,I=INF,D=DEL,O=ODS,V=VPRT,A=ANN       #LINES TOTAL     10253
   SEL# ACT  JOBNAME  ELAPS ROOM DESTID   ACCUMULATION DATE     TIME DBBLKS MREC
_     1       ADBPRT       1       LOCAL   FRI 03 MAY 2002.123   14:16      8  133
_     2       ADBPRT       3       LOCAL   FRI 03 MAY 2002.123   14:40      8  133
_     3     R BIGDB        2       LOCAL   FRI 03 MAY 2002.123   14:17      8  133
_     4       C112DBRP     1       LOCAL   FRI 03 MAY 2002.123   14:16     16  133
_     5       C112DBRP     0       LOCAL   FRI 03 MAY 2002.123   14:38      6  133
_     6       DBRPRPT      1       LOCAL   FRI 03 MAY 2002.123   14:16     16  133
_     7       DBRPRPT      2       LOCAL   FRI 03 MAY 2002.123   14:40      8  133
_     8       ENTRDBRP     3       LOCAL   FRI 03 MAY 2002.123   14:40      9  133
_     9       FIGRDBRP     1       LOCAL   FRI 03 MAY 2002.123   14:16      8  133
_    10       FIGRDBRP     0       LOCAL   FRI 03 MAY 2002.123   14:38      8  133
_    11       GEETAPEP     1       LOCAL   FRI 03 MAY 2002.123   14:38     10  133
_    12     * HIACC        0       LOCAL   FRI 03 MAY 2002.123   13:31      3  132
_    13     R HTAPEPRT     1       LOCAL   FRI 03 MAY 2002.123   14:39      6  133
_    14       INFOTAPE     0       LOCAL   FRI 03 MAY 2002.123   14:41      6  133
_    15     * JCLERR       0       LOCAL   FRI 03 MAY 2002.123   13:31      3  121
_    16       MAXLNIC      0       LOCAL   FRI 03 MAY 2002.123   13:32      6  133
_    17  P    MAXLNIN      0       LOCAL   FRI 03 MAY 2002.123   13:32      3   99
_    18       NOEXECC      0       LOCAL   FRI 03 MAY 2002.123   13:32      2  121
_    19  A    NOSNTAPE     0       LOCAL   FRI 03 MAY 2002.123   14:39      5  133
_    20     * NOTCATLG     0       LOCAL   FRI 03 MAY 2002.123   13:31      3  132
_    21       OUTTAPE      1       LOCAL   FRI 03 MAY 2002.123   14:40      5  133
_    22  O    PARTIAL      0       LOCAL   FRI 03 MAY 2002.123   13:31      1   85
_    23       PRTTAPE      0       LOCAL   FRI 03 MAY 2002.123   14:40      5  133
_    24       PRTTAPE      0       LOCAL   FRI 03 MAY 2002.123   14:40      5  133
_    25       RPTTAPE      2       LOCAL   FRI 03 MAY 2002.123   14:41     16  133
_    26       SCAN         0       LOCAL   FRI 03 MAY 2002.123   13:31      2  121
_    27     T SCANERR      0       LOCAL   FRI 03 MAY 2002.123   13:30      3  121
_    28       SCAN4CD      2       LOCAL   TUE 03 MAY 1999.124   13:32     35  133
_    29       SCAN4TX      2       LOCAL   FRI 03 MAY 2002.123   13:32     35  133
_    30 S  X SEETAPE1     0       LOCAL   MON 03 MAY 1998.124   14:40      5  133
_    31 S    SEETAPE2     1       LOCAL   FRI 03 MAY 2002.123   14:40      5  133
_    32     * SYSABEND     1       LOCAL   FRI 03 MAY 2002.123   13:31      3  132
_    33       TAPERPT      2       LOCAL   FRI 03 MAY 2002.123   14:41     27  133
_    34     * UNCONRS      1       LOCAL   FRI 03 MAY 2002.123   13:34      4  133
_    35   I* USRABEND      1       LOCAL   FRI 03 MAY 2002.123   13:32      3  132
******************************** END OF DATA *********************************
```

Figure 11 Work Selection Directory-(right portion)

Locate Command

Display of a multi-line directory may also be controlled through use of the **LOCATE** (or **L**) command. Locates may be done by entry number or alphabetically by the SORT field.

EXAMPLE:
L 132 *or* ***LOCATE 132***
RESULT:

The display is positioned so that the directory entry numbered '132' is the top line in the displayed entries.

To locate by SORT field, enter the LOCATE command in conjunction with a non-numeric value. The value entered will be compared with the values in the current sort field, and the entry which most closely matches the sort value, without exceeding it, is brought to the top of the screen display. (If entries are sorted in a descending manner, the entry which most closely matches, but is not 'less than' the entered value, is brought to the top of the screen). If no SORT value has been specifically entered, the JES name for the work is the default SORT value. For example, assume a directory of JOBs has been selected and has been sorted by CCODE (Completion Code). To locate the display to the first JCL ERROR, enter at the command line:

L JCL

This will position the selected entry at the second position on the screen.

The **LOCATE** command does not function with sort fields that are numeric (such as LINES, PAGES etc.), or with Date fields. The number located is assumed to be a directory entry number.

Sort Command

As noted above, work selection directories may be initially sorted by any display field by entering a **SORT** field value on the Primary Control Panel. After a directory of work has been presented, the work entries may be re-sorted on a different field by using the **SORT** command at the COMMAND line of the Work Selection Directory Panel. The last valid

command line sort parameters will be retained in the $AVRS primary panel **SORT FIELD** and **A/D** fields.

ENTRY:
SORT field or **SORT field,D**

where "field" is the name of the directory field that is desired to become the sort criteria. **NOTE**: Valid fields for the **SORT** command are identical to those listed for the Primary Control Panel **SORT** field listed previously.

Adding **,D** to the command forces the sort to be performed in descending sequence. If not entered, the sort is performed in ascending sequence.

Assume a JOB selection directory has been produced, and that it was initially sorted on the **LINES** field. Now the user wishes to quickly locate the jobs that specified his TSO userid in the NOTIFY field of the JCL.

ENTRY:
SORT NOTIFY

RESULT: The directory is redisplayed with the Jobs now sorted in NOTIFY value order. The user can then use the **LOCATE** command, specifying his userid, to quickly locate the directory display to the desired Jobs.

NOTE: Directory Sorting is valid only for selections of JOBs, STCs, TSUs or AH.

Refresh Command

Once a set of selection criteria has been entered from the initial panel, and a Work Selection Directory has been presented, the work entries initially selected remain static as long as the user remains on the selection directory or any of its sub functions. That is, if some of the work selected is deleted from the $AVRS data base while the directory remains selected, the entry line for the work is not removed from the directory display, although the entry is marked as "deleted" and is not available for selection. In addition, if $AVRS accumulates new work while the directory is held for display, the new work is not added to the selection, even if it would have met the selection criteria specified.

The **REFRESH** command allows the user to refresh the directory to the most current status without having to return to the Primary Control Panel.

ENTRY:
REFRESH or **REF**

RESULT: $AVRS re-selects the work from the data base, using the same criteria that were initially specified for the selection. Any work that has been deleted is removed from the directory, and any work that has been newly accumulated will be added.

The only Primary Control Panel field that is not necessarily used as it was originally entered is the **SORT** field. If a **SORT** command has been entered since the directory was selected, $AVRS uses the most recently entered SORT value in sorting the directory.

Chapter 5

DISPLAYING & PROCESSING ACCUMULATED WORK

Selecting work for processing from the Work Selection Directory is a simple process. Each process function has a single letter designated for selection of the function. Several process functions listed below are addressed in the remaining pages of this chapter. Default process function designators P, V, O, and D are detailed in Chapter 6, Printing, VTAM Printing, Outputting and Deleting Accumulated Work.

Default Process Function Designator	**Function**	**Invalid Selection Types**
A	Examine or update acknowledgment code and annotation for this work.	ST, FI, AH, FH
B	Browse all datasets of work entry as concatenated file.	
S	For selection of JOBs, STCs, TSUs, Syslog or Status Reports. Causes display of a FILE SELECTION PANEL for the entry. For Archive History and Fiche History, causes display the DETAIL PANEL for the entry.	
I	Display Step completion or JCL error information.	AH, FH, ST, FI
P	Print the lines for entry. Causes display of a Print Definition Panel. For Archive History Print the $AVRTAPE submission JCL is displayed.	FH
O	Output all lines to an output dataset. When this is specified, a Dataset Definition Panel is displayed.	AH, FH
D	Delete the entry and its associated lines from $AVRS database. Delete may be restricted by user according to installation definitions and security exit.	AH, FH,
R	Restore archived work from the archive dataset. **NOTE:** Valid only for Archive History.	
V	Print work via a VTAM Print interface. Causes display of VTAM Print Definition Panel. **NOTE**: Version 3.0B or above.	AH, FH
X	Restore archived work from the duplex archive dataset. NOTE: Valid only for Archive History.	

NOTE: Not all process functions are valid for each selection type. The Process Function Designators may be changed or eliminated by installation specification — see the $AVRFSEA Macro in the $AVRS Installation and Administrator's Manual. This document refers to the process functions by the default designators.

Entries may be selected for processing in either of two ways:

1) Entries may be processed by entering the letter of the desired function followed by the number of the desired entry in the COMMAND line.

   EXAMPLE:
   I 12 (or **I12**)

   RESULT: a display of the Step or Error Information Panel for entry number 12 in the current directory, if information is available for this work. When entering selections in this manner, the letter must precede the number.

2) Place the letter corresponding to the requested function in the input selection field to the left of the desired entry.

Multiple entries may be processed in this manner by entering a letter to the left of several work entries before pressing enter. As each function is completed (by entering the END PF Key to return to the directory), the next entry selected is automatically processed. Functions entered in this manner need not be of the same type, but may be mixed as desired. Any error which causes premature termination of an individual selection or which prevents selection of the function initially, results in "breaking" the multiple selections and returning to the directory, with an error message displayed. If multiple entries have been selected and it is then desired to terminate selections, the command "BREAK" or "BRK" may be entered from any sub-panel to return immediately to the directory.

$AVRS Browse Features

The **$AVRS Browse Feature** is used to display the output lines of a work entry that is currently in the $AVRS data base. Work presented via the BROWSE feature includes output lines from JOBs, STCs and TSUs, SYSLOG segments, Status Reports, and Fiche Index Reports. All the functions of browse described below are valid for examining these various types of work.

Entries selected from the $AVRS directories can be displayed for examination by placing a “B” in the input selection field to the left of the desired entry, as described above. Once BROWSE is invoked, there are several options available to the user:

- Data may be scrolled up or down and left or right to present the desired portion of the work on the viewing screen.

- Data may be moved to a desired line through use of the **Locate** command.

- Utilizing the **Lock** command, up to 60 columns of data can be "locked" on the screen while the remainder of the data is scrolled left or right.

- Utilizing the **Hex** command, the work may be displayed in hex format.

- Utilizing the **Cols** command, a columns line may be inserted at the top of the data.

- Utilizing the **CC** command (Carriage Control), the user may display carriage control to the left of the data.

- The **Find** command can be used to search for strings of data and can be limited through a variety of subparameters and subcommands. These are fully explained in the following paragraphs.

- The following screen commands facilitate quick screen navigation by "jumping" immediately to the requested function:

  FI - jumps to the FILES screen.
  AC - jumps to the ANNOTATIONS & ACKNOWLEDGMENT screen.
  IN - jumps to the INFORMATION screen.
  BR- jumps back to BROWSE from any of the above.

NOTE: Individual files from a selected JOB, STC or TSU may be browsed utilizing a FILE SELECTION PANEL, as explained in the section "Displaying JOBs, STCs, and TSUs". The following display is provided to illustrate the title line for the $AVRS BROWSE feature.

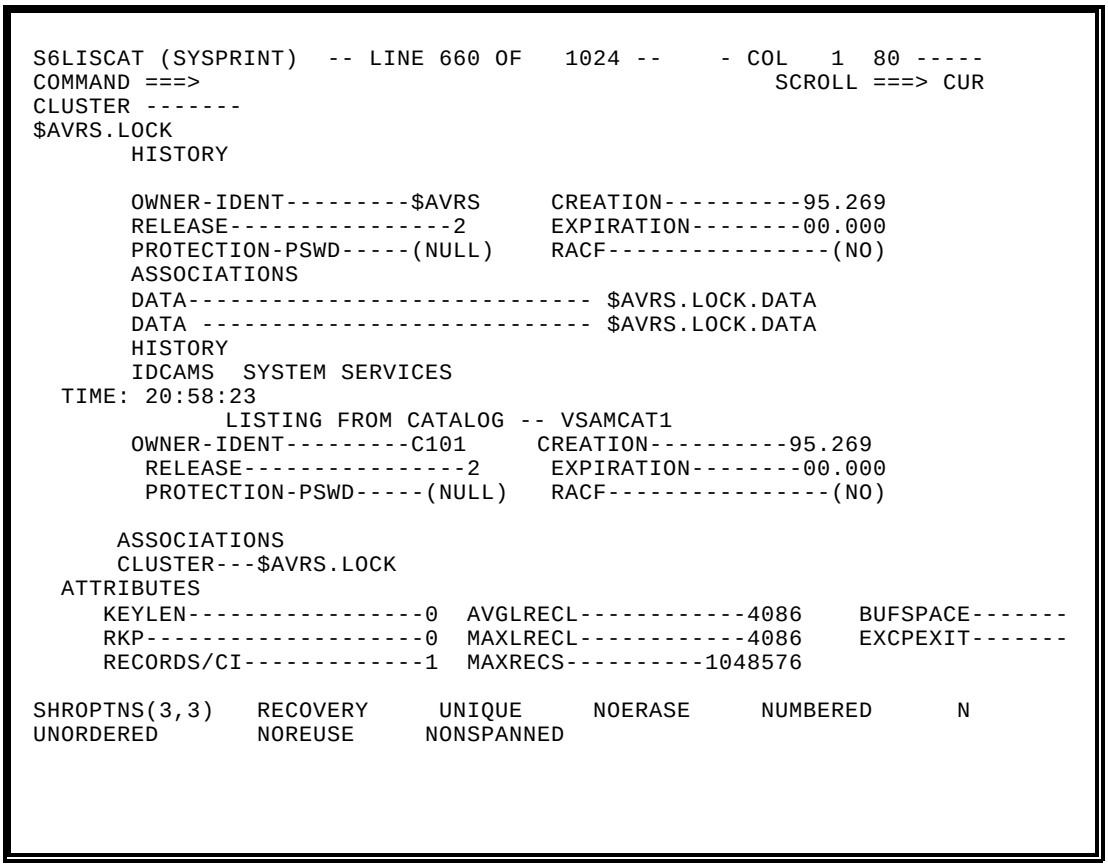


Figure 12 Browse Detail Display

The format of the header line on the $AVRS detail screen is as follows:

jesname (filename)----------LINE a OF bbb ------ COL d ee --L ff

jesname the name assigned by JES to the work.

filename names the file currently at the top of the screen.

a the number of the top line on the screen.

bbb the total number of lines in the file(s)

d and **ee** the beginning and ending columns of the data on the screen. If some number of columns are locked on the screen, 'COL' will appear as 'VCOL' and these numbers will represent the moveable columns only.

ff the number of columns 'locked' on this screen. If no columns are locked, this field is not shown.

NOTE: If a SYSLOG segment is displayed, the SYSLOG ID and the Start Date of the segment are displayed in the upper left corner instead of the JESNAME and FILENAME.

Locate Command

The **Locate** command is used to position the display to a specified line number. It is entered as follows:

ENTRY:
L nnnn or **LOC nnnn** or **LOCATE nnnn**

where **nnnn** is any number from **1** to **999999**.

RESULT: After the command has been entered, the line number entered becomes the top line in the current display.

Lock Command

The **Lock** command may be used to prevent a specified number of positions from being scrolled left or right.

ENTRY:
LOCK20
or
LOCK 20
where 20 is the number of positions/columns to be locked.

RESULT: If data is scrolled to the right or left, columns 1-20 of the data remain on the screen, while the rest of the data scrolls as specified. If a lock amount has been entered, the word COL on the header line is replaced by VCOL, (for variable columns), and the column amounts are followed by L 20, which indicates the number of locked columns/positions. In this case, the right-most portion of the header line would be **VCOL 21 80 L 20** (on an eighty column screen.) If the data were then scrolled 20 bytes to the right, the header would be **VCOL 41 100 L 20**. Positions 21 through 40 would no longer be displayed on the screen.

To unlock all positions:

ENTRY:
LOCK0

In the following example, a SYSLOG segment is displayed:

```
 SL 7777 -- 24 APR 1996.115 ----- LINE  1 OF    473 ------ COL   1  80 -----
 COMMAND ===>                                                SCROLL ===> CUR
FILE>   C112P.SYSLOG.JOB27656.D0000121.?,1996.123,16:07:09:02         <FILE
NC0000000 ESAM   96115 10:06:02.12 C118P    00000290  $CJ26340,P
NR0000000 ESAM   96115 10:06:02.94 JOB26340 00000090  $HASP608 C118AODR AWAITI
N 0200000 ESAM   96115 10:06:03.14 JOB26340 00000090  $HASP250 C118AODR IS PUR
NC0000000 ESAM   96115 10:06:14.15 INSTREAM 00000290  LOGON
N 0200000 ESAM   96115 10:06:18.35 TSU26363 00000081  $HASP100 C205     ON TSO
N 4000000 ESAM   96115 10:06:21.86 TSU26363 00000090  $HASP373 C205     STARTE
N 0000000 ESAM   96115 10:06:22.25 TSU26363 00000090  IEF125I C205 - LOGGED ON
N 0000000 ESAM   96115 10:06:23.00 CATALOG  00000290  IEF196I IEF237I 0166 ALL
N 0000000 ESAM   96115 10:06:23.10 CATALOG  00000290  IEF196I IGD104I CATALOG.
N 0000000 ESAM   96115 10:06:23.14 CATALOG  00000290  IEF196I DDNAME=SYS00067
N 0200000 ESAM   96115 10:07:41.39 JOB26364 00000081  $HASP100 C118AODR ON INT
N 0000000 ESAM   96115 10:07:41.65 JOB26364 00000290  IRR010I  USERID C118P
N 0020000 ESAM   96115 10:07:44.30 JOB26364 00000090  ICH70001I C118P    LAST
N 4000000 ESAM   96115 10:07:44.91 JOB26364 00000090  $HASP373 C118AODR STARTE
N 0000000 ESAM   96115 10:07:45.05 JOB26364 00000090  IEF403I C118AODR - START
N 0000000 ESAM   96115 10:07:45.19 CATALOG  00000290  IEF196I IEF237I 0166 ALL
N 0000000 ESAM   96115 10:07:45.25 CATALOG  00000290  IEF196I IGD104I CATALOG.
N 0000000 ESAM   96115 10:07:45.28 CATALOG  00000290  IEF196I DDNAME=SYS00068
N 0000000 ESAM   96115 10:07:48.32 LLA      00000290  IEF196I IEF237I 0164 ALL
N 0000000 ESAM   96115 10:07:49.38 LLA      00000290  IEF196I IEF285I   ESAM.A
N 0000000 ESAM   96115 10:07:49.41 LLA      00000290  IEF196I IEF285I   VOL SE
NC0000000 ESAM   96115 10:07:50.19 C118P    00000290  $CJ949,P
NR0000000 ESAM   96115 10:07:50.45 JOB00949 00000090  $HASP608 C118VCST AWAITI
N 0200000 ESAM   96115 10:07:50.50 JOB00949 00000090  $HASP250 C118VCST IS PUR
NC0000000 ESAM   96115 10:07:50.79 C118P    00000290  $CJ948,P
NR0000000 ESAM   96115 10:07:50.89 JOB00948 00000090  $HASP608 C118LINK AWAITI
N 0200000 ESAM   96115 10:07:52.15 JOB00948 00000090  $HASP250 C118LINK IS PUR
N C040000 ESAM   96115 10:07:52.43 JOB26364 00000090  ODDSESAM JOB=C118AODR ST
N 0004000 ESAM   96115 10:07:53.25 JOB26364 00000290  -
S                                                              ----PAGING COUNT
N 0004000 ESAM   96115 10:07:53.27 JOB26364 00000290  -JOBNAME  STEPNAME PROCS
S                                                     SERV  PG  PAGE  SWAP   V
N 0004000 ESAM   96115 10:07:53.30 JOB26364 00000290  -C118AODR TRM$     ASMCA
S                                                     4007   1     0     0
N 0000000 ESAM   96115 10:07:53.61 CATALOG  00000290  IEF196I IEF237I 0166 ALL
N 0000000 ESAM   96115 10:07:53.66 CATALOG  00000290  IEF196I IGD104I CATALOG.
N 0000000 ESAM   96115 10:07:53.69 CATALOG  00000290  IEF196I DDNAME=SYS00069
N C040000 ESAM   96115 10:08:29.85 JOB26364 00000090  ODDSESAM JOB=C118AODR ST
N 0004000 ESAM   96115 10:08:30.55 JOB26364 00000290  -C118AODR TRM$     ASMST
S                                                      125K   1     0     0
```

Figure 13 Lock Command Example: Panel 1

The user then enters "LOCK 36" from the command line to hold the time stamp on the screen. If the screen is then shifted right, it appears as follows:

```
 SL 7777 -- 24 APR 1996.115 ----- LINE      1 OF    473 ------VCOL  81 124  L 36
 COMMAND ===>                                              SCROLL ===> CUR
----+----1----+----2----+----3----+|----+----9----+----0----+----1----+----2----
FILE>        C112P.SYSLOG.JOB27656.D
NC0000000 ESAM     96115 10:06:02.12
NR0000000 ESAM     96115 10:06:02.94NG PURGE            PRIO  1 PURGE ANY
N 0200000 ESAM     96115 10:06:03.14GED
NC0000000 ESAM     96115 10:06:14.15
N 0200000 ESAM     96115 10:06:18.35INRDR
N 4000000 ESAM     96115 10:06:21.86D
N 0000000 ESAM     96115 10:06:22.25 - TIME=10.06.22
N 0000000 ESAM     96115 10:06:23.00OCATED TO SYS00067
N 0000000 ESAM     96115 10:06:23.10MVSVSAM.VCSUSER                      RETAINE
N 0000000 ESAM     96115 10:06:23.14
N 0200000 ESAM     96115 10:07:41.39RDR    C118P-VSE-TRMS-ASM    FROM TSU26317 C
N 0000000 ESAM     96115 10:07:41.65 IS ASSIGNED TO THIS JOB.
N 0020000 ESAM     96115 10:07:44.30ACCESS AT 09:25:13 ON WEDNESDAY, APRIL 24, 1
N 4000000 ESAM     96115 10:07:44.91D - INIT    1 - CLASS X - SYS ESAM
N 0000000 ESAM     96115 10:07:45.05ED - TIME=10.07.45
N 0000000 ESAM     96115 10:07:45.19OCATED TO SYS00068
N 0000000 ESAM     96115 10:07:45.25MVSVSAM.VCSUSER                      RETAINE
N 0000000 ESAM     96115 10:07:45.28
N 0000000 ESAM     96115 10:07:48.32OCATED TO SYS00037
N 0000000 ESAM     96115 10:07:49.38SMHV210.LINKLIB                        KEPT
N 0000000 ESAM     96115 10:07:49.41R NOS= ESAMV2.
NC0000000 ESAM     96115 10:07:50.19
NR0000000 ESAM     96115 10:07:50.45NG PURGE            PRIO  1 PURGE ANY
N 0200000 ESAM     96115 10:07:50.50GED
NC0000000 ESAM     96115 10:07:50.79
NR0000000 ESAM     96115 10:07:50.89NG PURGE            PRIO  1 PURGE ANY
N 0200000 ESAM     96115 10:07:52.15GED
N C040000 ESAM     96115 10:07:52.43EP=ASMCAT1 PGM=IEV90  RC=0000
N 0004000 ESAM     96115 10:07:53.25                          --TIMINGS (MINS.)--
S                                S---
N 0004000 ESAM     96115 10:07:53.27TEP    RC   EXCP   CONN    TCB    SRB  CLOCK
S                                IO SWAPS
N 0004000 ESAM     96115 10:07:53.30T1     00     73    342    .02    .00     .1
S                                 2     0
N 0000000 ESAM     96115 10:07:53.61OCATED TO SYS00069
N 0000000 ESAM     96115 10:07:53.66MVSVSAM.VCSUSER                      RETAINE
N 0000000 ESAM     96115 10:07:53.69
N C040000 ESAM     96115 10:08:29.85EP=ASMSTP2 PGM=IEV90  RC=0000
N 0004000 ESAM     96115 10:08:30.55P2     00    809   2521    .32    .01     .6
```

Figure 14 Lock Command Example: Panel 2

HEX Command

Records may be displayed in hexadecimal format through use of the **HEX** command. In HEX mode, lines are displayed as follows:

- 1st line - the actual record in display format.
- 2nd line - represents the hex zones portion of each character.
- 3rd line - represents the hex numerics portion of each character.

A divider line of dashes follows the third line. This format is repeated for as many lines as will fit on the current display screen. The SYSLOG Segment from the previous example appears as follows if displayed in Hex format:

```
 SL 7777 -- 24 APR 1996.115 ----- LINE     42 OF    473 ------ COL  19  98 -----
 COMMAND ===>                                                  SCROLL ===> CUR
 96115 10:08:30.90 CATALOG  00000290  IEF196I IEF237I 0166 ALLOCATED TO SYS00070
4FFFFF4FF7FF7FF4FF4CCECDDC44FFFFFFFF44CCCFFFC4CCCFFFC4FFFF4CDDDCCECC4ED4EEEFFFFF
096115010A08A30B900313136700000002900095619690956237900166013363135403602820007 0
 ------------------------------------------------------------------------------
 96115 10:08:30.95 CATALOG   00000290   IEF196I  IGD104I  CATALOG.MVSVSAM.VCSUSER
4FFFFF4FF7FF7FF4FF4CCECDDC44FFFFFFFF44CCCFFFC4CCCFFFC4CCECDDC4DEEEECD4ECEEECD444
096115010A08A30B950313136700000002900095619690974104903131367B4525214B5324259000
 ------------------------------------------------------------------------------
 96115 10:08:30.99 CATALOG  00000290  IEF196I DDNAME=SYS00070
4FFFFF4FF7FF7FF4FF4CCECDDC44FFFFFFFF44CCCFFFC4CCDCDC7EEEFFFFF4444444444444444444
096115010A08A30B990313136700000002900095619690445145E28200070000000000000000000
 ------------------------------------------------------------------------------
 96115 10:08:32.84 JOB26365 00000081  $HASP100 C118AODR ON INTRDR    C118P-NET2
4FFFFF4FF7FF7FF4FF4DDCFFFFF4FFFFFFFF445CCEDFFF4CFFFCDCD4DD4CDEDCD4444CFFFD6DCEF4
096115010A08A32B840162263650000000810 0B8127100031181649065095394900003118705532 0
 ------------------------------------------------------------------------------
                                    C118AODR
4444444444444444444444444444444444444CFFFCDCD4444444444444444444444444444444444
000000000000000000000000000000000000000311816490000000000000000000000000000000
 ------------------------------------------------------------------------------
 96115 10:08:32.95 JOB26365 00000290  IRR010I  USERID C118P    IS ASSIGNED TO TH
4FFFFF4FF7FF7FF4FF4DDCFFFFF4FFFFFFFF44CDDFFFC44EECDCC4CFFFD4444CE4CEECCDCC4ED4EC
096115010A08A32B950162263650000002900099901090042599403118700009201229755403603 8
 ------------------------------------------------------------------------------
 96115 10:08:33.16 JOB26365 00000090  $HASP520 C118AODR ON L1.JT1
4FFFFF4FF7FF7FF4FF4DDCFFFFF4FFFFFFFF445CCEDFFF4CFFFCDCD4DD4DF4DEF444444444444444
096115010A08A33B160162263650000000900 0B81275200311816490650 31B131000000000000000
 ------------------------------------------------------------------------------
 96115 10:08:33.34 JOB26364 00000090  ODDSESAM JOB=C118AODR STEP=GENRSTP3 PGM=IE
4FFFFF4FF7FF7FF4FF4DDCFFFFF4FFFFFFFF44DCCECECD4DDC7CFFFCDCD4EECD7CCDDEEDF4DCD7CC
096115010A08A33B340162263640000000900064425214016 2E3118164902357E755923730774E95
 ------------------------------------------------------------------------------
 96115  10:08:33.52  JOB26364  00000290   -C118AODR  TRM$       GENRSTP3       00
4FFFFF4FF7FF7FF4FF4DDCFFFFF4FFFFFFFF446CFFFCDCD4EDD544444CCDDEEDF4444FF44444FF44
096115010A08A33B52016226364000000290000311816490394B0000075592373000000000003000
 ------------------------------------------------------------------------------
                                          1562   1     0     0     2     0
4444444444444444444444444444444444444FFFF444F44444F44444F44444F44444F4444444444
000000000000000000000000000000000000000156200010000000000000000020000000000000000
 ------------------------------------------------------------------------------
 96115 10:08:34.12 JOB26364 00000090  ODDSESAM JOB=C118AODR STEP=BR14STP4 PGM=IE
```

Figure 15 Hexadecimal Browse Display

The **HEX** command can be entered as follows:

ENTRY: **HEX ON** *RESULT:* turns on hex mode.
ENTRY: **HEX OFF** *RESULT:* turns off hex mode.
ENTRY: **HEX** *RESULT:* switches from one mode to the other.

For example, if the display is in HEX mode, entering HEX without any sub-parameters switches the display to non-HEX mode.

Column Command

Enter, **"COL", "COLS"**, **"COL ON"** or **"COLS ON"**, at the command line to place a scale at the top of the display. This line appears as follows:

----+----1----+----2----+----3----+----4 etc.

A plus sign (+) is placed in the fives positions, and a digit is placed in the tens position. The columns line moves with the data as it is scrolled left or right to indicate the true column in the data.

To remove the column line from the display, enter: "RES" or "RESET"; "COL" or "COLS"; "COL OFF" or "COLS OFF".

ENTRY: **COL** *RESULT:* switches from one mode to the other.

CC Command

Enter, "**CC**" or "**CC ON**", at the command line to display any carriage control at the left of the display. Enter "**CC OFF**" to remove carriage control, or enter "**CC**" to switch from one mode to the other.

ENTRY: **CC** *RESULT:* switches from one mode to the other.

Find Command

The **Find** command can be used to search for strings of data. **Find** searches are normally carried out in a forward direction through the data, starting from the current display position. Searches may also be performed in a backward direction by use of a Find subparameter. Find strings can be specified in hex or display characters, and the search can be for upper or lower-case characters or "text" strings (disregarding case). Searches may also be performed for non-display characters.

Other sub-commands can be specified:

- to limit the search to specified columns or positions.
- to find the first or last occurrence of a string.
- to find all occurrences of a string.
- to find a string only when it occurs as a prefix, suffix, or word.

The maximum length of the entire FIND command including the end delimiter and sub parameters is 80 characters, depending on the form of the command used. If the command exceeds the maximum length 'STRING TOO LONG' will appear in the short message field.

NOTE: Searches may be continued with the current **Find** parameters by using the **RFIND** command or PF Key.

Find Subparameters

A **FIND** command string may be limited as to its placement within a word through use of FIND subparameters. If utilizing a **FIND** subparameter, it must be enclosed in quotes unless it is the only parameter entered. For example, to find the string 'PREV' with no qualifiers, FIND PREV may be entered. However, to find the same string in column 20, enter: **FIND 'PREV' 20**. To find the previous occurrence of the string '20', enter: **FIND PREV '20'** Subparameters of **FIND** are:

PRE or **PREFIX**
SUF or **SUFFIX**
PREV
WORD
FIRST
LAST
ALL

The chart below illustrates the purpose and use of each of the FIND subparameters.

FIND Subparameters

Subparameter	Purpose	Note	Example Entry
PRE or **PREFIX**	To limit the placement of a FIND command string within a word to the beginning.	Preceded by a non-alphanumeric character and followed by an alphanumeric character.	**F SAM PRE** *strings ("same ", "sammy" and* ".samson" *would be considered as matches. Strings "2same", and "sam)" would not.* **FIND SAM WORD** *only the string ' sam ' in the previous examples, is considered a match.*
SUF or **SUFFIX**	To limit the placement of a FIND command string within a word to the end.	Preceded by an alphanumeric character and followed by a non-alphanumeric character.	**F SAM SUF**
PREV	To enable searches to be carried backward through the data.		**FIND string PREV**
WORD	To limit a FIND command string to the entire word.	Preceded and followed by a non-alphanumeric character.	**FIND SAM WORD**
FIRST	Causes $AVRS to locate the first occurrence of the string within the data, regardless of current display position.		**FIND string FIRST**
LAST	Causes $AVRS to locate the last occurrence of the string within data.		**FIND string LAST**
ALL	Causes $AVRS to find all occurrences of a specified string.	A message is displayed on the top line of the screen indicating the number of times string occurs, and the number of lines on which it occurs. Cursor is positioned to first occurrence.	**FIND string ALL**

Entering FIND Strings

A string may be entered "as is" if it contains only alpha-numeric (non-blank) or national characters.

EXAMPLE:
F ABC$#LLC

If the string contains blanks, enclose it in single quotes.

EXAMPLE:
F 'the string '

If the string contains quotation marks, enclose it in single quotes or parentheses; if it contains single quotes, enclose in quotation marks or parentheses.

EXAMPLE:
F 'HE SAID "QUOTE ME"'
F "HENRY'S"
F (HE SAID "QUOTE ME")

Altering FIND Search Criteria Parameters

Once a find command has been entered, the same string may be searched using different search criteria through use of the **FIND *** command. Entering **FIND *** causes the search to be continued for the same string previously entered. Previously specified sub-parameters are no longer in effect. New sub-parameters may be specified with the command. For example, assume the following **FIND** command had been previously entered:

F DIFFERENT PREFIX PREV

If it is desired to continue the search in a forward direction, enter: F *. The sub-parameter 'PREFIX' is not in effect unless re-specified. To then find all occurrences of 'different' as a word, enter:

F * ALL WORD

CAPS Command

The **CAPS** command may be used to determine whether search strings and data are compared 'as is', or as if both are in upper case. The default mode for $AVRS browse is "**CAPS ON**". To set the mode enter:

CAPS ON or CAPS OFF

In **CAPS ON** mode, both the search string and the data are treated as if they are upper case. For example if the command F start is entered, strings 'start' 'START' and 'Start' are found.

In **CAPS OFF** mode, search strings and data are compared 'as is.' In the previous example only 'start' would be found. **CAPS OFF** mode may be overridden through use of the 'text' search described on the following pages.

NOTE: Searches for hex strings and non-display characters are not affected by the **CAPS** mode.

Limit Command

Upon entry of a **FIND** command, $AVRS searches only a limited number of lines. If the string is not found within the line limit, a message is issued indicating the number of lines searched.

The default limit is **10,000 lines**. This limit may be changed through use of the **LIMIT** command.

ENTRY:
LIM nnnnn or **LIMIT nnnnn**

nnnnn may be any number from **1** to **999999**.

RESULT: alters the search limit.

The search may be continued through the use of the **RFIND** command or **RFIND PFkey**. If the string is still not found, the number of lines searched is incremented until top or bottom of data is reached. The FIND LIMIT is part of your User ISPF Profile and the optional UPFL product, (USER PROFILE), so that it will be retained between sessions.

Limiting a Search by Columns

A **FIND** command may be limited to searching only for occurrences of a string beginning in a specified column, or occurring within a range of columns.

EXAMPLE:
F SPECIFIED 24

RESULT: finds the string 'specified' only when it begins in column 24.

EXAMPLE:
F SPECIFIED 12 60

RESULT: finds the same string if it occurs anywhere between columns 12 and 60.

NOTE: The string must both begin and end entirely within the specified column range. When columns are specified, the numbers given must be less than or equal to the maximum length of a record in the file currently displayed.

Finding HEX Characters

A **FIND** search may be performed for a hexadecimal character string, rather than a display character string by placing the string in single quotes preceded by an '**x**'.

EXAMPLE:
FIND x'C1C3F4'

would result in the same search criteria as entering:

FIND AC4

Hex characters need not be limited to corresponding display characters; the command **FIND x'012E1A'** is a valid **FIND** command. Characters specified within a hex string must be valid hex characters **0** through **9** and **A** through **F**. An even number of characters must be specified.

Hex searches are performed 'as is', and are not affected by the current CAPS mode.

Finding Text Strings

A **FIND** search may be performed to ignore upper or lower case characteristics of a string by specifying the search string as a '**Text**' string. To designate a string as a '**text**' string, place the string in apostrophes and precede it with a "**t**".

EXAMPLE:
FIND t'report'

RESULT: Both the search string and the report data searched are compared as if they contained only upper case characters. For this search argument, strings such as 'report', 'REPORT', and 'Report', would all be considered as matches. Without the 'text' designation, only the string 'report' would be considered a match. This type of search is effective only when operating in '**CAPS OFF**' mode (see '**CAPS COMMAND**'), since in **CAPS ON** mode, all searches are performed to ignore case, whether or not the string is entered as a '**text**' string.

Finding Non-Display Characters

Data may be searched for the occurrence of any non-display characters. (Hexadecimal character strings which do not have a display representation, such as x'01' or x'FF').

EXAMPLE:
F P'.'

RESULT: finds occurrences of non-display characters. Any such characters are found one at a time, and the hexadecimal representation of the character is displayed in the message at the top of the screen.

Information Panel

In order to display Step Completion or JCL error information, the **INFORMATION PANEL** may be selected by entering an "I" next to the desired entry on a Work Selection Panel, or by entering "I" followed by the entry number from the command line of the Work Selection Panel.

The following is a sample Information Panel listing containing Step Completion information:

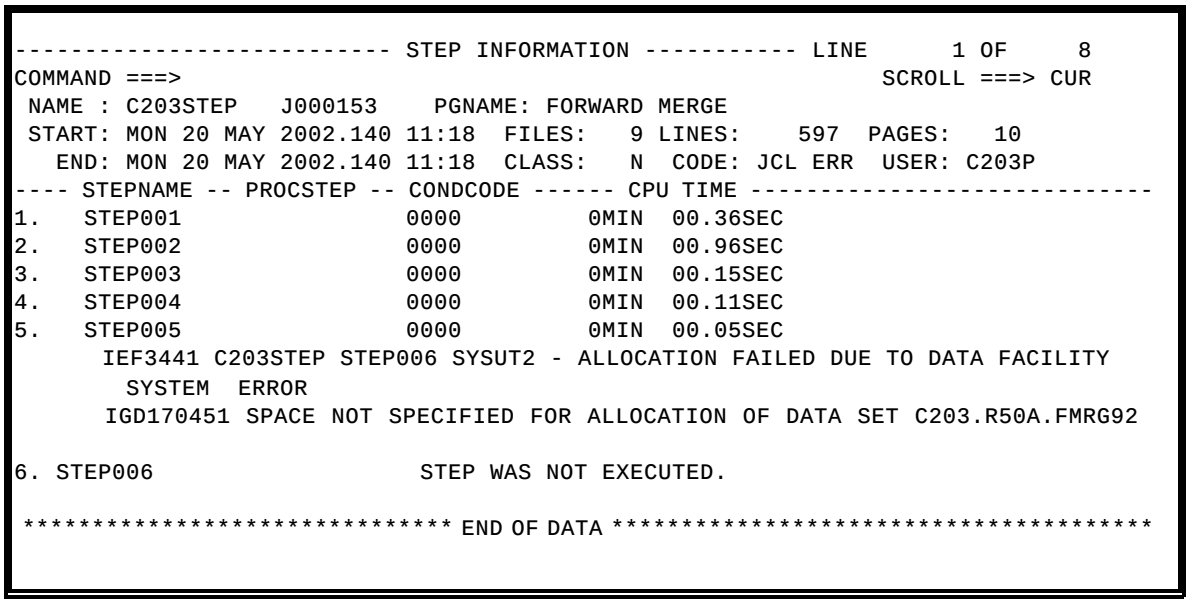


Figure 16 Info Panel, Example 1

The top two lines of the Information Panel are a header line and a command line, as with all other $AVRS panels. The third through fifth line present a summary of information about the selected entry, including JES name and number, start and end dates and times, completion code, programmer name, etc. The information presented starting on the sixth line is variable, depending on the disposition of the job.

If the JOB, STC or TSU has executed, regardless of completion code, the remainder of the display reflects a summary of step completion information. Steps are numbered for convenience in viewing. Each step lists the *STEPNAME, PROCSTEP NAME, COMPLETION CODE, and CPU TIME* for the step. If the step did not execute, the system message indicating this is displayed instead of the *COMPLETION CODE and CPU TIME*. At the top of the display, before the first step completion line, are any messages from the JESLOG file that your installation has specifically requested be displayed. (Refer to the Installation and Administrators Manual topic $AVRMLST for information about setting up $AVRS to extract specific messages). Intermixed with the Step Completion lines are any exceptional system IEF messages, plus other messages specifically designated through the customization procedure. By default, $AVRS displays any IEF messages from the SYSMSG file other than those associated with normal step completion. The IEF messages to be excluded from display are also alterable by installation customization. Exceptional messages

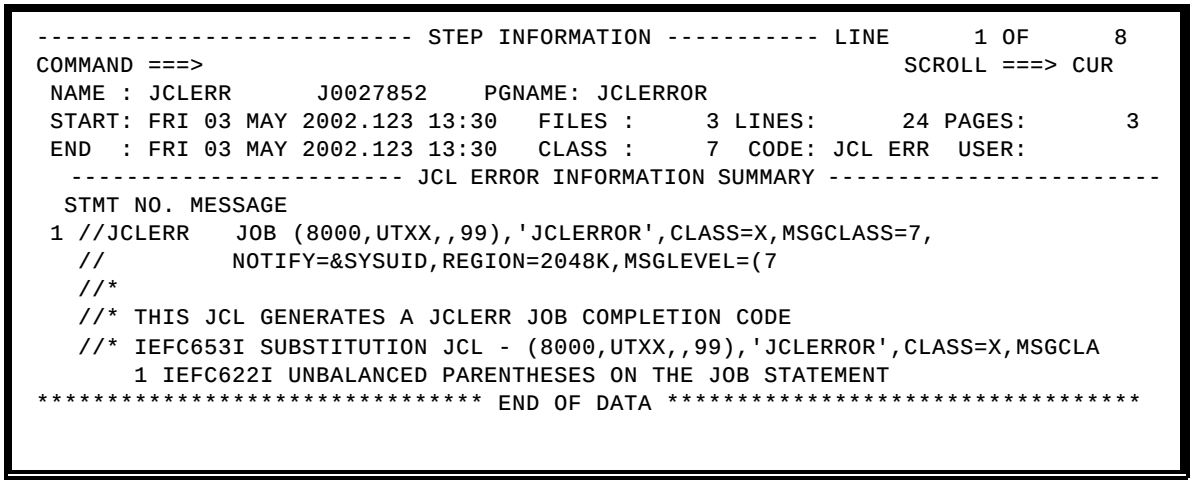
displayed in this manner appear either immediately before or after the Step Completion line for the step they relate to, depending on where they occur in the SYSMSG file.

Following the list of Step Completion lines and related messages is a listing of any $AVRS Special Control Statement from the JCL (//*$AVRS D= or //*$AVRS C=), if any are present.

If the JOB, STC or TSU did not execute due to a JCL error, the display beginning on the sixth line is a breakout of JCL error information. Listed in this case are the source JCL statements containing errors, followed by the system error message identifying the error. Any JCL error messages not relating to a specific statement (unnumbered messages), are listed in this section as they occur.

NOTE: $AVRS depends on the statement numbers present in the system JCL file to produce the JCL error display mentioned above. If the JCL file is not present, or if MSGLEVEL has been specified to suppress statement numbering ("MSGLEVEL=2"), the matching of error messages with JCL statements will not be possible, and no JCL statements will appear in the Information Panel Listing, although the error messages will still be displayed.

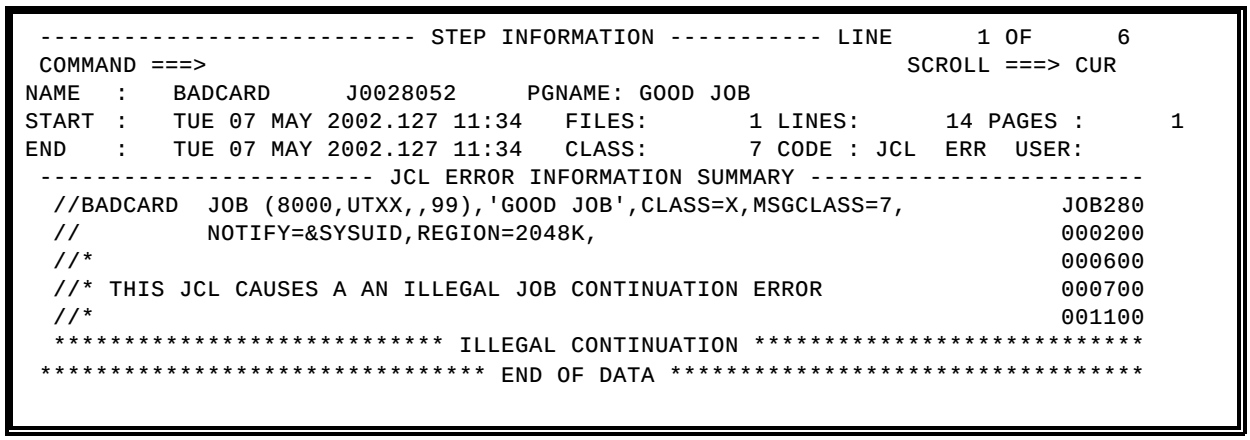
The following is a sample Information Panel containing JCL error information.



```
--------------------------- STEP INFORMATION ----------- LINE      1 OF      8
COMMAND ===>                                              SCROLL ===> CUR
 NAME : JCLERR      J0027852    PGNAME: JCLERROR
 START: FRI 03 MAY 2002.123 13:30   FILES :     3 LINES:      24 PAGES:       3
 END  : FRI 03 MAY 2002.123 13:30   CLASS :     7  CODE: JCL ERR  USER:
  ----------------------- JCL ERROR INFORMATION SUMMARY -----------------------
  STMT NO. MESSAGE
 1 //JCLERR   JOB (8000,UTXX,,99),'JCLERROR',CLASS=X,MSGCLASS=7,
   //         NOTIFY=&SYSUID,REGION=2048K,MSGLEVEL=(7
   //*
   //* THIS JCL GENERATES A JCLERR JOB COMPLETION CODE
   //* IEFC653I SUBSTITUTION JCL - (8000,UTXX,,99),'JCLERROR',CLASS=X,MSGCLA
       1 IEFC622I UNBALANCED PARENTHESES ON THE JOB STATEMENT
****************************** END OF DATA *********************************
```

Figure 17 Info Panel, Example 2

For some types of JCL errors (for example, Invalid Job Card), the only system file present is the JESLOG, and the error is identified within that file. For these types of errors, the section of the JESLOG listing the error is displayed on the Information Panel. The following is an example of an Information Panel with JCL error information extracted from the JESLOG:



```
 -------------------------- STEP INFORMATION ----------- LINE      1 OF      6
 COMMAND ===>                                          SCROLL ===> CUR
NAME   :   BADCARD     J0028052     PGNAME: GOOD JOB
START  :   TUE 07 MAY 2002.127 11:34   FILES:       1 LINES:      14 PAGES :      1
END    :   TUE 07 MAY 2002.127 11:34   CLASS:       7 CODE : JCL  ERR  USER:
 ----------------------- JCL ERROR INFORMATION SUMMARY -----------------------
  //BADCARD  JOB (8000,UTXX,,99),'GOOD JOB',CLASS=X,MSGCLASS=7,           JOB280
  //         NOTIFY=&SYSUID,REGION=2048K,                                 000200
  //*                                                                     000600
  //* THIS JCL CAUSES A AN ILLEGAL JOB CONTINUATION ERROR                 000700
  //*                                                                     001100
  *************************** ILLEGAL CONTINUATION ***************************
 ******************************** END OF DATA ********************************
```

Figure 18 Info Panel, Example 3.

If one or more of the requisite system datasets (JESLOG, JCL and/or SYSMSG) are not present in the output, the information panel will not be displayed when selected. The message `$AVR908A - STEP INFORMATION NOT AVAILABLE` is displayed instead.

Acknowledgment Code and Annotations

An acknowledgment code and/or free form annotations may be appended to the output of any Job on the $AVRS data base by selecting the **Annotation and Acknowledgment Feature** from the directory (default selection character "A"). The Acknowledgment Code is a six character field which may be used, for example, by a Programmer or Operator to acknowledge their examination of the work's output and/or annotations. The annotations may be used, for example, to note additional information about a Job, recommendations for follow-up in the case of a failing job, or for any other purpose desired. The Acknowledgment Code is listed as part of the directory entry for each Job for ease of reference and also for selection purposes.

When Annotation/Acknowledgment is selected, a panel (similar to the one below) is presented which lists the current acknowledgment code and any existing notations.

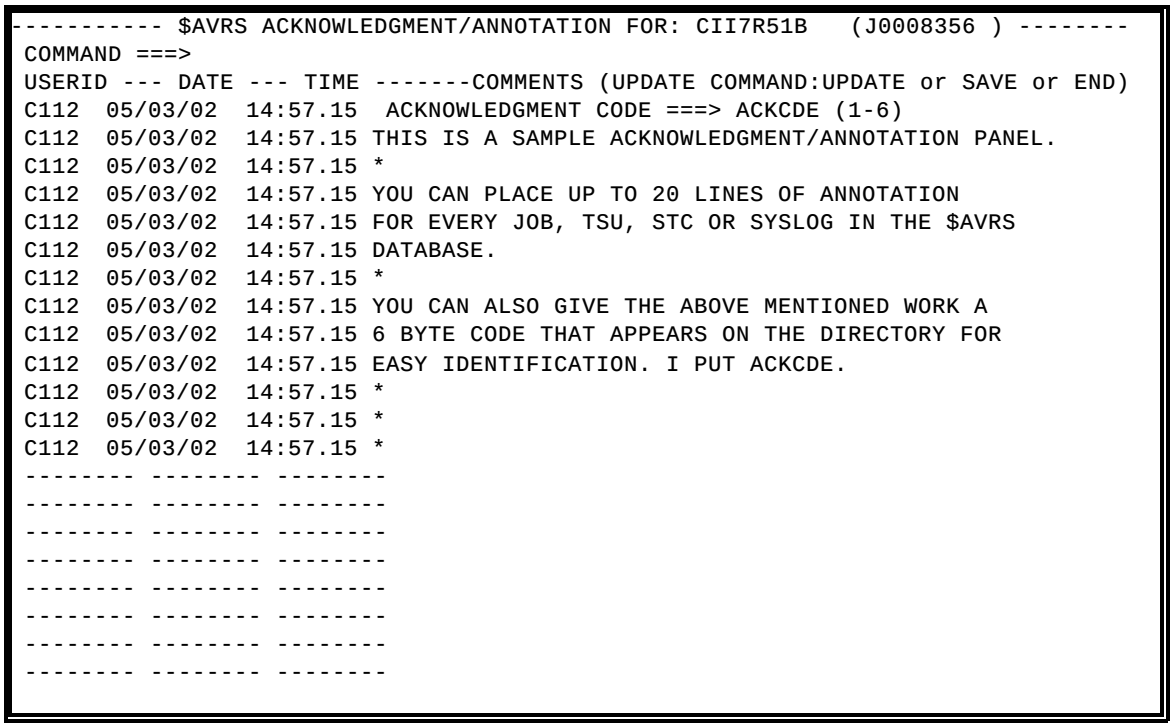


Figure 19 Annotation/Acknowledgment Panel

The userid of the last person to update the acknowledgment code and each notation line, along with the date and time of last update are listed to the left of each line. For any line which has not yet been used, dashes (-) appear in the userid, date and time fields. Twenty lines of 50 characters each are available for notation. To update an existing line, or to add a new line, simply use the NewLine key to place the cursor on the desired line and type in the desired data.

To save any updates, enter one of the following commands: UPDATE, SAVE, or END. Equivalent PK Keys to these commands may also be used. A message is issued requesting the user to press ENTER to confirm the updates. Press ENTER to record the updates and return to the directory. Press the END key to bypass the update and remain on the Annotation/Acknowledgment Panel. To cancel any entered changes, enter "CAN" or "CANCEL" from the command line.

To avoid data base contention, the $AVRS online interface does not directly update the Annotation/Acknowledgment record, but instead routes a transaction key request for update to the $AVRS Accumulation Task. Although the Accumulation Task normally processes this

request quickly, *it is possible for two or more users to attempt to update the same Acknowledgment/Annotation record at the same time.* Also, if the Accumulation Task is not currently active, the requests are not processed until it is activated. If the Acknowledgment Code or an existing notation line are updated by a second user before the first is processed, the last update overrides the first. If new lines are added by more than one user in this situation, the added lines from the last update are appended after any which were added by previous users as long as slots are available. If not enough slots are available, the requested lines will not be added. In any contention situation such as this, the $AVRS Accumulation Task issues a message indicating the action taken (see $AVRS message numbers 479, 481, 482, 483, or 484). Whenever potential contention exists, it may be prudent to verify update of the record by re-selection at a later time.

Displaying JOBs, STCs and TSUs

There are three primary methods of displaying output or information for JOBs, STCs or TSUs:

- All of the output lines may be browsed as a single **CONCATENATED** file by selecting the entry for BROWSE.

- A FILE SELECTION PANEL listing all the files present in the job output, as well as detailed information about each file may be selected for the entry. From this panel, individual files may be selected for browsing with all BROWSE options available.

- An INFORMATION PANEL may be selected which provides a breakout of information related to Step Completion, or information about JCL errors. If no Step information is available, the $AVR908A message will be displayed.

In order to provide the maximum utility and ease of use, $AVRS allows the user to switch back and forth between these three functions with a single command entered at the Command Line of the current function panel. The five commands, (with the valid short forms indicated in parentheses) are listed below:

Shortcuts For Screen Navigation

After a job has been selected from the $AVRS JOB SELECTION panel, enter any of the two character commands listed below on the command line to switch immediately to the desired panel.

(AC)k To switch to the Annotation and Acknowledgment panel
(BR)OWSE To switch to the concatenated Browse Facility
(FI)LES To switch to the File Selection Directory
(IN)FO To switch to the Information Panel
(PR)INT To switch to the Print Panel

For example, assume you have selected a directory of JOBs, and then select the Information panel for a particular JOB by placing an **I** to the left of the entry on the directory panel. Having verified step completion codes for the Job from this panel, you now wish to see what output files were produced. You may enter **FI** at the command line of the Information Panel. $AVRS immediately presents the File Selection Directory for the same job. You now wish to view detail lines of the job output, but prefer to be able to scroll through all the output without having to switch between files. You may enter **BR** at the command line of the File Selection Panel. You will then enter the $AVRS Browse function with all Job output lines listed for the same work.

The following paragraphs describe in greater detail the Browse Facility and the File Selection Directory, along with examples of each. Utilization of the Information Panel is described in previous paragraphs of this chapter.

(BR)OWSE Facility

Entering **B** (or an installation defined character) next to a JOB, STC or TSU entry on a Work Selection Directory results in the presentation of all output lines from the work as a concatenated file.

For browse purposes, lines are numbered as if they were contained in one file, that is, consecutively from the first line of the first file to the last line of the last file.

Two commands, **N** (NEXT) and **P** (PREVIOUS), may be used with browse. (Although this paragraph deals with concatenated browse, these two commands are valid for any type of

browse.) **N** and **P** under concatenated browse simply position the display to the first line of the next or previous file. The current file is considered to be the one which contains the line currently at the top of the screen. One other command, **TF** (Top of File), is helpful in positioning the display under concatenated browse. Entering this command results in the display being positioned to show the first line of the current file as the top line of the display.

The following example shows the output from a JOB as it would appear if selected for concatenated browse:

```
 DBRPRPT  (JESLOG) -------------- LINE   1 OF    623 ------ COL   1  80 -----
 COMMAND ===>                                                SCROLL ===> CUR
FILE>  C112P.DBRPRPT.JOB27881.D0000002.JESMSGLG,1996.124,14:16:47:95     <FILE
                  J E S 2  J O B  L O G  --  S Y S T E M  E S A M  --  N O

14.15.20 J0027881  IRR010I  USERID C112P    IS ASSIGNED TO THIS JOB.
14.15.21 J0027881  ICH70001I C112P   LAST ACCESS AT 14:15:11 ON FRIDAY, MAY 3,
14.15.23 J0027881  $HASP373 DBRPRPT  STARTED - INIT    4 - CLASS X - SYS ESAM
14.15.23 J0027881  IEF403I DBRPRPT - STARTED - TIME=14.15.23
14.16.33 J0027881  ODDSESAM JOB=DBRPRPT STEP=STEP1 PGM=$AVRDBRP  RC=0000
14.16.35 J0027881  -                                                --TIMINGS (M
14.16.35 J0027881  -JOBNAME  STEPNAME PROCSTEP  RC   EXCP   CONN    TCB    SRB
14.16.35 J0027881  -DBRPRPT  STEP1              00    223   1153    .04    .00
14.16.35 J0027881  IEF404I DBRPRPT - ENDED - TIME=14.16.35
14.16.36 J0027881  -DBRPRPT  ENDED.  NAME-ONLINE DB LIST     TOTAL TCB CPU TIM
14.16.36 J0027881  $HASP395 DBRPRPT  ENDED
------ JES2 JOB STATISTICS ------
03 MAY 1996 JOB EXECUTION DATE
 27 CARDS READ
617 SYSOUT PRINT RECORDS
  0 SYSOUT PUNCH RECORDS
 65 SYSOUT SPOOL KBYTES
1.24 MINUTES EXECUTION TIME
FILE> C112P.DBRPRPT.J0027881.D0000003.JESJCL,1996.124,14:16:47:95      <FILE
   1 //DBRPRPT  JOB (8000,UTXX,,99),'ONLINE DB LIST',CLASS=X,MSGCLASS=7,
     //      NOTIFY=&SYSUID,REGION=2048K
     IEFC653I SUBSTITUTION JCL - (8000,UTXX,,99),'ONLINE DB LIST',CLASS=X,M
   2 //STEP1   EXEC PGM=$AVRDBRP
   3 //STEPLIB DD  DSN=S600.R41A.LOADLIB,DISP=SHR
   4 //$AVRPRT  DD  SYSOUT=*
   5 //$AVRLOCK DD  DSN=C106.$AVRS.MULTI.LOCK,DISP=SHR
   6 //SYSPRINT DD  SYSOUT=*
   7 //SYSOUT   DD  SYSOUT=*
     //*
   8 //SORTMSG  DD  SYSOUT=*
   9 //SORTIN   DD  DSN=S600.$AVRSORT,DISP=SHR
  10 //SORTOUT  DD  DSN=S600.$AVRSORT,DISP=SHR
  11 //SORTWK01 DD  UNIT=SYSALLDA,
     //             SPACE=(CYL,(5))
  12 //SORTWK02 DD  UNIT=SYSALLDA,
     //             SPACE=(CYL,(5))
  13 //SORTWK03 DD  UNIT=SYSALLDA,
     //             SPACE=(CYL,(5))
```

Figure 20 Concatenated Browse: Example 1

Note the difference in LINE count on the top line of this display. Under concatenated browse, all lines of output for the job are counted as if they were consecutive. If **N** for NEXT is entered at the command line of this display, the screen would then appear as follows:

```
 DBRPRPT  (JCL) -------- LINE     22 OF    623 ------ COL   1  80 -----
 COMMAND ===>           SCROLL ===> CUR  FILE>
C112P.DBRPRPT.J0027881.D0000003.JESJCL,1996.124,14:16:47:95      <FILE
 1 //DBRPRPT  JOB (8000,UTXX,,99),'ONLINE DB LIST',CLASS=X,MSGCLASS=7,
   //         NOTIFY=&SYSUID,REGION=2048K
   IEFC653I SUBSTITUTION JCL - (8000,UTXX,,99),'ONLINE DB LIST',CLASS=X,M
 2 //STEP1   EXEC PGM=$AVRDBRP
 3 //STEPLIB DD  DSN=S600.R41A.LOADLIB,DISP=SHR
 4 //$AVRPRT  DD  SYSOUT=*
 5 //$AVRLOCK DD  DSN=C106.$AVRS.MULTI.LOCK,DISP=SHR
 6 //SYSPRINT DD  SYSOUT=*
 7 //SYSOUT   DD  SYSOUT=*
   //*
 8 //SORTMSG  DD  SYSOUT=*
 9 //SORTIN   DD  DSN=S600.$AVRSORT,DISP=SHR
10 //SORTOUT  DD  DSN=S600.$AVRSORT,DISP=SHR
11 //SORTWK01 DD  UNIT=SYSALLDA,
   //             SPACE=(CYL,(5))
12 //SORTWK02 DD  UNIT=SYSALLDA,
   //             SPACE=(CYL,(5))
13 //SORTWK03 DD  UNIT=SYSALLDA,
   //             SPACE=(CYL,(5))
   //*
14 //SYSABEND DD SYSOUT=*
15 //SYSIN    DD  *
   //*****************************************************************
FILE> C112P.DBRPRPT.J0027881.D0000004.JESYSMSG,1996.124,14:16:47:95  <FILE
ICH70001I C112P    LAST ACCESS AT 14:15:11 ON FRIDAY, MAY 3, 1996
IEF236I ALLOC. FOR DBRPRPT STEP1
IEF237I 0181 ALLOCATED TO STEPLIB
IEF237I 0166 ALLOCATED TO SYS00022
IEF237I JES2 ALLOCATED TO $AVRPRT
IEF237I 0181 ALLOCATED TO $AVRLOCK
IEF237I JES2 ALLOCATED TO SYSPRINT
IEF237I JES2 ALLOCATED TO SYSOUT
IEF237I JES2 ALLOCATED TO SORTMSG
IEF237I 0166 ALLOCATED TO SORTIN
IEF237I 0166 ALLOCATED TO SORTOUT
IGD100I 0195 ALLOCATED TO DDNAME SORTWK01 DATACLAS (        )
IGD100I 0197 ALLOCATED TO DDNAME SORTWK02 DATACLAS (        )
IGD100I 0195 ALLOCATED TO DDNAME SORTWK03 DATACLAS (        )
IEF237I JES2 ALLOCATED TO SYSABEND
IEF237I JES2 ALLOCATED TO SYSIN
```

Figure 21 Concatenated Browse: Example 2

Note that the display was positioned to the first line of the 'next' file, even though the next file was already displayed on the screen.

The **N** and **P** commands may be entered in conjunction with a numeric value to position the display a multiple number of files.

Example: N3 or P2

(FI)LE Selection Directory

The File Selection Directory for a JOB, STC or TSU is produced by:

- **!** entering an **S** (or your installation-defined command character) to the left of the desired entry on the Work Selection Directory,
- **!** by entering an **S** followed by the desired entry number at the Command line,
- **!** by entering the number of the desired entry at the Command line.

The following is a sample File Selection Directory:

```
 C101ALER (J04899) ---------- $AVRS FILE SELECTION --------------   ROW 1 OF 9
 COMMAND ===>                                                    SCROLL ===> CUR
    S=BRS,P=PRT,D=DEL,O=OUT,V=VPRT                          #LINES TOTAL       409
   SEL# ACT  FILE       C   FORM   FCB    UCS   DESTID   STEP   PROCSTEP LINES
_       1     JESLOG     2   STDX                  CSI   JES2                16
_       2     JCL        2   STDX                LOCAL   JES2                69
_       3     SYSMSG     2   STDX                LOCAL   JES2               145
_       4     SYSUT2     2   STDX                LOCAL   STEP001             20
_       5     SYSPRINT   2   STDX                LOCAL   STEP023              7
_       6     SYSUT2     2   STDX                LOCAL   STEP001            124
_       7     SYSPRINT   2   STDX                LOCAL   STEP001              4
_       8     SYSUT2     2   STDX                LOCAL   STEP001             20
_       9     SYSPRINT   2   STDX                LOCAL   STEP001              4
****************************** END OF DATA *********************************
```

Figure 22 File Selection Directory

The basic format of the File Selection Directory is similar to the Work Selection Directories. The top line of the panel lists the panel type (FILE SELECTION), along with the total number of files listed and the entry number of the file currently at the top of the

display. The left-most portion of the top line lists the JES NAME and JES NUMBER for the work selected. The second line contains the COMMAND area and SCROLL variable. The third lines lists valid functions and their command characters, along with the total number of output lines present in the files.

As with Work Selection Directories, entries are numbered for convenience. Rules for scrolling the entry portion of the panel are the same as those for Work Selection Directories. File Selection Directory entries are not sorted; neither the **SORT** nor **REFRESH** commands are valid. The **LOCATE** command may be used with a numeric value to position the display to a particular entry number (refer to LOCATE COMMAND in this manual).

File Selection Directory Fields

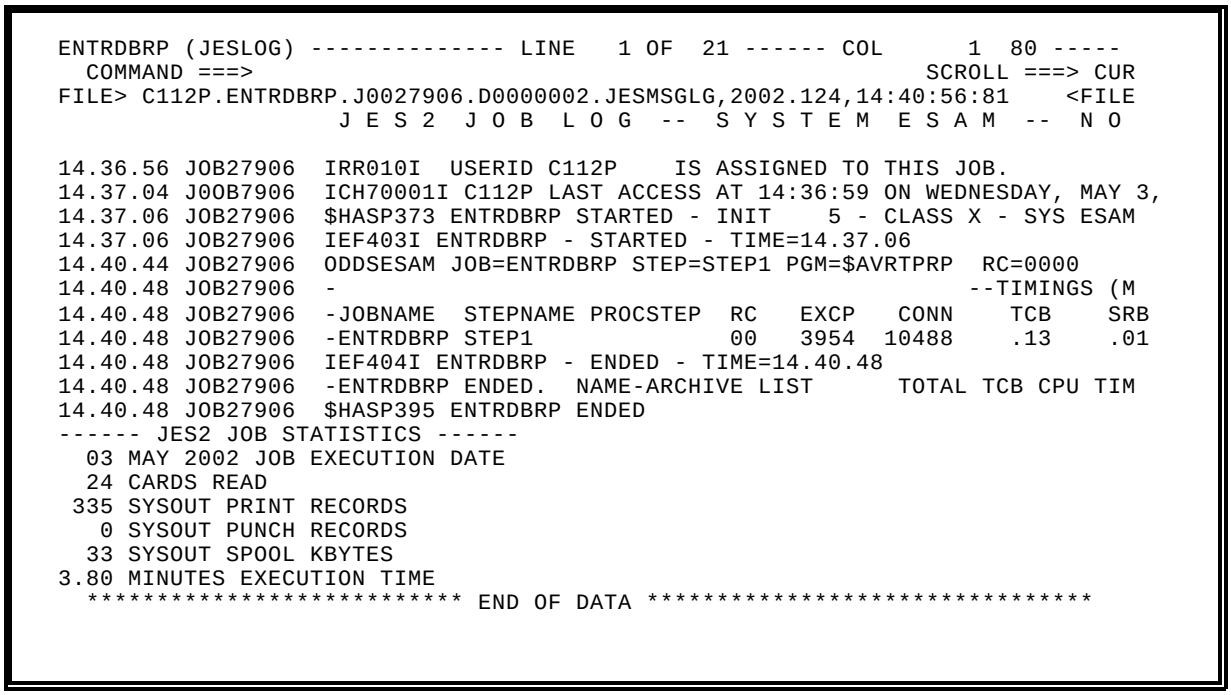
Each entry line on the directory lists information for a single file. Along with the name of the file, several fields of information are provided.

The following is a list of valid informational fields on the File Selection Directory. As with all $AVRS directories, the File Selection Directory heading may be customized by each installation, so all fields listed below will not necessarily be displayed at your installation.

Informational Field	**Explanation**
FILE	The FILENAME presented by JES during accumulation. The JES files are always JESLOG, JCL, and SYSMSG, regardless of your systems naming conventions. All dumps have the file name of DUMP when accumulated with release 5.0A#24 or higher.
C	The SYSOUT CLASS assigned to each Sysout dataset.
FORM	The SYSOUT FORM NUMBER assigned to each Sysout dataset.
FCB	The SYSOUT FORMS CONTROL BUFFER assigned to each Sysout dataset.
UCS	The SYSOUT UNIVERSAL CHARACTER SET assigned to each Sysout dataset.

DESTID	The SYSOUT DESTINATION assigned to each Sysout dataset.
STEP	The step name assigned to the PGM= statement in the JCL.
PROCSTEP	The step name that executes the proc.
LINES	The total number of LINES accumulated for each dataset.
PAGES	The total number of PAGES accumulated for each dataset.
COPYS	The number of copies for the file.
CHARS	The CHARACTER ARRANGEMENT TABLE if present.
COPY GROUPS	The copy groups value.
CPMOD	The COPYMOD ID assigned to each Sysout dataset.
MDTRC	0, 1, 2 or 3 TABLE REFERENCE CHARACTER for COPYMOD.
FLASH	The 3800 FLASH ID assigned to each Sysout dataset.
FLCNT	The Flash Count value.
MREC	The maximum record length of any record in the file.
DBBLKS	The number of data base blocks used by $AVRS to store the file.
WRITER	The output writer name assigned.

In addition to viewing file information from the File Selection Directory, individual files may be selected for Browse, either by entering an "S" to the left of the entry, or by entering the number of the desired entry at the Command line. The figure below is an example of a single file selected for browse from the File Selection Directory:



```
ENTRDBRP (JESLOG) ------------- LINE   1 OF  21 ------ COL      1  80 -----
  COMMAND ===>                                                SCROLL ===> CUR
FILE> C112P.ENTRDBRP.J0027906.D0000002.JESMSGLG,2002.124,14:40:56:81    <FILE
                    J E S 2  J O B  L O G  --  S Y S T E M  E S A M  --  N O

14.36.56 JOB27906  IRR010I  USERID C112P    IS ASSIGNED TO THIS JOB.
14.37.04 J0OB7906  ICH70001I C112P LAST ACCESS AT 14:36:59 ON WEDNESDAY, MAY 3,
14.37.06 JOB27906  $HASP373 ENTRDBRP STARTED - INIT    5 - CLASS X - SYS ESAM
14.37.06 JOB27906  IEF403I ENTRDBRP - STARTED - TIME=14.37.06
14.40.44 JOB27906  ODDSESAM JOB=ENTRDBRP STEP=STEP1 PGM=$AVRTPRP  RC=0000
14.40.48 JOB27906  -                                              --TIMINGS (M
14.40.48 JOB27906  -JOBNAME  STEPNAME PROCSTEP  RC   EXCP   CONN    TCB    SRB
14.40.48 JOB27906  -ENTRDBRP STEP1              00   3954  10488    .13    .01
14.40.48 JOB27906  IEF404I ENTRDBRP - ENDED - TIME=14.40.48
14.40.48 JOB27906  -ENTRDBRP ENDED.  NAME-ARCHIVE LIST      TOTAL TCB CPU TIM
14.40.48 JOB27906  $HASP395 ENTRDBRP ENDED
------ JES2 JOB STATISTICS ------
  03 MAY 2002 JOB EXECUTION DATE
  24 CARDS READ
 335 SYSOUT PRINT RECORDS
   0 SYSOUT PUNCH RECORDS
  33 SYSOUT SPOOL KBYTES
3.80 MINUTES EXECUTION TIME
  ************************** END OF DATA ******************************
```

Figure 23 Browsing Single Files: Example 1

From the Browse display panel for an individual file, the user may select other files from the same Job for Browse without returning to the File Selection Directory by using the **N** (for Next) and **P** (for Previous) commands. Entering **P** by itself results in selection for browse of the file entry previous to the current selected file. Entering **N** results in selection of the next file entry. Both of the commands may be entered in conjunction with a number to reposition file selection a multiple number of entries.

For example, assume you select an STC with 10 output files for viewing from the File Selection Directory. You may browse file number 8 (as numbered on the File Selection Directory) by entering an **S** next to entry eight. You then wish to browse file number 5. You can enter:

P3 or **P 3**

at the command line. The Browse would then be repositioned to present file 5. **P** or **N** may be entered in conjunction with a large number to force browse of either the first or last file listed. If **N** were entered at the command line of the previous example, the next file, in this case the JCL file, would be selected and the display screen would appear as follows:

```
ENTRDBRP (JCL) ------ LINE      1 OF     23 ------ COL   1  80 -----
COMMAND ===>                                          SCROLL ===> CUR
FILE> C112P.ENTRDBRP.J0027906.D0000003.JESJCL,2000.124,14:40:56:82 <FILE
 1 //ENTRDBRP JOB (8000,UTXX,,99),'ARCHIVE LIST',CLASS=X,MSGCLASS=7,
   //         NOTIFY=&SYSUID,REGION=2048K
   IEFC653I SUBSTITUTION JCL - (8000,UTXX,,99),'ARCHIVE LIST',CLASS=X,MSG
 2 //STEP1   EXEC PGM=$AVRTPRP
 3 //STEPLIB DD  DSN=S600.R41A.LOADLIB,DISP=SHR
 4 //$AVRPRT  DD  SYSOUT=*
 5 //$AVRTAPE DD  DSN=C112.$AVRS.UNLOAD,DISP=SHR
 6 //SYSPRINT DD  SYSOUT=*
 7 //SYSOUT   DD  SYSOUT=*
   //*
 8 //SORTMSG  DD  SYSOUT=*
 9 //SORTIN   DD  DSN=S600.$AVRSORT,DISP=SHR
10 //SORTOUT  DD  DSN=S600.$AVRSORT,DISP=SHR
11 //SORTWK01 DD  UNIT=SYSALLDA,
   //             SPACE=(CYL,(5))
12 //SORTWK02 DD  UNIT=SYSALLDA,
   //             SPACE=(CYL,(5))
13 //SORTWK03 DD  UNIT=SYSALLDA,
   //             SPACE=(CYL,(5))
   //*
14 //SYSABEND DD SYSOUT=*
15 //SYSIN    DD  *
************************** END OF DATA ******************************
```

Figure 24 Browsing Single Files: Example 2

Individual files from a JOB, STC or TSU may be selected for printing, deleting, or output. Refer to the sections, "PRINTING, VTAM PRINTING, OUTPUTTING & DELETING ACCUMULATED WORK", for more information on these process functions.

As with the Work Selection Directories, multiple files may be processed by entering the desired process function designator(s) to the left of multiple file entries. As one function is completed, the next file is automatically presented. Process function types may be mixed when processing files in this manner. If several files have been concurrently selected for processing in this manner, and it is then desired to terminate the selections at any point, the

BREAK or **BRK** command may be entered from any panel to cause immediate termination of function selection and return to the File Selection Directory.

Displaying Syslog (SL)

As you will recall from the information presented in an earlier chapter of this section, the only selection criteria required for selecting SYSLOG from the $AVRS data base is a **TYPE** field value of **SL**.

Once **SL** and any additional selection criteria have been entered, the user is presented with a SYSLOG Directory of accumulated SYSLOG segments satisfying the specified criteria. The following is a sample SYSLOG Directory. As with all $AVRS directories, the order and presence of individual fields may be altered by installation customization.

```
-   WED 03 MAY 2000.124 ----- $AVRS SYSLOG SELECTION -------- ROW 1 OF 29
 COMMAND ===>                                                 SCROLL ===> CUR
  B=BRS,S=FIL,P=PRT,D=DEL,O=ODS,V=VPRT               #LINES TOTAL   19061
    SEL# ACT  SLID JOB #    C ACK     START DATE          TIME       LINES
 -     1      ADVA 0027429  N       SUN 23 APR 2000.114 07:56.40      1873
 -     2      ADVA 0027429  N       SUN 23 APR 2000.114 09:01.02       179
 -     3      ADVA 0027429  N       SUN 23 APR 2000.114 10:01.06      1454
 -     4      ADVA 0027429  N       SUN 23 APR 2000.114 11:02.24       761
 -     5      ADVA 0027429  N       SUN 23 APR 2000.114 12:01.09       580
 -     6      ADVA 0027429  N       SUN 23 APR 2000.114 13:09.49       454
 -     7      ADVA 0027429  N       SUN 23 APR 2000.114 14:02.08       437
 -     8      ADVA 0027429  N       SUN 23 APR 2000.114 15:01.04       690
 -     9      ADVA 0027429  N       SUN 23 APR 2000.114 16:10.51       537
 -    10      ADVA 0027429  N       SUN 23 APR 2000.114 17:02.54       315
 -    11      ADVA 0027429  N       SUN 23 APR 2000.114 18:01.16      2597
 -    12      ADVA 0027429  N       SUN 23 APR 2000.114 19:13.17       651
 -    13      ADVA 0027429  N       SUN 23 APR 2000.114 20:05.01       284
 -    14      ADVA 0027429  N       SUN 23 APR 2000.114 21:01.30       561
 -    15      ADVA 0027438  N       WED 26 APR 2000.117 07:48.07      1896
 -    16      ADVA 0027438  N       WED 26 APR 2000.117 09:06.08       639
 -    17      ADVA 0027438  N       WED 26 APR 2000.117 10:01.01       482
 -    18      ADVA 0027438  N       WED 26 APR 2000.117 11:01.00       363
 -    19      ADVA 0027438  N       WED 26 APR 2000.117 12:01.09       543
 -    20      ADVA 0027438  N       WED 26 APR 2000.117 13:05.24        93
 -    21      ADVA 0027438  N       WED 26 APR 2000.117 14:13.04       144
 -    22      ADVA 0027438  N       WED 26 APR 2000.117 15:04.20        83
 -    23      ADVA 0027438  N       WED 26 APR 2000.117 16:15.13        60
 -    24      ADVA 0027438  N       WED 26 APR 2000.117 17:03.11       336
 -    25      ADVA 0027438  N       WED 26 APR 2000.117 18:01.03        60
 -    26      ADVA 0027438  N       WED 26 APR 2000.117 19:27.53       525
 -    27      ADVA 0027438  N       WED 26 APR 2000.117 20:04.08      2351
 -    28      ADVA 0027438  N       WED 26 APR 2000.117 21:46.12        14
 -    29      ADVA 0027438  N       WED 26 APR 2000.117 22:29.31        99
 ******************************* END OF DATA*****************************
```

Figure 25 Syslog Selection Panel (Portion)

All selection functions that are valid for JOBs STCs and TSUs, with the exception of the INFORMATION PANEL, are valid for SYSLOG. Individual segments may be browsed, printed, output to a dataset, or deleted (if the user has authority).

Because a SYSLOG segment contains only a single file which is used internally by $AVRS in accumulating the work, selection of a SYSLOG segment for presentation of the FILE SELECTION PANEL results in $AVRS immediately invoking a BROWSE of the file, rather than the normal presentation of the File Selection Directory. The File Selection Directory is presented upon return from the Browse of the data. This selection path is provided in this manner under the assumption that in most cases the user will not need to view the characteristics of the SYSLOG file. The file presentation panel can be avoided altogether by selecting the segment with the BROWSE selection character. The following is an example of a SYSLOG segment as displayed by the $AVRS BROWSE FEATURE. This facility allows for the viewing and manipulation of the data presented.

```
SL 7777 -- 24 APR 1996.115 ----- LINE  1 OF  602 ------ COL   1  80 -----
COMMAND ===>                                        SCROLL ===> CUR
 FILE>   C112P.SYSLOG.J0027656.D0000137.?,1996.123,16:07:17:91       <FILE
 N 0200000 ESAM 96115 14:08:14.51 JOB26416 00000081  $HASP100 C118ZAPR ON INT
 N 0000000 ESAM 96115 14:08:15.90 JOB26416 00000290  IRR010I  USERID C118P
 N C200000 ESAM 96115 14:08:16.75 JOB26416 00000090  $HASP520 C118ZAPR ON L1.
 N C200000 ESAM 96115 14:08:19.36 JES2     00000090  $HASP524 L1.JT1   INACTI
 N 0200000 ESAM 96115 14:08:19.71 JOB26416 00000090  $HASP250 C118ZAPR IS PUR
 NC0000000 ESAM 96115 14:08:23.65 INSTREAM 00000290  LOGON
 N 0200000 ESAM 96115 14:08:33.42 TSU26417 00000081  $HASP100 SEA7     ON TSO
 N 4000000 ESAM 96115 14:08:37.68 TSU26417 00000090  $HASP373 SEA7     STARTE
 NI0000000 ESAM 96115 14:08:38.07 JES2     00000290  SE '                1Q
 SI                                                  VSE , RC=0016, TIME=14:0
 N 0100000 ESAM 96115 14:08:38.32 JOB00974 00000090  $HASP540 C118ZAPR ON L1.
 S                                                   RECORDS
 N 0000000 ESAM 96115 14:08:38.59 TSU26417 00000090  IEF125I SEA7 - LOGGED ON
 N 0000000 ESAM 96115 14:08:39.75 CATALOG  00000290  IEF196I IEF237I 0166 ALL
 N 0000000 ESAM 96115 14:08:39.97 CATALOG  00000290  IEF196I IGD104I CATALOG.
 N 0000000 ESAM 96115 14:08:40.00 CATALOG  00000290  IEF196I DDNAME=SYS00112
 NI0000000 ESAM 96115 14:08:41.12 JES2     00000290  SE '14.08.40 JOB00974 $H
 SI                                                  LOGON,USER=(C118P)
 NI0000000 ESAM 96115 14:08:41.89 JES2     00000290  SE '                  1R
 SI                                                   MVS FOR MVS O-TR1',LOGO
 N 0200000 ESAM 96115 14:08:47.35 JOB26418 00000081  $HASP100 C118ZAPR ON INT
 N 0000000 ESAM 96115 14:08:47.46 JOB26418 00000290  IRR010I  USERID C118P
 N C200000 ESAM 96115 14:08:47.67 JOB26418 00000090  $HASP520 C118ZAPR ON L1.
 N C200000 ESAM 96115 14:08:49.59 JES2     00000090  $HASP524 L1.JT1   INACTI
 N 0200000 ESAM 96115 14:08:49.77 JOB26418 00000090  $HASP250 C118ZAPR IS PUR
 NI0000000 ESAM 96115 14:08:52.66 JES2     00000290  SE '                  1Q
 SI                                                  VSE , RC=0000, TIME=14:0
 N 0100000 ESAM 96115 14:08:52.96 JOB00975 00000090  $HASP540 C118ZAPR ON L1.
 S                                                   RECORDS
 NI0000000 ESAM 96115 14:08:53.21 JES2     00000290  SE '14.08.53 JOB00975 $H
 SI                                                  LOGON,USER=(C118P)
 NI0000000 ESAM 96115 14:08:54.21 JES2     00000290  SE '                  1R
 SI                                                  MVS FOR MVS O-TR1',LOGO
 NC0000000 ESAM 96115 14:09:02.24 C118P    00000290  $CJ974,P
 NR0000000 ESAM 96115 14:09:02.56 JOB00974 00000090  $HASP608 C118ZAPR AWAITI
 N 0200000 ESAM 96115 14:09:03.71 JOB00974 00000090  $HASP250 C118ZAPR IS PUR
 NC0000000 ESAM 96115 14:09:16.93 C118P    00000290  $CJ973,P
 NR0000000 ESAM 96115 14:09:17.02 JOB00973 00000090  $HASP608 C118VCST AWAITI
 N 0200000 ESAM 96115 14:09:17.08 JOB00973 00000090  $HASP250 C118VCST IS PUR
 N 0200000 ESAM 96115 14:09:47.15 JOB26419 00000081  $HASP100 C118VCST ON INT
```

Figure 26 Syslog Segment Under $AVRS Browse

Displaying Status Reports (ST)

Whenever $AVRS archives work from the active data base to an archive dataset, a status report summarizing the work archived is produced. These reports may be viewed online by specifying a **TYPE** field value of **ST** on the Primary Control Panel. For status report selection, specification of a DATE or TIME range is applied to the START and END dates of the work archived (**recap period**), rather than the DATE and TIME the archive job was run (**CREATION/RUN DATE)**. If any of the work in the archive run has START and/or END

DATEs and TIMEs meeting the criteria specified on the Primary Control Panel, the Status Report is selected for display.

A Status Report Directory is presented after selection criteria are entered. The figure below is an example of a Status Report Directory:

```
- WED 19 APR 2000.110 --------      $AVRS ONLINE STATUS RECAP         --------------ROW 225 OF 346
COMMAND ===>                                                                    SCROLL ===> CUR
 B=BRS,S=FIL,P=PRT,D=DEL,O=ODS,V=VPRT                                    #LINES TOTAL      2678
 SEL  # ACT LINES     RECAP PERIOD  START       RECAP PERIOD END      CREATION/RUN DATE    TIME
_   225        15 MON 04 MAY 1998.124 13:12 MON 04 MAY 1998.124 16:14 MON 04 MAY 1998.124  16:15.20
_   226        15 MON 04 MAY 1998.124 13:21 MON 04 MAY 1998.124 16:07 MON 04 MAY 1998.124  16:45.06
_   227        16 MON 04 MAY 1998.124 13:22 MON 04 MAY 1998.124 16:38 MON 04 MAY 1998.124  16:44.10
_   228        16 MON 11 MAY 1998.131 08:57 MON 11 MAY 1998.131 09:32 MON 11 MAY 1998.131  10:21.36
_   229        18 WED 20 MAY 1998.140 15:13 WED 20 MAY 1998.140 15:56 WED 20 MAY 1998.140  15:58.49
_   230        53 FRI 19 JUN 1998.170 09:47 MON 13 JUL 1998.194 09:30 MON 13 JUL 1998.194  09:34.58
_   231        31 TUE 22 SEP 1998.265 07:56 TUE 27 OCT 1998.300 14:44 TUE 27 OCT 1998.300  15:47.59
_   232        14 FRI 25 SEP 1998.268 09:25 FRI 25 SEP 1998.268 09:25 FRI 25 SEP 1998.268  09:30.01
_   233        29 FRI 25 SEP 1998.268 15:05 MON 12 OCT 1998.285 16:09 MON 12 OCT 1998.285  16:24.57
_   234        65 FRI 25 SEP 1998.268 15:05 TUE 12 DEC 2000.347 08:15 WED 02 FEB 2000.033  11:53.00
_   235       141 FRI 25 SEP 1998.268 15:05 THU 06 APR 2000.097 10:44 THU 06 APR 2000.097  11:00.49
_   236        16 MON 05 OCT 1998.278 10:00 MON 12 OCT 1998.285 16:28 MON 12 OCT 1998.285  17:19.51
_   237        15 TUE 27 OCT 1998.300 11:19 TUE 27 OCT 1998.300 12:08 TUE 27 OCT 1998.300  15:50.37
_   238        14 FRI 17 DEC 1999.351 16:17 FRI 17 DEC 1999.351 16:22 FRI 17 DEC 1999.351  16:49.11
_   239        57 FRI 17 DEC 1999.351 16:39 FRI 14 JAN 2000.014 16:27 TUE 18 JAN 2000.018  13:11.22
_   240      1391 WED 19 JAN 2000.019 16:56 TUE 12 DEC 2000.347 08:15 WED 09 FEB 2000.040  19:48.29
_   241        18 WED 09 FEB 2000.040 09:38 WED 09 FEB 2000.040 22:57 MON 14 FEB 2000.045  15:29.02
_   242        27 THU 10 FEB 2000.041 12:41 TUE 22 FEB 2000.053 09:58 WED 23 FEB 2000.054  14:32.54
_   243        14 MON 14 FEB 2000.045 15:50 MON 14 FEB 2000.045 15:50 MON 14 FEB 2000.045  15:58.38
_   244        30 TUE 22 FEB 2000.053 11:25 MON 28 FEB 2000.059 15:00 MON 28 FEB 2000.059  15:01.43
_   245        15 MON 28 FEB 2000.059 10:49 MON 28 FEB 2000.059 10:50 MON 28 FEB 2000.059  15:01.43
_   246        56 THU 06 APR 2000.097 10:16 WED 19 APR 2000.110 10:57 WED 19 APR 2000.110  11:00.52
```

Figure 27 Status Report Directory

Individual status reports may be selected for display as with any other work on the data base. Entries may be printed or output to a dataset. An entire status report may be deleted (if the user has installation defined authority). However, individual lines of a status report *may not* be deleted. The Information Panel is not available for Status Reports. As with SYSLOG, each STATUS REPORT contains only a single file, which may be displayed by entering the File Selection Directory selection character and then returning from the Browse which is immediately invoked.

Each piece of work archived is summarized within the status reports. The following is an example of a Status Report:

```
STATUS  -- 29 APR 2000.120 ----- LINE     13 OF     47 ------ COL   1  80 -----
 COMMAND ===>                                                 SCROLL ===> CUR
C203B05A J0027034  04/29/2000.120 12:41 04/29/2000.120 12:41 RC=0000          53
C203B05A J0027050  04/29/2000.120 12:42 04/29/2000.120 12:42 RC=0000          53
C203B05F J0027039  04/29/2000.120 12:41 04/29/2000.120 12:41 JCL ERR    R     54
C203B05F J00I27055 04/29/2000.120 12:43 04/29/2000.120 12:43 JCL ERR    R     54
C203B05J J00I27043 04/29/2000.120 12:42 04/29/2000.120 12:42 RC=0000          53
C203B05J J00I27059 04/29/2000.120 12:43 04/29/2000.120 12:43 RC=0000          53
C203C000 J00I27041 04/29/2000.120 12:41 04/29/2000.120 12:41 RC=0000          40
C203C000 J00I27057 04/29/2000.120 12:43 04/29/2000.120 12:43 RC=0000          40
C203INCP J00I27042 04/29/2000.120 12:42 04/29/2000.120 12:42 RC=0000         282
C203INCP J00I27058 04/29/2000.120 12:43 04/29/2000.120 12:43 RC=0000         282
C203URSP J00I27033 04/29/2000.120 12:40 04/29/2000.120 12:40 RC=0016          53
C203URSP J00I27049 04/29/2000.120 12:42 04/29/2000.120 12:42 RC=0016          53
C203X05F J00I27048 04/29/2000.120 12:42 04/29/2000.120 12:42 RC=0000          94
C203X05F J00I27064 04/29/2000.120 12:43 04/29/2000.120 12:43 RC=0000          94
C205PROD J00I27044 04/29/2000.120 12:42 04/29/2000.120 12:42 RC=0000          68
C205PROD J00I27060 04/29/2000.120 12:43 04/29/2000.120 12:43 RC=0000          68
C205RMRR J00I27071 04/29/2000.120 13:18 04/29/2000.120 13:18 PARTIAL           6
C205SIO  J00I27047 04/29/2000.120 14:33 04/29/2000.120 14:33 RC=3660          57
C205SIO  J00I27063 04/29/2000.120 14:33 04/29/2000.120 14:33 RC=3660          57
  NAME    NUMBER   START DATE/TIME        END DATE/TIME      ERROR TYPE R  LINES
JCLPS001 J00I27040 04/29/2000.120 12:40 04/29/2000.120 12:40 JCL ERR    R     16
JCLPS001 J00I27056 04/29/2000.120 12:41 04/29/2000.120 12:41 JCL ERR    R     16
S600B05C J00I27036 04/29/2000.120 12:40 04/29/2000.120 12:41 JCL ERR    R     28
S600B05C J00I27052 04/29/2000.120 12:41 04/29/2000.120 12:42 JCL ERR    R     28
BACKUP DSN=C203.R51A.BACKUP.G0006V00
VOLSER(S)=877158
FILESEQ=6
UNIT=CART
DUPLEX DSN=NULLFILE
DUPVOL(S)=
DUPFILE=
DUPUNIT=
DDNAME=$AVRS99
HISTORY RETENTION=0495
DATE/TIME=2000120,170527 TYPE=NORMAL
 ****************************** END OF DATA ******************************
```

Figure 28 Status Report

Note that at the end of each Status Report is a summary of information for the Primary and Duplex datasets, including the archive dataset name(s) and volume(s) used ($AVRS records up to 5 volumes), and type of backup that generated the Status Report. The types are:

Normal	task scheduled backup
Emergency	RRDS or KSDS full

Batch	batch backup
Operator	modify command from console.
Unknown	unable to determine initiation
Forward Merge	from the $AVRFMRG program
Merge Forward	from the $AVRMRGF program.

Displaying Archive History (AH)

A breakout of archive information by JES name, rather than archive run, may be produced by specifying a **TYPE** field of **AH** on the Primary Control Panel. For a more specific selection the **NAME**, **DATE**, **TIME**, **SELECT RULE, MSGCLASS, RETURN CODE**, **RANGE=Y** and **PGMER NAME** fields may also be used.

After selection criteria are entered, a directory of archived work satisfying the criteria entered is presented. The information in the Directory Header is a breakout of the information provided in the Status Report containing the work. A detailed information summary panel can be obtained for each entry by selecting it from the Archive History Selection Directory. The following is a sample ARCHIVE HISTORY Status Panel:

```
------------------ $AVRS ARCHIVE HISTORY STATUS --------- RELEASE 5.1B --
 COMMAND INPUT ===>
                  HISTORICAL DATA FOR GEETAPEP JOB NUMBER J0027908

   EXECUTION START DATE     WED 03 MAY 2000.124  START TIME      14:37
   EXECUTION END DATE       WED 03 MAY 2000.124  END TIME        14:38
   ENDED EXECUTION WITH     RC=0000              LINES ARCHIVED  368
   ELAPSED TIME             2                    CPU TIME        0.00.11
   ACCUMULATION CLASS       7

                ARCHIVE INFORMATION FOR THIS WORK IS:
BACKUP  DSNAME       DSI4.R50A.$AVRS99.G0001V00
        VOLUME(S)    876130
        FILESEQ      1
DUPLEX  DSNAME       DSI4.R50A.$AVRS99.DUPLEX.G0001V00
        VOLUME(S)    SMS001
        FILESEQ      DASD
        DATE         SAT 06 MAY 2000.127  (DATE ARCHIVED)
        TIME         13:25:04             (TIME ARCHIVED)
    PRESS END PFKEY TO RETURN OR TO DISPLAY THE NEXT SELECTION
```

Figure 29 Archive History Status Panel

RESTORING from ARCHIVE HISTORY

If the user has the proper authority, work may be Restored from the archive dataset to the online data base by entering an **R** next to the desired entry on the selection panel, or entering **R** followed by the directory entry number on the command line of the selection panel. Entering an **X** instead of an **R** will cause the duplex archive/backup dsname to be used in the restore. $AVRS uses the information retained about the work to check the catalog and build a Restore request for the entry. One or more pieces of work may be Restored before submission of the Restore Job.

If the primary archive file is not catalogued, the catalogued duplex file is automatically used. A restore request from a duplex will display as an **RX** in the Action field on the **AH** directory. If there is no duplex file or the primary duplex files are no longer catalogued, the following message is issued:

$AVR958W - *jobname, job number,* DSN = *dataset name* (1st 30 characters) - NOT CATLGD

NOTE: If multiple uncataloged restores are requested before hitting ENTER, only one **$AVR958W** message will be displayed on the selection panel.

The user can then:

- re-catalog the dataset and submit the restore at a later time.
- use the MVS tape management system to determine the volume number(s) and unit for the dataset and supply them on the submission panel.

When selection of work for Restore is complete, the user may submit the Restore Job by entering **SR**, **S R** or **SUBREST** at the command line of any Work Selection Directory or the Primary Control Panel. $AVRS then presents a submission panel containing the JCL JOBCARD produced for the Job. This JCL JOBCARD may be altered by the user and then submitted for execution by pressing enter, or submission may be aborted by pressing the "END" PF KEY from the submission panel. If the user does not explicitly request submission of the Restore Job by entering one of the commands listed above, the Submission Panel is automatically presented when $AVRS is exited. A sample ARCHIVE RESTORE job submission panel follows:

```
----- $AVRS ARCHIVE HISTORY CONFIRMATION AND SUBMISSION ------ RELEASE 5.1B -
 COMMAND INPUT ===>
               Modify JCL as required for Submit

      Sysout Class for Report ===> *          $AVRS Restore Report Class

      $AVRS Procedure         ===> $AVRREST   Procedure Name for Restore

   //C117REST JOB (YOUR ACCOUNTING),
   //             '$AVSELR RESTORE      '
   //              CLASS=X,MSGLEVEL=(1,),
   //              MSGCLASS=T,NOTIFY=C117P,TYPRUN=HOLD
   //*
   //*
   //*
   //*
   //*


          Press ENTER to Submit Restore Job
          or  END PFkey to ABORT Restore(s)
```

Figure 30 Restore Job Confirmation and Submit Panel

After submitting the restore, any uncataloged backup data sets encountered during the conformation process will be identified by message $AVR958W. The following will be displayed for each uncataloged data set being processed.

```
--- $AVRS ARCHIVE HISTORY RESTORE DATASET NOT CATALOGUED--- RELEASE 5.1B -
COMMAND INPUT ===>
 $AVR958W  - KS50ASL, 1417, DSN=$AVR.R50A.BACKUP.G0169V00 NOT CATLGD

    MEDIA TYPE (TAPE OR DASD)?     ===> TAPE

    UNIT NAME?                     ===> CART
    Maximum 8 characters

    VOLSER?                        ===> 877479,877480
    Maximum 5 VOLSER's

    FILE SEQUENCE #                ===> 3
    Build JES2 SETUP statement     ===> Y
    (Yes or No)

         Press ENTER to Submit Restore Job
         or  END PFkey to ABORT Restore(s)
```

Figure 31 Restore Dataset Not Catalogued

MEDIA TYPE	The physical unit where the backup dataset resides, TAPE or DASD. Required.
UNIT NAME	The particular type of I/O device defined on your system. The 8 character name may be a: Device Address (e.g. 590), Device Type (e.g.Cart), or a Group Name (e.g. Temp). Required.
VOLSER	The specific volumes where the backup dataset resides. Each Volume Serial Number is 1 to 6 alphanumerics. For Example: CART01 or 877890,877891. The maximum that can be specified is 5. Required.
FILE SEQUENCE #	The dataset position on the tape volume. The file sequence number is a numeric number. 1- 255.
Build JES2 SETUP	A VALUE OF "Y"ES causes the submit routine to build JES2/* setup control cards for the job being submitted. If the END PFKey is entered, the currently identified dataset will not be processed. The confirmation process will continue.

There are no deletes permitted from the Archive History (AH) directory.

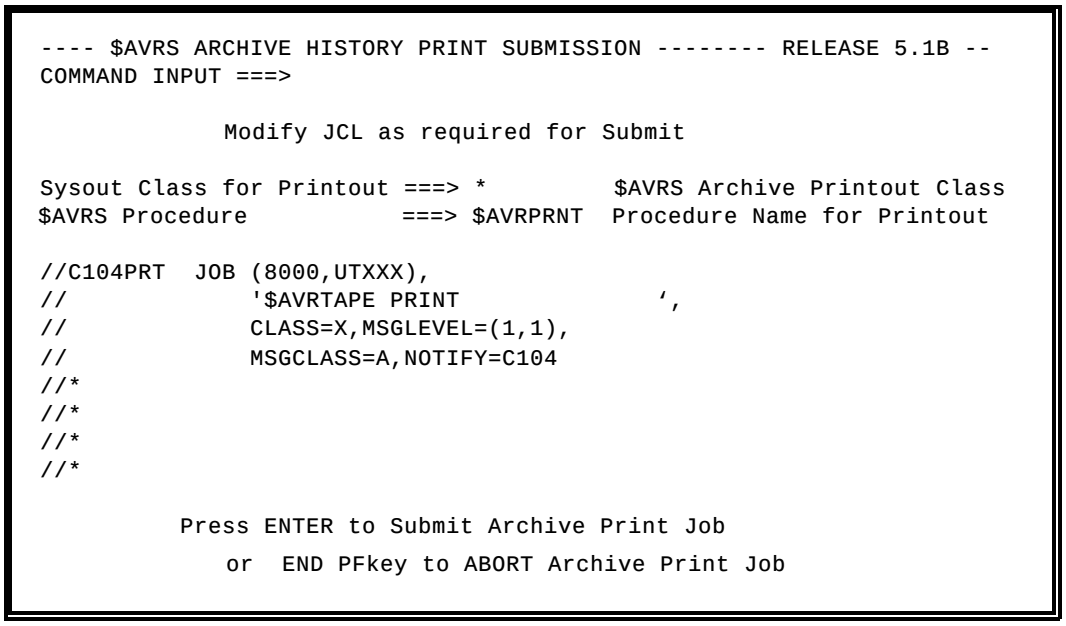
If you have the necessary authority, history deletions may be done from the Status History (**ST**) directory (at the Archive/backup report level) or using the programs $AVRFMRG or $AVRMRGF (to delete individual job history).

Output and Information functions are also not available from Archive History (**AH**).

PRINTING FROM ARCHIVE HISTORY

If the user has the proper authority, work may be Printed from the archive data set by entering a **P** next to the desired entry on the selection panel, or entering **P** followed by the directory entry number at the command line of the selection panel. $AVRS uses the information retained about the work to check the catalog and build a Print request for the entry using the $AVRTAPE program.

When selection of work for Print is complete, the user may submit the Print Job ($AVRTAPE) by entering **SP, S P** or **SUBPRINT** at the command line of any Work Selection directory or the Primary Control Panel. $AVRS then presents a submission panel containing the JCL JOBCARD produced for the Job. This JCL JOBCARD may be altered by the user and then submitted for execution by pressing enter, or submission my be aborted by pressing the END PF KEY from the submission panel. If the user does not explicitly request submission of the Print Job by entering one of the commands listed above, the Submission Panel is automatically presented when $AVRS is exited. A Sample ARCHIVE PRINT job submission panel follows:



```
---- $AVRS ARCHIVE HISTORY PRINT SUBMISSION -------- RELEASE 5.1B --
COMMAND INPUT ===>

             Modify JCL as required for Submit

Sysout Class for Printout ===> *         $AVRS Archive Printout Class
$AVRS Procedure           ===> $AVRPRNT  Procedure Name for Printout

//C104PRT  JOB (8000,UTXXX),
//              '$AVRTAPE PRINT              ',
//              CLASS=X,MSGLEVEL=(1,1),
//              MSGCLASS=A,NOTIFY=C104
//*
//*
//*
//*

          Press ENTER to Submit Archive Print Job
             or  END PFkey to ABORT Archive Print Job
```

Figure 32 Archive Print Job Submission Panel

After submitting the print, any uncataloged backup data sets encountered during the conformation process will be identified by message $AVR958W. The following will be displayed for each uncataloged data set being processed.

```
--- $AVRS ARCHIVE HISTORY PRINT DATASET NOT CATALOGUED--- RELEASE 5.1B -
COMMAND INPUT ===>
 $AVR958W  - KS50ASL, 1417, DSN=$AVR.R50A.BACKUP.G0169V00 NOT CATLGD

    MEDIA TYPE (TAPE OR DASD)?     ===> TAPE

    UNIT NAME?                     ===> CART
    Maximum 8 characters

    VOLSER?                        ===> 877479,877480
    Maximum 5 VOLSER's

    FILE SEQUENCE #                ===> 3
    Build JES2 SETUP statement     ===> Y
    (Yes or No)

         Press ENTER to Submit Print Job
         or  END PFkey to ABORT Restore(s)
```

Figure 33 Print Dataset Not Catalogued

MEDIA TYPE	The physical unit where the backup dataset resides, TAPE or DASD. Required.
UNIT NAME	The particular type of I/O device defined on your system. The 8 character name may be a: Device Address (e.g. 590), Device Type (e.g.Cart), or a Group Name (e.g. Temp). Required.
VOLSER	The specific volumes where the backup dataset resides. Each Volume Serial Number is 1 to 6 alphanumerics. For Example: CART01 or 877890,877891. The maximum that can be specified is 5. Required.
FILE SEQUENCE #	The dataset position on the tape volume. The file sequence number is a numeric number. 1- 255.
Build JES2 SETUP	A VALUE OF "Y"ES causes the submit routine to build JES2/* setup control cards for the job being submitted.

If the END PFKey is entered, the currently identified dataset will not be processed. The confirmation process will continue.

There are no deletes permitted from the Archive History (AH) directory.

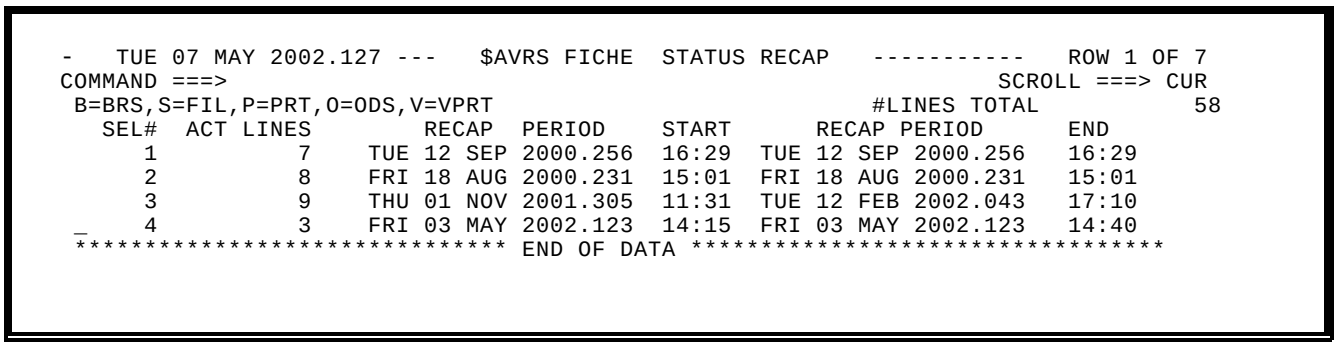
If you have the necessary authority, history deletions may be done from the Status History (**ST**) directory (at the Archive/backup report level) or using the programs $AVRFMRG or $AVRMRGF (to delete individual job history).

Output and Information functions are also not available from Archive History (**AH**).

Displaying Fiche Index Reports (FI)

An output summary from each run of the $AVRFICH program is saved in a FICHE INDEX REPORT and these reports are accumulated on the $AVRS data base. These reports may be viewed by entering a **TYPE** field of **FI** on the Primary Control Panel.

A directory of FICHE INDEX REPORTS is produced listing those reports which satisfy the selection criteria entered. Individual reports may be browsed similarly to any other work in the data base. FICHE INDEX REPORTS may be printed or output to a dataset. An entire FICHE INDEX REPORT (FI) may be deleted if the user has installation defined authority. However, individual lines of the FICHE INDEX REPORT *may not* be deleted The Information Panel is not available for FICHE INDEX. The following is an example of a Fiche Index Directory:



```
-   TUE 07 MAY 2002.127 ---   $AVRS FICHE  STATUS RECAP   -----------   ROW 1 OF 7
COMMAND ===>                                                  SCROLL ===> CUR
 B=BRS,S=FIL,P=PRT,O=ODS,V=VPRT                          #LINES TOTAL          58
   SEL#  ACT LINES        RECAP  PERIOD    START      RECAP PERIOD      END
      1          7    TUE 12 SEP 2000.256  16:29  TUE 12 SEP 2000.256   16:29
      2          8    FRI 18 AUG 2000.231  15:01  FRI 18 AUG 2000.231   15:01
      3          9    THU 01 NOV 2001.305  11:31  TUE 12 FEB 2002.043   17:10
 _    4          3    FRI 03 MAY 2002.123  14:15  FRI 03 MAY 2002.123   14:40
 ****************************** END OF DATA *********************************
```

Figure 34 Fiche Index Directory

The Reports list the work processed by the fiche creation run. The following is an example of a Fiche Index Report:

```
 STATUS  -- 03 MAY 2002.127 ----- LINE      1 OF     18 ------ COL   1  80 -----
 COMMAND ===>                                                 SCROLL ===> CUR
FILE>       C112P.FICHCRET.J0028046.D0000106.?,2002.127,11:23:54:46        <FILE
  NAME     NUMBER     START DATE/TIME      END DATE/TIME    ERROR TYPE FICHE R/C
ADBPRT    J0027878 03 MAY2002.123 14:15 03 MAY2002.123 14:16 RC=0000       1  A1
BIGDB     J0027879 03 MAY2002.123 14:15 03 MAY2002.123 14:17 RC=0000       1  J1
FIGRDBRP  J0027883 03 MAY2002.123 14:15 03 MAY2002.123 14:16 RC=0000       1  G2
ADBPRT    J0027902 03 MAY2002.123 14:36 03 MAY2002.123 14:39 RC=0000       1  D3
ENTRDBRP  J0027906 03 MAY2002.123 14:37 03 MAY2002.123 14:40 RC=0000       1  A4
FIGRDBRP  J0027907 03 MAY2002.123 14:37 03 MAY2002.123 14:37 RC=0000       1  K4
GEETAPEP  J0027908 03 MAY2002.123 14:37 03 MAY2002.123 14:38 RC=0000       1  H5
PRTTAPE   J0027912 03 MAY2002.123 14:38 03 MAY2002.123 14:38 RC=0000       1  G6
PRTTAPE   J0027913 03 MAY2002.123 14:39 03 MAY2002.123 14:39 RC=0000       1  A7
RPTTAPE   J0027915 03 MAY2002.123 14:39 03 MAY2002.123 14:41 RC=0000       1  G7
SEETAPE1  J0027916 03 MAY2002.123 14:39 03 MAY2002.123 14:39 RC=0000       1  K8
SEETAPE2  J0027917 03 MAY2002.123 14:39 03 MAY2002.123 14:40 RC=0000       1  E9
BACKUP DSN=$AVRS.R50A.FICH.MAY123
VOLSER(S)=629711
FILESEQ=
DATE/TIME=20022127,111030
 ******************************** END OF DATA *********************************
```

Figure 35 Fiche Index Report

Displaying Fiche History (FH)

Fiche creation and index information may be extracted from the FICHE INDEX REPORTS according to the JES NAME of the work processed by specifying a **TYPE** field of **FH** on the Primary Control Panel, along with the **NAME** of the work requested.

A directory of work meeting the selection criteria is presented. The Directory Heading fields are extracted from the entry for the work in the corresponding FICHE INDEX REPORT. A partial Fiche History Selection Panel appears as follows:

```
-THU 07 MAR 2002.066   --------    $AVRS FICHE  HISTORY SELECTION    ------------------                    ROW 1 OF 4
COMMAND ===>                                                                                          SCROLL ===> CUR
SELECT AN ENTRY WITH CODE:   S FOR DISPLAY

 SEL# ACT  JOBNAME     JOB#  TYP      START DATE    TIME       END DATE      TIME   CCODE    CREATIONDATE  TIME  FI#
 _    1      C203DMP 0001260  JOB  TUE15FEB2000.046  10:52  TUE15FEB2000.046  10:52   RC=0000  07MAR2000.067   16:03     1
 _    2      C203DMP 0001264  JOB  TUE15FEB2000.046  10:54  TUE15FEB2000.046  10:54   SI/O      07MAR2000.067   16:03     1
 _    3      C203DMP 0001453  JOB  WED16FEB2000.047  11:49  THU17FEB2000.048  11:49   SI/O      07MAR2000.067   16:03     1
 _    4      C203DMP 0001668  JOB  THU17FEB2000.048  15:14  THU17FEB2000.048  15:14   SI/O      07MAR2000.067   16:03     1
************************************** END OF DATA ****************************************
```

Figure 36 Fiche History Selection Panel

A detail summary screen for each entry may be selected by entering an **S** to the left of the entry, or entering **S** followed by the entry number from the command line of the Directory Panel. The following is a sample FICHE HISTORY DETAIL PANEL:

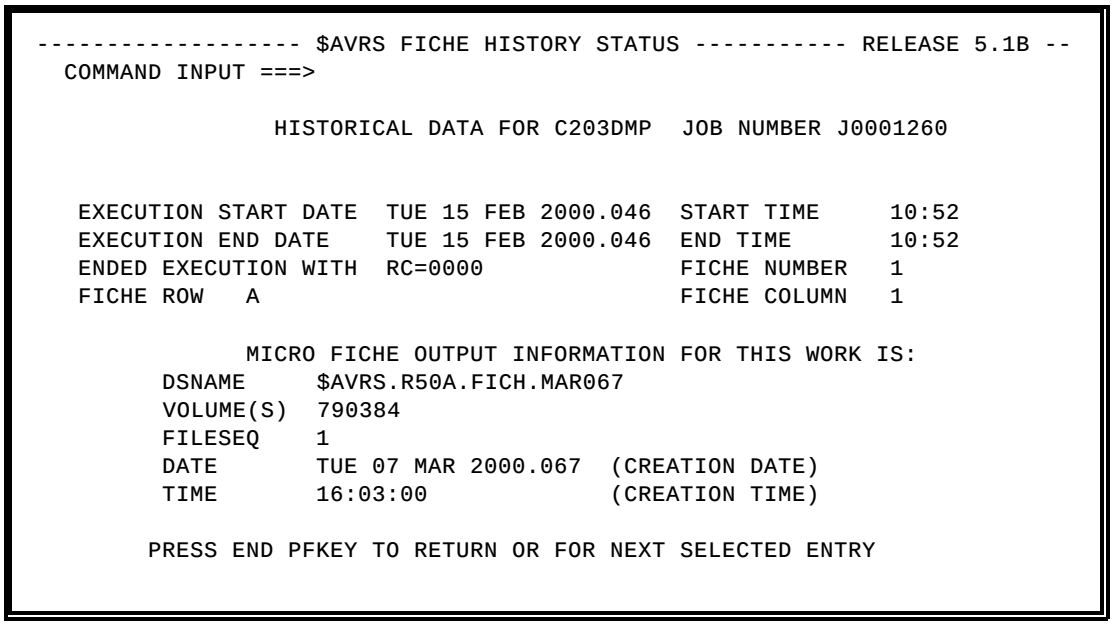


Figure 37 Fiche History Detail Panel

Selection of the Detail Panel is the only valid processing function for FICHE HISTORY selections.

Printing, output to a dataset, and delete are not valid functions for FICHE HISTORY (FH). Deletes may be done at the (FI) FICHE INDEX REPORT level, if the user has installation defined authority.

Selecting Work VIA Quick Path

Quick Path is a method to *quickly* get into $AVRS *from another application* and either Browse a job or display either the Files panel or the Info panel for the selected job *without going through the primary panel* or, potentially, a directory panel. Optionally, this method includes a Quick Exit that returns the User to the original panel they were on when they entered the Quick Path command.

By entering the Quick Path command on the Command Line from another application, the user will be presented with

- a directory list of selected jobs, if multiple jobs matched the selection criteria
- the desired $AVRS Panel (Info, Files or Browse) when a single job matched the selection criteria.

If the Quick Exit option has been set to **Y**(yes), pressing PF3/F3 will return to the original panel you were on before you entered the Quick Path Command.

Quick Path requires the use of the CLIST that contains the parameters needed to select the desired work. Two sample CLISTs are provided in the $AVRS installation library.

Two defaults *must* be implemented in the CLIST:

- DC(??) to present the desired panel when a single job is selected
- QX(?) to allow or prevent Quick Exit.

Contact your $AVRS Administrator and staff for information about the Clist name(s) and options allowable for your installation.

Selection Parameters

All of the possible Quick Path CLIST parameters are listed below. Examples are shown after the CLIST parameters:

J# The JES number of the work. This field is one to seven numeric digits. Only work with an exactly matching job number will qualify for selection.
Valid for JN, SN, TN.
Example: J#(3428) | J#(16749) | J#(0003971)
Default: NONE

JN The JES job name. No masking allowed. However, jobs can be selected by their prefix. The entire name need not be specified. This field is one to eight characters.
Example: JN(PAY03020) | JN(C2) | JN(A110B.)
Default: If neither SN or TN is entered the system will default to JN(*).

SN The Started task name. No masking is allowed. However, tasks can be selected by their prefix. The entire name need not be specified. This field is one to eight characters.
Example: SN($AVRS) | SN(CICS3211) | SN(TRMS)
Default: JN(*).

TN The TSO job name. No masking is allowed. However, TSO jobs can be selected by their prefix. The entire name need not be specified. This field is one to eight characters.
Example: TN(BBMCK) | TN(C203P8519)
Default: JN(*).

ST The starting time of the work to be selected. When less than four (4) digits are entered $AVRS assumes "minutes" and converts the minutes to hhmm. unless the delimiter is included (h:mm). For an example if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for JN, SN, TN.
Example: ST(1230) | ST(91)
Default: 00:00 (midnight)

SD The starting date of the work to be selected. This date can be a five (5) byte Julian date (yyddd), a seven (7) byte Julian date (yyyyddd), an asterisk (*), for today's date or a negative number for today minus a specified number of days
Valid for JN, SN, TN.
Example: SD(98016) | SD(2000119) | SD(*) | SD(-5)
Default: Lowest possible

DC The display command specifies which type of panel will be shown for the selected work when a single job is selected. The possible panel types are the informational panel (**IN**) , the browse panel (**BR**) and the files panel (**FI**).

IN Info or Informational panel
BR Browse panel
FI Files panel

Valid for JN, SN, TN.
Example: DC(IN) | DC(BR) | DC(FI)
Default: DC(IN) in the sample CLIST

QX The quick exit parameter allows the user to press the PF3/F3 key and $AVRS will exit back to the screen where the initial command was entered. The values for this parameter are **Y** for yes and **N** for no.
Valid for JN, SN, TN.
Example: QX(Y) | QX(N)
Default: QX(Y) in the sample CLIST

Examples:

TSO SQP J#(2348) JN(PAY02678) DC(FI)
This will present a file directory for job **PAY02678** that has a JOB number of 2348 using the File panel. The $AVRS primary panel will not be presented.

TSO SQP J#(2348) JN(PAY02) DC(BR)
This will present a job that begins with **PAY02** and that has a JOB number of 2348 using the Browse panel without any intermediate $AVRS selection panels.

TSO SQP JN (PAY02) SD(98003)
This will present a directory list of all jobs that begins with **PAY02** and were executed on or after January 3, 1998. The $AVRS primary panel will not be presented.

TSO SQP JN(PAY02) ST(1600) SD(2001138)
This will present a directory list of all jobs that begin with **PAY02** that were executed on or after 4:00 PM on May 15, 2001.

Chapter 6

PRINTING, VTAM PRINTING, OUTPUTTING & DELETING ACCUMULATED WORK

For work accumulated in the $AVRS data base, utility process functions are available to:

- **! PRINT** — whole JOBs **(JN)**, STCs **(SN)**, TSUs **(TN)**, SYSLOG segments **(SL)**, Status Reports **(ST)**, FICHE Reports **(FI)**, or individual files from any of these.
- **! VTAM PRINT** — whole JOBs **(JN)**, STCs **(SN)**, TSUs **(TN)**, SYSLOG segments **(SL)**, Status Reports **(ST)**, FICHE Reports **(FI),** or individual files from any of these.
- **! OUTPUT** — work to an existing or dynamically allocated dataset.
- **! DELETE** — work other than Archive History **(AH)**, or Fiche History **(FH)** from the data base (authorized users).

PRINTING

Enter a **P** to the left of an entry on a Work Selection or File Selection Directory to request $AVRS to route the entry's output lines to print. PRINT may also be requested by entering **P** followed by the entry number at the command line of a Selection Directory. Print is valid for all work on the data base except Archive and Fiche History. When print of an entire JOB, STC or TSU is requested, the user is presented with the following Print Attributes Panel.

NOTE: ARCHIVE HISTORY (AH) uses the Archive History Print function.

```
 -------------------- $AVRS PRINT FACILITY ------------ RELEASE 5.1B -
COMMAND INPUT ===>

 PRESS ENTER TO PRINT TESTATTS (J0029750) WITH THE FOLLOWING ATTRIBUTES
         OR CHANGE THE PRINT ATTRIBUTES AND PRESS ENTER.

SYSOUT CLASS ==> H               BANNER PAGES ==> N  (Y, N or 1-9)
HOLD         ==> N               PGMR NAME    ==>
FORMS ID     ==> STDXPRNT        ROOM NUMBER  ==>
FCB ID       ==>                 FLASH COUNT  ==>
UCS ID       ==>                 FORMDEF(AFP) ==>
COPIES       ==>                 PAGEDEF(AFP) ==>
DEST         ==>
COPY GROUPS  ==>
FLASH        ==>                 MEMBER(PRE)  ==>
BURST        ==> NO              MEMBER(POST) ==>
CHARS        ==>
COPYMOD      ==>                 MODTRC       ==>
WTRNAME      ==>                 SYSOUT PROGRAM NAME
OUTPUT CC    ==> A               PRINT CC - A=ANSI, M=MACHINE, N=NONE
OUTPUT NAME  ==>                 OUTPUT STMT Name (in TSO logon proc)
```

Figure 38 Job Print Attributes Panel

Print Panel Fields

The following are guidelines for the values on the Print Attributes Panel:

Value	Explanation
SYSOUT CLASS	Any valid SYSOUT PRINT CLASS.
HOLD	**Y** to Create HELD SYSOUT **N** to Create NON-HELD SYSOUT.
FORMS ID	1 - 8 char FORMS ID ("blank" uses installation default).
FCB ID	1 - 4 char FORMS CONTROL BUFFER ID
UCS ID	1 - 4 char UNIVERSAL CHARACTER SET ID ("blank" uses installation default).
DEST	JES Destination or JESDEST.SUBDEST.
COPIES	Total number of copies to reproduce.

COPY GROUPS	1 - 8 groups, Copy group specifications, i.e. 123, 3, 12.
FLASH	1 - 4 char FLASH ID ("blank" indicates no FLASH).
FLASH COUNT	1 - 255 FLASH COPIES.
BURST	**Y** to BURST stack on a 3800. **N** or BLANK continuous forms stacker.
CHARS	1 - 4 CHARACTER ARRANGEMENT TABLE NAMES for a 3800.
COPYMOD	COPYMOD id or enter **OPTJ** for OPTCD J only.
MODTRC	0, 1, 2 or 3 TABLE REFERENCE CHARACTER for COPYMOD.
WTRNAME	Sysout Program Name to assign to the printout.
OUTPUT CC	The kind of carriage control to use for this printout.
OUTPUT NAME	1 - 8 character name of a MVS OUTPUT JCL statement that exists in your logon proc. This printout will assume all print attributes specified on this JCL statement. OUTPUT NAME is mutually exclusive with FORMDEF and PAGEDEF. FORMDEF and PAGEDEF cannot be used to override the attributes in OUTPUT NAME.
BANNER PAGES	**Y** to Produce, **N** to Suppress $AVRS banner pages, or number 1 through 9 for multiple banner pages.
PGMR NAME	Programmer Name to place on $AVRS banner pages.
ROOM NUMBER	1 - 8 character Room Number to be printed on $AVRS banner pages.
MEMBER (PRE)	A member name from the $AVRFSEA Print PDS dataset to precede this printout.
MEMBER (POST)	A member name from the $AVRFSEA Print PDS dataset to follow this printout.

FORMDEF 1 - 6 character AFP forms overlay library member that contains statements to control the Print Services Facility for printing data sets on a page-mode printer. This field is mutually exclusive of OUTPUT NAME. It will not override the attributes assigned in OUTPUT NAME.

PAGEDEF 1 - 6 character AFP page overlay library member that contains statements to control the Print Services Facility for page formatting data sets on a page-mode printer. This field is mutually exclusive of OUTPUT NAME. It will not override the attributes assigned in OUTPUT NAME.

Fill in or change Print Characteristics as desired and press enter to route output to print.

Closing the Print File

Multiple entries, of the same or different types, may be queued for print in the same Sysout dataset. As each entry is selected, the Print Attributes Panel is presented. As long as the Print Attributes are not altered, the work is queued in the same data set. If the Attributes are changed, the previously printed work is released to JES and a new Print Dataset is allocated. Accumulated Printed work may also be released by entering the command **CLOSE PRT** at the command line of any directory, or the Primary Control Panel. If the user does not explicitly close the print dataset, it will automatically be closed upon leaving $AVRS.

ENTRY:
CLOSE PRT
C PRT
CPRT

When printing a single file, rather than an entire JOB, the Print Attributes Panel presented is as follows:

```
 TESTATTS (J0029750) ------ $AVRS PRINT FACILITY -------- RELEASE 5.1B--
 COMMAND ===>
  FILENAME C FORM FCB  UCS  DEST       FLASH BURST CPYMD PAGES     LINES
  SYSMSG   N FO3C           U113                          1        26
-------------------------------------------------------------------------
MODIFY IF DESIRED, PRESS ENTER TO PRINT, ELSE PRESS END PFKEY TO ESCAPE
ENTER Y TO DISPLAY COLLECTED ATTRIBUTES ==> N
ENTER Y TO PRINT WITH COLLECTED ATTRIBUTES    ==> N
 -------- REQUIRED ---------            ---------- OPTIONAL ---------------- --
   SYSOUT CLASS ==> H       BANNER PAGES==> N     (Y, N OR 1-9)
   HOLD         ==> N       PGMR NAME   ==>
   BURST        ==> NO      ROOM NUMBER ==>
   OUTPUT CC    ==> A       CARRIAGE CTL  -  A=ANSI, M=MACHINE, N=NONE
-------- OPTIONAL --------- MEMBER(PRE) ==>
   FORMS ID     ==> STDXPRNT MEMBER(POST)
   FCB ID       ==>         FORMDEF(AFP)==>
   UCS ID       ==>         PAGEDEF(AFP)==>
   DEST         ==> U513    PRINT       ==>   P=PAGES,L=LINES
   COPIES       ==>         FROM        ==>   THRU ==>
   COPY GROUPS  ==>
   FLASH        ==>         FLASH COUNT ==>
   CHARS        ==>
   COPYMOD      ==>         MODTRC      ==>
   OUTPUT NAME  ==>         WTRNAME     ==>
```

Figure 39 File Print Attributes Panel

The user may enter Print Attributes from the panel, as with the full JOB Print Panel, or may elect to Print the file with its accumulated attributes by entering **Y** in the appropriate input field.

File level print can be limited to specific lines or pages. When printing a partial file using the lines or pages criteria, please specify both the **FROM** and the **THRU** fields numeric values. File level print using the **FROM** and **THRU** fields for Syslog segments **(SL)**, Status Reports **(ST)**, and Fiche Reports **(FI)** may be done by selecting the desired work and then using PF3 to obtain a file selection panel.

VTAM Printing

This release of $AVRS is supported by VTAM Print version 3.0B or above. Enter a **V** to the left of an entry on a Work Selection or File Selection Directory to request $AVRS to route the entry's output lines to the optional **VTAM Print** product. To use this feature, the $AVRS Environment Area must contain the name of the **VTAM Print** RRDS dataset. The **VTAM Print** started task does not need to be running for $AVRS to place work in the **VTAM Print** RRDS, but it does need to be running to actually print the work. The **VTAM Print** feature may also be requested by entering **V** followed by the entry number at the command line of a Selection Directory. **VTAM Print** is valid for all work on the data base except Archive and Fiche

History. When an entire JOB, STC or TSU is requested, the user is presented with the following **VTAM Print** Attributes Panel:

```
------------------- $AVRS VTAM PRINT FACILITY ---------- RELEASE 5.1B--
COMMAND INPUT ===>
      JOBNAME  C FORM FCB   UCS   DEST   FLASH BURST CPYMD PAGES LINES
      C203ATTS N F03C             U513                          5    94
--------------------------------------------------------------------------

  PRESS ENTER TO PRINT C203ATTS ( J0008036 ) WITH THE FOLLOWING ATTRIBUTES
            OR CHANGE THE PRINT ATTRIBUTES AND PRESS ENTER.

 VTAM PRINTER ==> P45F       (VTAM Print ID)
 SYSOUT CLASS ==> H
 HOLD         ==> N
 FORMS ID     ==> ABCD
 FCB ID       ==>
 MEMBER (PRE) ==>            (Print Control Member Name, PRE-PRINT)
 MEMBER (POST)==>            (Print Control Member Name, POST PRINT)
 BANNER PAGES ==> N          (Y, N or 1-9)
```

Figure 40 Job VTAM Print Attributes Panel

VTAM Print Panel Fields

The following are guidelines for the values on the VTAM Print Attributes Panel:

Value	Explanation
VTAM PRINTER	The name of a printer defined in the **VTAM Print** product.
SYSOUT CLASS	Any valid SYSOUT PRINT CLASS for the specified printer.
HOLD	**Y** to hold in **VTAM Print**, **N** to make available for printing.
FORMS ID	1 - 8 char FORMS ID ("blank" uses installation default).
FCB ID	1 - 4 char FORM CONTROL BUFFER ID ("blank" uses installation default).
MEMBER NAME	A member name from the $AVRFSEA Print PDS dataset to precede or follow this printout.

BANNER PAGES **Y** to Produce.
N to Suppress $AVRS banner pages, or number 1 through 9 for multiple banner pages.

Fill in or change the **VTAM Print** Characteristics as desired and press enter to route the output to **VTAM Print.** It is written to the **VTAM** database. There is no need to close the print file because each request is separate under **VTAM Print**.

When printing a single file, rather than an entire JOB, the Print Attributes Panel presented is as follows:

```
C203ATTS ( J0005265 ) ------ $AVRS VTAM PRINT FACILITY ---- RELEASE 5.1B--
COMMAND ===>
    FILENAME C FORM FCB  UCS  DEST   FLASH BURST CPYMD PAGES    LINES
    JCL      N STDX           U513                        1       25
-------------------------------------------------------------------------
 MODIFY IF DESIRED, PRESS ENTER TO PRINT, ELSE PRESS END PFKEY TO ESCAPE
 ENTER Y TO PRINT WITH COLLECTED ATTRIBUTES    ==> N
-------- REQUIRED ---------
 VTAM PRINTER  ==> P45F
 SYSOUT CLASS  ==> T
 HOLD          ==> N
---------------------------------- OPTIONAL -----------------------------
 FORMS ID      ==>
 FCB ID        ==>
 MEMBER(PRE)   ==>                          MEMBER(POST) ==>
 BANNER PAGES  ==> Y        (Y, N OR 1-9 )
 PRINT         ==>          P=PAGES,L=LINES
 FROM          ==>          THRU ==>
```

Figure 41 File Print Attributes Panel

The user may enter Print Attributes from the panel, as with the full JOB Print Panel, or may elect to Print the file with its accumulated attributes by entering **Y** in the "print with collected attributes" field.

File level print can be limited to specific lines or pages. When printing a partial file using the pages or lines criteria, please specify both the **FROM** and the **THRU** fields numeric values. File level print using the **FROM** and **THRU** fields for Syslog segments **(SL)**, Status Reports **(ST)**, and Fiche Reports **(FI)** may be done by selecting the desired work and then using PF3 to obtain a file selection panel.

Screen Printing - (Non ISPF)

THIS SECTION DOES NOT APPLY TO ISPF

A screen print facility is provided by ISPF for users signed on to $AVRS through that interface.

When $AVRS is accessed through either the Command Processor (TSO, ROSCOE) or VCSI, a screen print facility is provided by $AVRS.

ENTRY: **PRINT**
RESULT: **Causes $AVRS to copy the current screen to a Print dataset.**
*ENTRY***: CLOSE PRT**, **C PRT**, **CPRT**

Multiple screens may be printed to the same dataset. When the first **PRINT** command is entered, a Print Attributes Panel, similar to those presented for printing work from the data base is displayed. The user may fill in or change the attributes to define the print dataset. For subsequent screen PRINTs, the Attribute Panel is not displayed. Print is accumulated to the same dataset until a **CLOSE PRT** command is issued. Work may be released by entering the command **CLOSE PRINT** at the command line of any directory, or the Primary Control Panel. If the user does not explicitly close the dataset, it will automatically be closed upon leaving $AVRS. After a Print dataset has been closed and the work queued to JES, the next screen PRINT will again cause display of the Print Attributes Panel.

NOTE: $AVRS Utility panels may not be printed. These include: the Delete Confirmation Panel, the Dataset Allocation Panel, the Job Submission Panel and the Print Attributes Panels. All other $AVRS panels and directories, including HELP Panels may be printed through the $AVRS Screen Print facility.

```
------------------    $AVRS SCREEN PRINT  --------- RELEASE 5.1B--
COMMAND INPUT ===>
    PRESS ENTER TO PRINT SCREEN WITH THE FOLLOWING ATTRIBUTES
         OR CHANGE THE PRINT ATTRIBUTES AND PRESS ENTER.

  SYSOUT CLASS ==> T           BANNER PAGES ==> N     (Y, N OR 1-9)
  HOLD         ==> N           PGMR NAME    ==>
  FORMS ID     ==>             ROOM NUMBER  ==>
  FCB ID       ==>             MEMBER NAME  ==>
  UCS ID       ==>
  DEST         ==>
  COPIES       ==>
  COPY GROUPS  ==>
  FLASH        ==>             FLASH COUNT  ==>
  BURST        ==> N
  CHARS        ==>
  COPYMOD      ==>             MODTRC       ==>
  WTRNAME      ==>             SYSOUT PROGRAM NAME
  OUTPUT CC    ==> A           PRINT CC - A-ANSI, M=Machine, N=None
  OUTPUT NAME  ==>
```

Figure 42 $AVRS Screen Print Panel

Outputting Work

The process function designator **O** is used to request $AVRS to output an accumulated JOB **(JN)**, STC **(SN)**, TSU **(TN)**, SYSLOG Segment **(SL)**, Status Report **(ST)**, Fiche Report **(FI)**, or file to a system dataset. When output is requested, the following Dataset Allocation Panel is presented:

```
------------- $AVRS OUTPUT DATASET ALLOCATION ----------- RELEASE 5.1B--
    COMMAND INPUT ===>

       DATASET NAME      ===> $AVRS.OUTPUT.DATA
       MEMBER NAME       ===>       (BLANK ELSE MEMBER NAME FOR "PDS")
       DISPOSITION       ===> OLD   (N/NEW,M/MOD OR O/OLD)
       NORMAL DISP.      ===>       (C/CATLG OR K/KEEP)
       SPACE UNITS       ===>       (C/CYLS OR T/TRKS)
       PRIMARY QUAN.     ===>       (IN ABOVE UNITS)
       SECONDARY QUAN.   ===>       (IN ABOVE UNITS)
       DIRECTORY BLOCKS  ===>       (ZERO OR BLANK FOR SEQUENTIAL)
       RECORD FORMAT     ===>
       RECORD LENGTH     ===>       (MAX 255 RECFM=F, MAX 259 RECFM=V)
       BLOCK SIZE        ===>       (ZERO FOR SYS DETERMINED BLOCKSIZE)
       VOLUME SERIAL     ===>       (BLANK FOR AUTHORIZED DEFAULT VOLUME)
       UNIT              ===>       (BLANK FOR AUTHORIZED DEFAULT UNIT)
       SMS STORAGE CLASS ===>
       SMS MNGMNT CLASS  ===>
       SMS DATA CLASS    ===>
       EXPDT/RETPD       ===>       (YYDDD = EXPDT, 9999 = RETPD)
```

Figure 43 $AVRS Dataset Allocation Panel

Dataset Allocation Panel Fields

The following are guidelines for the fields of the Dataset Allocation Panel.

If disposition is ***old*** or ***mod*** the following fields are valid:

Field	Explanation
DATASET NAME	Required. The dataset name may be fully qualified within single quotes or if single quotes are omitted, your prefix will be supplied for the dataset.
MEMBER NAME	Required if the dataset is a PDS.
VOLUME SERIAL	Optional (works in conjunction with UNIT).
UNIT	Optional (works in conjunction with VOLUME SERIAL).

If disposition is ***new*** the following fields are valid:

DATASET NAME	Required. The dataset name may be fully qualified within single quotes or if single quotes are omitted, your prefix will be supplied for the dataset.
MEMBER NAME	Required if the dataset is a PDS.
NORMAL DISP.	Required. Value must be CATLG or KEEP.
SPACE UNITS	Required. Value must be CYLS or TRKS.
PRIMARY QUAN.	Required. # of primary allocation space units.
SECONDARY QUAN.	Optional. # of secondary allocation space units.
DIRECTORY BLOCKS	Required for PDS allocation.

RECORD FORMAT	Required. Valid values: F or FB with or without carriage control -- A or M V or VB with or without carriage control -- A or M.
RECORD LENGTH	Required. Maximum of 255 for RECFM=F, maximum for 259 RECFM=V.
BLOCK SIZE	BLANK (defaults to 13030), ZERO (System Determined BLOCKSIZE) or any BLOCKSIZE that will fit on a track.
VOLUME SERIAL	Optional (works in conjunction with UNIT).
UNIT	Optional (works in conjunction with VOLUME SERIAL).
SMS STORAGE CLASS	Optional. Specifies the storage class for the dataset. It is used in lieu of the **UNIT** and **VOLUME** parameters.
SMS MNGMNT CLASS	Optional. Specifies the Management class for the data set.
SMS DATA CLASS	Optional. Specifies the Data class for the dataset.
EXPDT/RETPD	Optional. Specifies the Expiration date or the Retention period for the dataset. Numeric values of 5 or 7 bytes are treated as Expiration dates, (yyddd or yyyyddd) . Other numeric values less than 5 bytes in length are considered retention periods. These parameters override corresponding **DATA** class values if **MANAGEMENT** class allows it.

Closing the Output Dataset

Once allocated, the output dataset *remains open* for further output requests *until it is closed by command or the user exits $AVRS*. The following command may be entered at the command line of any of the Directory Panels or the Primary Control Panel to cause the output dataset to be closed and freed.

ENTRY: **CLOSE DS, C DS, CDS**

Deleting Work

Individual files or whole jobs may be deleted from the $AVRS data base by entering the process function designator **D** either to the left of the entry, or at the command line in conjunction with the entry number.

NOTE: Users must have installation defined authority to perform deletions.

If the **CONFIRM DELETIONS** field on the Primary Control Panel has been specified as **Y** (as recommended), the user is presented with the Delete Confirmation Panel as follows:

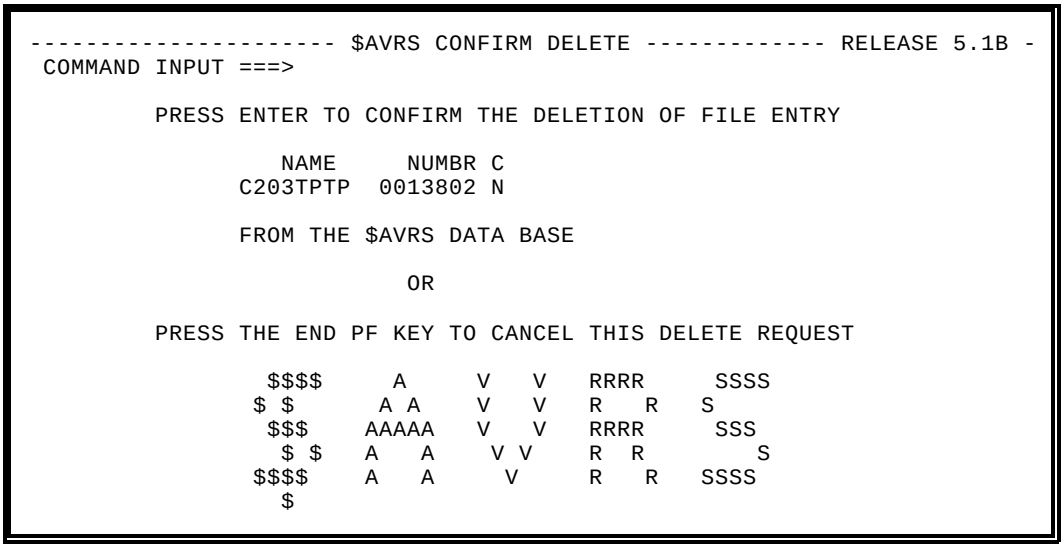


Figure 44 Delete Confirmation Panel **(Job Level)**

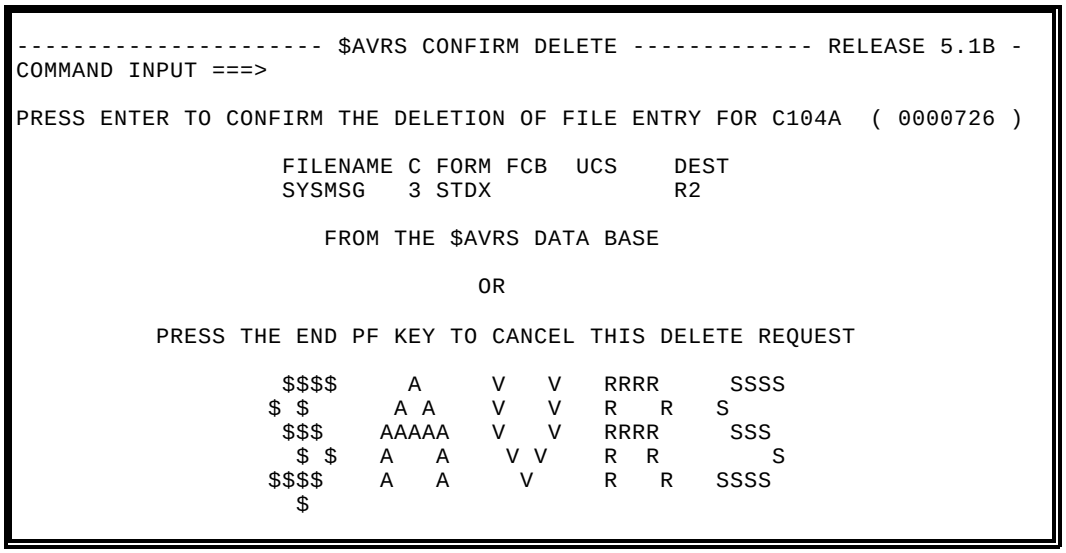


Figure 45 Delete Confirmation Panel **(File Level)**

The delete function is not valid (with default parameters) for Status Reports, **(ST)** or Fiche Index Reports, **(FI)**. Special delete authority is usually required to delete these special $AVRS work types.

Online deletes from Archive History, **(AH)** or Fiche History, **(FH)** is not possible.

Chapter 7

DISPLAYING STATUS OF ZAPS AND CUSTOMIZATIONS

Maintenance Zap Status Display Panel

Specification of a type of **DZ** or **DC** in the **TYPE** field on the $AVRS Primary Control Panel results in the display of the status for maintenance or customization zaps that have been applied to $AVRS. A type of DZ presents a maintenance Zap Status Display Panel similar to the one below.

```
------------- $AVRS Maintenance   ZAP Status Display ---- Release 5.0A-06 --
   COMMAND ===>
   $AVR226I - LEVEL SET 06 INCLUDES APARS THROUGH 30
   001    021
   002-1  022
   003    023
   004    024
   005    025
   006-1  026
   007
   008    028
   009    029
   010
   011
   012-2
   013
   014
   015
   016
   017
   018
   019
```

Figure 46 Maintenance Zap Status Display Panel (DZ)

When a maintenance or customization zap is applied to the current release of $AVRS, the appropriate field on the panel above is updated with the zap identification number. The applied zap identification numbers are listed positionally in columns on these panels. If a position on the panel is blank, the zap has not been applied. A revision of a zap has been applied if the zap identification number is followed by a dash and a value from 1 - 9. For example, 012-2 indicates that revision number 2 has been applied to zap number 012. If more than 9 revisions

have been applied to a particular zap, an asterisk (*) appears in the display. For example, 001-* would indicate that 10 or more revisions have been applied to zap 001. To view zap numbers 261 through 520, press the *Down PFkey*. To return to zap numbers 1 through 260 press the *Up PFkey*.

Maintenance Customization Display Panel

A type of DC presents a Maintenance Customization Display Panel Similar to the one below. If no customizations have been applied, the following message is displayed.

$AVR475I - NO CUSTOMIZATION ZAPS ARE IN EFFECT FOR NUMBERS 1 THRU 260

To view the custom numbers for 261 through 520, press the *Down Pfkey*. To return to the custom numbers 1 through 260, press the *UP PFkey*.

```
---------- $AVRS Customization ZAP Status Display ---------- Release 5.1B ---
COMMAND ===> ----------

                                        178

020
```

Figure 47 Maintenance Customization Zap Status Display Panel

Section B

BATCH PROGRAMS

Overview - Chart of Programs

Below is an alphabetical chart of all current $AVRS programs - batch, utility, exits and main driver type programs. Following the chart all the Group 3 User Programs are discussed. Information about Group 1 and Group 2 Programs are only included in the Installation and Administrator's Manual.

Program	**Function**	**Location, Group, or Section** **A = Admin**	**U = User**
$$DELDMP	User program to delete dumps based on age	A - Group 1 Chapter 1	N/A
$$DELKEY	User program to delete jobs based on jobname and type	A - Group 1 Chapter 2	N/A
$AVRBANR	User program to create custom $AVRS banner pages	A - Group 1 Chapter 3	N/A
$AVRBKPR	Validates backup control statements	A - Section A Chapter 6	N/A

Program	Function	Location, Group, or Section A = Admin	U = User
$AVRBKUP	Backup/Archive	A- Section A Chapter 6	N/A
$AVRCLEN	Recover dead blocks, Broken Chains, Orphan KSDS records, etc,.	A-Group 1 Chapter 4	N/A
$AVRCOND	Used to produce different numeric return codes while testing $AVRS installation	A - Group 2 Chapter 1	N/A
$AVRCRSE	Creates the VCSI user/password table	A - Section A Chapter 7	N/A
$AVRDBRP	Report on contents of database	A - Group 2 Chapter 2	N/A
$AVRECOV	Replaced by $AVRCLEN in 5.1B.	N/A	N/A
$AVRFDEL	Create delete records for microfiche	A - Group 1 Chapter 6	N/A
$AVRFICH	Create Microfiche file	A - Group 1 Chapter 5	N/A
$AVRFMRG	Selects archived jobs forward merge	A - Group 1 Chapter 7	N/A
$AVRFMRR	Formats database; load archived/unload files	A - Section A Chapter 4	N/A
$AVRFOLD	Converts 80 character input	A - Group 2 Chapter 3	N/A
$AVRGRPT	Report on options in effect	A - Group 2 Chapter 4	N/A
$AVRINIT	Validate, report, store user options, accumulate sysout, operator commands, initiate backups restores respins	A - Section A Chapter 5	N/A
$AVRLIDR	Displays the IDRDATA on the loadlib modules	A - Group 2 Chapter 5	N/A
$AVRLOCK	Reset the $AVRS locking mechanism, sync the database	A - Group 1 Chapter 8	N/A
$AVRMRGF	Selects archived jobs for merging forward	A - Group 1 Chapter 9	N/A
$AVRMXPR	Validate and syntax check changes to the MAXLINE Definition Table	A - Section A Chapter 5	N/A

Program	Function	Location, Group, or Section A = Admin	 U = User
$AVRPULL	Batch interface, extract work from data base	A- Group 3 Chapter 1	U- Group 3 Chapter 1
$AVRSCP	TSO interface ROSCOE/ETSO interface entry	A- Section B Chapter 2	N/A
$AVRSELR	Selective restore	A - Group 3 Chapter 2	U - Group 3 Chapter 2
$AVRSTMP	Force a WTO time stamp	A - Group 2 Chapter 6	N/A
$AVRSXPR	Validates and syntax check changes to the scan argument definition records	A - Section A Chapter 5	N/A
$AVRTAPE	Extract work from archive batch mode	A - Group 3 Chapter 3	U - Group 3 Chapter 3
$AVRTPRP	Report on contents of backup tape files	A - Group 3 Chapter 4	U - Group 3 Chapter 4
$AVRUNLD	Unload database to sequential file for dif. device types	A - Group 2 Chapter 7	N/A
$AVRUX1	Backup user exit	A- Appendix A	N/A
$AVRUX2	Extract authorization user exit	A- Appendix A	N/A
$AVRUX3	Maxlines user exit	A- Appendix A	N/A
$AVRUX4	Work deletion authorization user exit	A- Appendix A	N/A
$AVRUX5	Print authorization user exit	A- Appendix A	N/A
$AVRUX6	Token creation user exit	A- Appendix A	N/A
$AVRUX7	Respin disposition user exit	A- Appendix A	N/A
$AVRUX8	Dataset respin attributes user exit	A- Appendix A	N/A
$AVRUX9	Message modification user exit	A- Appendix A	N/A
$AVRUXA	Job submit JCL user exit	A- Appendix A	N/A
$AVUMTRR	Empties an RRDS file	A - Group 1 Chapter 10	N/A
$AVURMRR	Removes an RRDS file	A - Group 1 Chapter 11	N/A

Program	Function	Location, Group, or Section A = Admin	U = User
CONTRUX7	During accumulation prints one page summary of accumulated job	A- Appendix A	N/A
DSPLYGBL	Formatted display of the KSDS control record	A - Group 2 Chapter 8 Use only under direction of support personnel	N/A
DSPLYLCK	Formatted display of the Lock control record	A - Group 2 Chapter 9 Use only under direction of support personnel	N/A
EQQUX010	OPC restart User Exit	A- Appendix A	N/A
FRONTEND	Customized intermediate primary panel usage	A - Group 2 Chapter 10	N/A
ISP$AVRS	SPF/ISPF dialog manager	A- Section B Chapter 2	N/A
USERPGM	User program to access the $AVRS database	A - Group 2 Chapter 11	N/A

The programs not discussed during the installation and accumulation portion of the Installation and Administrator's Manual have been divided into three groups and listed alphabetically within the groups.

- Group one programs (Administrator's Use Only) would commonly be used only by the $AVRS Administrator and staff. These programs alter the $AVRS database, its contents, or its functioning.

- Group two programs (Administrator's Informational and Utility Programs) provide database informational type reports. These programs will not interest the normal $AVRS user.

- Group three programs (Administrator's and User's Programs) could be valuable for Users or the $AVRS Administrator and staff.

Group #1

The list of programs below perform administrative functions. Their execution should be performed only under the control of the $AVRS Administrator.

$$DELDMP
$$DELKEY
$AVRBANR
$AVRCLEN
$AVRFDEL
$AVRFICH
$AVRFMRG
$AVRLOCK
$AVRMGRF
$AVUMTRR
$AVURMRR

Group #2

Other batch jobs included in the Administrators manual that perform non-user related functions:

$AVRCOND
$AVRDBRP
$AVRFOLD
$AVRGPRT
$AVRLIDR
$AVRSTMP
$AVRUNLD
DSPLYGBL
DSPLYLCK
FRONTEND
USERPGM

Group #3

The following batch jobs could be valuable for Administrators or Users and are included in the Users Manual and the Administrator Manual;

$AVRPULL
$AVRSELR
$AVRTAPE
$AVRTPRP

NOTE: Only Group #3 Program information is included in the User's Manual

Chapter 1

$AVRPULL - Online Database Extract

The **$AVRPULL** Program is used to Extract Work From The Online Database. This program does not extract work from Archive History files - use the $AVRTAPE program for that purpose.

Return Codes

$AVRPULL sets the following return codes.

- Zero - normal processing
- Four - processing completed but no work was selected
- Eight - processing terminated with invalid parameter values
- Twenty - all parameters failed validation or none were specified.

It may be initiated in any of the following ways:

- **!** As a started task or batch job with a **PARM** Value of **WTOR** (**PARM=WTOR**).
- **!** As a batch job without a specified PARM value.
- **!** As a TSO foreground program executed via a CLIST.

Regardless of which way $AVRPULL is used, $AVRPULL can print work from the online database in two separate ways. The RSLOCAL DDNAME is used when printing to either hardcopy or to a file. The RSLOCAL DDNAME may be changed to any valid DDNAME.

The $AVRVPRT DDNAME is used when sending print to the **VTAM Print** product. The $AVRVPRT DDNAME, if used, should point to the **VTAM Print** RRDS dataset. **VTAM Print** does not have to be active for $AVRPULL to send work to it, but it must be active for the selected work to actually print.

Security in the **$AVRPULL** Program is handled in four ways:

- **!** A response from the operator may be required to message '**$AVR701A**" for a password if **PARM=WTOR**.
- **!** The opening of the VSAM DATA file if the VSAM password facility is being used allowing access via **$AVR1ST**.
- **!** The authorization by the **$AVRUX2** user exit.
- **!** The installations' external security package via the **$AVRSECU** module.

Using $AVRPULL with a PARM of "WTOR", (PARM=WTOR)

The program issues **WTOR**s for input requests from the user and **WTO**s as output replies back to the user. The user should respond to message "**$AVR703A**" with the desired **$AVRPULL** selection criteria and the selected output will be written to DDNAME "RSLOCAL" or sent to DDNAME "$AVRVPRT". The DCB attributes for RSLOCAL can specify a RECFM of either "**V**" or "**F**", blocked or unblocked and either "A", "M" or no carriage control. The BLKSIZE can be any value supported by the output device type and the LRECL must be 18 or larger, up to 255 characters. The RSLOCAL output dataset is written with QSAM and may be allocated to any QSAM supported device or to sysout. The default DCB attributes for RSLOCAL are RECFM(VBM), LRECL(259), and BLKSIZE(263).

A procedure "**OPERPULL**" is supplied in the $AVRS.rel#.INSTLIB. Modifications must be made to the procedure to supply your dataset name for the $AVRLOCK DDNAME. $AVRKSDS and $AVRRRDS names are no longer required. If $AVRS custom banners are not desired, a **PARM** value of "**NOBANNER**" (**PARM=NOBANNER**), can be used to suppress the $AVRS BANNER PAGES.

The following is a Sample JCL - WTO Parm for the $AVRPULL Program execution. This JCL is also provided in "$AVRS.rel#.INSTLIB (OPERPULL)".

```
//$AVRPULL PROC OUT=A
//$AVRS    EXEC PGM=$AVRPULL,PARM=WTOR,REGION=768K
//*$AVRS   EXEC PGM=$AVRPULL,PARM='WTOR,NO BANNER',REGION=768K
//STEPLIB  DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR
//*
//*$AVRVPRT DD DISP=SHR,DSN=Your.VTAM.Print.RRDS
//*
//*    $AVRVPRT DD IS THE OUTPUT VTAM PRINT RRDS
//*
//*    RSLOCAL DD IS THE OUTPUT SELECTION PRINT FILE
//*
//RSLOCAL  DD  SYSOUT=&OUT
//*
//*    $AVRLOCK DD IS ASSIGNED TO THE LOCK FILE
//$AVRLOCK DD  DSN=$AVRS.rel#.LOCK,DISP=SHR
//*
//*
//*  THE $AVRKSDS AND $AVRRRDS DD  STATEMENTS ARE NO LONGER
//*  REQUIRED.
//*
//*    $AVRACCM DD IS ASSIGNED TO THE RRDS FILE
//*
//* $AVRWORK DD  UNIT=SYSALLDA,SPACE=(CYL,(2,2))
//*
//*  $AVRWORK DD IS THE $AVRS TEMP WORK FILE.  IT IS DYNAMICALLY
//*    ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL
//*
//SYSPRINT DD  DUMMY
//*
//*       OUTPUT $AVRPULL MESSAGE ARE TO THE CONSOLE
//*
//SYSIN    DD  DUMMY
//*
//*       INPUT SOURCE FOR THE SELECTION PARAMETERS IS THE CONSOLE
```

Figure 48 $AVRPULL WTOR PARM Example JCL (OPERPULL)

$AVRPULL - BATCH

$AVRPULL — Batch: The Input Selection Criteria is read from DDNAME "SYSIN" and the output responses are written to DDNAME "SYSPRINT". The selected work is written to either DDNAME "RSLOCAL" or DDNAME "$AVRVPRT". The RSLOCAL DDNAME may be written in a variety of ways. See the JCL member or the example on the following pages. The DCB attributes for RSLOCAL can specify a RECFM of either "**V**" or "**F**", blocked or unblocked and either "**A**", "**M**" or no carriage control. The BLKSIZE can be any value supported by the output device type and the LRECL can be from 18 to 255 characters excluding the RDW for variable records. The RSLOCAL output dataset is written with QSAM and may be allocated to any QSAM supported device or to sysout. The default DCB attributes for RSLOCAL are RECFM(VBM), LRECL(259), and BLKSIZE(263). The **$AVRPULL** selection criteria is defined below.

The following is a Sample JCL - Procedure for the $AVRPULL program execution. The JCL is also provided in member $AVRPULL in the INSTLIB.

```
//$AVRPULL JOB ('ACCOUNTING INFO')"$AVRS BATCH PRINT',MSGCLASS=N,
//             CLASS=X,TIME=10,NOTIFY=$SYSUID
//*
//*********** A PARM OF "NOBANNER" IS OPTIONAL TO SUPRESS BANNERS
//*
//STEP001   EXEC PGM=$AVRPULL,REGION=768K   ,PARM=NOBANNER
//*
//STEPLIB  DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR  -- $AVRS LOADLIB --
//*
//*               - $AVRS' DATABASE -
//$AVRLOCK DD DISP=SHR,DSN=$AVR.rel#.LOCK         -- $AVRS LOCK -
//*
//* THE $AVRKSDS DD STATEMENT IS NO LONGER RECOGNIZED
//* THE $AVRRRDS DD STATEMEMT IS NO LONGER RECOGNIZED
//*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//*      $AVRWORK DD IS THE $AVRS TEMP WORK FILE.  IT IS DYNAMICALLY
//*      ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL.
//*
//*  $AVRWORK DD   UNIT=SYSALLDA,SPACE=(cyl,(2,2))
//***********               ------- SELECT WORK TO: -------
//***********               ------- VTAM PRINT ------------
//*$AVRVPRT DD SDISP=SHR,DSN=PRTV.$AVRS.RRDS OR  ------
//RSLOCAL   DD SYSOUT*
//*
//***       (RSLOCAL MAY BE MODIFIED AS SHOWN BELOW)
//*
//**RSLOCAL DD DCB=(LRECL=133,BLKSIZE=133,RECFM=FM),
//***================>            ------- SYSOUT -------
//***       SYSOUT=A           ,HOLD=YES------- OR -------
//***================>       -------TO AN OLD DATASET -------
//*     DD=$AVRS.$AVRPULL.LIST,DISP=OLD --- OR ---
//***================>        ------- TO A NEW DATASET -------
//*     DD=$AVRS.$AVRPULL.LIST,DISP=(,CATLG,CATLG),
//*     DCB=(LRECL=133,BLKSIZE=15476,RECFM=FBA),
//*     UNIT=SYSDA,SPACE=(CYL,(25,1),RLSE)
//***********            ------- INPUT CONTROL VIA -------
//***********            ------- DATASET - PARTITIONED -
//***********            -------    OR -    SEQUENTIAL -
//$AVRPCTL DD DSN=$avr.REL#.CNTL(APDSNAME),DISP=SHR
//************           ------- OR SYSIN          -------
//SYSIN    DD *
ONEMATCH=YES
* *********** ONEMATCH=YES SELECT ONLY ONE MATCHING JOB
* *********** FOR EACH SELECTION CRITERIA BELOW
* *********** SELECT ALL JOBNAMES WITH PREFIX OF "SEA" THAT
* *********** RAN YESTERDAY BETWEEN 00:01 AND 23:59 HOURS
* *********** AND SEND PRINT TO VTAM PRINT QUEUE FOR PRINTER
* *********** P45F.  (NOTE THAT THE $AVRVPRT DD STATEMENT
* *********** MUST BE UNCOMMENTED TO DO THIS).
*TYPE=JN,NAME=SEA,SD=1,ED=-1,ST=0001,ET=2359,LUNAME=P45F
/*
```

Figure 49 $AVRPULL Batch Procedure Example

A procedure BATCPULL member is supplied in the INSTLIB. Modifications must be made to the procedure to supply your dataset name for the "$AVRLOCK". If $AVRS custom banners are not desired, a **PARM** value of "**NOBANNER**" (**PARM=NOBANNER**), can be used to suppress the $AVRS BANNER PAGES.

```
//$AVRBATC PROC OUT=A
//$AVRS    EXEC PGM=$AVRPULL,REGION=3848K
//STEPLIB  DD  DSN=$AVRS.REL#.LOADLIB,DISP=SHR
//*
//*       RSLOCAL DD IS THE OUTPUT SELECTION PRINT FILE
//*            (DDNAME RSLOCAL MAY BE MODIFIED)
//*
//*$AVRVPRT DD DSN=YOUR.VTAM.PRINT.RRDS,DISP=SHR
//*
//*      $AVRVPRT DD IS THE OPTIONAL VTAM PRINT RRDS
//*
//$AVRLOCK DD  DSN=$AVRS.REL#.LOCK,DISP=SHR
//*
//*       $AVRLOCK DD IS ASSIGNED TO THE LOCK FILE
//*
//SYSPRINT DD  SYSOUT=&OUT
//*
//*       SYSPRINT DD IS THE OUTPUT $AVRPULL MESSAGE FILE
//*
//* $AVRWORK DD  UNIT=SYSALLDA,SPACE=(CYL,(2,2))
//*
//* $AVRWORK DD IS THE $AVRS TEMP WORK FILE.  IT IS DYNAMICALLY
//*   ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL
//*
//************         -------INPUT CONTROL VIA -------
//************         -------DATASET - PARTITIONED -
//************         -------    OR-SEQUENTIAL -
//$AVRPCTL DD DSN=$AVRS.rel#.CNTL(APDSNAME),DISP=SHR
//************         -------OR SYSIN          -------
//SYSIN    DD  DUMMY
//*
//* SYSIN DD IS THE INPUT SOURCE FOR THE SELECTION PARAMETERS
//*
```

Figure 50 $AVRPULL Procedure Example (BATCPULL)

TSO FOREGROUND PROGRAM EXECUTED FROM A CLIST

$AVRPULL — TSO: The program issues TGETs for input, and TPUTs for output requests respectively when invoked from a CLIST. **This type of use should only be for "TTY" type terminals**. To direct the **$AVRPULL** program to read SYSIN input instead of using TGETs, supply a PARM of "BATCH" on the call statement. To suppress banners when output is to sysout and also read from SYSIN, provide a **PARM** of "**BATCH,NOBANNER**". The CLIST "**$AVRSTTY**" can be found in INSTLIB. The CLIST should be moved to your CLIST library as member "**$AVRSTTY**" and must be modified to provide the correct dataset name defined to the DDNAME "$AVRLOCK". The CLIST (see the next figure) can then be executed in any one of the following ways:

%$AVRSTTY — This option routes the "RSLOCAL" job selection output to the TSO terminal.

%$AVRSTTY H(Y) — This option routes the "RSLOCAL" job selection output to SYSOUT "A" and makes it held output.

%$AVRSTTY H(N) — This option routes the "RSLOCAL" job selection output to SYSOUT "A" and makes the output available for print.

%$AVRSTTY H(Y) DEST (RMT1) — This option routes the "RSLOCAL" job selection output to Sysout "A" destination "RMT1" and makes it held output.

%$AVRSTTY H(N) S(T) — This option routes the 'RSLOCAL" job selection output to Sysout "T" and makes the output available for print.

All of the distributed ways the **$AVRSTTY** clist is supplied route the "SYSPRINT" output to the TSO terminal.

The following is an example CLIST, Sample JCL - TSO Foreground, for the $AVRPULL program execution. This JCL is also provided in member $AVRSTTY in the INSTLIB.

```
PROC  0  DEBUG(NO) HOLD(T) SYSOUT(A) DEST(LOCAL)
/*                                                                 */
/****NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***/
/*                                                                 */
/*THIS INTERFACE SHOULD BE USED FOR TTY OR LINE OUTPUT TERMINALS.  */
/*                                                                 */
/*  USE THE $AVRSCP COMMAND PROCESSOR UNDER TSO FULLSCREEN MODE    */
/*  OR THE SPF/ISPF INTERFACE UNDER SPF/ISPF.                      */
/*                                                                 */
/****NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***NOTE***/
/*****************************************************************/
/*                                                               */
/*     IF YOU WISH TO USE THE VTAM PRINT INTERFACE WITH          */
/*     THIS CLIST, YOU MUST UNCOMMENT ALL LINES THAT             */
/*     HAVE "$AVRVPRT" IN THEM.                                  */
/*                                                               */
/*****************************************************************/
/*                                                               */
CONTROL NOFLUSH END(ENDO)
IF &SUBSTR(1:1,&DEBUG) = Y THEN CONTROL LIST CONLIST SYMLIST
CONTROL NOMSG
         FREE FI(RSLOCAL)ATTRLIST($AVRATTR)
         FREE FI($AVRLOCK)
/*       FREE FI($AVRVPRT)     */
CONTROL MSG
/*****************************************************************/
/*     THE FOLLOWING ALLOCATION MUST BE ALLOCATED                */
/*     TO YOUR $AVRLOCK VSAM RRDS DATASET.                       */
/*****************************************************************/
           ALLOC FI($AVRLOCK) DS('$AVRS.REL#.LOCK') SHR
/*****************************************************************/
/*                                                               */
/*     THE PREVIOUS ALLOCATION MUST BE                           */
/*     ALLOCATED TO YOUR $AVRLOCK VSAM DATASET.                  */
/*****************************************************************/
/*       ALLOC FI($AVRVPRT) DS('YOUR.VTAM.PRINT.RRDS') SHR       */
/*                                                               */
         ALLOC FI(SYSPRINT) DS(*)
IF &SUBSTR(1:1,&HOLD) = N THEN DO
         ALLOC FI(RSLOCAL) SYSOUT(&SYSOUT) DEST(&DEST)
              GOTO CALLPULL
   ENDO
IF &SUBSTR(1:1,&HOLD) = Y THEN DO
         ALLOC FI(RSLOCAL) SYSOUT(&SYSOUT) HOLD DEST(&DEST)
              GOTO CALLPULL
   ENDO
         ATTR $AVRATTR BLKSIZE(262) LRECL(258) RECFM(V)
         ALLOC FI(RSLOCAL) DS(*) USING($AVRATTR)
CALLPULL: CALL '$AVRS.REL#.LOADLIB($AVRPULL)'
CONTROL NOMSG
         FREE FI(RSLOCAL)ATTRLIST($AVRATTR)
         FREE FI($AVRLOCK)
/*       FREE FI($AVRVPRT)     */
```

Figure 51 $AVRPULL Foreground Program CLIST Example

Selection Criteria

$AVRPULL Input parameters may be expressed as a SYSIN (DDNAME) input data stream or an input dataset with the DDNAME of $AVRPCTL. The Input parameters are all keyword driven and free form. Each new statement must start in column 1. If a continuation is required the previous line must end in a comma and the next keyword must start in position 2 or greater of the next line. Each parameter must be completed within the line in which it starts and cannot be split across multiple lines.

Several of these keyword parameters can have their values **masked**. A **masked** value means a string of alpha-numeric or national characters that may be from 1 to 8 characters in length. Any position may have its significance overridden by using the character "*****". When no additional significant characters are needed to compose a name mask, a "**-**" may be used in lieu of filling the remaining positions with asterisks. A "." (period) in the mask denotes an absolute name. The remainder of the name field is filled with trailing blanks. This mask character, ".", prevents a name shorter than 8 characters from being generic. Any name given is considered **generic** unless it is completely specified for its entire length or if masking characters are used to compose the name. All work matching that type of request will be selected if all other selection criteria is met.

Examples of these control statements follow the explanation of all the possible keywords.

ONEMATCH= This parameter is used by multiple programs: $AVRPULL, $AVRSELR, $AVRTAPE. This parameter is a program control parameter that controls how many times a selection control statement(s) can be used. If used, this parameter may only appear once in the input stream per execution of the program and should be on a record by itself. Any selection criteria that is on the same record as this parameter is ignored.

YES is designed to work with a **fully qualified** NAME=parameter and any other use may result in excessive output. **YES** will select only the first work that is matched by each control statement. With ONEMATCH=YES, after all control statements are matched, the program completes successfully, even if the input file is not at EOF.

EX: ONE=YES
TYPE=JN,NAME=C106PAY2 (selects only first occurrence of this fully qualified job).
TYPE=JN,NAME=C203BB. (selects only first occurrence of this fully qualified job. **NOTE:** The ending period makes this fully qualified).
TYPE=JN,NAME=C203BB (selects first occurrence of every job that begins with C203BB - C203BB, C203BBA, C203BBB, etc.).

ONEMATCH=Y and LAST=Y are mutually exclusive parameters. If used together, the results are unpredictable.
NO will select all work that matches each control statement.
Shortest Abbreviation: ON
Example: **`ONE=YES | ON=Y`**
Default: **`ONE=NO`**

TYPE= Identifies the kind of work to be selected. The valid work types are:

JN - Batch Jobs
TN - TSO Users
SN - Started Tasks
SL - SYSLOG (Today's date if SD is not specified for Syslog)
ST - Status Reports
FI - Fiche Indices
AH - Archive History
FH - Fiche History

Multiple work types may be selected by enclosing all desired types in parentheses and separating each individual type with a comma. This parameter determines which base keys in the data base are allowed for selection except for **AH** and **FH**. For Archive and Fiche History, this parameter indicates selection is based off individual data lines instead of base keys.
Shortest Abbreviation: TY
Example: **`TY=JN | TYPE=(JN,SN,AH)`**
Default: **`TYPE=(JN,TN,SN)`**

NAME= The highest level name of the work. This field can be **masked**. It is possible to use this field for SYSLOG selection if you follow $AVRS SYSLOG naming conventions: i.e. SLxxxxy where "xxxx" is the system id (ESAM) and "y" is the sysout class (N)
Valid for Types: JN, TN, SN, AH, FH, SL
Shortest Abbreviation: NA
Example: **`NAME=RIC- | NA=JOBNAME. | NAM=A*BCD |NAME=SLESAMN`**
Default: **`NAME=-`**

SDATE= This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: SD
Example: **`SD=99001 | SD=* | SD=-30`**
Default: **`Lowest possible, except SYSLOG which uses today.`**

EDATE= The ending date of the work to be selected. This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for todays date or a negative number for today minus a specified number of days. For SYSLOG segments this date determines when printing of SYSLOG stops. For Status Reports and Fiche Indices this date is the Recap Period ending date of the work. For all other work this is the ate the work ended execution.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: ED
Example: **`ED=2001001 | ED=* | ED=-30`**
Default: **`Highest possible, except SYSLOG which uses today.`**

STIME= The starting time of the work to be selected. For SYSLOG this is the time at which printing begins. Status Reports and Fiche Indices use the Recap Period start time to determine selection. For all other work this is the time the work started execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). An example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.

Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: ST
Example: **`ST=1000 | ST=1500`**
Default: **`00:00 (midnight)`**

ETIME= The ending time of the work to be selected. For SYSLOG this is the time at which printing stops. Status Reports and Fiche Indices use the Recap Period end time to determine selection. For all other work this is the time the work ended execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). An example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: ET
Example: **`ET=2300 | ET=1700`**
Default: **23:59 (11:59 p.m)**

NOTE: If both start and end times are specified, the work must have either started, ended or both started and ended within the specified time period. If the work was executing across both bounds of the time period, it will not qualify for selection.

SELRULE Date and Time selection may now be based on the job end time only, the start time only, both the start and end times together or any start or end time (default). This parameter is valid for all selection types. All four date and time values (SD, ST, ED, ET) must be specified to use SELRULE. Failure to supply dates and times will cause the SELRULE to be ignored or use the default dates and times for the TYPE of work specified.

START Only jobs that *started* between the start date/time and the end date/time from above will be selected.

END Only jobs that *ended* between the start date/time and the end date/time from above will be selected.

BOTH Only jobs that both *started* and *ended* between the start date/time and the end date/time from above will be selected.

blank Ignore SELRULE
Valid for Types: All
Examples: SELRULE=END | SELRULE=BOTH
Default: Any entry other than "Start", "End" or "Both" will be ignored.

RANGE= A **Yes** or **No** value that governs how the starting and ending date and time fields are used. **Yes** means all work within the starting date and time and the ending date and time range is selected. **No** means all work within the start date and end date range *that fell within the start and end time of each day is selected* a SLICE of time. When RANGE=No end time must be greater than start time.
Valid for Types: **JN, TN, SN, SL, ST, FI, AH, FH**
Shortest Abbreviation: RA
Example: **`RA=YES | RA=N`**
Default: **`Yes`**

LAST= The **YES** value means to find the last execution of the work indicated by the TYPE and NAME= selection criteria on this control statement. For best results, NAME= should be fully qualified, not generic. SDATE and EDATE may be used with LAST to find the last execution of the work for specific dates. **NO** means the parameter is ignored. ONEMATCH=Y and LAST=Y are mutually exclusive parameters. If used together, the results are unpredictable.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: LA
Example: **`LA=YES | LA=N`**
Default: **`No`**

ACKNOWLEDGMENT=
The acknowledgment code of the work. This six byte character field can be masked. It can also contain two special values: **N/A** and **@ALL@**. **N/A** selects all work that does not have any code and **@ALL@** selects all work that has a code.
Valid for Types: JN, TN, SN
Shortest Abbreviation: AC
Example: **`AC=123456 | AC=1*PR. | AC=N/A`**
Default: **`None`**

JO= Job Owner of the work. This is an eight byte field that can be masked. Work is considered "owned" if this field matches either the NOTIFY or USER of the work.
Valid for Types: JN or when TYPE not specified
Shortest Abbreviation: JO
Example: **`JO=RIC- | JOB=JOBOWNR. | JOB=AB*CD`**
Default: **`None`**

ID= The System Id of the SYSLOG to be selected. This is a four byte field that can be masked. All SYSLOG accumulated by $AVRS has a seven byte "constructed" name as follows: 'SL' in cc 1 - 2; System Id in cc 3 - 6 and the sysout class in cc 7. This field matches against the System Id or SID part of a SYSLOG name.
Valid for Types: SL
Shortest Abbreviation: ID
Example: **`ID=IPO1 | ID=E*A`**
Default: **`None`**

CCODE= The Completion or Condition code of the work. This is a seven byte field that can be masked. “NZRSPN” will select respun work with a non-zero condition code.
Valid for Types: JN, TN, SN, AH, FH
Shortest Abbreviation: CC
Example: **`CC=PARTIAL | CC=***0000`**
Default: **`None`**

NUMBER= The JES number of the work. Leading zeroes do not have to be specified in this five byte numeric field, but the number specified must exactly match the work for selection.
Valid for Types: JN, TN, SN, SL, AH, FH
Shortest Abbreviation: NU
Example: **`NU=5 | NU=967`**
Default: **`None`**

STEPNAME= The step name of the work. This eight byte field can be masked.
Valid for Types: JN, TN, SN
Shortest Abbreviation: STE
Example: **`STE=- | STEP=STEP.`**
Default: **`None`**

PROCSTEP= The procedure step name of the work. This eight byte field can be masked.
Valid for Types: JN, TN, SN
Shortest Abbreviation: PR
Example: **`PR=- | PR=PROC-`**
Default: **`None`**

DDNAME= The DDNAME name of the work. This eight byte field can be masked. In addition to actual DDNAMES, five special values are allowed; JEJCLIN, JESLOG, JESJCL, SYSMSG AND JESDS. JESDS selects all JES datasets from the work. The other special values selects only their respective JES data sets.
Valid for Types: JN, TN, SN
Shortest Abbreviation: DD
Example: **`DD=- | DD=SYSMSG | DD=SY*P-`**
Default: **`None`**

LOCATE= Defines an argument to be found in the work. In SYSLOG, each line containing the argument is printed including any multi-line or continued messages requested. All other work prints a special header before the line showing where the argument was found in the work. This is a 30 byte field. If the argument contains blanks, the argument must be enclosed in parentheses, single or double quotes. **LUNAME** forces the entire file(s) where the argument is found to print without the special headers. This parameter is mutually exclusive with **FA=** and **SCAN=**. This parameter is mutually exclusive with LUNAME for TYPE=SL.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: LO
Example: **`LO=TEXT | LO='A LOT OF TEXT'`**
Default: **`None`**

FA= or **SCAN=** Defines an argument to be found or scanned for in the work. In SYSLOG, each line continuing the argument is printed including any multi-line or continued messages requested. For all other types of work, if the argument is found, the entire work is printed. If the **DDNAME=** selection parameter was also used for this scan, only the file is printed instead of the entire work. This is a 30 byte field. If the argument contains blanks, the argument must be enclosed in parentheses, single or double quotes. This parameter is mutually exclusive with **LO=**. This parameter is mutually exclusive with LUNAME for TYPE=SL.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: FA or SC
Example: **`FA=TEXT | SC=(A LOT OF TEXT)`**
Default: **`None`**

SSTART= The starting position for any type of scan (LO, FA or SC). This is a three byte numeric field that can range from 1 through 255, inclusive. The first printable position is position 1. Leading zeroes are not required.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: SS
Example: **`SS=5 | SS=100`**
Default: 1

SRANGE= The range from the starting position that is scanned for the argument. This is a three byte numeric field ranging from 1 through 255, inclusive. All scans (LO, FA or SC) try to locate the argument from the starting position for this amount. If the argument starts within this range, it is considered found.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: SS
Example: **`SR=5 | SR=100`**
Default: **`255`**

ANNOTATIONS=

Annotations print switch. This governs how annotation print is done. **Yes** means print the annotations (if present) and the selected work. **No** means do not print annotation, just the selected work. **Only** means only print annotations for selected work. Acknowledgment codes are included as part of an annotation.
Valid for Types: JN, TN, SN
Shortest Abbreviation: AN
Example: **`AN=Y | AN=ONLY | ANN=NO`**
Default: **`Yes`**

MSGCLASS= MSGCLASS checks the JOB level class indicator, the first file accumulated from the job exstablishes the field, (KYCLASS in the $KSDSKEY record). The MSGCLASS is the class that is displayed on JOB level directories. This is a one byte field. CLASS and MSGCLASS are mutually exclusive for $AVRPULL.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: MS
Example: **`MA=A | MSGC=5`**
Default: **`None. All classes are processed.`**

CLASS= CLASS checks the individual sysout class of the FILE, (FICLASS in the $KSDS101 record). Only work accumulated in this sysout class is selected by this parameter. This is the class that displays on FILE level directories. This is a one byte field. MSGCLASS and CLASS are mutually exclusive for $AVRPULL.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: CL
Example: **`CL=5 | CL=A`**
Default: **`None. All classes are processed.`**

FORM= Sysout Forms Id. Work must generically match this value to be selected. This field is a maximum of eight characters. This parameter is not valid for SYSLOG.
Valid for Types: JN, TN, SN
Shortest Abbreviation: FO
Example: **`FO=STD5 | FO=STD`**
Default: **`None`**

FCB= Sysout Forms Control Buffer. Work must match this value to be selected. This field is a maximum of four bytes. This parameter is not valid for SYSLOG.
Valid for Types: JN, TN, SN
Shortest Abbreviation: FC
Example: **`FC=STD1 | FC=STD`**
Default: **`None`**

UCS= Sysout Universal Character Set. Work must match this value to be selected. This field is a maximum of four bytes. This parameter is not valid for SYSLOG.
Valid for Types: JN, TN, SN
Shortest Abbreviation: UC
Example: **`UC=TN | UC=PN`**
Default: **`None`**

AHWORK= Archive History Work type. This value selects the types (JN, SN, TN, or SL) of Archive History. This parameter requires TYPE=AH selections. Multiple types may be selected by enclosing all desired types in parentheses and separating them with commas.
Valid for Types: AH
Shortest Abbreviation: AHW
Example: **`AHW=JN | AHW=(JN,SN)`**
Default: **`None`**

AHTYPE= Allows Archive History selection for either non-zero completion codes (**NZHIST**) or respun work (**REHIST**).
Valid for Types: AH
Shortest Abbreviation: AHT
Example: **`AHT=NZHIST | AHT=REHIST`**
Default: **`None`**

LUNAME= The name of a **VTAM Print** defined printer where this selection will print. This parameter requires that the DD statement $AVRVPRT be present in the execution JCL of $AVRPULL. The maximum length of this field is eight bytes. LUNAME is mutually exclusive with LO/SC/FA when TYPE=SL.
Valid for Types: JN, SN, TN, SL, ST, FI
Shortest Abbreviation: LU
Example: **`LU=P45F`**
Default: **`None`**

VPCLASS= A sysout class for this **VTAM Print** bundle. All work sent to **VTAM Print** from a selection record with this parameter will have this sysout class shown in the **VTAM Print** directories. This parameter is ignored if the LUNAME parameter is not used or its requirements are not met.
Valid for Types: JN, SN, TN, SL, ST, FI
Shortest Abbreviation: VPC
Example: **`VPC=A | VPCLASS=3`**
Default: **`The accumulation sysout class`**

VPFORM= The FORMS ID of this **VTAM Print** bundle. All work sent to **VTAM Print** from a selection record with this parameter will have this Sysout Form ID shown in the **VTAM Print** directories. This parameter is ignored if the LUNAME parameter is not used or its requirements are not met.
Valid for Types: JN, SN, TN, SL, ST, FI
Shortest Abbreviation: VPFO
Example: **`VPFO=STD | VPFORM=FRM1`**
Default: **`None`**

VPFCB= The FCB ID of this **VTAM Print** bundle. All work sent to **VTAM Print** from a selection record with this parameter will have this Sysout FCB ID shown in the **VTAM Print** directories. This parameter is ignored if the LUNAME parameter is not used or its requirements are not met.
Valid for Types: JN, SN, TN, SL, ST, FI
Shortest Abbreviation: VPFC
Example: **`VPFC=STD5 | VPFCB=STD`**
Default: **`None`**

VPHOLD= The initial HELD state of this **VTAM Print** bundle. This parameter can be **Yes** or **No**. All work sent to **VTAM Print** from a selection record with this parameter will have held state set to this value. This parameter is ignored if the LUNAME parameter is not used or its requirements are not met.
Valid for Types: JN, SN, TN, SL, ST, FI
Shortest Abbreviation: VPH
Example: `VPH=YES | VPHOLD=NO`
Default: `None`

REP= This is only valid for the Foreground execution of the program. This command causes the last command to be reissued.
Valid for: TSO Foreground execution only
Default: None

Selection Examples

`TYPE=(JN,SN),NAME=PAY,SD=00001,ED=2000003,ST=0800,ET=1700`
This example selects all jobs and started tasks whose name start with PAY. This work would have to start, end or both start and end in the time period of 2000001.0800 through 2000003.1700.

`TY=SL,ID=ESAM,SD=-1,ED=*,ST=0700,ET=0900,RANGE=NO`
All SYSLOG with a system id of ESAM dated between yesterday at 7:00 a.m. through 9:00 a.m. and today between 7:00 a.m. and 9:00 a.m. would select from this example.

```
TYPE=JN,NAME=LOTS,STEP=ST*1,DD=PR1FILE.,
 SCAN="MY TEXT ARGUMENT",SS=10,SR=5,ACK=@ALL@,
 CL=7
```

The work for this example must be a batch job, start with the name LOTS, have a stepname starting with ST and a 1 in the fourth stepname position, have a DDNAME of exactly PR1FILE, be acknowledged and have been accumulated in class 7. The sysout from DDNAME PR1FILE will be scanned for the string MY TEXT ARGUMENT starting in position 10 through 15. If the argument is found, All sysout from DDNAME PR1FILE is printed.

`TYPE=JN,NAME=COMPILE,LAST=YES`
The last executed batch job with a job name COMPILE will be selected.

`TYPE=JN,NAME=COMPILE,LAST=YES,SDATE=2000325,EDATE=2000325`
The last batch job named COMPILE that started and ended execution on November 20,2000 will be selected.

Sample Work Selection List and Error Report

```
RPTID=$AVRPULL-REPORT        WORK SELECTION LIST AND ERROR REPORT           TIME=10:50:02   DATE=2002.128     PAGE     1
<<< $AVRS - USER TITLE LINE >>>                    **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                   **** SEA INC   1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010

                         TYPE=JN,NA=D-
                         TYPE=SL,SDATE=1999115,STIME=1601,EDATE=1999115,ETIME=23591,SELRULE=BOTH
RPTID=$AVRPULL-REPORT    WORK SELECTION LIST AND ERROR REPORT              TIME=10:50:02   DATE=2002.128       PAGE   2
<<< $AVRS - USER TITLE LINE >>>                    **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                   **** SEA INC   1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010

$AVR719I - REQUEST SELECTED  DBRPRPT (J0027881) 15 PAGE(S), 623 LINE(S), FOR SELECTION 1
$AVR719I - REQUEST SELECTED  DBRPRPT (J0027905) 8 PAGE(S), 237 LINE(S), FOR SELECTION 1
$AVR715I - PULL COMPLETE     FOR SELECTION 1
$AVR719I - REQUEST SELECTED  SL77777 (S0027656) 21 PAGE(S),91,095 LINE(S), FOR SELECTION 2
$AVR715I - PULL COMPLETE     FOR SELECTION 2
                                        * * *   E N D   O F   R E P O R T   * * *
```

Chapter 2

$AVRSELR - Restore Archived Work For Viewing

Restoring Work to the $AVRS Data base — The **$AVRSELR** program is used to restore work from $AVRS backup or unload files to the online data base. Selections of the work to be restored is requested by control statements read from the DDNAME SYSIN. All work is restored using the original name, number, start/end dates and times, attributes, etc. However, the accumulation date/time of the work will be the restore date and time.

Backup/Archive files from previous releases of $AVRS are accepted as input on DDNAME SYSUT1 in this release of **$AVRSELR** and mixed release files may be concatenated as a single logical input file.

$AVRSELR checks the available space in the data base for all work about to be restored. If the work about to be restored needs more space than exists in the data base, then message **$AVR594** is issued. **$AVRSELR** continues processing by skipping the work detailed in the **$AVR594** message. The restore job will end with a return code of no less than 8, if work could not be restored for space limitations.

The input **$AVRSELR** selection criteria is read from DDNAME SYSIN and the output responses are written to DDNAME SYSPRINT. A sample of the output in the SYSPRINT is included at the end of this topic. The selected work is written directly to the RRDS file(s) of the database. The KSDS transaction keys are written to the JES spool with the first ACCMCLS (ACCDEST and ACCFORM if applicable) and a writer name of $AVRSELR. The KSDS transaction keys must be accumulated before the restore will be accessible.

Return Codes

$AVRSELR sets the following return codes.

- Zero - normal processing
- Four - processing completed but no work was selected
- Eight - processing terminated with invalid parameter values or adequate space was not available in the database
- Twenty - all parameters failed validation or none were specified.

A modification must be made to the online JCL procedure, ($AVRREST in the Install library), to supply your dataset names on the DD statements. The restore procedure should be moved to your systems proclib. This will prevent problems during future maintenance or release upgrades.

WARNING * WARNING *** WARNING**

It is imperative that the out put from any job that runs the $AVRSELR restore program is not purged prior to the $AVRS accumulator picking up the KSDS transaction keys for the restored work.

If the $AVRSELR transaction keys are not put into the data base by the accumulator, the work restored into the RRDS DATA file(s) and the space taken by the work is lost. The work will not appear in the job directory until the KSDS transaction keys are accumulated.

In order to recover the lost space in the RRDS DATA file(s), the $AVRUNLD program (fully explained in the Installation Manual) must be run followed by data base deletions, re-definitions, and running $AVRFMRR using the output file created by $AVRUNLD or by running the $AVRCLEN program (described in the Installation Manual).

The $AVRSELR program issues ENQ requests as follows:

Major Name=$AVMAINT, Minor Name=*LOCK DSNAME*
Major Name=$ELRCLEN, Minor Name=*LOCK DSNAME*

Sample Batch JCL

The following is sample JCL for the *Batch* **$AVRSELR** execution. This JCL is also provided in "*$AVRS.rel#.INSTLIB($AVRSELR)*".

```
//$AVRSELR JOB (ACCOUNTING INFO),'$AVRSELR',CLASS=X,
//             MSGCLASS=X,NOTIFY=&SYSUID
//JOBLIB   DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR*** $AVRS LOADLIB ***
//STEP1    EXEC PGM=$AVRSELR,REGION=768K
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//ABNLIGNER DD DUMMY
//*
//* BATCH SELECTIVE RESTORE
//*
//$AVRLOCK DD DSN=$AVRS.REL#.LOCK,DISP=SHR
//*
//* $AVRLOCK DD IS ALLOCATED TO THE LOCK FILE
//*
//SYSUT2  DD  DSN=&SYSUT2,DISP=(,DELETE),UNIT=SYSALLDA,
//    SPACE=(CYL,(5,1))
//*
//* SYSUT2 DEFINES A SORT WORK FILE
//*
//SYSUT1  DD  DSN=    ,DISP=OLD
//        DD  DSN=    ,UNIT=AFF=SYSUT1
//        DD  DSN=    ,UNIT=TAPE
//        VOL=SER=
//*
//*   SYSUT1 DEFINES FILE(S) CREATED BY:
//*    1. $AVRBKUP OR $AVRUNLD, OR
//*    2. $AVRBKUP FROM A PREVIOUS RELEASE(R3.0A OR LATER)OF $AVRS,OR
//*    3. $AVRBKUP THAT IS NOT CATALOGUED
//*
//* SYSIN  DD IS THE INPUT SOURCE FOR THE SELECTION PARAMETERS
//*
//SYSIN    DD  *
/*
/
```

Figure 52 $AVRSELR Example JCL Batch

In the example above, the user would supply the load library, lock file, SYSUT 1, and SYSIN.

Sample JCL Procedure

The following is a sample procedure used to build JCL for an online restore from the AH (Archive History) directory. SYSUST1 and SYSIN data are supplied from online information. This sample procedure is also provided in "$AVRS,rel#.INSTLIB ($AVRREST)".

```
//$AVRREST PROC RRTCLS='*'
//             DUMPCLS='*'
//*
//SELRSTEP EXEC PGM=$AVRSELR,REGION=768K
//*
//STEPLIB DD DSN=$AVRS.REL#.LOADLIB,DISP=SHR**$AVRS LOADLIB**
//*
//SYSDUMP DD SYSOUT=&DUMPCLS
//*
//SYSPRINT DD SYSOUT=&RPTCLS
//*
//*
//$AVRLOCK DD DSN=$AVRS.REL#.LOCK,DISP=SHR     <==================
//*
//*     $AVRLOCK DD IS ALLOCATED TO THE LOCK FILE
//*
//*     $AVRRRDS DD IS NO LONGER RECOGNIZED
//*     $AVRKSDS DD IS NO LONGER RECOGNIZED
//*
//SYSUT2 DD DISP=(,DELETE,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,2))
//*
//SYSUT1 DD DCB=(RECFM=VB,BLKSIZE=32760)
```

Figure 53 $AVRREST Example Procedure

A $AVRREST procedure should be added to your systems proclib by the $AVRS Administrator and staff. The name of this procedure should match the JOBPROC value in your $AVRFSEA module.

Selection Criteria

$AVRSELR SYSIN parameters are all keyword driven and free form. Each new statement must start in column 1. If a continuation is required the previous line must end in a comma and the next keyword must start in position 2 or greater of the next line. Each parameter must be completed within the line in which it starts and cannot be split across multiple lines.

Several of these keyword parameters can have their values **masked**. A **masked** value means a string of alpha-numeric or national characters that may be from 1 to 8 characters in length. Any position may have its significance overridden by using the character "*****". When no additional significant characters are needed to compose a name mask, a "**-**" may be used in lieu of filling the remaining positions with asterisks. A "." (period) in the mask denotes an absolute name. The remainder of the name field is filled with trailing blanks. This mask character, ".", prevents a name shorter than 8 characters from being generic.

Any name given is considered **generic** unless it is completely specified for its entire length or if masking characters are used to compose the name. All work matching that type of request will be selected if all other selection criteria is met.

Examples of these control statements are shown at the end of the selection control parameters and before a copy of the $AVRSELR report.

ONEMATCH= The parameter is used by multiple programs: $AVRPULL, $AVRSELR, $AVRTAPE. This parameter is a program control parameter that controls how many times a selection control statement(s) can be used. If used, this parameter may only appear once in the input stream per execution of the program and should be on a record by itself. Any selection criteria that is on the same record as this parameter is ignored.

YES is designed to work with a **fully qualified** NAME=parameter and any other use may result in excessive output. **YES** will select only the first work that is matched by each control statement. With ONEMATCH=YES, after all control statements are matched, the program completes successfully, even if the input file is not at EOF.

EX: ONE=YES
TYPE=JN,NAME=C106PAY2 (selects only first occurrence of this fully qualified job).

TYPE=JN,NAME=C203BB. (selects only first occurrence of this fully qualified job. **NOTE:** The ending period makes this fully qualified).

TYPE=JN,NAME=C203BB (selects first occurrence of every job that begins with C203BB - C203BB, C203BBA, C203BBB, etc.).

NO will select all work that matches each control statement.
Shortest Abbreviation: ON
Example: **`ONE=YES | ON=Y`**
Default: **`ONE=NO`**

TYPE= Identifies the kind of work to be selected. The valid work types are:

- **JN** - Batch Jobs
- **TN** - TSO Users
- **SN** - Started Tasks
- **SL** - SYSLOG (Today's date if SD is not specified for Syslog)
- **ST** - Status Reports
- **FI** - Fiche Indices

Multiple work types may be selected by enclosing all desired types in parentheses and separating each individual type with a comma. This

parameter determines which base keys in the data base are allowed for selection.
Shortest Abbreviation: TY
Example: TY=JN | TYPE=(JN,SN)
Default: **`TYPE=(JN,TN,SN)`**

NAME= The highest level name of the work. This field can be **masked**. It is possible to use this field for SYSLOG selection if you follow $AVRS SYSLOG naming conventions: i.e. SLxxxxy where "xxxx" is the system id (ESAM) and "y" is the sysout class (N).
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: NA
Example: **`NAME=RIC- | NA=JOBNAME. | NAM=A*BCD |NAME=SLESAMN`**
Default: **`NAME=-`**

SDATE= The starting date of the work to be selected. This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days. This is the starting date of the segment for SYSLOG. For Status Reports and Fiche Indices this date is the Recap Period start date of the work. For all other work this is the date the work started execution.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: SD
Example: **`SD=02001 | SD=* | SD=-30 | SD=2000005`**
Default: **`Lowest possible, except SYSLOG which uses today.`**

EDATE= The ending date of the work to be selected. This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days. For SYSLOG segments this date is the ending date of the segment. For Status Reports and Fiche Indices this date is the Recap Period ending date of the work. For all other work this is the date the work ended execution.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: ED
Example: **`ED=96001 | ED=* | ED=-30 | ED=1996001`**
Default: **`Highest possible, except SYSLOG which uses today.`**

STIME= The starting time of the work to be selected. For SYSLOG this is the starting time of the segment. Status Reports and Fiche Indices use the Recap Period start time to determine selection. For all other work this is the time the work started execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). Example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: ST
Example: **`ST=1000 | ST=1500`**
Default: **`00:00 (midnight)`**

ETIME= The ending time of the work to be selected. For SYSLOG this is the ending time of the segment. Status Reports and Fiche Indices use the Recap Period end time to determine selection. For all other work this is the time the work ended execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). Example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: ET
Example: **`ET=2300 | ET=1700`**
Default: 23:59 (11:59 p.m)

NOTE: If both start and end times are specified, the work must have either started, ended or both started and ended within the specified time period. If the work was executing across both bounds of the time period, it will not qualify for selection.

SELRULE= Date and Time selection may now be based on the job end time only, the start time only, both the start and end times together or any start or end time (default). This parameter is valid for all selection types. All four date and time values (SD, ST, ED ET) must be specified to use SELRULE. Failure to supply dates and times will cause the SELRULE to be ignored or use the default dates and times for the TYPE of work specified.

START Only jobs that *started* between the start date/time and the end date/time from above will be selected.

END Only jobs that *ended* between the start date/time and the end date/time from above will be selected.

BOTH Only jobs that both *started* and *ended* between the start date/time and the end date/time from above will be selected.

blank Ignore SELRULE

Valid for Types: All
Examples: SELRULE=END | SELRULE=BOTH
Default: Any entry other than "Start", "End" or "Both" will be ignored.

RANGE= A **Yes** or **No** value that governs how the starting and ending date and time fields are used. **Yes** means all work within the starting date and time and the ending date and time range is selected. **No** means all work within the start date and end date range *that fell within the start and end time of each day is selected* a SLICE of time.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: RA
Example: **`RA=YES | RA=N`**
Default: **`Yes`**

ACKNOWLEDGMENT=
The acknowledgment code of the work. This six byte character field can be **masked**. It can also contain two special values: **N/A** and **@ALL@**. **N/A** selects all work that does not have any code and **@ALL@** selects all work that does.
Valid for Types: JN, TN, SN
Shortest Abbreviation: AC
Example: **`AC=123456 | AC=1*PR. | AC=N/A`**
Default: **`None`**

JO= Job Owner of the work. This is an eight byte field that can be **masked**. Work is considered "owned" if this field matches either the NOTIFY or USER of the work.
Valid for Types: JN or when TYPE not specified
Shortest Abbreviation: JO
Example: **`JO=RIC- | JOB=JOBOWNR. | JOB=AB*CD`**
Default: **`None`**

ID= The System Id of the SYSLOG to be selected. This is a four byte field that can be **masked**. All SYSLOG accumulated by $AVRS has a seven byte "constructed" name as follows: 'SL' in cc 1 - 2; System Id in cc 3 - 6 and the sysout class in cc 7. This field matches against the System Id or SID part of a SYSLOG name.
Valid for Types: SL
Shortest Abbreviation: ID
Example: **`ID=IPO1 | ID=E*A`**
Default: **`None`**

CCODE= The Completion or Condition code of the work. This is a seven byte field that can be **masked**. "NZRSPN" will select respun work with a non-zero condition code.
Valid for Types: JN, TN, SN
Shortest Abbreviation: CC
Example: `CC=PARTIAL | CC=***0000`
Default: `None`

NUMBER= The JES number of the work. Leading zeroes do not have to be specified in this five byte numeric field, but the number specified must exactly match the work for selection.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: NU
Example: `NU=5 | NU=967 | NU=7490038`
Default: `None`

Selection Examples

`TYPE=(JN,SN),NAME=PAY,SD=96001,ED=1996003,ST=0800,ET=1700`
This example selects all jobs and started tasks whose name starts with PAY. This work would have to start, end or both start and end in the time period of 1996001.0800 through 1996003.1700.

`TY=SL,ID=ESAM,SD=-1,ED=*,ST=0700,ET=0900,RANGE=NO`
All SYSLOG with a system id of ESAM dated between yesterday at 7:00 a.m. through 9:00 a.m. and today between 7:00 a.m. and 9:00 a.m. would select from this example.

```
TYPE=JN,NAME=LOTS,
 ACK=N/A
```
The work for this example must be a batch job, start with the name LOTS and NOT be acknowledged.

```
TYPE=JN,NAME=LOTS,STEP=ST*1,DD=PR1FILE.,
 SCAN="MY TEXT ARGUMENT",SS=10,SR=5,ACK=@ALL@,
 CL=7
```
The work for this example must be a batch job, start with the name LOTS, have a stepname starting with ST and a 1 in the fourth stepname position, have a DDNAME of exactly PR1FILE, be acknowledged and have been accumulated in class 7. The sysout from DDNAME PR1FILE will be scanned for the string MY TEXT ARGUMENT starting in position 10 through 15. Each line that contains the argument is printed along with a header line detailing where the line was found. By default, the annotations of this work will also be selected.

`TYPE=ST,ST=0700,SD=2000010,ET=0700,ED=2000011`
This example restores a STATUS HISTORY report with the Recap Periods of Jan10, 2000 at 7am through January 11, 2000 at 7am. Start and end date and times for SATUS HISTORY are recap periods **NOT** creation/run date and time. These dates are displayed on the TYPE=ST panel.

Sample Selective Restore Status Report

```
RPTID=$AVRSELR                 SELECTIVE RESTORE STATUS REPORT              TIME=11:11:47   DATE=2002.128 PAGE     1
<<< $AVRS - USER TITLE LINE >>>                 **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                **** SEA INC   1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010

                        ONE=YES
                        TYPE=JN,NAME=A112-
                        TYPE=SN,NU=9362
                        TYPE=TN,NAME=C112
RPTID=$AVRSELR                 SELECTIVE RESTORE STATUS REPORT              TIME=11:11:47   DATE=2002.128   PAGE   2
<<< $AVRS - USER TITLE LINE >>>                 **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                **** SEA INC   1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010

$AVR719I - REQUEST SELECTED  A112JOBA (J000723) 3 PAGE(S), 38 LINE(S), FOR SELECTION 1
$AVR719I - REQUEST SELECTED  C112 (T000041) 1 PAGE(S), 2 LINE(S), FOR SELECTION 3
$AVR719I - REQUEST SELECTED  $MS40A (S009362) 3 PAGE(S), 515 LINE(S), FOR SELECTION 2
$AVR715I - PULL COMPLETE    FOR SELECTION 1
$AVR715I - PULL COMPLETE    FOR SELECTION 2
$AVR715I - PULL COMPLETE    FOR SELECTION 3
$AVR463I - NUMBER OF LINES RESTORED TO DATA BASE                555
$AVR464I - NUMBER OF PAGES RESTORED TO DATA BASE                  7
$AVR465I - NUMBER OF FILES RESTORED TO DATA BASE                  7
$AVR466I - NUMBER OF JOB KEYS RESTORED                            1
$AVR467I - NUMBER OF STC KEYS RESTORED                            1
$AVR468I - NUMBER OF TSU KEYS RESTORED                            1
$AVR469I - NUMBER OF SYSLOG KEYS RESTORED                         0
$AVR470I - NUMBER OF STATUS RPT KEYS RESTORED                     0
$AVR478I - NUMBER OF FICHE RPT KEYS RESTORED                      0
$AVR471I - NUMBER OF MESSAGE KEYS RESTORED                        0
$AVR473I - NUMBER OF ANNOTATION KEYS RESTORED                     0
$AVR472I - NUMBER CONTROL KEYS RESTORED                           0
$AVR499I - END OF SELECTIVE RESTORE PROCESSING
```

Chapter 3

$AVRTAPE - Backup/Unload file extract

The **$AVRTAPE** program is used to Extract Work From The Backup/Unload Files created by $AVRS and may be used in two ways: selections for work may either be supplied from the DDNAME "SYSIN" or may be entered as a reply to a WTOR interface.

- **!** As a started task or batch job with a **PARM** value of **WTOR** (**PARM=WTOR**).
- **!** As a batch job without a specified **PARM** value.

Return Codes

$AVRTAPE sets the following return codes.

- Zero - normal processing
- Four - processing completed but no work was selected
- Eight - processing terminated with invalid parameter values
- Twenty - all parameters failed validation or none were specified.

Security in the **$AVRTAPE** program is handled in these ways:

- **!** When **PARM=WTOR**, a response from the operator is required for message **"$AVR701A"**.
- **!** Authorization by the **$AVRUX2** user exit, if used by your $AVRS Administrator.
- **!** Authorization by the **$AVRSECU** security exit.

Backup/archive files from previous releases of $AVRS are accepted as input to this release of **$AVRTAPE** and mixed release files may be concatenated as a single logical input file.

Custom banners may be produced or suppressed for the work that is selected by the **$AVRTAPE** program. The option to suppress the custom banners is controlled by the word "NOBANNER" in the parm field on the execute statement.

To change the input DDNAME from $AVRTAPE to SYSUT1, assign the parm value on the execute statement as *PARM=SYSUT1*. When the procedure is executed with the PARM=SYSUT1 declared, $AVRTAPE reads its input using DDNAME SYSUT1 instead of DDNAME $AVRTAPE.

Started Task or a Batch JOB with a PARM of WTOR

The program issues **WTOR**s for input selection(s) when a **PARM** field of "**WTOR**" is used. WTOs will be issued to inform the user as to the success of the request for work. The user should respond to message "**$AVR703A**" with the desired **$AVRTAPE** selection criteria or END. The selected output will, as a result, be written to DDNAME "RSLOCAL". The default DCB attributes for RSLOCAL are RECFM(VBM) LRECL(259) BLKSIZE(263). The RSLOCAL DCB attributes may be overridden to use a RECFM of V or FB with or without A or M carriage control. The LRECL has a maximum of 255 for RECFM(F) or 259 for RECFM(V). Any valid BLKSIZE may be specified. The RSLOCAL output dataset is written with QSAM and may be allocated to any QSAM supported device or sysout.

A sample procedure "OPERTAPE" is supplied in $AVRS.rel#.INSTLIB. A modification must be made to the procedure to supply your dataset name to the "**$AVRTAPE**" DD. If **$AVRS** custom banners are not desired, a PARM value of "**NOBANNER**" can be used to suppress the **$AVRS BANNER PAGES**.

Sample Procedure - WTOR Parm

```
//$AVROTAP PROC OUT=A
//$AVRS    EXEC PGM=$AVRTAPE,PARM=WTOR,REGION=768K
//STEPLIB  DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR
//*
//$AVRTAPE DD  DSN=$AVRS.BACKUP(+0),DISP=SHR
//*
//*       $AVRTAPE DD IS THE INPUT BACKUP BEING SELECTED FROM
//*
//RSLOCAL  DD  SYSOUT=&OUT
//*
//*       RSLOCAL DD IS THE OUTPUT SELECTION PRINT FILE
//*
//SYSPRINT DD  DUMMY
//*
//*       OUTPUT MESSAGE FILE IS TO THE CONSOLE
//*
//* $AVRWORK DD  UNIT=SYSALLDA,SPACE=(CYL,(2,2))
//*
//* $AVRWORK DD IS THE $AVRS TEMP WORK FILE.  IT IS DYNAMICALLY
//*    ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL.
//*
//SYSIN    DD  DUMMY
//*
//*  INPUT FOR THE SELECTION PARAMETERS IS FROM THE CONSOLE
//*
```

Figure 54 $AVRTAPE WTOR PARM (INSTLIB member OPERTAPE)

Batch Procedure with Input from SYSIN

The input **$AVRTAPE** selection criteria is read from DDNAME "SYSIN" and the output responses are written to DDNAME "SYSPRINT". A sample of the SYSPRINT output is at the end of this chapter. The selected work is written to DDNAME "RSLOCAL". The default DCB attributes for RSLOCAL are RECFM(VBM) LRECL(259) BLKSIZE(263). The RSLOCAL DCB attributes may be overridden to use a RECFM of V or FB with or without A or M carriage control. The LRECL has a maximum of 255 for RECFM(F) or 259 for RECFM(V). Any valid BLKSIZE may be specified. The RSLOCAL output dataset is written with QSAM and may be allocated to any QSAM supported device or sysout. The **$AVRTAPE** selection criteria is defined below. A sample procedure "**BATCTAPE**" is supplied in $AVRS.rel#.INSTLIB. A modification must be made to the procedure to supply your dataset name to the "**$AVRTAPE**" DD statement. If $AVRS custom banners are not desired, a PARM value of "**NOBANNER**" can be used to suppress the $AVRS BANNER PAGES.

```
//$AVRBTAP PROC OUT=A
//$AVRS    EXEC PGM=$AVRTAPE,REGION=768K,
//STEPLIB  DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR
//*
//$AVRTAPE DD  DSN=$AVRS.rel#.BACKUP(+0),DISP=SHR
//*
//*       $AVRTAPE DD IS THE INPUT BACKUP BEING SELECTED FROM
//*
//RSLOCAL  DD  SYSOUT=&OUT
//*
//*       RSLOCAL DD IS THE OUTPUT SELECTION PRINT FILE
//*
//SYSPRINT DD  SYSOUT=&OUT
//*
//*  SYSPRINT DD IS THE OUTPUT $AVRPULL MESSAGE FILE
//*
//* $AVRWORK DD  UNIT=SYSALLDA,SPACE=(CYL,(2,2))
//*
//*       $AVRWORK DD IS THE $AVRS TEMP WORK FILE. IT IS DYNAMICALLY
//*       ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL.
//*
//SYSIN   DD DUMMY
//*
//*       SYSIN DD IS THE INPUT SOURCE FOR THE SELECTION PARAMETERS
//*
```

Figure 55 $AVRTAPE Procedure (INSTLIB member BATCTAPE)

Batch JOB with Input from SYSIN

Following is Sample JCL for Batch Execution of **$AVRTAPE**

```
//$AVRTAPE JOB ('ACCOUNTING INFORMATION'), 'TAPE PRINT',REGION=512K,
//             CLASS=XMSGCLASS=N,NOTIFY=&SYSUID
//STEP1    EXEC PGM=$AVRTAPE,REGION=786K
//STEPLIB  DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR
//SYSDUMP  DD SYSOUT*
//*
//$AVRTAPE DD  DSN=$AVRS.rel#.BACKUP(+0),DISP=SHR
//*
//*        $AVRTAPE DD IS THE INPUT BACKUP BEING SELECTED FROM
//*
//RSLOCAL  DD  SYSOUT=*
//*
//*        RSLOCAL DD IS THE OUTPUT SELECTION PRINT FILE
//*
//SYSPRINT DD  SYSOUT=*
//*
//*  SYSPRINT DD IS THE OUTPUT $AVRTAPE MESSAGE FILE
//*
//* $AVRWORK DD  UNIT=SYSALLDA,SPACE=(CYL,(2,2))
//*
//*       $AVRWORK DD IS THE $AVRS TEMP WORK FILE. IT IS DYNAMICALLY
//*       ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL.
//*
//SYSIN   DD *
//*
//*       SYSIN DD IS THE INPUT SOURCE FOR THE SELECTION PARAMETERS
//*
```

Figure 56 $AVRTAPE Procedure (INSTLIB member $AVRTAPE)

Selection Criteria

$AVRTAPE SYSIN parameters are all keyword driven and free form. Each new statement must start in column 1. If a continuation is required the previous line must end in a comma and the next keyword must start in position 2 or greater of the next line. Each parameter must be completed within the line in which it starts and cannot be split across multiple lines.

Several of these keyword parameters can have their values **masked**. A **masked** value means a string of alpha-numeric or national characters that may be from 1 to 8 characters in length. Any position may have its significance overridden by using the character '*****'. When no additional significant characters are needed to compose a name mask, a "**-**" may be used in lieu of filling the remaining positions with asterisks. A "." (period) in the mask denotes an absolute name. The remainder of the name field is filled with trailing blanks. This mask character, ".", prevents a name shorter than 8 characters from being generic.

Any name given is considered **generic** unless it is completely specified for its entire length or if masking characters are used to compose the name. All work matching that type of request will be selected if all other selection criteria is met.

Examples of these control statements are shown at the end of all the parameters and before a copy of the $AVRTAPE Report.

ONEMATCH= This parameter is used by multiple programs: $AVRPULL, $AVRSELR, $AVRTAPE. This parameter is a program control parameter that controls how many times a selection control statement(s) can be used. If used, this parameter may only appear once in the input stream per execution of the program and should be on a record by itself. Any selection criteria that is on the same record as this parameter is ignored.

YES is designed to work with a **fully qualified** NAME=parameter and any other use may result in excessive output. **YES** will select only the first work that is matched by each control statement. With ONEMATCH=YES, after all control statements are matched, the program completes successfully, even if the input file is not at EOF.

EX: ONE=YES
TYPE=JN,NAME=C106PAY2 (selects only first occurrence of this fully qualified job).
TYPE=JN,NAME=C203BB. (selects only first occurrence of this fully qualified job.
NOTE: The ending period makes this fully qualified).

TYPE=JN,NAME=C203BB (selects first occurrence of every job that begins with C203BB - C203BB, C203BBA, C203BBB, etc.).

NO will select all work that matches each control statement.

Shortest Abbreviation: ON
Example: **`ONE=YES | ON=Y`**
Default: **`ONE=NO`**

TYPE= Identifies the kind of work to be selected. The valid work types are:

JN - Batch Jobs
TN - TSO Users
SN - Started Tasks
SL - SYSLOG (Today's date if SD is not specified for Syslog)
ST - Status Reports
FI - Fiche Indices
AH - Archive History
FH - Fiche History

Multiple work types may be selected by enclosing all desired types in parentheses and separating each individual type with a comma. This parameter determines which base keys in the data base are allowed for selection except for **AH** and **FH**. For Archive and Fiche History, this parameter indicates selection is based off individual data lines instead of base keys.
Shortest Abbreviation: TY
Example: **`TY=JN | TYPE=(JN,SN,AH)`**
Default: **`TYPE=(JN,TN,SN)`**

NAME= The highest level name of the work. This field can be **masked**. It is possible to use this field for SYSLOG selection if you follow $AVRS SYSLOG naming conventions: i.e. SLXXXXY where XXXX is the system id (ESAM) and "Y" is the sysout class (N).
Valid for Types: JN, TN, SN, AH, FH, SL
Shortest Abbreviation: NA
Example: **`NAME=RIC- | NA=JOBNAME. | NAM=A*BCD |NAME=SLESAMN`**
Default: **`NAME=-`**

SDATE= The starting date of the work to be selected. This field can be a five byte julian date, (yyddd), a seven byte julian date, (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days. SYSLOG uses this date to determine when printing begins. For Status Reports and Fiche Indices this date is the Recap Period start date of the work. For all other work this is the date the work started execution.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: SD
Example: **`SD=99001 | SD=* | SD=-30 | SD=1999001`**
Default: **`Lowest possible, except SYSLOG which uses today.`**

EDATE= The ending date of the work to be selected. This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days. For SYSLOG segments this date determines when printing of SYSLOG stops. For Status Reports and Fiche Indices this date is the Recap Period ending date of the work. For all other work this is the date the work ended execution.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: ED
Example: **`ED=00001 | ED=* | ED=-30 | SD=2000001`**
Default: **`Highest possible, except SYSLOG which uses today.`**

STIME= The starting time of the work to be selected. For SYSLOG this is the time at which printing begins. Status Reports and Fiche Indices use the Recap Period start time to determine selection. For all other work this is the time the work started execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). An example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: ST
Example: **`ST=1000 | ST=1500`**
Default: **`00:00 (midnight)`**

ETIME= The ending time of the work to be selected. For SYSLOG this is the time at which printing stops. Status Reports and Fiche Indices use the Recap Period end time to determine selection. For all other work this is the time the work ended execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). An example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: ET
Example: **`ET=2300 | ET=1700`**
Default: 23:59 (11:59 p.m)

NOTE: If both start and end times are specified, the work must have either started, ended or both started and ended within the specified time period. If the work was executing across both bounds of the time period, it will not qualify for selection.

SELRULE Date and Time selection may now be based on the job end time only, the start time only, both the start and end times together or any start or end time (default). This parameter is valid for all selection types. All four date and time values (SD, ST, ED, ET) must be specified to use SELRULE. Failure to supply dates and times causes the SELRULE to be ignored or use default dates and times for TYPE of work specified.

START Only jobs that *started* between the start date/time and the end date/time from above will be selected.

END Only jobs that *ended* between the start date/time and the end date/time from above will be selected.

BOTH Only jobs that both *started* and *ended* between the start date/time and the end date/time from above will be selected.

blank Ignore SELRULE
Valid for Types: All
Examples: SELRULE=END | SELRULE=BOTH
Default: Any entry other than “Start”, “End” or “Both” will be ignored.

RANGE= A **Yes** or **No** value that governs how the starting and ending date and time fields are used. **Yes** means all work within the starting date and time and the ending date and time range is selected. **No** means all work within the start date and end date range *that fell within the start and end time of each day is selected* a SLICE of time. When RANGE=NO end time must be greater than start time.
Valid for Types: JN, TN, SN, SL, ST, FI, AH, FH
Shortest Abbreviation: RA
Example: **`RA=YES | RA=N`**
Default: **`Yes`**

ACKNOWLEDGMENT=
The acknowledgment code of the work. This six byte character field can be **masked**. It can also contain two special values: **N/A** and **@ALL@**. **N/A** selects all work that does not have any code and **@ALL@** selects all work that does.
Valid for Types: JN, TN, SN
Shortest Abbreviation: AC
Example: **`AC=123456 | AC=1*PR. | AC=N/A`**
Default: **`None`**

JO= Job Owner of the work. This is an eight byte field that can be **masked**. Work is considered "owned" if this field matches either the NOTIFY or USER of the work.
Valid for Types: JN or when TYPE not specified
Shortest Abbreviation: JO
Example: **`JO=RIC- | JOB=JOBOWNR. | JOB=AB*CD`**
Default: **`None`**

ID= The System Id of the SYSLOG to be selected. This is a four byte field that can be **masked**. All SYSLOG accumulated by $AVRS has a seven byte "constructed" name as follows: 'SL' in cc 1 - 2; System Id in cc 3 - 6 and the sysout class in cc 7. This field matches against the System Id or SID part of a SYSLOG name.
Valid for Types: SL
Shortest Abbreviation: ID
Example: `ID=IPO1 | ID=E*A`
Default: `None`

CCODE= The Completion or Condition code of the work. This is a seven byte field that can be **masked**. "NZRSPN" will select respun work with a non-zero condition code.
Valid for Types: JN, TN, SN, AH, FH
Shortest Abbreviation: CC
Example: `CC=PARTIAL | CC=***0000`
Default: `None`

NUMBER= The JES number of the work. Leading zeroes do not have to be specified in this five byte numeric field, but the number specified must exactly match the work for selection.
Valid for Types: JN, TN, SN, SL, AH, FH
Shortest Abbreviation: NU
Example: `NU=5 | NU=967`
Default: `None`

STEPNAME= The step name of the work. This eight byte field can be **masked**.
Valid for Types: JN, TN, SN
Shortest Abbreviation: STE
Example: `STE=- | STEP=STEP.`
Default: `None`

PROCSTEP= The procedure step name of the work. This eight byte field can be **masked**.
Valid for Types: JN, TN, SN
Shortest Abbreviation: PR
Example: `PR=- | PR=PROC-`
Default: `None`

DDNAME= The DDNAME name of the work. This eight byte field can be **masked**. In addition to actual DDNAMES, five special values are allowed; JEJCLIN, JESLOG, JESJCL, SYSMSG AND JESDS. JESDS selects all JES datasets from the work. The other special values selects only their respective JES data sets.
Valid for Types: JN, TN, SN
Shortest Abbreviation: DD
Example: `DD=- | DD=SYSMSG | DD=SY*P-`
Default: `None`

LOCATE= Defines an argument to be found in the work. In SYSLOG, each line containing the argument is printed. All other work prints a special header before the line showing where the argument was found in the work. This is a 30 byte field. If the argument contains blanks, the argument must be enclosed in parentheses, single or double quotes. This parameter is mutually exclusive with **FA=** and **SCAN=**.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: LO
Example: `LO=TEXT | LO='A LOT OF TEXT'`
Default: `None`

FA= or **SCAN=** These parameters function the same as the **LO=** parameter for $AVRTAPE.

SSTART= The starting position for any type of scan (LO, FA or SC). This is a three byte numeric field that can range from 1 through 255, inclusive. The first printable position is position 1. Leading zeroes are not required.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: SS
Example: **`SS=5 | SS=100`**
Default: 1

SRANGE= The range from the starting position that is scanned for the argument. This is a three byte numeric field ranging from 1 through 255, inclusive. All scans (LO, FA or SC) try to locate the argument from the starting position for this amount. If the argument starts within this range, it is considered found.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: SS
Example: **`SR=5 | SR=100`**
Default: **`255`**

ANNOTATIONS=
Annotations print switch. This governs how annotation print is done. **Yes** means print the annotations (if present) and the selected work. **No** means do not print annotation, just the selected work. **Only** means only print annotations for selected work.
Valid for Types: JN, TN, SN
Shortest Abbreviation: AN
Example: **`AN=Y | AN=ONLY | ANN=NO`**
Default: **`Yes`**

CLASS= Sysout class. Only work accumulated in this sysout class is selected by this parameter. This is a one byte field.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: CL
Example: **`CL=5 | CL=A`**
Default: **`None`**

FORM= Sysout Forms Id. Work must match this value to be selected. This field is a maximum of four bytes. This parameter is not valid for SYSLOG.
Valid for Types: JN, TN, SN
Shortest Abbreviation: FO
Example: **`FO=STD5 | FO=STD`**
Default: **`None`**

FCB= Sysout Forms Control Buffer. Work must match this value to be selected. This field is a maximum of four bytes. This parameter is not valid for SYSLOG.
Valid for Types: JN, TN, SN
Shortest Abbreviation: FC
Example: **`FC=STD1 | FC=STD`**
Default: **`None`**

UCS= Sysout Universal Character Set. Work must match this value to be selected. This field is a maximum of four bytes. This parameter is not valid for SYSLOG.
Valid for Types: JN, TN, SN
Shortest Abbreviation: UC
Example: **`UC=TN | UC=PN`**
Default: **`None`**

AHWORK= Archive History Work type. This value selects the types (JN, SN, TN, or SL) of archive history. This parameter requires TYPE=AH selections. Multiple types may be selected by enclosing all desired types in parentheses and separating them with commas.
Valid for Types: AH
Shortest Abbreviation: AHW
Example: **`AHW=JN | AHW=(JN,SN)`**
Default: **`None`**

AHTYPE= Allows Archive History selection for either non-zero completion codes (**NZHIST**) or respun work (**REHIST**).
Valid for Types: AH
Shortest Abbreviation: AHT
Example: **`AHW=NZHIST | AHW=REHIST`**
Default: **`None`**

Selection Examples

`TYPE=(JN,SN),NAME=PAY,SD=00001,ED=2000003,ST=0800,ET=1700`
This example selects all jobs and started tasks whose name starts with PAY. This work would have to start, end or both start and end in the time period of 2000001.0800 through 2000003.1700. By default, any annotations of this work will also be selected.

`TY=SL,ID=ESAM,SD=-1,ED=*,ST=0700,ET=0900,RANGE=NO`
All SYSLOG with a system id of ESAM dated between yesterday at 7:00 a.m. through 9:00 a.m. and today between 7:00 a.m. and 9:00 a.m. would select from this example. By default, any annotations of this work will also be selected.

```
TYPE=JN,NAME=LOTS,STEP=ST*1,DD=PR1FILE.,
 SCAN="MY TEXT ARGUMENT",SS=10,SR=5,ACK=@ALL@,
 CL=7
```
The work for this example must be a batch job, start with the name LOTS, have a stepname starting with ST and a 1 in the fourth stepname position, have a DDNAME of exactly PR1FILE, be acknowledged and have been accumulated in class 7. The sysout from DDNAME PR1FILE will be scanned for the string MY TEXT ARGUMENT starting in position 10 through 15. Each line that contains the argument is printed along with a header line detailing where the line was found. By default, the annotations of this work will also be selected.

Sample Work Selection List and Error Report

```
RPTID=$AVRTAPE-REPORT     WORK SELECTION LIST AND ERROR REPORT             TIME=11:01:47   DATE=2002.128       PAGE     1
<<< $AVRS - USER TITLE LINE >>>                      **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                     **** SEA INC   1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010

                          ONE=YES
                          TYPE=JN,NAME=A-
                          TYPE=JN,NAME=NOSN-
                          TYPE=SN,NAME=NOTFOUND
                          TYPE=SL,SDATE=96115,ST=1601
                          TYPE=JN,NAME=B-,AN=N
RPTID=$AVRTAPE-REPORT     WORK SELECTION LIST AND ERROR REPORT             TIME=11:01:47   DATE=2002.128       PAGE   2
<<< $AVRS - USER TITLE LINE >>>                      **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                     **** SEA INC   1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010

$AVR719I - REQUEST SELECTED  ADBPRT (J0027878) 6 PAGE(S), 235 LINE(S), FOR SELECTION 1
$AVR719I - REQUEST SELECTED  BADCARD (J0028052) 1 PAGE(S), 14 LINE(S), FOR SELECTION 5
$AVR719I - REQUEST SELECTED  NOSNTAPE (J0027911) 4 PAGE(S), 143 LINE(S), PLUS ANNOTATIONS FOR SELECTION 2
$AVR719I - REQUEST SELECTED  SL77777 (S0027656) 21 PAGE(S), 1,095 LINE(S), FOR SELECTION 4
$AVR715I - PULL COMPLETE    FOR SELECTION 1
$AVR715I - PULL COMPLETE    FOR SELECTION 2
$AVR720W - REQUESTED ARGUMENT(S) NOT ON FILE FOR SELECTION 3 OR MATCHED BY ANOTHER SELECTION
$AVR715I - PULL COMPLETE    FOR SELECTION 4
$AVR715I - PULL COMPLETE    FOR SELECTION 5
                                        * * *   E N D   O F   R E P O R T   * * *
```

Chapter 4

$AVRTPRP - Archive Tape Report (Contents of a $AVRS Backup or Unload File)

This program can be used at any time to generate a report on the contents of a $AVRS Backup or Unload File. Concatenated files may be processed by this program and the contents report indicates which work is contained in each dataset that is processed.

$AVRTPRP selection criteria is read in through DDNAME "SYSIN" and output responses are written to DDNAME "SYSPRINT" in the **$AVRTPRP PARSE REPORT**. Any errors with selection parameters also appear on this report immediately following the statement in error. The **Archive Tape Index Report** from the $AVRPRT DDNAME is similar to the **RECAP BY JOBNAME** report produced by a $AVRS ARCHIVE/BACKUP. Samples of the two $AVRTPRP Reports are included at the end of this chapter.

Return Codes

$AVRTPRP sets the following return codes.

- Zero - normal processing (including no work found)
- Eight - processing terminated with invalid parameter values
- Twenty - all parameters failed validation or none were specified.

The following is Sample JCL required to execute $AVRTPRP. This JCL is also provided in $AVRS.rel#.INSTLIB($AVRTPRP).

```
//$AVRTPRP JOB ('ACCOUNTING INFORMATION'),'TAPE RPT',REGION=512K
//JOBLIB   DD  DSN=$AVRS.rel#.LOADLIB,DISP=SHR
//STEP1    EXEC PGM=$AVRTPRP
//SYSUDUMP DD SYSOUT=*
//*                -- $AVRTPRP WORK FILES --
//*
//$AVRPRT  DD  SYSOUT=A          *** RECAP BY JOBNAME      ***
//SYSPRINT DD  SYSOUT=A          *** PARSE REPORT          ***
//*
//*       $AVRTAPE DD IS ALLOCATED TO THE BACKUP TAPE FILE
//*       FILES MAY BE CONCATENATED IF DESIRED
//*
//$AVRTAPE DD  DSN=$AVRS.BACKUP(+0),DISP=SHR
//*
//*
//*    SYSIN DD IS USED TO READ IN THE $AVRTPRP CONTROL
//*     PARAMETERS
//*
//* $AVRWORK DD  UNIT=SYSALLDA,SPACE=(CYL,(2,2))
//*
//*    $AVRWORK DD IS THE $AVRS TEMP WORK FILE.  IT IS DYNAMICALLY
//*    ALLOCATED UNLESS THE DD IS PRESENT IN THE JCL
//*
//SYSIN    DD  *
```

Figure 57 $AVRTPRP Example JCL

Selection Criteria

$AVRTPRP SYSIN parameters are all keyword driven and free form. Each new statement must start in column 1. If a continuation is required the previous line must end in a comma and the next keyword must start in position 2 or greater of the next line. Each parameter must be completed within the line in which it starts and cannot be split across multiple lines.

Several of these keyword parameters can have their values **masked**. A **masked** value means a string of alpha-numeric or national characters that may be from 1 to 8 characters in length. Any position may have its significance overridden by using the character "*****". When no additional significant characters are needed to compose a name mask, a "**-**" may be used in lieu of filling the remaining positions with asterisks. A "." (period) in the mask denotes an

absolute name. The remainder of the name field is filled with trailing blanks. This mask character, ".", prevents a name shorter than 8 characters from being generic.

Any name given is considered **generic** unless it is completely specified for its entire length or if masking characters are used to compose the name. All work matching that type of request will be selected if all other selection criteria is met.

Examples of the control statements are shown at the end of all the parameters and before a copy of the $AVRTPRP Report.

TYPE= Identifies the kind of work to be selected. The valid work types are:

JN - Batch Jobs
TN - TSO Users
SN - Started Tasks
SL - SYSLOG (Today's date if SD is not specified for Syslog)
ST - Status Reports
FI - Fiche Indices
AH - Status Reports
FH - Fiche Indices

NOTE: A type of **AH** is identical to **ST** and a type of **FH** is identical to **FI.**

Multiple work types may be selected by enclosing all desired types in parentheses and separating each individual type with a comma. This parameter determines which base keys in the data base are allowed for selection.
Shortest Abbreviation: TY
Examples: **`TY=JN | TYPE=(JN,SN)`**
Default: **`TYPE=(JN,TN,SN)`**

NAME= The highest level name of the work. This field can be **masked**. It is possible to use this field for SYSLOG selection if you follow $AVRS SYSLOG naming conventions: i.e. SLxxxxy where "xxxx" is the system id (ESAM) and "y" is the sysout class (N).
Valid for Types: JN, TN, SN
Shortest Abbreviation: NA
Examples: **`NAME=RIC- | NA=JOBNAME. | NAM=A*BCD |NA=SLESAMN`**
Default: **`NAME=-`**

SDATE= The starting date of the work selected. This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days. This is the starting date of the segment for SYSLOG. For Status Reports and Fiche Indices this date is the Recap Period start date of the work. For all other work this is the date the work started execution.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: SD
Examples: **`SD=99001 | SD=* | SD=-30 | SD=1999001`**
Default: **`Lowest possible, except SYSLOG which uses today.`**

EDATE= The ending date of the work to be selected. This field can be a five byte julian date (yyddd), a seven byte julian date (yyyyddd), an asterisk (*) for today's date or a negative number for today minus a specified number of days. For SYSLOG segments this date is the ending date of the segment. For Status Reports and Fiche Indices this date is the Recap Period ending date of the work. For all other work this is the date the work ended execution.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: ED
Examples: **`ED=00001 | ED=* | ED=-30 | ED=2000001`**
`Default: Highest possible, except SYSLOG which uses today.`

STIME= The starting time of the work to be selected. For SYSLOG this is the starting time of the segment. Status Reports and Fiche Indices use the Recap Period start time to determine selection. For all other work this is the time the work started execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). An example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: ST
Examples: **`ST=1000 | ST=1500`**
Default: **`00:00 (midnight)`**

ETIME= The ending time of the work to be selected. For SYSLOG this is the ending time of the segment. Status Reports and Fiche Indices use the Recap Period end time to determine selection. For all other work this is the time the work ended execution. When less than four (4) digits are entered, $AVRS assumes "minutes" and converts the minutes to hhmm, unless the delimiter is included (h:mm). An example of this is: if 90 is entered, $AVRS converts this to 01:30; 91 becomes 01:31.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: ET
Examples: **`ET=2300 | ET=1700`**
Default: 23:59 (11:59 p.m)

NOTE: If both start and end times are specified, the work must have either started, ended or both started and ended within the specified time period. If the work was executing across both bounds of the time period, it will not qualify for selection.

SELRULE Date and Time selection may be based on the job end time only, the start time only, both the start and end times together or any start or end time (default). This parameter is valid for all selection types. All four date and time values (SD, ST, ED, ET) must be specified to use SELRULE. Failure to supply dates and times will cause the SELRULE to be ignored or use the default dates and times for the TYPE of work specified.

START Only jobs that *started* between the start date/time and the end date/time from above will be selected.

END Only jobs that *ended* between the start date/time and the end date/time from above will be selected.

BOTH Only jobs that both *started* and *ended* between the start date/time and end date/time from above will be selected.

blank Ignore SELRULE

Valid for Types: All
Examples: SELRULE=END | SELRULE=BOTH
Default: Any entry other than "Start", "End" or "Both" will be ignored.

RANGE= A **Yes** or **No** value that governs how the starting and ending date and time fields are used. **Yes** means all work within the starting date and time and the ending date and time range is selected. **No** means all work within the start date and end date range *that fell within the start and end time of each day is selected* a SLICE of time. When RANGE=NO end time must be greater than start time.
Valid for Types: JN, TN, SN, SL, ST, FI
Shortest Abbreviation: RA
Examples: **`RA=YES | RA=N`**
Default: **`Yes`**

ACKNOWLEDGMENT=
The acknowledgment code of the work. This six byte character field can be **masked**. It can also contain two special values: **N/A** and **@ALL@**. **N/A** selects all work that does not have any code and **@ALL@** selects all work that does.
Valid for Types: JN, TN, SN
Shortest Abbreviation: AC
Examples: **`AC=123456 | AC=1*PR. | AC=N/A`**
Default: **`None`**

ERROR= The Completion or Condition code of the work. This field is the same as the online field **CCODE**. This is a seven byte field that can be **masked**.
Valid for Types: JN, TN, SN
Shortest Abbreviation: ER
Examples: **`ER=PARTIAL | ER=***0000`**
Default: **`None`**

LINES= The number of sysout lines the work must contain. This is a seven byte field. Valid values are 1 through 9999999. All work must contain this amount or a greater number of sysout lines for selection.
Valid for Types: JN, TN, SN
Shortest Abbreviation: LI
Examples: **LI=1 | LI=9999999**
Default: **None**

MESSAGES= This field allows selection of work by any of the vertical title messages that appear on the $AVRPRT DDNAME. An explanation and sample of this report is included at the end of the chapter. You may select either the *pro* or *con* of each message and multiple messages may be selected by enclosing all messages in parentheses separated by commas. The "*pro*" of the message is selected by entering the vertical title itself. All work that has an asterisk (*) in that message column is then selected. The "*con*" of the message is the vertical title prefixed with "**NO**". All work that does **not** have an asterisk in that message column is selected. When multiple messages are selected, they are treated as an "AND" condition. You may select any combination of "*pro*" or "*con*" messages, but cannot select the "*pro*" and "*con*" of the same message.
Valid for Types: JN, TN, SN
Shortest Abbreviation: ME
Examples: **ME=RSPN | ME=(RSPN,NZCC,NOMLEX)**
Default: **None**

NUMBER= The JES number of the work. Leading zeroes do not have to be specified in this five byte numeric field, but the number specified must exactly match the work for selection.
Valid for Types: JN, TN, SN, SL
Shortest Abbreviation: NU
Examples: **NU=5 | NU=967**
Default: **None**

ID= The System Id of the SYSLOG to be selected. This is a four byte field that can be **masked**. All SYSLOG accumulated by $AVRS has a seven byte "constructed" name as follows: 'SL' in cc 1 - 2; System Id in cc 3 - 6 and the sysout class in cc 7. This field matches against the System Id or SID part of a SYSLOG name.
Valid for Types: SL
Shortest Abbreviation: ID
Examples: `ID=IPO1 | ID=E*A`
Default: `None`

Selection Examples

```
TYPE=(JN,SN),NAME=PAY,SD=2000001,ED=2000003,ST=0800,ET=1700
```
This example selects all jobs and started tasks whose name starts with PAY. This work would have to start, end or both start and end in the time period of 2000001.0800 through 2000003.1700.

```
TY=SL,ID=ESAM,SD=-1,ED=*,ST=0700,ET=0900,RANGE=NO
```
All SYSLOG with a system id of ESAM dated between yesterday at 7:00 a.m. through 9:00 a.m. and today between 7:00 a.m. and 9:00 a.m. would select from this example.

```
TYPE=JN,NAME=LOTS,
 ACK=@ALL@,LI=1500,ME=(RSPN,NOMLEX,NZCC)
```
The work for this example must be a batch job, start with the name LOTS and be acknowledged. It must have at least 1500 sysout lines and have an asterisk (*) under the vertical titles RSPN and NZCC. The vertical column under MLEX must be blank.

Sample $AVRTPRP Parse Report

```
$AVR001-06      TUE 07 MAY 2002.127         $AVRTPRP PARSE REPORT                       TIME=12:08:41          PAGE     1
<<< $AVRS - USER TITLE LINE >>>                 **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                                **** SEA  INC.   1230 HEMPSTEAD TURNPIKE FRANKLIN SQUARE, N.Y. 11010

                        NAME=-,ACK=N/A,MESS=MLEX
                        NA=OHF,SD=1992300,ED=1992325
                        TYPE=TN
                        TYPE=SN,ME=(NZCC,NOMLEX,472I)
                        TYPE=(ST,FI)
```

Sample Archive Tape Index Report

```
RPTID=ARCHIVE TAPE INDEX                  RECAP BY JOBNAME              TIME=12:08:41   DATE=05/07/2002.127   PAGE   1
<<< $AVRS - USER TITLE LINE >>>          **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                         **** SEA  INC.    1230 HEMPSTEAD TURNPIKE FRANKLIN SQUARE, N.Y. 11010
        BACKUP DSN=C112.$AVRS.UNLOAD          VOLSER(S)=CSISEA                                FILESEQ=DASD
                                                                                              R M N U M B O D 8 4 4 4 2 6 2 2 U U
                                                                                              S L Z M L K N R 6 5 7 5 4 9 8 5 P P
                                                                                              P E C S B U L E 3 1 2 X 4 0 7 1 F F
 JOBNAME   JOB  NO. JOB  START   JOB      END EXEC MINS. CPU  TIME ACKCDE  ERROR TYPE    LINES N X C G M P N Q I I I I I I I I B O
 C112LOAD  J002085  1992223.1213 1992223.1213       0     0.00.00          SI/O             48 * *       *         *
 C112REFR  J002086  1992223.1216 1992223.1216       0     0.00.00          SI/O             51 * *       *         *
 C112REFR  J002110  1992223.1356 1992223.1356       0     0.00.00          SI/O             17 * *       *         *
 C112      T000041  1992211.1113 1992211.1113       0     0.00.00          PARTIAL           2           *   *
 C112      T002205  1992226.0905 1992226.0905       0     0.00.00          PARTIAL           2           *   *
 CICS42B7  S001505  1992218.1240 1992218.2124     524     2.34.45          U2307           456 *   *     *         *
 CSARDEMO  S009286  1992206.0814 1992206.0814       0     0.00.00          SI/O            134   *       *         *
 CSARDEMO  S002029  1992223.0817 1992223.0817       0     0.00.00          SI/O            134 * *       *         *
 DITTO     S002201  1992224.0900 1992224.0900       0     0.00.00          SI/O             17 * *       *         *
 DITTOTP   S002826  1992226.0900 1992226.0902       2     0.00.38          S222             60 *   *     *         *
 DITTOTP   S002848  1992226.0951 1992226.0952       1     0.00.29          S222             55 *   *     *         *
 INIT      S002020  1992223.0817 1992223.0817       0     0.00.00          SI/O            132 * *       *         *
 INITTAPE  S002731  1992225.1643 1992225.1650       7     0.00.34          S222             72 *   *     *         *   *
 ISPFLMF   S009285  1992206.0814 1992206.0814       0     0.00.00          SI/O             49   *       *         *
 ISPFLMF   S002028  1992223.0817 1992223.0817       0     0.00.00          SI/O            221 * *       *         *
 JRP       S009290  1992206.0816 1992206.0816       0     0.00.00          SI/O             38   *       *         *
 JRP       S002032  1992223.0819 1992223.0819       0     0.00.00          SI/O             38 * *       *         *
 NET       S009283  1992206.0814 1992206.0814       0     0.00.00          SI/O             67   *       *         *
 NET       S002026  1992223.0817 1992223.0817       0     0.00.00          SI/O            182 * *       *         *
 PRTVTAM   S007653  1992195.1301 1992195.1301       0     0.11.03          S23E            110     *     *         *
RPTID=ARCHIVE TAPE INDEX                 RECAP BY JOBNAME              TIME=12:08:41   DATE=05/07/2002.127   PAGE   1
<<< $AVRS - USER TITLE LINE >>>            **** LICENSED BY * $AVRS (C) 1982 - 2002 CSI         RELEASE 5.1B-00
                                           **** SEA  INC.        1230 HEMPSTEAD TURNPIKE   FRANKLIN SQUARE, N.Y. 11010
        BACKUP DSN=C112.$AVRS.UNLOAD                    VOLSER(S)=CSISEA                              FILESEQ=DASD
    HISTORY         LINES      RECAP PERIOD START        RECAP PERIOD END            CREATION/RUN DATE TIME
 -------------   -----------   ------------------------ ------------------------   ---------------------------
 ONLINE STATUS          344    TUE 21 JAN 1992.021 16:04 THU 30 APR 1992.121 15:49   THU 30 APR 1992.121 11:18.49
 ONLINE STATUS          700    THU 11 JUN 1992.163 15:51 THU 11 JUN 1992.163 08:48   THU 11 JUN 1992.163 09:23.31
 ONLINE STATUS        1,034    MON 13 JUL 1992.195 14:17 MON 13 JUL 1992.195 12:46   MON 13 JUL 1992.195 12:50.15
 ONLINE STATUS          450    SAT 18 APR 1992.109 09:33 MON 27 JUL 1992.209 13:25   MON 27 JUL 1992.209 13:27.26
 FICHE INDEX              8    SAT 11 APR 1992.102 09:12 SAT 11 APR 1992.102 09:03   MON 20 JUL 1992.202 12:14.15
                                        * * *   E N D   O F   R E P O R T   * * *
```

Archive Tape Index Report Description

Field Names and Descriptions

JOBNAME The name of the JOBs, STCs, TSUs, or SYSLOG which was accumulated.

JOB NUMBER The number that the JES had assigned to the JOBs, STCs, TSUs, or SYSLOG.

JOB START The start execution time of the JOBs, STCs, or TSUs. If jobname is for a SYSLOG, the JOB START is the start time for the segment of the SYSLOG which was accumulated.

JOB END The end execution time of the JOBs, STCs, or TSUs. If jobname is for a SYSLOG, the JOB END is the END time for the segment of the SYSLOG which was accumulated.

EXEC MINS The elapsed (clock) time used by the job.

CPU TIME The total cpu time in the form mmmm.ss.hs (where mmmm = cpu minutes, ss = cpu seconds, and hs = cpu hundredths of seconds). This field is not used for SYSLOG.

ACKCDE The user supplied acknowledgment code 1 to 6 bytes in length.

ERROR TYPE ERROR TYPE prints the highest level error encountered during execution of the JOBs, STCs, or TSUs. Neither SCAN nor SCANERR are reasons for an automatic respin; these conditions are reported only for convenience. A list of the error types follow:

1. *ALLOCATION*
   The operator responded "NO" to a device allocation recovery message.
2. *JCL ERR*
   A JCL error occurs during execution.
3. *NOTCTLG*
   NOT CATLGD error occurred during deallocation.
4. *RC=nnnn, HC=nnnn, IC=nnnn, TX=nnnn, RQ=nnnn*
   The highest return code from any step executed.

5. *Snnn*
   A system abend occurred during execution.
6. *SI/O*
   The $AVRS SYNAD exit was entered or message IEF376I was not found while processing the SYSMSG dataset for the work.
7. *Unnn*
   A user abend occurred during execution.
8. *PARTIAL*
   The JES msgs dataset was not available for this key at the time of its accumulation.
9. *INCMPLT*
   MAX LINES was reached during accumulation of this work.
10. *SCAN*
    TYPRUN=SCAN was specified on the job card for this accumulated work and NO errors were detected by the scan.
11. *SCANERR*
    TYPRUN=SCAN was specified on the job card for this accumulated work and errors were detected by the scan. The most critical error encountered is printed in the field. This field is not used for SYSLOG.
12. *ABRVMSG*
    If the message number specified in the initialization parameter ABRVMSG was found in this job, this number will be reported here instead of the job completion code.

LINES The number of lines accumulated in a job.

Vertical Titles

NOTE: When the condition specified by the "VERTICAL TITLE" exists, an asterisk is printed under the vertical title.

RSPN work was respun. This indicates that the work was respun due to an error.

MLEX max lines exceeded. The "MAXLINES" value was exceeded during accumulation for this job.

NZCC non-zero COND CODE. A non-zero condition code was encountered while processing the job.

UMSG user messages. At least one "ABRVMSG" was encountered while processing this job.

MLBM max lines before message dataset completed. The "MAXLNES" value was exceeded prior to accumulating the JES2 message dataset. Some key information had to be generated. If this field is flagged the "MAXLNES" value might be too small or the job may need to be checked for a potential problem.

BKUP job was written to backup. This job was written to the backup file during this backup run.

ONLN job remains in the online data base

DREQ dump respun/requeued. A dump was encountered during accumulation of this job. The job had to be specially handled by $AVRS.

863I message IEF8631 encountered. Job waited on datasets.

451I message IEF451I or JDJ451I encountered. Job was ended by a condition code.

472I message IEF472I or JDJ472I encountered. Job abended.

45XI message IEF45XI encountered. Either a IEF452I, JDJ452I or a IEF453I, JDJ453I, IEF657I, JDJ657I, IEF722I, or JDJ722I message was encountered. Job failed with a JCL error.

244I message IEF244I encountered. Job went into allocation recovery.

690I message IEF690I encountered. Job waited on volumes.

287I message IEF287I, IEF371I, IEF378I, or IGD17101I encountered. Job had a "NOT- CATLGD" on a dataset. Message is limited to just the NOT-CATLG type messages. "NOT-RECATLGD" is not included in this category.

251I message IEF251I encountered. Cancel was responded to an allocation recovery.

UPFB $AVRUX1 user exit, requested that the job not be written on the $AVRBKUP dataset (backup file).

UPFO $AVRUX1 user exit, requested that the job not be written on the online data base.

APPENDIX A

$AVRS ABENDS

$AVRS ABENDS

$AVRS user abends normally refer to associated messages for explanation and resolution of the abend. The associated messages may have different alphabetic suffixes than those shown depending on the program that issues the abend.

The $AVRS Message referenced in the following pages can be found in Appendix B of this manual.

ABEND	DESCRIPTION
0001	See the $AVRS Message $AVR001C for the cause of this abend and the proper action for recovery.
0002	See the $AVRS Message $AVR002 for the cause of this abend and the proper action for recovery.
0004	An invalid parameter was passed the sort module $AVRCSRL. Please contact SEA for assistance. No $AVRS error message is issued for this User 0004 Abend.
0020	See the $AVRS Message $AVR020C for the cause of this abend and the proper action for recovery.
0018	See the $AVRS Message $AVR018C for the cause of this abend and the proper action for recovery.
0022	See the $AVRS Message $AVR022C for the cause of this abend and the proper action for recovery.
0023	See the $AVRS Message $AVR023C for the cause of this abend and the proper action for recovery.
0024	See the $AVRS Message $AVR024C for the cause of this abend and the proper action for recovery.
0026	See the $AVRS Message $AVR026C for the cause of this abend and the proper action for recovery.
0030	See the $AVRS Message $AVR030C for message description. Invalid allocation attributes passed from a user exit. Check your modified user exit. **NOTE**: There are multiple $AVR030 messages.
0042	See the $AVRS Message $AVR042C for the cause of this abend and the proper action for recovery.
0050	See the $AVRS Message $AVR050 for the cause of this abend and the proper action for recovery.

ABEND	DESCRIPTION
0053	See the $AVRS Message $AVR053 for the cause of this abend and the proper action for recovery.
0060	See the $AVRS Message $AVR060C, $AVR576C or $AVR577C for the cause of this abend and the proper action for recovery.
0101	Dynamic allocation failed for DDNAME "$AVREDYN" in program $AVRINIT. Message $AVR030C gives the reason for the failure. Make the necessary corrections to your modified user exit, and restart $AVRS. NOTE: There are multiple $AVR030 messages.
0102	See the $AVRS Message $AVR025C or $AVR102C for the cause of this abend and the proper action for recovery. The initialization parameters/errors are written to the $AVRERR file. This file is in the first accumulation class (ACCMCLS) of the $AVRS task. At the end of the file is a list of all possible $CRD### messages.
0110	See the $AVRS Message $AVR110C for the cause of this abend and the proper action for recovery.
0205	See the $AVRS Message $AVR205C for the cause of this abend and the proper action for recovery.
0206	See the $AVRS Message $AVR206 for the cause of this abend and the proper action for recovery.
0210	See the $AVRS Message $AVR210C for the cause of this abend and the proper action for recovery.
0211	See the $AVRS Message $AVR211C for the cause of this abend and the proper action for recovery.
0214	See the $AVRS Message $AVR214C for the cause of this abend and the proper action for recovery.
0232	See the $AVRS Message $AVR202C, $AVR203, $AVR232, or $AVR234C for the cause of this abend and the proper action for recovery.

ABEND	DESCRIPTION
0233	See the $AVRS Message $AVR233C for the cause of this abend and the proper action for recovery.
0255	See the $AVRS Message $AVR255W for the cause of this abend and the proper action for recovery.
0348	See the $AVRS Message $AVR348C for the cause of this abend and the proper action for recovery.
0368	One or more errors have occurred while parsing the $AVRBKUP control parameters. See the $AVRBKUP parse report for the message(s) which identify the error(s) that have produced the U0368 abend. Correct the error(s) and rerun the job. Use the $AVRBKPR program to validate the $AVRBKUP parameters. Reference $AVRS Messages $AVR305C through $AVR359C, $AVR368C or $AVR369C.
0370	See the $AVRS Message $AVR370 for the cause of this abend and the proper action for recovery.
0373	See the $AVRS Message $AVR373 for the cause of this abend and the proper action for recovery.
0380	See the $AVRS Message $AVR389C for the cause of this abend and the proper action for recovery.
0382	See the $AVRS Message $AVR382E for the cause of this abend and the proper action for recovery.
0383	See the $AVRS Message $AVR383 for the cause of this abend and the proper action for recovery.
0384	See the $AVRS Message $AVR384E and $AVR385E for the cause of this abend and the proper action for recovery.
0404	Dynamic allocation failure for RSLOCAL/RSREMOTE DDNAME when a job with errors is being respun. Message $AVR030C gives the reason for the failure. **NOTE:** There are multiple $AVR030 messages. Make the necessary corrections and restart $AVRS.

ABEND	DESCRIPTION
0406	See the $AVRS Message $AVR406 for the cause of this abend and the proper action for recovery.
0407	Dynamic unallocation failure for a JES2 data set after accumulation. Message $AVR030C gives the reason for the failure. Restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0412	See the $AVRS Message $AVR412C for the cause of this abend and the proper action for recovery.
0413	See the $AVRS Message $AVR413C or $AVR414C for the cause of this abend and the proper action for recovery.
0424	Dynamic allocation failure for a DDNAME required by the $AVRBKUP program. Message $AVR030C gives the DDNAME and the reason for the failure. Restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0425	Dynamic unallocation failure for either the $AVRBKUP or $AVRDUPX DDNAME. Message $AVR030C gives the reason for the failure. Restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0426	Dynamic unallocation failure for a JES SYSOUT dataset. Message $AVR030C gives the reason for the failure. Restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0427	Dynamic allocation failure for a JES SYSOUT dataset. Message $AVR030C gives the reason for the failure. Restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0428	An internal error has occurred during the accumulation of a job. Restart $AVRS. If the abend is repeated, contact SEA for assistance with the resolution.
0430	See the $AVRS Message $AVR430I for the cause of this abend and the proper action for recovery.
0431	See the $AVRS Message $AVR431C for the cause of this abend and the proper action for recovery.

ABEND	DESCRIPTION
0433	An open failed for a JES dataset during accumulation for a job. Restart $AVRS.
0434	Key maintenance was called but no valid Writer Name was assigned to the SYSOUT work.
0435	An internal error has occurred during compression of a SYSOUT dataset for a job. See message $AVR435C for corrective action if the abend is repeated for the same job. Restart $AVRS.
0440	A SYNAD exit was taken during processing for a SYSLOG dataset. See message $AVR444I for the corrective action.
0449	A $AVRS backup file is being created on DASD and the dataset name is not a GDG. Call SEA for technical support.
0459	See the $AVRS Message $AVR459C for the cause of this abend and the proper action for recovery.
0479	See the $AVRS Message $AVR459C for the cause of this abend and the proper action for recovery.
0497	JES returned a file with an invalid dsname. $AVRS is unable to continue with operator action. See the $AVRS message $AVR497C for additional details. Also see 0499 below for a further explanation.
0498	See the $AVRS Message $AVR498C for the cause of this abend and the proper action for recovery.
0499	A JES request error has occurred. Place the job $AVRS is attempting to accumulate on hold and reinitialize $AVRS. The WRKINFO=YES parameter in the initialization parms will display the current job $AVRS is accumulating. ($AVRS message $AVR455I) Also see 0497 above. The DSNAME may contain blanks or invalid characters.
0500	Place this jobs output on hold and restart the $AVRS task.
0503	See the $AVRS Message $AVR503C for the cause of this abend and the proper action for recovery.

ABEND	DESCRIPTION
0504	See the $AVRS Message $AVR504C for the cause of this abend and the proper action for recovery.
0505	See the $AVRS Message $AVR505C for the cause of this abend and the proper action for recovery.
0506	See the $AVRS Message $AVR506C for the cause of this abend and the proper action for recovery.
0507	See the $AVRS Message $AVR507C for the cause of this abend and the proper action for recovery.
0516	See the $AVRS Message $AVR516C for the cause of this abend and the proper action for recovery.
0518	See the $AVRS Message $AVR518C for the cause of this abend and the proper action for recovery.
0524	See the $AVRS Message $AVR524C for the cause of this abend and the proper action for recovery.
0525	See the $AVRS Message $AVR525C for the cause of this abend and the proper action for recovery.
0576	See the $AVRS Message $AVR576C for the cause of this abend and the proper action for recovery.
0577	See the $AVRS Message $AVR577C for the cause of this abend and the proper action for recovery.
0581	See the $AVRS Message $AVR581C for the cause of this abend and the proper action for recovery.
0584	See the $AVRS Message $AVR584C for the cause of this abend and the proper action for recovery.
0591	See the $AVRS Message $AVR591C for the cause of this abend and the proper action for recovery.

ABEND	DESCRIPTION
0592	See the $AVRS Message $AVR592C for the cause of this abend and the proper action for recovery.
0601	Open failure for RSREMOTE DDNAME when a job with errors is being respun. Reference messages $AVR030C, $AVR601C and $AVR602C for the reason for the failure. Make the necessary corrections and restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0602	Open failure for RSLOCAL DDNAME when a job with errors is being respun. Message $AVR030C gives the reason for the failure. Make the necessary corrections and restart $AVRS. **NOTE**: There are multiple $AVR030 messages.
0650	See the $AVRS Message $AVR650C for the cause of this abend and the proper action for recovery.
0680	See the $AVRS Message $AVR680C or $AVR682C for the cause of this abend and the proper action for recovery.
0681	See the $AVRS Message $AVR681C for the cause of this abend and the proper action for recovery.
0683	See the $AVRS Message $AVR683C for the cause of this abend and the proper action for recovery.
0700	See the $AVRS Message $AVR700C for the cause of this abend and the proper action for recovery.
0702	See the $AVRS Message $AVR702C for the cause of this abend and the proper action for recovery.
0705	See the $AVRS Message $AVR705C for the cause of this abend and the proper action for recovery.
0707	See the $AVRS Message $AVR707 for the cause of this abend and the proper action for recovery.
0718	See the $AVRS Message $AVR718 for the cause of this abend and the proper action for recovery.

ABEND	DESCRIPTION
0720	See the $AVRS Message $AVR720C for the cause of this abend and the proper action for recovery.
0730	See the $AVRS Message $AVR730 for the cause of this abend and the proper action for recovery.
0741	See the $AVRS Message $AVR741C for the cause of this abend and the proper action for recovery.
0759	See the $AVRS Message $AVR759C for the cause of this abend and the proper action for recovery.
0970	$AVRS requires a minimum release of the operating system on the mainframe computer. The operating system does not meet the minimum required. Contact SEA for technical support.
0999	Call to a $AVRS program from a module with an invalid parameter list. $AVRS modules are ineligible to be called by non-$AVRS programs.
4095	The $AVRBLKL Block Lettering program has an internal error. Call SEA for technical support.
1nnn	nnn = the return code from CAMLIST LOCATE SVC when issued to determine the dataset name for the backup file.

APPENDIX B

$AVRS MESSAGES

$AVRS Messages

$AVRS Messages are also available in the Installation Library in the QUIKHELP member.

This member contains the $AVRS 5.1B messages in a format that can be used as input into the Chicago-Soft, LTD MVS/QUICKREF System (for instructions on building a User database, refer to the MVS/QUICKREF Manual). All the messages in this file may not be included in the MVS/QUICKREF Release currently used at your site.

The **$AVRS** system identifies messages with the following 4 header formats:

Format #1

- **$AVR*nnnx*:** where *nnn* is the numeric identifier and *x* is one of the following message suffix codes:

Code	Meaning
A	**ACTION** message - *action is required.*
I	**INFORMATIONAL** message - *no action required.*
E	**ERROR** message - *run usually cannot continue*.
W	**WARNING** message - *not serious, run continues, user should consult documentation for description of the message*.
C	**CRITICAL** message - *run cannot continue.*
U	**USER** message - *this message issued via USER control, usually within a User Exit.*

NOTE: Message suffices may be different than listed in this appendix when issued by several different $AVRS programs. A message may be a critical message (suffix C) for program A and an information message (suffix I) for program B.

- Messages that are generated by the new format message method will have the issuing modules name in parenthesis following the message number as shown below:

  $AVR680C ($AVRMRGI) Above errors have occurred.
  Correct and rerun the job.
- *Italicized* character strings represent variables in the $AVRS messages. For example, *##* is a numeric variable
- The text is one of the messages defined in the following list. An explanation and, if required, and when possible a suggested correction is given.

Format #2

- **$AVRFM*nn*:** where *nn* is the numeric identifier:
  $AVRFM01 - SYST1 FAILED TO OPEN SUCCESSFULLY
  This type of message header comes from $AVRFMRG, the $AVRS Forward Merge Program

Format #3

- **$CRDnnnX**: where nnn is the numeric identifier and X is one of the following message suffix codes:

Code	Meaning
C	**CRITICAL** message - *Initialization cannot continue. User response required.*
I	**INFORMATIONAL** message - *no action required.*

The following message members appear as prefixes to your initialization parameters in the $AVRERR initialization report.

For example:

$CRD000 ACCDEST=U513
This type message number suffix of $CRD000I would indicate that the accdest parameter of U513 is valid on your system.

$CRD001 ACCDEST=U513
This type message number suffix of $CRD001C would indicate that the accdest parameter of U513 is invalid on your system. This is a critical error that will stop your task initialization.

The $CRD001C message at the bottom of the $AVRERR initialization report lists the text of the error message. For additional help, the text of the error message, a more detailed explanation of the error, and the users response is in the Messages Appendix that follows.

Format #4

- **UPFLnnnX:** where *nnn* is the numeric identifier and *X* is one of the following message suffix codes:

Code	Meaning
C	**CRITICAL** message - *UPFL cannot continue*.
E	**ERROR** message - *UPFL usually cannot continue.*
I	**INFORMATIONAL** message - *no action required.*
W	**WARNING** message - *not serious, run continues, user should consult documentation for description of the message.*

- Messages that are generated by the new format message method will have the issuing modules name in parenthesis following the message number as shown below:

UPFL203C (UPFLBFMT) **ATTEMPTING TO FORMAT A NON-EMPTY KSDS FILE - TERMINATING**

- Italicized character strings represent variables in the UPFL messages. For example, ## is a numeric variable.

- The text is one of the messages defined in the following list. An explanation if required, and when possible a suggested correction is given.

- The UPFL messages are grouped by category into numeric ranges. The ranges by numeric category are listed below:

  UPFL000x - UPFL099x messages are intended to be displayed when UPFL is operating in the test mode. If any of them is displayed, it indicates that it inadvertently did not get removed prior to shipping the product for this release. These messages should be reported to SEA so they can be eliminated in the next release.

  UPFL100x - UPFL199X messages are produced when there are problems detected in the UPFL KSDS database by the UPFL access method programs. They may occur in either the UPFL batch system or the application online system.

UPFL200x - UPFL299x messages are produced when there are problems detected by the UPFL batch programs. They may occur only in the UPFL batch system.

UPFL300x - UPFL399x messages are produced when there are problems detected by the UPFL online programs. They may occur only in the UPFL online system. The User Profile online system will be deactivated for the remainder of this session and the application ($AVRS, TRMS, etc.) System will attempt to restart without the User Profile System being activated.

UPFL390x UPFL399x messages are a subset of the UPFL300x thru UPFL399x and are produced when there are problems detected by the UPFL online interface logic. They are displayed because the UPFL facility is either not available under ISPF or a UPFL error has occurred. The environment area you are using has specified a non-blank UPFLDSN keyword which causes the application programs to attempt to activate UPFL. The User Profile online system will be deactivated for the remainder of this session and the application ($AVRS, TRMS, etc) system will attempt to restart without the User Profile System being activated.

UPFL400x - UPFL499x messages are produced by errors detected by UPFL common programs (programs that can operate in both the batch system and the online system.) They may occur in either the UPFL batch or the online application.

UPFL500x - UPFL999x messages in the range UPFL500 through UPFL999 are not used in the current release of the UPFL system.

The complete text of the User Profile messages, the explanation, system response, and user response are listed in the **UPFL** manual.

Substitutional Variables

The following is a list of the Substitutional Variables used in $AVRS messages. They will appear in the $AVRS MESSAGES APPENDIX as lower case and italicized.

#.##	The $AVRS release is 5.1B.
#.##-##	The $AVRS release is 5.1B.10 maintenance level 10.
##	A decimal number.
aaaa	An alphabet or alphanumeric value.
addr1	Program address number 1, number 2, etc.
block type	One of two types of block numbers: RRDS High Performance block number-HIPERBLK# or RRDS DATA block number - DATABLK#
card image	The problematic sysin input is shown.
chain type	One of three chain types: RRDS High Performance block/report chains-HIPO BLKCHAIN or HIPO RPTCHAIN. RRDS Data chains - RRDS BLKCHAIN.
class	The specific sysout class specified.
classes	The multiple sysout classes specified.
code	The abend code involved.
command	The command issued.
criteria 1	Criteria 1, criteria 2, etc.
csect	The program csect where the error occurred.
dataset name	The specific dataset name involved.
day	The day of the week - ie Monday.
ddname	The JCL DDNAME.
destination	The operating system specified destination ie R10.
enque name	The $AVRS enque names ie ENQ$AVRS, $AVRACTV, etc.
error code	The EC code returned to $AVRS from an attempted open, close, etc.
f1f2f3f4	The hexadecimal SDWA input flags.
fdbk	The feed back code from the function being attempted.
file name	The $AVRS file name ie KSDS, RRDS, etc.
file status	The file status or action - open or closed.
file type	The KDSD, LOCK or RRDS type file.
fmid	The current operating system FMID.
form	The operating system specified forms name or number.
function	The function being attempted when the problem occurred - ie open.
generic key	The KSDS generic key being accessed when the problem occurred.
information code	The IC code returned to $AVRS from the failing function.
jes file	The name of the JES file involved in the problem.
job id	The JES job number involved in the problem.
job name	The JES job name involved in the problem.
job number	The JES job number involved in the problem.
location	The program label name where the problem occurred.
member name	The JES file name involved in the problem.
mm ddd yyy, mmddyyy	One of several Y2K compliant date formats.
module	The program name involved in the problem. This may or may not be a $AVRS program.
nnn	A decimal number. N should only be used as # is used to indicate a number previously in the same message.
offset	The location in storage where the problem occurred.
panel	The panel name involved in the problem.

parameter	The specific parameter in error is, for example - ACCMCLS.
pool name	The name of the pool involved in the reported error.
proc name	The JCL procedure name involved in the problem.
profile record key data	The KSDS key currently being used.
program name	The program name involved in the problem. This may or may not be a $AVRS program.
psw	The Program Status Word at the time of abend.
rc	The return code passed or issued by $AVRS for a failing program or function.
reason	The abend reason code
record length	The record length of the KSDS record.
rel#	The $AVRS release number ie, 5.1B.
routine name	The program routine involved in the reported error
selection card image	The problematic sysin input is shown.
step name	The JCL step name from the job.
stat	A value indicating where the initialization failed.
status	The RRDS file status is Active, Drained, or Remove.
stc name	The JCL task name.
stc number	The task job number.
storage length	A getmain value of zero or greater than 32K was requested.
system	The current operating system
ttt	This indicated time in the date format.
type	The type of work - either STC, JOB, TSU, LOG, FILE or REPORT.
userid	The users sign on identifier.
vaddr	The virtual address causing the translation exception.
variable name	The name of the variable involved in the reported error.
xxxx	Variable message text specific to this problem or it may be user specified text
yyyy.ddd *yyyyddd,hhmmss*	One of several Y2K compliant date formats.
yyyy-mn-dd	One of several Y2K compliant date formats.
X'##'	The format of hex error code. It can be 1 to 4 hex digits.

$AVR001I - $AVRS HAS SUCCESSFULLY TERMINATED

Explanation: The $AVRS task has ended in a normal or controlled manner.

System Action: $AVRS terminates normally. It does not abend.

User Response: No operator action is required. Restart $AVRS when desired.

$AVR001C - INSUFFICIENT VIRTUAL STORAGE FOR LOADING PASSWORD MODULE

Explanation: The region size is too small to contain the user password module.

System Action: Abend U0001 is issued.

User Response: Increase the region size by the size of the assembled module and restart $AVRS.

$AVR002I This message is accompanied by one of the following phrases

S 1) *type* RECORDS PROCESSED FOR *function=##*

Explanation: The message is issued by $AVRCLEN to report statistics on its processing.
type of records is FILE, JOB, or REPORT.
function is RECOVER, VALIDATE or DELETION
is the number of base (x'01-06') file (x'65'), and acknowledgment (x'67') KSDS records. The x'00' X'66' records are not processed.

System Action: None.

User Response: None.

S 2) INVALID CONTROL STATEMENT, IGNORED: *card image*

Explanation: The input statement is invalid, *card image* is the SYSIN Control Statement(s) that is in error.

System Action: None.

User Response: Correct errors and re-run.

$AVR002E - CONTROL STATEMENT IN ERROR - statement

Explanation: The control statement containing an error is printed as an aid in error resolution.

System Action: Typically, a non-zero return code is returned.

User Response: Correct the error.

$AVR002E - THE CONTROL STATEMENT MUST BEGIN WITH EITHER VALIDATE OR RECOVER

Explanation: The input control statement is not correct.

System Action: The job terminates with a return code of 4.

User Response: Correct the input control statement and re-submit the job.

$AVR003C - GETMAIN FAILURE IN PROGRAM *program name* AT LOCATION *location*

or

- (*module*) GETMAIN ERROR AT OFFSET X ‹xxxx’

Explanation: A GETMAIN error has prevented a $AVRS program from obtaining necessary storage.

System Action: Current $AVRS function is terminated.

User Response: Ensure that any necessary region requirements for this function have been met. If not, increase the region size and retry the function. Otherwise, call for technical support.

$AVR006C - "BKUPTIM" PARAMETER NOT SPECIFIED

Explanation: The initialization parameter "BKUPTIM" was not specified.

System Action: Abends U0102.

User Response: Specify a value for "BKUPTIM" and restart $AVRS.

$AVR007C - MUST BIT SET BUT MESSAGE ADCON FOR MESSAGE EQUAL TO ZERO.

Explanation: The internal check for control parameters is in error.

System Action: Abends U0102.

User Response: Call for technical support.

$AVR008C - "ACCMCLS" "*class*" MUST NOT BE EQUAL TO ANY "REQCLAS" ENTRY OR THE "MAXLCLS"

Explanation: An ACCMCLS= value was equal to a REQCLAS or MAXLCLS value.

System Action: Abends U0102.

User Response: Specify a different class for one of the parameters and restart $AVRS.

$AVR009C - "ERRSRTN = NO", BUT SCAN HAS RESPINS. NO RESPINS WILL BE PRINTED.

Explanation: The "ERRSRTN" initialization parameter specified no respins of any accumulated work, but the scan argument definition records have potential respins.

System Action: Abends U0102.

User Response: Correct the inconsistency between the "ERRSRTN" parameter and the scan argument definitions. Restart $AVRS.

$AVR010C - LOCATE FOR BACKUP DSN FAILED WITH RETURN CODE *X'##'*

Explanation: A LOCATE for the backup DSNAME failed.

System Action: Abends U0102.

User Response: Consult OS/VS2 SYSTEM PROGRAMMING LIBRARY: DATA MANAGEMENT for an explanation of the return code.

$AVR011C - NO GENERATION INDEX EXISTS FOR BKUPDSN= ***dataset name***

Explanation: A LOCATE for the backup generation dataset name DSNAME failed.

System Action: Abends U0102.

User Response: Check the DSN in the backup controls. Correct the error and rerun the backup.

$AVR012C - "ACCMCLS" PARAMETER NOT SPECIFIED

Explanation: The initialization parameter "ACCMCLS" was not specified.

System Action: Abends U0102.

User Response: Specify a value for "ACCMCLS" and restart $AVRS.

$AVR016I - $AVRS RELEASE *#.##-##* INITIALIZATION COMPLETE

Explanation: All initialization parameters are valid and have been accepted.

System Action: Accumulation begins when work is available.

User Response: None required.

$AVR017C - SYSLOG CLASS "*class*" FOR "SID=*aaaa*" MUST BE DECLARED AS ONE OF THE "ACCMCLS" ENTRIES

Explanation: The class assigned in the LOGCLAS=class, SID=aaaa parameter was not an accumulation class assigned on the ACCMCLS= parameter.

System Action: Abends U0102.

User Response: The SYSLOG CLASS indicated must be specified as a valid ACCMCLS.

$AVR018C - THE $AVRS ENVIRONMENT DOES NOT HAVE APF AUTHORIZATION

Explanation: $AVRS initialization was attempted from a load library that did not exist in the current list of APF libraries at the time of the last IPL, or authorization code 1 has been removed from the $AVRINIT load module.

System Action: Abends U0102.

User Response: Ensure that the $AVRS LOADLIB is in the IEAAPFxx list, or ensure that an unauthorized library is not in the steplib concatenation.

$AVR019C - INVALID REPORT DESTINATION FOR THIS COMPLEX DESTINATION=*destination*

Explanation: The destination supplied on the "RPTDEST=" initialization parameter is invalid for this particular complex.

System Action: Abends U0102.

User Response: Modify the initialization parameter "RPTDEST."

$AVR020C - READ OR WRITE ERROR $AVRS STOPPING

Explanation: A physical read or write error occurred on one of the VSAM files.

System Action: Message $AVR021C will also be issued to show the exact file and type of error. Abend U0020 is issued.

User Response: If the VSAM file must be relocated, first run $AVRUNLD as a batch job before the existing files are altered; second, redefine the VSAM databases; then run $AVRFMRR program using the file fromthe DDNAME $AVRUNLD that was created by the $AVRUNLD job. After the above steps are performed, $AVRS may be restarted.

$AVR021C - 'up to 128 characters of VSAM SYNAD *message text*'

Explanation: Physical error returned by VSAM.

System Action: The SYNAD message text that is formatted by VSAM is displayed as the text for this message.

User Response: Correct the problem and retry the function.

$AVR022C - (*module*) *function* FAILED FOR *dataset name* WITH RC=X‹##', FDBK=X‹##'

or

- function FAILED RC=X'##' FDBK=X'##' IN MODULE *module* AT LOCATION *location*.

Explanation: A VSAM logical error occurred. *Function* is the VSAM function, *module* is the issuing module, *dataset name* is the DDNAME, and *location* is the location within the module.

RC = the register 15 return code.
FDBK = the feedback code associated with the "RC"

System Action: Abends U0020, U0022.

User Response: The location is given to aid in resolution if vendor support is required. Please have message text at hand if vendor support is required. If running $AVRFMRR check KSDS, RRDS, and lock dataset names in the JCL. Retry with corrected names.

$AVR023C - *function* FAILED WITH CODE X'##'IN MODULE *module* AT LOCATION *location.*

Explanation: A VSAM logical error occurred for either MODCB or SHOWCD.

module is the issuing module and location is the location within the module.

RC = the register 15 return code.

System Action: Abends U0020.

User Response: The location is given to aid in resolution if vendor support is required. Please have message text at hand if vendor support is required. Restart $AVRS.

$AVR024C - *programname,location,module,location,function,*RC=X'##', FDBK=X'##",R0=X'##'

Explanation: An I/O error has occurred in a $AVRS program at the location and CSECT indicated..
RC = the register 15 return code.
FDBK = is the feedback code returned from $AVRSAM.
R0 = is the error code returned from $AVRSAM.

System Action: Abends U0024.

User Response: Check the error codes for possible VSAM or $AVRSAM error and correct as indicated.

See Appendix C - $AVRS Access Method Return and Feedback Codes for descriptions.

If you are unable to correct the error vendor support may be required. Please have message text at hand if vendor support is required.

$AVR025C - BOTH BACKUP AND DUPLEX DSNS ARE NOT GDGS OR NON-GDGS

Explanation: The catalog structure of the BACKUP dataset and the DUPLEX dataset are not the same.

System Action: Abends U0102.

User Response: The datasets must *both* be either generation datasets, or neither of them may be generation datasets. Correct the datasets' catalog specification and restart the task.

$AVR026C - THERE IS CURRENTLY AN ACTIVE $AVRS FOR THIS DATA BASE:

Explanation: The accumulation program $AVRINIT or the dead block recovery program $AVRCLEN tried to do an ENQUEUE on the $AVRS LOCK and KSDS datasets and the ENQUEUE failed.

System Action: Abend U0026.

User Response: The $AVRS accumulator must be stopped before attempting this procedure. Or, the $AVRS accumulator will not start because the $AVRCLEN program is executing. Wait until $AVRCLEN completes execution and restart the $AVRS accumulator.

$AVR027C - "*enque name*" ENQ FAILED FOR THE LOCK FILE: *dataset*

Explanation: There was an attempt to ENQUEUE on the $AVRS LOCK dataset and the ENQUEUE failed.

System Action: Abend U0026**.**

User Response: The $AVRS accumulator must be stopped before attempting this procedure. Or, the $AVRS accumulator will not start because the $AVRCLEN program is executing. Wait until $AVRCLEN completes execution and restart the $AVRS accumulator.

$AVR029I - stc name(*stc num*) $AVRRRDS IS AT ##.#% UTILIZATION

Explanation: The RRDS data file(s) for $AVRS (stc name(stc num)) has reached the percentage value specified for the initialization parameter WARN=. The ##.#% in the message is the actual percentage of the data file in use. This message is repeated a minimum of every WARNINT= minutes.

System Action: Informational.

User Response: None needed. When $AVRS reaches 100% utilization, it automatically does an emergency backup and tries to delete what work it can. This message is intended to notify the user of a specified percentage so the user may do another type of backup and avoid the emergency one.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - ALLOCATE ERROR RC=X'##',EC=X'####',IC=X"####' DDNAME=*dataset name*

Explanation: Dynamic allocation (svc99) failed.

RC = register 15 return code
EC = SVC99 error code
IC = SVC99 informational code (usually text unit key)
DDNAME = the DDNAME for which the allocation failed

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Consult the IBM OS/390 Authorized Assembler Services Guide for further diagnosis of the message text. Restart $AVRS.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DATASET HAS NO DSORG SPECIFICATION

Explanation: The dataset name specified in this request does not have a valid dataset organization.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Specify a valid DSORG and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DATASET NAME NOT FOUND

or

- DSNAME NOT FOUND - DSN= *dataset name*

Explanation: The dataset name specified in this request was not found.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Correct the dataset name specified and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DATASET NAME IS ALREADY CATALOGED

Explanation: The dataset name specified in this request is already known to the system.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Either specify a different dataset name or change the DISP to OLD and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DATASET NAME SPECIFICATION IS INVALID

Explanation: The dataset name specified in this request is invalid.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Correct the dataset name value and do a PRINT REQUEST.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DATASET NAME UNAVAILABLE

Explanation: The dataset name specified in this request is unavailable to $AVRS.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Check that the dataset is not currently in use by something else or specify another dataset name.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DEVICE NAME AND VOLUME REQUESTED ARE INCOMPATIBLE

Explanation: The device name and volume are not compatible.

System Action: The job is terminated with an Abend U0030.

User Response: Correct the unit and volume to make them compatible. Most common is the UNIT=value specified in the $AVRXXEA if the failure is in online.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DIRECT ACCESS DEVICE IN NOT AVAILABLE

Explanation: The volume requested was not available for allocation.

System Action: The error is displayed and the task is terminated.

User Response: Be sure you have access to the device requested or that the device is eligible for allocation by you.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DSN FROM $AVRFSEA ASSEM NOT FOUND DD=*ddname* [SMS=smsrc] or [ERR=svc99rc]

Explanation: Dynamic allocation failed for the specified DDNAME.

System Action: The job terminates.

User Response: Determine if the dataset exists. The "smsrc" is the DFSMS failure reason code. The DFSMS reason code can be found in DFSMS/MVS Macro Instruction for Data Sets manual. The "svc99rc" is the SVC99 reason code. The SVC99 reason code can be found in the OS/390 MVS Authorized Assembler Services Guide manual. Supporting detailed error $AVR589I messages can be found on your SYSLOG. Update your $AVRFSEA data file with the correct dataset name. Assemble and linkedit your updated $AVRFSEA data file. Re-try this function again. If unable to proceed, call Tech Support at SEA.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - DUPLICATE NAME ON VOLUME

Explanation: The dataset name specified in this request is already known to the volume.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Either specify a different dataset name, change the DISP to OLD, or specify a different volume and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - INVALID "UADS" UNIT NAME AND $AVRFSEA UNIT NOT DEFINED

Explanation: An invalid UNIT name has been specified in the UADS dataset and no UNIT name was specified as a $AVRS installation default.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Either correct the UADS entry or specify a $AVRS default in the $AVRFSEA macro and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - REQUESTED ALLOCATION WAS DENIED BY AN OPERATOR

Explanation: The operator denied the allocation request.

System Action: This message is issued and a user Abend U0030 for the Batch. The error is displayed in a panel and the task is terminated.

User Response: Determine the reason the operator responded negatively to allocation request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - REQUESTED VOLUME NOT AVAILABLE

Explanation: The volume requested for allocation was not available.

System Action: The error is displayed and the task is terminated.

User Response: Make sure you have access to the device requested or that the device is eligible for allocation by you.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - REQUIRED VOLUME NOT ON AN AVAILABLE DEVICE

Explanation: You requested a volume for which you have no access authority.

System Action: The error is displayed and the task is terminated.

User Response: Obtain authority for the volume/unit that is required to complete the allocation.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - SMS PARAMETER SPECIFICATION ERROR

Explanation: One or more of the System Managed Storage parameters specified for this request is invalid at this time.

System Action: The Dataset Allocation Panel is redisplayed.

User Response: Correct any typographical errors in the SMS fields, verify that SMS is up and running and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - SPACE REQUIRED NOT AVAILABLE ON ELIGIBLE VOLUME

Explanation: Not enough space was found on an eligible volume to satisfy this request.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Change the amount of space requested or the volume list specified and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - UNIT SPECIFICATION IS INVALID

Explanation: An invalid UNIT= value has been specified.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Correct the UNIT= value and do a PRINT RESET, “CPRT’.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030C - VOLUME REQUESTED IS UNAVAILABLE OR NOT MOUNTED

Explanation: The VOLUME parameter specified in this request is invalid at this time.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Either choose a different volume or ready the volume desired and retry the request.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030E - (*module*) ALLOCATION ERROR OF KSDS OR RRDS FILES FAILED ($AVRALLC)

Explanation: $AVRS was unable to allocate one or more of the data base files.

System Action: The job is terminated with an abend code of U0030.

User Response: Look for other messages that may provide additional information. If a Format or Format/Reload ($AVRFMRR) was recently done, verify that all step condition codes were zeros.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030I - DESTINATION SPECIFIED IS UNDEFINED ON THIS JES CONFIGURATION

Explanation: An invalid DEST= value has been specified.

System Action: Abend U0030 or, when using a full screen interface, the abend may not occur.

User Response: Correct the DEST= value and do a PRINT RESET,"CPRT".

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR030I - GETMAIN FOR BASEKEY FOR JOBNAME *jobname*, JOB# *jobnumber*

Explanation: Reference $AVR003 error message.

System Action: Reference $AVR003 error message.

User Response: Reference $AVR003 error message.

NOTE: THERE ARE MULTIPLE $AVR030 MESSAGES. PLEASE MATCH THE TEXT BEFORE DETERMINING THE USER RESPONSE.

$AVR031C - ALLOCATE|DEALLOCATE ERROR RC=X‹##', EC=X‹####', IC=X‹####', DDNAME=*ddname* [DEST=*dest*], SMSTEXT, or SVC99*error text*, SMSRC or SVC99*rc*

Explanation: Dynamic allocation failed for the specified DDNAME.

System Action: The job terminates with U0030 during batch processing. During online processing, the request is ignored.

User Response: Determine if the dataset exists. Re-try this function again or attempt another $AVRS function. If unable to proceed, call Tech Support at **SEA.**

$AVR042C - USER EXIT RETURNED A PRINT CLASS WHICH DUPLICATES "ACCMCLS=" VALUE

Explanation: A call to user exit $AVRUX8 has resulted in $AVRS trying to respin sysout into a sysout class that $AVRS accumulates. If $AVRS were to allow this, it would create an infinite loop. $AVRS can NOT respin sysout to a sysout class that it also accumulates.

System Action: Abend U0042.

User Response: Either change the accumulation or the respin classes. Restart $AVRS.

$AVR044C - (module) OPEN ERROR FOR DDNAME ddname

Explanation: Open failed for the specified DDNAME.

System Action: Abend U0044

User Response: Check the JCL to insure that all required JCL statements are present and that the DDNAMEs are spelled correctly and re-run.

$AVR045C - INVALID PARM SPECIFIED - MUST BE BLANK OR BEGIN WITH "ADD"

Explanation: The parm value is not valid.

System Action: Abend U0045

User Response: Correct the PARM= value on the JCL EXEC Statement for $AVRFMRR and re-run.

$AVR046I - (*module*) INVALID FUNCTION REQUESTED - NO ACTION PERFORMED

Explanation: Invalid request (parm) passed to the $AVRS Open/Close module

System Action: No Action was performed

User Response: Notify Tech Support at **SEA**

$AVR047W - (*module*) INVALID RECORD ENCOUNTERED - RECORD IGNORED

Explanation: Invalid record was encountered during the reload of work to the $AVRS data base.

System Action: The record was ignored, no attempt was made to add it to the data base.

User Response: No action required

$AVR048I - $AVRS *data set name* HAS BEEN CHANGED FROM [ACTIVE|DRAINED] TO [ACTIVE|DRAINED]

Explanation: The status of an RRDS file has change, based on the request issued via the modify command.

System Action: Depending on the status of the RRDS, the system will either allow (ACTIVE) work to be placed on the RRDS or not allow (DRAINED) new work to placed on the RRDS.

User Response: No action required.

$AVR049I - $AVRS THE REQUEST TO [SETACTIVE|DRAIN] AN RRDS DATA FILE REQUIRES AN INDEX VALUE - IGNORED.

Explanation: A modify command to change the status of an RRDS Data File was issued without the required index value.

System Action: The request was ignored.

User Response: Use the "F *$avrstaskname,*STATUS,RR" console command to obtain a status of all defined RRDS data files. Then, using the INDEX Value associated with the specific RRDS in question, re-enter the original command.

$AVR050W - job name (*job number*), HOLDS "*Function*" LOCK NEEDED BY *job name*2(*job number*)

Explanation: Job "jobname2 (jobnumber)" is trying to lock the $AVRS database but the lock is already held by job "jobname (jobnumber)".

System Action: Function will be replaced by the name of the function trying to lock the database.

Valid values for **Function** are:

ACCM - The accumulation task.
BKUP - A backup task.
UNLD - A database unload job.
FORM - The database format job.

Message **$AVR051A** follows.

User Response: Respond to message $AVR051A if issued.

$AVR051A - REPLY "CONT" TO FORCE OR "STOP" TO QUIT

Explanation: A preceding message has indicated a lockout condition on the $AVRS database between two tasks.

System Action: The issuing task is now waiting for direction before continuing.

User Response: Reply "CONT" to force the lock to be reset for this task or "STOP" to terminate this task. A "CONT" reply should be used only to recover from a previous abnormal termination and **extreme care** should be exercised to ensure no other active $AVRS task is holding this lock.

$AVR051W - REPLY "STOP" TO TERMINATE THIS TASK
OR
CALL TECH SUPPORT, BEFORE REPLY OF "CONT" OR "STOP"

Explanation: A preceding message has indicated a lockout condition on the $AVRS database between two tasks.

System Action: The issuing task is now waiting for direction before continuing.

User Response: This is a customized version of message $AVR051A - follow the rules established by your shop, or use the instructions that follow. Reply "CONT" to force the lock to be reset for this task or "STOP" to terminate this task. A "CONT" reply should be used only to recover from a previous abnormal termination and **extreme care** should be exercised to ensure no other active $AVRS task is holding this lock.

$AVR052I - INVALID REPLY SPECIFIED

Explanation: An invalid reply has been given to a request from a $AVRS task.

System Action: Message $AVR051 is reissued.

User Response: Retry with the correct reply.

$AVR053C - UNLOCK/LOCK FUNCTION HAS FAILED

Explanation: An attempt to unlock /lock the $AVRS database has failed.

System Action: Execution is terminated.

User Response: Determine what other $AVRS job is holding the LOCK. If the job has completed or abended by now, use the $AVRLOCK program to reset the lock indicated. Otherwise, wait for the job to finish. - See message $AVR050W.

$AVR054I - TERMINATING DUE TO REPLY OF STOP TO $AVR051W

Explanation: A reply of "STOP" was given to a lock warning message.

System Action: The task is terminating.

User Response: None required.

$AVR057I - $AVRS ACTIVE RRDS DATA FILE(S) UTILIZATION: ##%

Explanation: Displays the percent of data blocks used for ACTIVE RRDS Files

System Action: No Action Required

User Response: No Action Required

$AVR058I - ALL RRDS DATA FILES: ACTIVE BLOCKS FORMATTED=nnnn, ACTIVE BLOCKS USED=nnnn

Explanation: Informational message providing the total number of formatted and used RRDS data blocks for the “Active” RRDS files.

System Action: None.

User Response: None.

$AVR059I - $AVRS REQUEST TO [SETACTIVE|DRAIN] RRDS index # IGNORED - THE FILE IS IN “REMOVE” STATUS AND CANNOT BE CHANGED VIA A MODIFY CMD

Explanation: An attempt was made to change the status of an RRDS which was in “REMOVE” status. The status of an “REMOVED” RRDS cannot be altered.

System Action: The command was ignored

User Response: Verify that the INDEX # entered was correct.

$AVR060C - FUNCTION FAILED, PROGRAM= *program name*, LOCATION= *location*

Explanation: An I/O request against the $AVRS database has failed. *function* indicates the type of operation requested, *program name* indicates what program issued the message, and *location* indicates where in the program the error was detected.

System Action: This message is accompanied by message $AVR061C, message $AVR062C, message $AVR063C, and message $AVR064C. Abends U0060.

User Response: Collect the error codes from message $AVR062C and refer to Appendix C - $AVRS Access Method Return and Feedback Codes for a description of the codes and instructions on correcting the error.

$AVR061C - $AM MODULE=*module*, LOCATION=*location*

Explanation: An I/O request against the $AVRS database has failed. *module* indicates the $AVRSAM module that detected the error and *location* indicates where in the module the error was detected.

System Action: This message is accompanied by message $AVR060C, message $AVR062C, message $AVR063C, and message $AVR064C.

User Response: Collect the error codes from message $AVR062C and refer to message Appendix C - $AVRS Access Method Return and Feedback Codes for description of the codes and instructions on correcting the error.

$AVR062C - REG15=X'##', FDBK=X'####', REG0=X'##'

Explanation: An I/O request against the $AVRS database has failed.
REG15=X'##' indicates the contents of register 15 at the time of the error.
FDBK=X'####' indicates the VSAM or $AVRSAM feedback code
REG0=X'##' indicates the contents of register 0 at the time of the error.

System Action: This message is accompanied by message $AVR060C, message $AVR061C, message $AVR063C, and message $AVR064C.

User Response: Refer to message Appendix C - $AVRS Access Method Return and Feedback Codes for a description of the codes and instructions on correcting the error.

$AVR063C - MESSAGE = (*Variable VSAM synad message*)

Explanation: An I/O request against the $AVRS database has failed. **MESSAGE** = shows any SYNAD error message associated with the error.

System Action: This message is accompanied by message $AVR060C, message $AVR061C, message $AVR062C, and message $AVR064C.

User Response: Collect the error codes from message $AVR062C. Refer to message Appendix C - $AVRS Access Method Return and Feedback Codes for a description of the codes and instructions on correcting the error.

$AVR064C - Continuation from $AVR063C above

Explanation: An I/O request against the $AVRS database has failed. This is a continuation of the SYNAD message displayed in message $AVR063C.

System Action: This message is accompanied by message $AVR060C, message $AVR061C, message $AVR062C, and message $AVR063C.

User Response: Collect the error codes from message $AVR062C. Refer to message Appendix C for a description of the codes and instructions on correcting the error.

$AVR065C - NO RRDS ALLOCATED - UNABLE TO CONTINUE. THE LOCK FILE MAY BE DAMAGED OR INCORRECT.

Explanation: $AVRS was unable to allocate the RRDS files based on the information contained in the $AVRS LOCK file. The second sentence of the message may not display when using $AVRS online.

System Action: $AVRS terminates.

User Response: Verify that the LOCK file DD Statement is correct. Verify that the RRDS file(s) are available and have not been deleted.

$AVR066C - *module* MODULE NOT FOUND

Explanation: The $AVRS load module specified by *module* was not found when $AVRS tried to load it. This usually identifies a optional $AVRS module that applies only when certain customizations are applied to the product.

System Action: Processing continues, but any special processing done by the module identified is skipped.

User Response: Verify the module is in the $AVRS load library. If not, assemble and link the specified module from the $AVRS installation library. If you can not find the module in the installation library, contact SEA for assistance.

$AVR067C - YEAR 2000 EXPDT FAILED, *dataset name* SET TO PERMANENT RETENTION

$AVR067E - YEAR 2000 EXPDT FAILED

Explanation: $AVRS tried to dynamically allocate a dataset that had an expiration date on or after January 1, 2000. The operating system rejected the request and pointed to the expiration date as the reason for the failure.

System Action: $AVRBKUP continues processing by setting the backup file identified by "dataset name" to a permanent expiration date (99366). The online function of outputting to a dataset terminates.

User Response: Your operating system is not year 2000 enabled. Contact your system programmer for an upgrade. You will not be able to use $AVRS for year 2000 processing until the operating system supports it.

$AVR068E - THE (*key maintenance*) KEY MAINT FROM JOB *jobname, jobnumber* IS FROM A PRIOR RLSE OF $AVRS: RCDS DELETED

Explanation: The $AVRS accumulator attempted to accumulate $AVRS Key Maintenance Records created with a prior release of $AVRS.

System Action: The maintenance records are ignored and deleted.

User Response: Re-run the job that created the key maintenance and insure that the Key Maintenance records are directed to the correct JES class associated with the $AVRS data base where the maintenance records are created. Determine why this error occurred to prevent possible deadblocks or loss of data in your database.

$AVR069E - THE *key maint* RCDS FROM JOB *job name, job number*
$AVR070E - WAS CREATED FOR THE $AVRS DATABASE WITH LOCK=*lock dsname*. THE LOCK DSN FOR THIS $AVRS DATABASE IS lock *dsname*. RECS DELETED

Explanation: $AVRS maintenance was created for one $AVRS data base but was eligible for accumulation to another $AVRS database with the same accumulation attributes.

NOTE: $AVR070E is a continuation of $AVR069E.

System Action: The maintenance records are ignored and deleted.

User Response: Re-run the job that created the key maintenance and insure that the Key Maintenance records are accumulated by the Correct $AVRS database. Determine why this error occurred to prevent possible deadblocks or loss of data in your database.

If the keys were created by $AVRSELER, insure that the selective restore proc contains the correct LOCK file. Also consider changing the $AVRFSEA parameter to PROCDEF=YES. This will refresh the AH job submission panel procedure name with the JOBPROC value.

$AVR070E - SEE $AVR069E ABOVE

$AVR071I - JOB= *job name*,N=*job number*, HAD NO DATA FILES - DROPPED

Explanation: During the database unload function, it was discovered that this job did not have any data files associated with the base key.

System Action: This base key has been dropped from the unload process. The return code is set to '4'.

User Response: None required.

$AVR072I - JOB *job name,job number* HAS # OF # DATA FILES MISSING

Explanation: During the database unload function, it was discovered that this job had data files missing. This is usually the result of 'forced' deletions.

System Action: The unload continues.

User Response: None required.

$AVR073W - THE SSIB JES NAME WAS NOT LOCATED

Explanation: The JES2 spool interface does not match the $AVRHMAP.

System Action: $AVRS continue to accumulate with incorrect job and file attributes.

User Response: Reassemble the $AVRHMAP with the JES2 macro libraries that are in the same order and same PTF level as those used by your JES2 operating system. After reassembly stop and restart the $AVRS task. Failure to correct this error will result in attribute fields with incorrect, blank, or garbage values.

$AVR074W - UNABLE TO LOCATE SJB, NO ATTRIBUTES WILL BE COLLECTED

Explanation: The JES2 spool interface does not match the $AVRHMAP.

System Action: $AVRS continue to accumulate with incorrect job and file attributes.

User Response: Reassemble the $AVRHMAP with the JES2 macro libraries that are in the same order and same PTF level as those used by your JES2 operating system. After reassembly stop and restart the $AVRS task. Failure to correct this error will result in attribute fields with incorrect, blank or garbage values.

$AVR075W - PSO POINTER IS ZERO IN THE SJB FOR THE TASK

Explanation: The JES2 spool interface does not match the $AVRHMAP.

System Action: $AVRS continue to accumulate with incorrect job and file attributes.

User Response: Reassemble the $AVRHMAP with the JES2 macro libraries that are in the same order and same PTF level as those used by your JES2 operating system. After reassembly stop and restart the $AVRS task. Failure to correct this error will result in attribute fields with incorrect, blank or garbage values.

$AVR076I - INVALID INITIALIZATION PARAMETER: "XXXXXXXX". DEFAULT: "YYYYYYY" USED.

Explanation: During $AVRS initialization processing, $AVRS detected an invalid initialization parameter "XXXXXXX". $AVRS is using the default value "YYYYYYY" for processing.

System Action: $AVRS continues processing with the default parameter.

User Response: None required.

$AVR077W - $AVRS WILL NOT BE POSTED WHEN WORK IS AVAILABLE

Explanation: If the JESPOST initialization parameter has been set to NO, this highlighted message is issued when $AVRS initializes. It is a reminder that $AVRS will not check if any work becomes available until its sleep/wait time interval is complete, every one minute.

System Action: None.

User Response: None - a warning message only. It acknowledges the user chosen parameter value and lets the user know that $AVRS may not be accumulating because it is in a controlled sleep/wait state.

$AVR101W - OPEN FAILED FOR INITIALIZATION PARAMETERS

Explanation: The SYSIN DD file failed to open during $AVRS initialization. Values defined during the last $AVRS initialization will be used. When the TRIALPD= is required $AVRS will terminate due to a TRIAL PERIOD EXPIRATION.

System Action: Execution continues.

User Response: Correct the SYSIN specification and restart $AVRS if TRIALPD= is required.

$AVR102C - FATAL ERRORS IN INITIALIZATION PARAMETERS

Explanation: One or more critical errors were encountered while processing the user options during initialization. A list of the initialization parameters for this execution was created on DDNAME "$AVRERR" along with a complete list of the initialization options error messages.

System Action: Abends U0102.

User Response: Correct errors and restart $AVRS.

$AVR103W - PREVIOUS $AVRS INITIALIZATION VALUES BEING USED

Explanation: The SYSIN DD file failed to open during $AVRS initialization. Values defined during the last $AVRS initialization will be used. When the TRIALPD= is required $AVRS will terminate due to a TRIAL PERIOD EXPIRATION.

System Action: Execution continues.

User Response: Correct the SYSIN specification and restart $AVRS if TRIALPD= parameter is required.

$AVR104I - THE REQUEST TO DRAIN '*rrds dataset name*' IGNORED BECAUSE NO RRDS FILE WOULD BE LEFT ACTIVE

Explanation: An attempt to DRAIN the last ACTIVE RRDS file was made via the Operator Modify Command. $AVRS does not allow all RRDS files to be placed in the "DRAINED" status.

System Action: The modify command is ignored.

User Response: No action required.

$AVR105I - THE REQUEST TO DRAIN '*rrds dataset name*' IGNORED BECAUSE NO RRDS BLOCKS WOULD BE AVAILABLE

Explanation: The request to drain the RRDS could not be honored because this action would not leave enough RRDS Data Blocks available for $AVRS to continue.

System Action: The modify command is ignored.

User Response: No action required

$AVR110I - *parameter* IS AN INVALID PARAMETER FOR $AVRLOCK

Explanation: An invalid request, "parameter", has been passed to the $AVRLOCK program.

System Action: The request has been ignored.

User Response: Retry with a correct request.

$AVR111W - *function* LOCK UPDATE REQUESTED

Explanation: A request has been made to update the lock file for the lock word specified by function.

Valid values for function are:

ACCM - The accumulation task.
BKUP - A backup task.
UNLD - A database unload job.
FORM - The Format/Reload job.

System Action: Execution completes based on response to $AVR051W message.

User Response: See message $AVR050W.

$AVR201C - CISIZE OF LOCK MUST BE 4096 - TERMINATING

or

- CISIZE OF RRDS MUST BE [4096|2048|1024] - TERMINATING

Explanation: The Record Size for the LOCK File must be 4089 and the CISIZE must be 4096. The Record Size/CISIZE for the RRDS File(s) may be 4089/4096, 2041/2948 or 1017/1024 but may NOT be mixed. ALL RRDS Files must be defined with the same Record Size/CISIZE values. Any RRDS file added to the existing $AVRS database must be defined with the same Record Size/CISIZE values as the previously defined RRDS file(s).

System Action: The job terminates with a U0201 abend.

User Response: Correct the JCL and resubmit the job.

$AVR202C - *ddname1* AND *ddname2* SPECIFY THE SAME DSNAME *dataset name* - TERMINATING

Explanation: Two DD Statements in the $AVRS Format job ($AVRFMRR) have the same DSNAME specified.

System Action: The job terminates with a U0232 abend.

User Response: Correct the JCL and resubmit the job.

$AVR203I - ATTEMPTING TO FORMAT A NON-EMPTY [LOCK|KSDS|RRDS] FILE - TERMINATING

Explanation: $AVRS detected a non-empty file during the format ($AVRFMRR) process. The $AVRS format process requires newly created VSAM file(s). Previously used files must be deleted and re-defined before they may be used by the format job.

System Action: The format job terminates with a U0203 abend.

User Response: Re-submit the format job using only a newly defined VSAM file(s)

$AVR204I - RRDS SUCCESSFULLY FORMATTED - DDNAME "*ddname*", DSNAME "*dataset name*"

Explanation: This message is issued for each RRDS successfully formatted during the $AVRFMRR format job.

System Action: None - informational

User Response: None

$AVR205I - *ddname* LRECL OF #### AND/OR CISIZE OF #### NOT VALID - TERMINATING

Explanation: A $AVRS VSAM data base file was defined with incorrect Record Size/CISIZE values. Valid combinations of Record Size & CISIZE are: 4089/4096; 2041/2048; 1017/1024

System Action: The job terminates with Abend U0205

User Response: Correct the JCL, re-define the VSAM file and re-submit the format job.

$AVR206E - LOCK FILE FULL - INCREASE ALLOCATION - TERMINATING

Explanation: The lock is full and no new RRDS files may be added without a larger LOCK file.

System Action: The job terminates with Abend U0206.

User Response: If this occurred during the formatting of a new $AVRS data base, increase the size of the LOCK file, delete and re-define all data base files and re-submit the format job. If this occurred while attempting to add an RRDS to an existing $AVRS data base, $AVRS must be stopped, unloaded and the existing data base files deleted, re-defined (with a larger LOCK file). Then run the format/reload job, $AVRFMRR. The correct size for the LOCK file is CYLS(5 0) no secondary space should be defined.

$AVR207I - DATA BASE FORMAT PROCESSING COMPLETE

Explanation: The $AVRS format process completed normally

System Action: None

User Response: None

$AVR208I - RECORD LENGTH OF TEXT DATA FILE *####*

Explanation: Displays the record length defined for the RRDS text data file(s)

System Action: None

User Response: None

$AVR209I - COUNT OF CONTROL BLOCKS FORMATTED *####*
COUNT OF MAP BLOCKS FORMATTED *####*
COUNT OF TEXT BLOCKS FORMATTED *####*

Explanation: Displays the total Control Blocks, Map Blocks and Text Blocks formatted.

System Action: None

User Response: None

$AVR210C - DDNAME "$AVRR*###*" ALREADY DEFINED IN THE $AVRS DATA BASE

Explanation: There is a duplicate DDNAME in the JCL for the format job.

System Action: The job terminates with an Abend U0210.

User Response: Correct the JCL and re-submit the format job. The VSAM LOCK, KSDS and RRDS file(s) may need to be deleted and re-defined.

$AVR211C - THE LRECL OF *####* FOR RRDS-*ddname* DOES NOT MATCH THE LRECL OF *####* FOR THE FIRST RRDS

Explanation: The $AVRS format program accepts the Record Size/CISIZE for the first RRDS DD Statement encountered as the desired parameters. All subsequent RRDS files must be defined with these same size parameters. The format job has encountered a file which does not match the record size or CISIZE of the first RRDS.

System Action: The format job terminates with Abend U0211.

User Response: If formatting an entire $AVRS data base, delete and re-define all $AVRS VSAM files (LOCK, KSDS, and RRDS) with the correct Record Size and CISIZE combinations and re-submit the format job. If adding a single RRDS file to an existing $AVRS data base, delete and re-define this single file with the correct Record Size/CISIZE and re-submit the job.

$AVR212I - COUNT OF [MAP|CONTROL] BLOCKS USED ####

Explanation: Displays the count of map blocks and control blocks used by the reload of work during a Format/Reload job.

System Action: None

User Response: None

$AVR213I - COUNT OF *xxxxxx* LOADED TO THE DATA BASE ###

Explanation: Displays the count of LINES, PAGES, FILES, JOBS, STCS, TSUS, SYSLOGS, STATUS REPORTS, FICHE KEYS, MESSAGE KEYS, CONTROL KEYS and ANNOTATION KEYS loaded to the data base during the Format/Reload process.

System Action: None

User Response: None

$AVR214C - LOCK DSNAME (*dataset name*) AND KSDS DSNAME (*dataset name*) ARE EQUAL

Explanation: The format job has determined that the LOCK file and the KSDS file have the same dataset name.

System Action: Format terminates with Abend U0214

User Response: Delete and redefine the $AVRS data base files with the correct data set names and re-submit the format job.

$AVR216W - RRDS DDNAME "*ddname*" IS NOT WITHIN THE RANGE OF 001 THROUGH 256 AND WAS NOT FORMATTED

Explanation: The $AVRS format job requires RRDS DDNAMEs to be in the format of $AVRRnnn where nnn may range from 001 to 256. $AVRS format encountered a DD statement with the '$AVRR' format but the last three characters did not fall within the numeric range.

System Action: Sets a return code of 4 and continues.

User Response: Correct the JCL statement and re-submit the job. If formatting a complete data base you may need to delete and re-define all VSAM data base files.

$AVR217I - $AVRS THE RRDS INDEX ENTERED WAS NOT [VALID|NUMERIC]. RE-ENTER WITH THE CORRECT VALUE

Explanation: The index value entered with the modify command was either not valid, the operator entered an index number that did not match an RRDS file, or was not numeric. The RRDS index must be numeric.

System Action: The modify command is ignored.

User Response: Re-enter the command with the correct value.

$AVR218I - $AVRS NO MATCH FOUND FOR THE INDEX VALUE ENTERED (###); RE-ENTER WITH THE CORRECT VALUE

Explanation: The index value entered did not match any of the RRDS files. $AVRS could not perform the requested action.

System Action: The modify command is ignored.

User Response: Either re-enter the command with the correct RRDS Index number or issue the "F *$avrstaskname*,STATUS,RR" modify command to obtain a list of all RRDS files with their associated index.

$AVR219I - RRDS DATA FILE NOT FOUND, SOME DATA WILL BE MISSING

Explanation: One of more of the job(s) files can not be located to copy to the $AVRLUND output or accessed from online. See message $AVR220I if available for more information.

System Action: Sets a return code of 4 and continues.

User Response: None required.

$AVR220I - SOME DATA HAS BEEN LOST FOR JOB *job name, job number,* FILE *file name*

Explanation: The referenced files are no longer on the RRDS and cannot be copied to the $AVRUNLD output.

System Action: Sets a return code of 4 and continues.

User Response: None required.

$AVR226I - LEVEL SET ## INCLUDES APARS THROUGH *nnn*

Explanation: The user has displayed the maintenance zaps applied to the $AVRS system. This message indicates the highest APAR number (*nnn*) included in the level set (##) for the user's system.

System Action: None.

User Response: Not required.

$AVR227I - PERCENTAGE OF [TEXT FILE|KSDS INDEX COMPONENT|KSDS DATA COMPONENT] IN USE ##.#%

Explanation: This messages are issued by the $AVRS Format/Reload job when work is reloaded to the data base. It reports on the amount of space used after the reload of a data base.

System Action: None

User Response: None

$AVR228I - RECOVERED BLOCKS=*##*, CURRENT USED=*##*, PREVIOUS USED=*##*, DSN=*dataset name*

Explanation: Informational message showing statistics by *dataset name* for the RRDS File(s) processed.

System Action: None.

User Response: None.

$AVR229I - BLOCKS RECOVERED=*##*

Explanation: Informational message. ## is the total number of blocks recovered by the program.

System Action: None.

User Response: None.

$AVR231E - *module* UNABLE TO RECOVER DEAD BLOCKS DUE TO PREVIOUS ERROR(S) - TERMINATING

Explanation: A critical error has occurred in the module indicated..

System Action: The job will terminate with a non-zero return code.

User Response: Review the error message and manually delete the work with errors or re-submit the job using the "CLEANERRORS' subparameter.

$AVR232E - NO RRDS FILES WERE SUCCESSFULLY FORMATTED

Explanation: No RRDS files could be formatted.

System Action: Terminates with Abend U0232.

User Response: Look for other error messages in this report and correct as required.

$AVR233C - THERE WERE NO INPUT CONTROL STATEMENTS-- AT LEAST ONE IS REQUIRED

Explanation: No input control statement was read and at lease one is required.

System Action: An error message is issued and, normally, the user abend U0233 is invoked.

User Response: Refer to the documentation, create the appropriate input control statement(s) and rerun.

$AVR234E - KSDS MUST SPECIFY KEYS(64 0) - TERMINATING

Explanation: The KSDS file was defined with the wrong KEYS value. $AVRS requires a KEYS value of "(64 0)".

System Action: $AVRS format terminates with Abend U0234.

User Response: Correct the JCL then delete and re-define all $AVRS data base files and re-submit the format job.

$AVR235W - (*RRDS dataset name*) ALREADY DEFINED IN THE $AVRS DATA BASE

Explanation: $AVRS format job encountered an RRDS with a duplicate dataset name.

System Action: $AVRS forces a return code of 4 and continues.

User Response: If this occurred during the initial format of the $AVRS data base, delete and re-define all $AVRS data base files before re-submitting the corrected format job. If this occurred during an add or add active format, check the online database before restarting the accumulator and correct the JCL for future jobs. Only the "new" previously unformated RRDS file names should be included in the format JCL when the parameter ADD or ADDACTIVE is used.

$AVR236I - BLOCKS TOTAL##, USED=##, ACCOUNTED FOR=##, INDEX=##, DSN=*dataset name*

Explanation: Informational message. ## is a decimal number. *Dataset name* is the rRDS processed.

System Action: None.

User Response: None.

$AVR238I - "requested function" CONTROL STATEMENT VALIDATED - the statement

Explanation: The syntax of the control statement is valid.

System Action: Processing continues.

User Response: None required.

$AVR239C - NO *type* RECORD FOUND FOR THE JOB SPECIFIED

Explanation: An attempt to get a record was made and no record was found. Either a specific job was requested or the database is empty.

System Action: Sets a non-zero return code and continues.

User Response: None.

$AVR240E - MISSING *text* RECORDS FOR JOB *job name*, JES# *job number*, DDN *ddname*

Explanation: *job name* is the JES job name of the work which has missing records. *job number* is the JES job number of the work. *ddname* is the file name involved. *Text* may be:

1. INVALID HIPER POINTER

System Action: The job will continue but will set a non-zero return code.

User Response: None.

$AVR241I - FILE COUNT DATA BLK MISMATCH FOR JOB *jobname*, JES *jobnumber*, B/KEY=*nnn*,ACTUAL=*nnn*-B/KEY UPDATED

Explanation: Informational message showing that a mismatch was found between the basekey and the actual files or data.

System Action: The message indicates the action taken. Usually the basekey is updated to reflect the actual files or data blocks found.

User Response: None.

$AVR242I - (module) NUMBER OF CONTROL STATEMENTS READ: #

Explanation: This a count of the control statements that were processed during the execution of this job.

System Action: None.

User Response: None.

$AVR243I - (module) NUMBER OF ERRORS ENCOUNTERED: #

Explanation: This is the total number of errors that were encountered while processing the control statements for this job.

System Action: None.

User Response: Review the control statements and possible other messages and correct the errors.

$AVR244I This message is accompanied by one of the following phrases:

- **1) JOBNAME = *job name,* JES# = *job number,* KRRYSED=*# of blocks***
  2) JOBNAME = *job name,* JES# =*job number,* FILE =*ddname,* FI#RRBLK = *# of blocks*
  3) *chain type* = x'##' x'##' .....
  4) GETMAIN FAILURE ATTEMPTING TO BUILD A BITMAP TABLE FOR RRDS = RRDS *dataset name*
  5) JOBNAME = *job name* JES#= *job number* FILE = *ddname block type = x'xx##'*
  6) GETMAIN FAILURE ATTEMPTING TO PROCESS JOB= *job name* JES = *job number* - BYPASSED

Explanation:
1) Identifies the JES *job name*, *number* and the number of RRDS Data Blocks used by this job.
2) Identifies the JES *job name*, *number*, the *file name*, and the *number* of RRDS Data Blocks used by this file.
3) Displays the HIPO BLKCHAIN, HIPO RPTCHAIN, or RRDS BLKCHAIN, Forward Chain Pointers for debugging. -
4) Unable to obtain a Getmain area large enough to hold the Map Blocks for this RRDS.
5) Produced when tracing has been invoked. JES *job name*, *job number*, *block type*, *file*, and *block number* are displayed. The *x'xx##'* is *xx* for the RRDS index and *##* for the *block number*.
6) The total number of Hiper Records was too large for the available free storage.

System Action:
1) Job continues to run.
2) Job continues to run.
3) Job continues to run.
4) The job terminates with a high return code.
5) Job continues to run.
6) This file is bypassed and the job continues to run.

User Response:
1) None.
2) None.
3) None.
4) Re-run the job in a partition with more memory available.
5) None.
6) None.

$AVR247E - *job name job number text*

Explanation: Informational error message with three *text* variation. The indicated action/information *text* below is for the stated JES *job name* and *number*:

1) HAS NO DATA FILES (KEYS)
2) ATTEMPT TO DELETE FAILED
3) DELETED - NO FILE KEYS

System Action: The Job will continue but sets a non-zero return code.

User Response: None.

$AVR248I - JOB(*jobname*), JES(*jobnumber*), DDNAME(*ddname*), "*various messages*"

Explanation: Informational message(s).

System Action: The message indicates the action taken.

User Response: None.

$AVR249I - END OF FORMAT PROCESSING - NUMBER OF ERRORS DETECTED: ###

Explanation: Informational message identifying the end of the $AVRS format processing and giving the total number of errors detected during the process.

System Action: None

User Response: None

$AVR255W - RRDS DD STATEMENT(S) ARE MISSING OR WERE INVALID

Explanation: Informational message warning that errors have been detected.

System Action: The job terminates with Abend U0255

User Response: Correct the errors specified in the previous error message(s).

$AVR257W - RRDS *dataset name* NOT ADDED TO THE LOCK FILE DUE TO INSUFFICIENT MAP BLOCKS

Explanation: Informational message identifying the end of the $AVRS format processing and giving the total number of errors detected during the process.

System Action: None

User Response: None

$AVR304I - ADDITIONAL PARAMETERS FOUND ON THE ONEMATCH STATEMENT ARE IGNORED

Explanation: The ONEMATCH statement contained additional parameters or data. This parameter should be on an input statement by itself.

System Action: None, the job continues to run.

User Response: Change the input statements and remove any additional parameters on the “ONEMATCH” statement to another input statement.

$AVR305C - INITIAL VALUE FOR A PARAMETER MUST START IN COLUMN 1

Explanation: The initial value for a control card statement must start in column 1.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Either the initial value started past column 1 or a continuation comma was left off and the continued parameter was taken as an initial parameter. Correct as required.

$AVR306C - KEYWORD "*xxxxxxx* " UNKNOWN OR INVALID

Explanation: The keyword specified in the message was unknown or was defined in an invalid format.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Check the keyword and correct as required.

$AVR307C - CONTINUATION CARDS CANNOT START IN COLUMN 1

Explanation: A continuation was expected due to a continuation comma on the previous control card but the continued card started in column 1.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the column number as required.

$AVR308I - BLANK INPUT RECORD WILL BE TREATED AS A COMMENT RECORD

Explanation: A blank input record was found.

System Action: The blank record will be treated as a comment record.

User Response: None required.

$AVR309C - PARM AREA COUNT PASSED INDICATES NO PARMS

Explanation: A parameter area was passed to the parse routine but it indicated that there was no parm being passed.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the parameter that may be causing the problem and retry the job. Call **SEA** if correction of the parameter and retry fails.

$AVR310C - PARAMETER HAS A LENGTH OF ZERO PARM=*parameter*

Explanation: A parameter area was passed to the parse routine but it indicated that there was a zero length for the parameter being passed.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the parameter that may be causing the problem and retry the job. Call **SEA** if correction of the parameter and retry fails.

$AVR311C - PARAMETER TYPE MUST BE A 1 OR A 2

Explanation: The only valid parameter types are **1** and **2**.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Ensure that these values start in position 1 of the control statement followed by the DDname with which they are associated.

$AVR312C - DDNAME SPECIFIED DOES NOT CONFORM TO REQUIREMENTS

Explanation: The DDNAME specified after the TYPE 1 or TYPE 2 parameter does not conform to the format **$AVRS##**.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Respecify the parameter in the correct format.

$AVR313C - NON-NUMERIC VALUE SPECIFIED FOR *parameter*

Explanation: A numeric value was expected for the parameter in the message.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem with the parameter and resubmit the job.

$AVR314C - BLKSIZE VALUE MUST BE A NUMBER BETWEEN 6233 AND 32760

Explanation: The block size specified is not between the allowable ranges of 6233 and 32760.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Respecify the block size.

$AVR315C - *"xxxx* " WAS SPECIFIED BUT IS AN UNKNOWN KEYWORD

Explanation: A keyword was specified but is unknown to the $AVRS parse routine.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the invalid parameter and rerun the job.

$AVR316C - NO RECORDS WRITTEN TO SORT FILE - ALL PARAMETERS FAILED VALIDATION

Explanation: There were no valid parameter control cards. This message is usually accompanied by $AVR348. The text in message $AVR348 indicates what kind of control cards are in error.

System Action: The sort will not be performed. Abend U0368 happens if this error is produced by a $AVRS backup.

User Response: Correct the problem with the Input Parameter and rerun the job.

$AVR317C - KEYWORD MISSING OR VALUE WAS INVALID FOR - xxxxxxxx

Explanation: The specified keyword was either not defined or the value specified for the keyword was invalid.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem with the Input Parameter and rerun the job.

$AVR318C - KEYWORDS ARE MUTUALLY EXCLUSIVE -

Explanation: The keywords in the message text are mutually exclusive.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and re-execute the program.

$AVR319W - ** WARNING ** NO RETPD OR EXPDT GIVEN FOR ARCHIVE FILE DDNAME

Explanation: A warning is issued to indicate that no retention or expiration was specified for the archive file being created.

System Action: Execution continues.

User Response: Add a RETPD or EXPDT to eliminate the message.

$AVR320C - SPACE TYPE DEFINED BUT PRIMARY QUANTITY NOT SPECIFIED OR INVALID

Explanation: The keyword SPACE was defined on the TYPE 1 control statement but no primary space value was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR321C - PRIMARY SPACE SPECIFIED BUT SPACE TYPE NOT SPECIFIED OR INVALID

Explanation: The keyword PRIMARY was defined on the TYPE 1 control statement but no SPACE keyword was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR322C - SPACE= VALUE MUST BE "CYLS", "TRKS" OR "BLKS"

Explanation: The keyword SPACE was defined on the TYPE 1 control statement but the value specified was not one of the following:

CYLS Cylinder allocation
CYLR Cylinder allocation with Release
TRKS Track allocation
TRKR Track allocation with Release
BLKS Block allocation
BLKR Block allocation with Release

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR323C - DBDAYS OR DBCYCLES VALUE MUST BE BETWEEN 0 AND 9999

Explanation: The value for the database retention must be a value between 0 and 9999.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR324C - PARAMETER HAS BEEN PREVIOUSLY SPECIFIED - *parameter*

Explanation: The parameter in the message text has been previously specified for this control statement.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR325C - DISP= PARAMETER IS INVALID, VALUE MUST BE - *parm*

Explanation: For the DISP= parameter the value specified must be as stated in the message text.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR326C - DATASET NAME CANNOT HAVE TWO CONSECUTIVE PERIODS

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR327C - FIRST CHARACTER OF DATASET NAME IS INVALID

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR328C - LAST CHARACTER OF DATASET NAME IS A DOT (".")

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR329C - DATASET NAME GDG INFO COMES IMMEDIATELY AFTER A DOT (".")

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR330C - FIRST CHARACTER OF LOWER LEVEL INDEX IS INVALID

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR331C - LENGTH OF ONE OR MORE DATASET NAME INDEXES IS GREATER THAN 8

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR332C - GDG DSNAME CANNOT HAVE MORE THAN 35 CHARACTERS BEFORE "G0000V00"

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR333C - MAXIMUM NUMBER OF INDEX LEVELS ALLOWED IS 8

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR334C - DSNAME TEXT CONTAINS ILLEGAL OR UNDEFINED CHARACTERS

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR335C - GDG VERSION MUST BE "(+1)"

Explanation: An invalid dataset name was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR336C - EXPDT= VALUE MUST BE 5 CHARACTERS

Explanation: An expiration date value must be five numeric characters in length.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and rerun the job.

$AVR337C - VALUE NOT GIVEN FOR - *param*eter

Explanation: No value was given for the parameter specified in the message text.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR338C - RETPD= VALUE MUST BE A NUMBER BETWEEN 1 AND 9999

Explanation: A retention period value must be numeric and a value between 1 and 9999.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR339C - FILE= VALUE MUST BE A NUMBER BETWEEN 1 AND 255, NEW OR NEWTAPE

Explanation: The FILE= parameter value must be a numeric value between 1 and 255 or the value "**NEW**" or "**NEWTAPE**".

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR340C - HISTONL= VALUE MUST BE A NUMBER BETWEEN 1 AND 9999

Explanation: The HISTONL initialization parameter value must be a numeric value between 1 and 9999.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR341C - MSVGP= VALUE IS NOT ALPHANUMERIC, $, @ OR #

Explanation: The value specified for MSVGP was not a combination of alphanumerics, $, @ or #.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR342C - VALUE CANNOT EXCEED 8 CHARACTERS - PARM=*param*eter

Explanation: The value specified for the keyword parameter specified in the message cannot exceed 8 characters.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR343C - UNIT NAME EXCEEDS 8 CHARACTERS

Explanation: The UNIT name value cannot exceed 8 characters in length.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR344C - UNKNOWN VALUE SPECIFIED FOR "UNIT=" VALUE=*param*eter

Explanation: An unknown value was specified for the UNIT= keyword.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR345C - A MAXIMUM OF 5 VOLSERS MAY BE SPECIFIED

Explanation: More than 5 volsers were specified for the VOLSER= keyword.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Reduce the number of volsers, correct the problem, and resubmit the job.

$AVR346C - ONE OR MORE VOLSERS EXCEEDS 6 CHARACTERS

Explanation: Volsers specified more than the allowable 6 character maximum.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR347C - INVALID VOLSER SPECIFIED OR A BLANK DETECTED IN A VOLSER STRING

Explanation: Either an invalid volume serial number was specified or a blank was detected in a volser string.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR348C - ERRORS DETECTED IN *xxxxxxxxxxx*

Explanation: Errors have been detected in the parameters specified for the *xxxxxxxxxxx* variable. This variable can be:

BACKUP INPUT CONTROL PARAMETERS,
MAXLINE INPUT CONTROL PARAMETERS or
EXTENDED SCAN INPUT CONTROL PARAMETERS.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete if this message is produced from a $AVRS backup. Maxline or Extended scan versions of this message cause processing to continue as if no control parameters had been entered.

User Response: Refer to other error messages in the Parse Report, correct and resubmit the job. Use the $AVRBKPR program to validate the $AVRBKUP table parameters.
Use the $AVRMXPR program to validate the $AVRMAXL table parameters.
Use the $AVRSXPR program to validate the $AVRSCAX table parameters.

$AVR349C - NO PRIMARY (TYPE 1) DD DEFINITION INPUT FOUND

Explanation: There were TYPE 2 control definitions that referenced TYPE 1 DDnames, but the TYPE 1 input definitions either did not exist or contained errors.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR350C - DATA NOT FOUND ON VOLSER=*xxxxxxxx*, DSN= *dataset name*

Explanation: A VOLSER= and DSN= parameter was specified for a DASD dataset, but the dataset specified as DISP=OLD but was not found.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR351C - VOLSER(S) SPECIFIED AND NO UNIT NAME DEFINED FOR DDNAME

Explanation: No UNIT= parameter was specified when a VOLSER parameter was specified.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR352C - NO SELECTION (TYPE 2) INPUT FOUND

Explanation: There were no TYPE 2 selection input control parameters specified or all of the TYPE 2 parameters had errors.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR353C - NO PRIMARY DDNAME DEFINED FOR DUPLEX

Explanation: A duplex DDNAME was specified eg; $AVRS##D, but no matching primary DDNAME was specified or the primary DDNAME had errors.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR354C - The parm passed to $AVRALLC did not include xxxxxxxx

Explanation: A user written program invoked program $AVRALLC and did not pass the correct parm value(s).

System Action: $AVRALLC terminates with a User Abend

User Response: Review the coding example provided in “USERPGM” found in the installation library for the correct parm list format.

$AVR355C - MULTIPLE ENTRIES FOUND FOR PRIMARY DDNAME

Explanation: The same primary DDNAME was specified multiple times in the input control parameters.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR356C - MISSING PRIMARY DDNAME FOR SELECTION DDNAME=*ddname*

Explanation: A TYPE 2 selection control parameter specifies a DDNAME that does not exist in the control parameters or the primary DDNAME parameter had errors.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR357C - NUMBER OF WORK SELECTION DEFINITIONS EXCEEDS 32767

Explanation: The maximum number of work selection definitions cannot exceed a maximum of 32767.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Refine the selection criteria (TYPE 2) definitions using generic or masked definitions. Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Resubmit the job.

$AVR358C - PRIMARY DDNAME $AVRS99 MUST BE DEFINED IN THE INPUT CONTROL DEFINITIONS

Explanation: The primary input parameter (TYPE 1) $AVRS99 must be defined in the control parameter definitions.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR359C - FILE STACKING OR FILE=NEWTAPE REQUESTED FOR A NON-GDG DATASET, DDNAME=

Explanation: File stacking can only be done using GDG format datasets.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Use the $AVRBKPR program to validate the $AVRBKUP table parameters. Correct the problem and resubmit the job.

$AVR360C - DUPLICATE DSN DEFINED FOR DDNAME=*ddname* AND DDNAME= *ddname*

Explanation: Duplicate dataset names were defined for the DDnames given in the message text.

System Action: None.

User Response: Correct the problem and resubmit the job.

$AVR361C - GDG DATASET MUST HAVE A DISPOSITION OF "NEW"

Explanation: A GDG (+1) dataset was specified but the disposition was not **NEW**.

System Action: None.

User Response: Correct the problem and resubmit the job.

$AVR362C - CATALOG LOCATE FAILED FOR DSN= *dataset name*

Explanation: A disposition **OLD** or GDG dataset locate failed. If the dataset is **OLD** then the dataset does not exist in the catalog and no VOLSER= parameter was specified. If the dataset was a GDG then the base index was not defined. Ensure that the high level index is connected to the catalog that contains the dataset being defined.

System Action: None.

User Response: Correct the problem and resubmit the job.

$AVR363C - UNCATALOG FAILED FOR DSN=*dataset name*

Explanation: An uncatalog failed for the specified dataset.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR364C - "DISP=NEW" DATASET FOUND ON VOLSER= , DSN= *dataset name*

Explanation: A dataset specified as DISP=NEW, with a VOLSER parameter was found on the specified volume.

System Action: None.

User Response: Correct the problem and resubmit the job.

$AVR365C - SCRATCH FAILED ON VOLSER= , DSN=*dataset name*

Explanation: The scratch attempt failed for the dataset on the volser specified in the message text.

System Action: None.

User Response: Correct the problem and resubmit the job.

$AVR366I - UNCATALOG HAS BEEN DONE FOR "DISP=(NEW,CATLG)" DATASET, DSN=dataset name

Explanation: The dataset specified in the message text has been uncataloged.

System Action: None.

User Response: Informational.

$AVR367C - NUMBER OF RECORDS TO BE SORTED EXCEEDS 32767 FOR WORK= AND REQUIRES AN EXTERNAL SORT PROGRAM

Explanation: The number of like work keys records exceeds the maximum of 32767 and requires an external sort program.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: After terminating the $AVRS task, update your $AVRS initialization parameter SORTPGM= with the external sort program used at your location. Restart the $AVRS task.

$AVR368C - ERRORS ENCOUNTERED IN BACKUP PARSE/SELECTION PROCESS

Explanation: Errors were encountered during the backup/archive parse routines.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Run $AVRBKPR to validate the Backup parameters. Correct the errors and resubmit the job.

$AVR369C - DENSITY VALUE MUST BE "6250" OR "1600"

Explanation: The DENSITY= value can only be 6250 or 1600.

System Action: Parameter validation continues. Processing will terminate with abend U0368 after parameter validation is complete.

User Response: Correct the problem and resubmit the job.

$AVR370C - PROCESSING ERROR OCCURRED IN $AVRBK08 LOCATION FINDBK1

Explanation: An error has occurred during the allocation process for the backup/archive being run.

System Action: Abends U0370.

User Response: Refer to other messages produced to describe the processing error that occurred. Correct the problem and resubmit the job.

$AVR371W - WARNING ERRORS HAVE OCCURRED DURING BACKUP PROCESS

Explanation: An error has occurred during backup/archive processing.

System Action: Processing continues.

User Response: Refer to a previous message to determine the processing error. Correct the problem and resubmit the job.

$AVR372W - ** WARNING ** OPERATOR HAS REPLIED CANCEL TO BACKUP ALLOCATION DDNAME=*ddname*

Explanation: The operator has replied CANCEL to an outstanding dynamic allocation message. This message is issued to indicate the operator cancel.

System Action: Processing continues with other archives.

User Response: None required.

$AVR373C - $AVRBK99D ** DDNAME ALREADY OPEN ** INTERNAL ERROR

Explanation: An internal error has occurred. A dataset being opened was already open.

System Action: Processing terminates and Abend U0373 occurs.

User Response: Call SEA for support and problem resolution. Please have the dump and the archive run available prior to making the call.

$AVR374W - OPERATOR REPLIED SHUTDOWN ** ALL PREVIOUS BACKUP PROCESSING IGNORED

Explanation: The operator replied SHUTDOWN to the $AVR449A message.

System Action: All previous archive files created this run will not be accounted for in the archive history.

User Response: Rerun archive.

$AVR375I - DUE TO EMERGENCY ARCHIVE, DDNAME=*ddname,* HAS MAX RETPD DATE= , CYCLES=

Explanation: The database indicated previously has filled and the DDNAME specified in the message has had its max retention and cycles set as defined in the message.

System Action: Processing continues.

User Response: Informational.

$AVR376I - DUE TO EMERGENCY ARCHIVE, ALL RETENTION PERIODS REDUCED BY #

Explanation: The retention periods specified have been reduced by the number of days specified in the message due to an emergency backup/archive situation.

System Action: Processing continues.

User Response: Informational.

$AVR377I - RECORD NOT FOUND ON REREAD FOR WORK - *xxxxxxxxx*

Explanation: Record **xxxxxxxxxx** was marked for backup during initial $AVRS backup processing. Later in backup processing, $AVRS tried to re-read the record and received a "not found" error.

System Action: Processing continues.

User Response: None. The record was probably deleted online or during a batch backup.

$AVR378C - PARAMETER VALUE EXCEEDS MAXIMUM LENGTH - *xxxxxxxxxx*

Explanation: The value for the keyword xxxxxxxxxx is longer than the maximum allowed for the field.

System Action: The control record is rejected.

User Response: Consult the appropriate place in this manual for details on the maximum value length of the keyword xxxxxxxxxx. After correcting the error, rerun the job.

$AVR380C - MAXIMUM INPUT CONTROL STATEMENTS REACHED

Explanation: The maximum number of control statements for a single execution of $AVRFICH has been reached.

System Action: This statement and additional control records are skipped.

User Response: Break up the control records into multiple executions of $AVRFICH.
User Abend U380.

$AVR381E - SMS ALLOCATIONS MUST BE IN TRACKS

Explanation: A $AVRS backup dataset has requested CYLinder allocation for an SMS dataset.

System Action: Processing Terminates

User Response: Either remove the STORCLAS parameter from the $AVRS backup statement or remove the SPACE, UNIT, and VOLSER parameters. No SPACE, UNIT, or VOLSER keyword should be used with STORCLAS.

$AVR382E - SMS SUBSYSTEM REQUIRED, BUT NOT AVAILABLE

Explanation: A $AVRS backup requires that the SMS subsystem be operational, but SMS is either down or not found.

System Action: Processing terminates with a U382 abend.

User Response: Either remove the STORCLAS parameter from the $AVRS backup statement(s) or start the SMS subsystem. Re-run the backup.

$AVR383C - NO ACCUMULATED WORK EXISTS FOR ARCHIVE

Explanation: A backup/archive was attempted and no accumulated work exists in the database. (i.e. no JOB, TSU, STC or SYSLOG sysout)

System Action: Backup terminates. If this backup was initiated as an emergency backup from the accumulation task, the accumulation task terminates as well. The return code is 5.

User Response: Accumulate work before attempting a backup. If this is an emergency backup, the database must be expanded by activating another RRDS files for accumulation or by manually deleting $AVRS Status/Fiche reports and restarting the accumulator.

If the $AVR441I - KSDS Full message was issued before the emergency backup occurred, use the KEYREPRO member from the $AVRS INSTALL library to rebuild the KSDS index and reorganize the KSDS file.

$AVR384E - THE DSN *dsname* ENTERED ON THE PARM DOES NOT MATCH

Explanation: Refer to the continuation of this message in $AVR385E.

$AVR385E - THE DSN *dsname* THAT $AVRALLC FAILED TO ALLOCATE

Explanation: During the process to FORCE an RRDS to a status of DRAIN and EMPTY, $AVRS attempts to validate the dsname entered on the PARM against the first RRDS that module $AVRALLC fails to allocate. This is to insure that the dsname entered on the parm was done without any typing errors. It is assumed that the FORCE is being done because an RRDS was deleted or is no longer available and $AVRS cannot be started.

System Action: $AVUMTRR terminates with a U0384 abend.

User Response: Verify that the dsname entered on the PARM is correct and re-submit the job.

$AVR386E - This message is accompanied by one of the following phrases:

1) RRDS NOT IN LOCK FILE
2) RRDS MUST BE DRAINED
3) RRDS ALREADY EMPTIED
4) RRDS NOT IN REMOVE STATUS
5) $AVRLOCK GET BAD
6) $AVEMPTY GET BAD

Explanation:

1) The RRDS specified by the $AVEMPTY or $AVREMOV ddname is not part of the data base specified by the $AVRLOCK ddname.
2) The RRDS specified by the $AVEMPTY ddname is not in drain status.
3) The RRDS specified by the $AVEMPTY ddname has already been emptied.
4) The RRDS specified by the $AVREMOV ddname is not in remove pending status.
5 & 6) An attempt to retrieve a record from the RRDS specified by the ddname failed.

System Action: The $AVUMTRR or $AVURMRR program terminates.

User Response:

1) Provide the correct LOCK file or target RRDS file in the JCL and rerun the utility.
2) Issue a modify command to the $AVRS started task to drain the RRDS. Rerun the $AVUMTRR utility.
3) None. The RRDS has already been emptied.
4) Drain the RRDS and empty it via the $AVUMTRR utility. Rerun the $AVURMRR utility.
5 & 6) Contact **SEA** Support for instructions in resolving the problem.

$AVR389C - *ddname* DDNAME NOT SPECIFIED

Explanation: A file required by the job has not been specified via the appropriate DD Statement.

System Action: Job terminates.

User Response: Correct the JCL and resubmit the job.

$AVR390C - INSUFFICIENT BLOCKS IN DATA BASE TO EMPTY RRDS-REQUIRED=X'########',AVAILABLE=X'########'

Explanation: The $AVUMTRR Utility has determined that insufficient data blocks are available to allow relocation of blocks from the RRDS to be emptied to other RRDS files in the $AVRS Data Base.

System Action: $AVUMTRR Utility terminates.

User Response: Make additional blocks available by deleting files from the database or by adding an additional RRDS to the database or by activating RRDS files.

$AVR393C - RRDS MAINTENANCE FUNCTION ENQUEUE FAILED - TRY AGAIN LATER

Explanation: The $AVUMTRR and the $AVURMRR RRDS maintenance utilities require exclusive control of the LOCK File that defines the data base being maintained. The Enqueue is currently held by some other $AVRS process (ie., started task or restore).

System Action: The utility terminates with a return code of 393.

User Response: Resubmit the utility job after all processes holding the enqueue have terminated.

$AVR394I - THERE ARE NO FILES IN THE RRDS TO BE EMPTIED - EMPTY REQUEST IGNORED

Explanation: There are no active files in the RRDS to be emptied.

System Action: Job terminates.

User Response: None.

$AVR395C - EMPTY FAILED FOR RRDS *dsname* RTNCODE=X'########'

Explanation: The $AVUMTRR Utility was unable to empty the specified RRDS. The RTNCODE value indicates the nature of the error.

System Action: $AVUMTRR Utility terminates.

User Response: Contact **SEA** Support for instructions in resolving the problem.

$AVR396C - RELOCATION OF FILE *file name* RDRNUM #### FAILED

Explanation: An internal error prevented relocation of the file within the $AVRS data base by the $AVUMTRR Utility.

System Action: None.

User Response: Contact **SEA** Support for instructions in resolving the problem.

$AVR397I - RRDS *dataset name* EMPTIED AND PLACED IN REMOVE PENDING STATUS

Explanation: The $AVUMTRR Utility has successfully relocated all files in the RRDS and has indicated that the RRDS may now be removed from the $AVRS data base.

System Action: None.

User Response: The RRDS may now be removed from the $AVRS data base via the $AVURMRR Utility.

$AVR398I - RRDS *dataset name* HAS BEEN REMOVED FROM THE DATA BASE

Explanation: The $AVURMRR Utility has completed the removal of the RRDS from the $AVRS data base.

System Action: None.

User Response: None.

$AVR399C - FILE *file name* RDRNUM #### NOT RELOCATED, #### BLOCKS REQUIRED, #### BLOCKS AVAILABLE

Explanation: The $AVUMTRR Utility was unable to relocate the file while emptying the RRDS due to a shortage of Data Base Blocks in other RRDS files in the Data Base.

System Action: $AVUMTRR continues to process other files that may have lesser requirements.

User Response: Ensure additional blocks are available in the data base and rerun $AVUMTRR.

$AVR401I - *job name* HAS BEEN DELETED DURING ACCUMULATION OF JOB

Explanation: While processing "jobname" a user deleted all or part of the job currently being accumulated.

System Action: All or part of the jobs SYSOUT from the "ACCMCLS" may not have been accumulated.

User Response: Informational.

$AVR402I - PREVIOUS DAYS(S) BACKUP/ARCHIVE RUN SKIPPED DUE TO INSTALLATION SPECIFICATIONS

Explanation: $AVRS is informing the user that at least 1 normal backup has passed while $AVRS was not active. However, the backup(s) would not have been taken anyway due to the **BKUPDAY** initialization parameter.

System Action: Processing continues.

User Response: Informational.

$AVR403I - DUMP FOR *job name* JOB# #### REQUEUED TO CLASS *class*

Explanation: A dump was encountered while processing a job's SYSOUT.

System Action: Due to special processing caused by encountering the dump, the entire job will be requeued to the "REQCLAS". No automatic return other than this requeue will be done for this job.

User Response: Informational.

$AVR404C - INVALID RESPIN DESTINATION FOR THIS COMPLEX - DESTINATION= *destination*

Explanation: An invalid "LCLDEST" has been specified for local respins or the remote destination is not valid on this JES complex.

System Action: Abends U0404.

User Response: If "LCLDEST" is invalid, correct the "LCLDEST" specification and restart $AVRS.

$AVR405I - RESPIN DESTINATION FOR -*destination* - SET TO LOCAL

Explanation: A dynamic allocation of a sysout dataset has returned an error indicating that the remote destination is in error. $AVRS resets the destination to the default (LOCAL) and retries the allocation.

System Action: Reset destination and retry the allocation.

User Response: Informational.

$AVR406I - OUTPUT DDNAME - *ddname* - WAS AN INVALID REFERENCE

Explanation: A dynamic allocation of a sysout dataset has returned an error indicating that the output ddname is in error. $AVRS resets the output ddname to the default and retries the allocation.

System Action: Reset output ddname and retry the allocation.

User Response: Informational.

$AVR407I - MAXLINES OF ZERO (0) ENCOUNTERED FOR JOBNAME *jobname, jobnumber*

Explanation: The MAXLINE table has instructed $AVRS to not accumulate any lines for this job name. This job is either deleted or respun depending on other MAXLINE parameters.

System Action: Processing continues.

User Response: Informational.

$AVR408W - SMF RECORD WOULD NOT FIT INTO AN EMPTY SMF DATASET

Explanation: This was the indication given by the return code from SMFWTM.

System Action: The SMF record is not written and processing continues.

User Response: Correct the problem with your SMF.

$AVR409W - SMF RECORD HAS RDW LESS THAN 18 BYTES

Explanation: This was the indication given by the return code from SMFWTM.

System Action: The SMF record is not written and processing continues.

User Response: Correct the problem.

$AVR410W - SMF RECORD SCHEDULED BUT SMF IS NOT ACTIVE

Explanation: This was the indication given by the return code from SMFWTM.

System Action: The SMF record is not written and processing continues.

User Response: Correct the SMF problem.

$AVR411W - $AVRSMF DISCONTINUED DUE TO IEFU83 SMF EXIT

Explanation: This was the indication given by the return code from SMFWTM.

System Action: The SMF record is not written and processing continues.

User Response: Correct the SMF problem if SMF records are desired.

THERE WILL BE NO FURTHER ATTEMPT TO RECORD SMF RECORDS FOR THIS EXECUTION OF $AVRS.

$AVR412C - S S O B RETURN NON-ZERO REGISTER 15=*xx* FOR *job name, job number*

Explanation: This message is issued in conjunction with message $AVR405C under certain conditions.

System Action: Abend U0412 is issued.

User Response: See the OS/390 MVS using the subsystem interface #SC281729 for a description of the return code and the corrective action.
$AVRS may be restarted after this job is HELD.

$AVR413C - S S O B RETURN CODE UNACCEPTABLE "SSOBRETN=##" FOR *job name, job number*

Explanation: ## = SSOB return code

System Action: Abends U0413.

User Response: See the OS/390 MVS using the subsystem interface #SC281729 for a description of the return code and the corrective action.
$AVRS may be restarted after this job is HELD .

$AVR414C - ACCDEST=*destination* DESTINATION NOT DEFINED TO THIS JES COMPLEX

Explanation: An SSOB return code of 28 has been detected.

System Action: Abends U0413.

User Response: See the OS/390 MVS using the subsystem interface #SC281729 for a description of the return code and the corrective action.
Restart $AVRS.

$AVR415A - ENTER PASSWORD FOR MODIFY

Explanation: The current user options require the operator password before execution of the command is attempted.

System Action: If password given, processing continues. If not, the requested function is ignored.

User Response: Enter the operator password.

$AVR416I - INVALID PASSWORD MODIFY/STOP COMMAND DISALLOWED

Explanation: The correct password was not given in response to message $AVR415A.

System Action: The content and the syntax of the MODIFY is ignored, and the STOP request is also ignored.

User Response: Enter the correct password.

$AVR417I - $AVRS ACKNOWLEDGES OPERATOR STOP COMMAND

Explanation: The STOP request is acknowledged.

System Action: $AVRS will stop as soon as logically possible.

User Response: None required.

$AVR418I - $AVRS ACKNOWLEDGES OPERATOR REQUEST FOR BACKUP

Explanation: The MODIFY request for backup/archive is acknowledged.

System Action: Backup/archive will proceed as soon as possible.

User Response: None required.

$AVR419I - $AVRS INVALID MODIFY COMMAND ENTERED

Explanation: The correct password was supplied if password option is in effect but the command is unknown to $AVRS.

System Action: MODIFY is ignored, and the STOP request is also ignored.

User Response: Reissue the correct command.

$AVR420I - $AVRS CURRENT *Commandword* COMMAND IS:
[DRAINED |
DRAINING |
WAITING |
WORKING |
RESPIN |
MAXLINE |
BACKUP |
KEYMAINT]

Explanation: This message is in response to a F *$avrstaskname,commandword* command. This message is followed by **$AVR428I**. The various messages mean the following:

DRAINED - $AVRS is waiting for more work, but has been drained by operator command.
DRAINING - $AVRS has been drained by operator command, but continues a current function in progress. $AVRS completely drains after the current function completes.
WORKING - $AVRS is ready for work and is currently accumulating sysout.
RESPIN - $AVRS is ready for work and is currently in RESPIN processing.
MAXLINE - $AVRS is ready for work and is currently in MAXLINE processing.
BACKUP - $AVRS is ready for work and is currently in BACKUP processing.
KEYMAINT - $AVRS is ready for work and is currently in KEYMAINT processing.

System Action: None.

User Response: Informational Only.

$AVR421C - LOCATE FOR BACKUP DATASET NAME FAILED ABEND EQUALS THE LOCATE RETURN CODE+1000

Explanation: The locate that is required to determine the full dataset name when a generation dataset is being used for the backup failed.

System Action: A user abend is issued equal to the locate return code+1000.

User Response: Correct the locate error and restart $AVRS.

$AVR422I - NO NEW DATA WRITTEN TO ONLINE FILES - BACKUP WILL NOT BE TAKEN

Explanation: In response to a backup request, it is detected that no additional data has been written to the online files.

System Action: Message $AVR423I is also issued and processing terminates.

User Response: Informational.

$AVR423I - NEXT NORMAL BACKUP MAY BE TAKEN ON *yyyy.ddd*

Explanation: This message is issued in addition to msg $AVR422I or $AVR424I to tell when next normal backup may be taken.

System Action: If additional data is written on the online files and another backup request is made before YYYY.DDD is reached, that backup will be taken and processing will terminate.

User Response: Informational.

$AVR424I - DUE TO INSTALLATION SPECIFICATIONS BACKUP WILL NOT BE TAKEN FOR *day YYYY.DDD*

Explanation: $AVR424I is issued at the "BKUPTIM" if the day has been excluded via the "BKUPDAY" parameter.

System Action: Message issued and processing continues.

User Response: Informational.

$AVR425I - $AVRS STOPPING FOR NORMAL BACKUP BY REQUEST

Explanation: $AVR425I is issued at the "BKUPTIM" if the day is not excluded via the "BKUPDAY" parameter and the "BKUPTIM=nn,STOP" was specified.

System Action: $AVRS terminates normally.

User Response: Informational.

$AVR426I - PREVIOUS DAY(S) BACKUP/ARCHIVE RUN BYPASSED

Explanation: $AVR426I is issued when $AVRS is initializing and determines that the BKUPTIM= time has passed since the accumulation task was previously running.

System Action: Message $AVR427A is issued for an operator response.

User Response: Run the bypassed archive or do not run the archive.

$AVR427A - REPLY "CONT" TO TAKE THE BACKUP OR "SKIP" TO BYPASS

Explanation: $AVR425A or $AVR425I is issued in conjunction with $AVR426I.

System Action: If "CONT" is replied, backup continues. If "SKIP" is replied, backup is bypassed.

User Response: Respond "CONT" if the archive that has been bypassed is to be taken at this time or "SKIP" if the bypassed archive is to be ignored and not taken.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - $AVRRRDS IS AT ##.#% UTILIZATION

Explanation: This message is in response to a **F *$avrstaskname*,STATUS** command. The utilization message is displayed on all forms of the STATUS response. This shows the percentage of the $AVRS RRDS data file currently in use.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - WAITING FOR WORK

Explanation: This message is in response to a **F *$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates that $AVRS is ready for work, but has none to process.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - JOB= *job name(job number)*, [STEP= *step name*,] [PROC=*proc name*,] DDNAME= *ddname* [ACCUMULATING | RESPINNING | RESPOOLING | BACKING UP]

Explanation: This message is in response to a F ***$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates that $AVRS is ready for work and is currently processing sysout in one of the functions listed in the message text. **STEP** and **PROC** are optional in the message text. They are displayed only if present. **ACCUMULATING** means $AVRS is currently accumulating the sysout into the database. **RESPINNING** means the sysout is currently respinning from the database. **RESPOOLING** the sysout is currently in MAXLINE RESPIN processing. **BACKING UP** means the sysout is actually being backed up from the database.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - ##### LINES, ##### PAGES ACCUMULATED

Explanation: This message is in response to a F ***$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates that $AVRS is ready for work and is currently accumulating sysout. This message form shows the number of lines and pages accumulated for the sysout so far.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - ##### LINES PROCESSED

Explanation: This message is in response to a **F *$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates that $AVRS is ready for work and is currently in a respinning, backing up or key maintenance function for the sysout. This message form shows the number of lines processed for the sysout so far.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - MARKING FOR BACKUP JOB= *job name(job number*)

Explanation: This message is in response to a **F *$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates that $AVRS is ready for work and is currently preparing the database for a backup.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - [RESTORING | [JOB= *job name(job number*)
DELETING |
UPDATING |
ACKNOWLEDGE/ANNOTATING]

Explanation: This message is in response to a **F *$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates that $AVRS is processing key maintenance. The specific key maintenance function is listed in the message text. You may see the text **NEXT BACKUP DATE** in place of **JOB=**. This means $AVRS is updating the next backup date in the control record.

System Action: None.

User Response: Informational Only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - CURRENT ACCUMULATION CLASSES ARE: *x,y,z,...*

Explanation: This message is in response to a **F *$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates the current primary Accumulation Classes, the ACCMCLS initialization parameter. This message does not include any classes associated with the optional ACCMTBL - "Accumulation Table" parameters. Information on these classes are included in the Initialization Report.

System Action: None.

User Response: None. Informational only.

Note: There are Multiple versions of the $AVR428I message.

$AVR428I - CURRENT ACCMDEST IS: *destination*

Explanation: Informational message in response to a **F *$avrstaskname*,STATUS** command. This form of the **$AVR428I** message indicates the current ACCDEST initialization parameter. This message does not include any destinations associated with the optional Accumulation Table parameters - ACCMTBL,DEST=. Information about these destinations are included in the Initialization Report, $AVRERR DD.

System Action: None.

User Response: None. Informational only.

Note: There are Multiple versions of the $AVR428I message.

$AVR429A - ENTER $AVRS ($=) AND JES (J=) PROCESSING OPTIONS OR "CONTINUE"

Explanation: The operator has INTERRUPTED a $AVRS function via the MODIFY command. The function interrupted allows options to be specified for the sysout being processed for $AVRS and for JES. Valid responses are **CONTINUE** or **$=[K or D],J=[K or D]**.

System Action: Depends on the response. If **CONTINUE**, $AVRS continues as normal. Otherwise, $AVRS **K**eeps or **D**eletes the sysout in $AVRS and JES as specified in the response.

User Response: Respond with **CONTINUE** or **K**eep and **D**elete as desired for both $AVRS and JES. Processing is suspended in $AVRS until this message is replied to. If you specify **J=K** to keep the JES sysout, $AVRS terminates allowing you to process the sysout in JES manually.

$AVR430I - $AVRKMNT - I/O ERROR ON SPOOL FILE RECORD IGNORED

Explanation: An I/O error on the JES SPOOL FILE was detected during key maintenance processing.

System Action: The record causing the I/O error was ignored and processing continued.

User Response: Determine the problem with the JES spool. Correct as required.

$AVR431C - NO SYSLOG CLASS MATCH L#CLSID=*classes* SSSOCLSL=*class*

Explanation: SYSLOG has been selected from a SYSOUT class and that class was not found in the user defined "L#CLSID".

classes = user defined SYSLOG classes
class = classes used for selection of SYSLOG.

System Action: $AVRS issues abend U0431.

User Response: Restart $AVRS after this SYSLOG is made unavailable for selection, OR add a valid LOGCLS# control parameter for all $AVRS classes that will contain SYSLOGs.

$AVR432I - SYSLOG IN CLASS *class* HAS BEEN ASSIGNED A CPU-ID OF "NOID"

Explanation: **class** = class of this SYSLOG

A SYSLOG has been encountered in this class and "**L#CLSID**" was not specified in the initialization options, OR the $AVRLOGI member contains an invalid location(s) of the System ID.

System Action: Processing continues and this SYSLOG may be accessed as "NOID".

User Response: Add a valid LOGCLS# control parameter for all $AVRS classes that will contain SYSLOGs OR correct the $AVRLOGI table entry(s) and reassemble the table.

$AVR433I - OPEN FAILED FOR JES INPUT ACB

Explanation: The JES2 input ACB for $AVRS failed to open successfully.

System Action: This request is bypassed and processing continues.

User Response: None required.

$AVR434W - $AVRKMNT CALLED WITH INVALID WRITER NAME

Explanation: Internal $AVRS errors.

System Action: Processing continues and bypasses the output dataset.

User Response: Call for technical support.

$AVR435C - $AFINDL1 INTERNAL ERROR DURING COMPRESSION FOR *job number - job name*

Explanation: **$AFINDL1** is the location where a compression error was detected. **jobnumber** = the JOB, STC, or TSU number being processed. **job name** = the JOB, STC, or TSU name being processed.

System Action: Abend U0435 will be issued for this error.

User Response: Restart $AVRS and if message $AVR435C is repeated with the same jobname and job number, the job must be made unavailable for selection by $AVRS.

$AVR436I - DATA FILE FULL AND CANNOT BE EXTENDED. ABNORMAL BACKUP WILL BE TAKEN

Explanation: The $AVRS RRDS DATA FILE has filled while accumulating a job. An abnormal backup is one that is taken automatically in response to a Full RRDS or KSDS condition.

System Action: The job being processed is retained and after the backup the job will be re-selected for accumulation.

User Response: #1 If the RRDS fills frequently - add another RRDS file, decrease online retention, or take more frequent backups.
#2 Control the RRDS abnormal backups using the BUMXRD# initialization parameter, $AVRC009, or $AVRC020. These three methods will require manual intervention. They should not be considered if an automation product will restart $AVRS.

$AVR437I - MESSAGE TABLE EXCEEDED IN KSDS RECORD

Explanation: The table that contains user specified messages is filled for this job.

System Action: Normal processing proceeds without any more additions to the"ABRVMSG" table in the KSDS record for this job.

User Response: None required.

$AVR438I - MAXLINES EXCEEDED FOR *job name job number*

Explanation: The maximum number of lines allowed for accumulation was exceeded for job "**jobname**" job number "**jobnnnnn**".

System Action: The work will be respun to the respin class as a file under the $AVRS accumulator.

User Response: None required.

$AVR439A - ENTER JES (J=) PROCESSING OPTIONS OR "CONTINUE"

Explanation: The operator has INTERRUPTED a $AVRS function via the MODIFY command. The function interrupted allows options to be specified for the sysout being processed for JES only. Valid responses are **CONTINUE** or **J=[K or D]**.

System Action: Depends on the response. If **CONTINUE**, $AVRS continues as normal. Otherwise, $AVRS **K**eeps or **D**eletes the sysout in JES as specified in the response.

User Response: Respond with **CONTINUE** or **K**eep and **D**elete as desired for JES. Processing is suspended in $AVRS until this message is replied to. If you specify **J=K** to keep the JES sysout, $AVRS terminates allowing you to process the sysout manually.

$AVR440A - ENTER "I" TO CONFIRM INTERRUPT OR "CONTINUE"

Explanation: The operator has INTERRUPTED a $AVRS function via the MODIFY command. Valid responses are **CONTINUE** or **I**.

System Action: Depends on the response. If **CONTINUE**, $AVRS continues as normal. Otherwise, $AVRS terminates the function and either continues other processing or terminates.

User Response: Respond with **CONTINUE** or **I**. Processing is suspended in $AVRS until this message is replied to.

$AVR441I - KSDS FILE CANNOT BE EXTENDED. ABNORMAL BACKUP WILL BE TAKEN

Explanation: The KSDS FILE does not have space in the key range requested and no more space to split (create a new key range). There is no more room on the pack, no secondary space was allowed, the maximum number of extents has been reached, or the 4.2 GIG limit is exceeded. An abnormal backup is one that is taken automatically in response to a Full RRDS or KSDS condition.

System Action: The daily backup will also be taken at the specified time. The job being processed is retained and after the backup the job will be re-selected for accumulation.

User Response: #1 If the KSDS fills frequently, read - Section A Chapter 9 - Maintaining your KSDS.
#2 Control the KSDS abnormal backups using the BUMXKD# initialization parameter, $AVRC009, or $AVRC020. These three methods will require manual intervention. They should not be considered if an automation product will restart $AVRS.

$AVR442I - INTERRUPT NOT APPLICABLE TO *function* FUNCTION

Explanation: The operator attempted to INTERRUPT a $AVRS function via a MODIFY command. This function does not allow INTERRUPTED processing.

System Action: None.

User Response: Informational Only.

$AVR443I - JES *####, job name* NO RESPIN INDICATED

Explanation: To activate this message the user must install the $AVRUX9 and allow this message to be displayed.

System Action: No respin was indicated but the job was accumulated.

User Response: None required.

$AVR444I - JES *####*, job name RESPUN DUE TO - *reason* <===

Explanation: Reasons for this message are listed below in order of severity:

NOTCTLG if a not cataloged database(NOT- CATLG) dataset was encountered. Controlling initialization parameters: Errors Routine (ERRSRTN) and Accumulation Class (ACCMCLS) OR Requeue SYSOUT class (REQCLAS) and Accumulation Class (ACCMCLS)

JCL ERR if the job failed with a JCL ERROR. Controlling initialization parameter: Errors Routine (ERRSRTN) OR Requeue SYSOUT class (REQCLAS) and Accumulation Class (ACCMCLS)

Abend Code if the job ABENDED. Controlling initialization parameters: Requeue SYSOUT class (REQCLAS) and Accumulation Class (ACCMCLS)
SI/O the $AVRS SYNAD exit was entered or message IEF3761 was not found while processing the SYSMSG dataset for the work. Controlling initialization parameters: Requeue SYSOUT class (REQCLAS) and Accumulation Class (ACCMCLS) HC=#### job exceeded HIGHACC value Controlling initialization parameter: Highest Accumulation Condition Code (HIGHACC) or Exception Table, if defined.

RC=#### an unacceptable CONDITION CODE was detected for this job.

RQ=#### a respin print was requested for this job in the $AVREXCC exception table, $AVRUX7, or the $AVRS synad was entered. Controlling initialization parameter: Highest Accumulation Condition Code (HIGHACC) and the $AVREXCC table.

MSGID if this message was encountered. Controlling initialization parameters: Abbreviated message (ABRVRTN) and Abbreviated message text (ABRVMSG).

AR=#### respin defined SYSOUT class(es) relating to Accumulation Class. Controlling initialization parameters: Automatic Respin (AUTORSP) and Accumulation Class (ACCMCLS)

System Action: The work is respun to the assigned respin.

User Response: None required.

$AVR445I - $AVRS REQUEST TO [DRAIN|ACTIVE] RRDS *index#* IGNORED - ALREADY SET TO [DRAIN|ACTIVE]

Explanation: An operator modify command was issued to either DRAIN or set ACTIVE an RRDS. The command could not be completed because the RRDS was already set to the desired status.

System Action: The modify command is ignored.

User Response: Re-enter the command using the correct RRDS Index number.

$AVR446I - EMERGENCY BACKUP PROCESS BEING INVOKED

Explanation: See $AVR436I or $AVR441I messages for explanation.

System Action: Invokes emergency backup.

User Response: None required.

$AVR447I - $AVRS WILL NOT TERMINATE AFTER BACKUP

Explanation: This message is in response to a MODIFY command to $AVRS that requested a "**BACKUP,NOSTOP**" in order to perform an archive and to continue normal operation.

System Action: Backup processing will proceed and $AVRS will not stop after completion.

User Response: None required.

$AVR448I - JES *#####*, job name CONTAINED *xxxxxxxx* but RESPIN CLASS SUPPRESSED <=====

Explanation: Job "**jobname** " job number "**JES #####**" contained respin reason of "xxxxxxxx".

System Action: The respin class is suppressed and $AVRS was unable to respin the work.

User Response: None required.

$AVR449A - ENTER ACTION FOR REPLY OF "NO" TO MOUNT

Explanation: The operator has replied "NO" to a requested tape mount for backup processing.

System Action: $AVRS is waiting for instructions to continue.

User Response: Enter a reply as follows:

STOP	Terminate $AVRS processing
SHUTDOWN	Terminate $AVRS processing
SHUTDN	Terminate $AVRS processing
END	Terminate $AVRS processing
SKIP	Skip backup processing for this day. This day is treated as a normal skip day. This response is either allowed or disallowed based on the value of the initialization parameter, AUTOREP.

Any Other Response - Retry the mount.

$AVR450I - DUE TO OPERATOR REPLY OF "SKIP" BACKUP WILL NOT BE TAKEN FOR DAY YYYY.DDD

Explanation: The operator has replied "**SKIP**" to message $AVR449A.

System Action: $AVRS will skip backup processing. "**day**" indicates the day of the week the backup was skipped.

User Response: None required.

$AVR451A - REPLY OPTION IS INVALID

Explanation: The operator has given an invalid reply to a message.

System Action: The reply is requested again and processing continues.

User Response: Enter the correct response to the message.

$AVR453I - PROCESSING INTERRUPTED BY OPERATOR COMMAND, JES FILE DELETED.

Explanation: The operator INTERRUPTED $AVRS processing for this sysout via the MODIFY command. The JES file this sysout was accumulated from was deleted by operator command.

System Action: None.

User Response: Informational Only.

$AVR454I - $AVRS TERMINATING DUE TO JES=KEEP PROCESSING OPTION.

Explanation: The operator INTERRUPTED a $AVRS function and either specified or defaulted to a KEEP option on a JES sysout. $AVRS must terminate to avoid processing the sysout again.

System Action: None.

User Response: Informational Only.

$AVR455I - Jobname (*Jobnumber*) ACCUMULATING,WTR NAME=wtrname

Explanation: This Jobname (Jobnumber) has started to accumulate. This message will not display if the $AVRS initialization parameter WRKINFO is set to NO.
NOTE: WTR NAME is an optional portion of the message that is added when available.

System Action: Accumulating.

User Response: None required.

$AVR456A - VSAM *dataset name* UNAVAILABLE FOR OPEN FOR PAST 5 MINUTES RESPOND "WAIT" TO WAIT OR "CANCEL" TO ABEND

Explanation: $AVRS attempted to open the specified dataset and received an indication that the file was unavailable to be opened for output. The open request has been attempted every 30 seconds for 10 attempts prior to issuing the message.

System Action: A request is made to the operator to either keep trying to open the file every 30 seconds for another 5 minutes (WAIT) or to terminate the task (cancel).

User Response: Respond either "WAIT" to continue trying to open the specified file or respond "CANCEL" to terminate the $AVRS task with a $AVR060C message series and abend U0060. Find the cause of the conflicting open and correct it.

$AVR459C - A $AVRS DATA BASE SYNCHRONIZATION ERROR HAS OCCURRED. THE FAILING DATA BASE FILE NAMES ARE:
KSDS NAME = *failing.ksds.name*
RRDS NAME = *failing.rrds.name*
LOCK NAME = *failing.lock.name*
THE SYNCHRONIZED DATA BASE FILE NAMES ARE:
KSDS NAME = *sync.ksds.name*
RRDS NAME = *sync.rrds.name*
LOCK NAME = *sync.lock.name*

Explanation: An access to the $AVRS data base has been attempted and one or more of the $AVRS database DDNAMES in the JCL or environment area do not match those saved in the database itself. The **failing** file names from the JCL/environment area are listed in the message text. The names that should be used are also listed as the **sync** file names.

System Action: The access to the $AVRS database is terminated with a U0459 abend.

User Response: Verify the **failing** file names in the JCL/environment area ($AVRFSEA). Correct and re-run the function, if appropriate. If the **sync** names are in error, contact SEA.

$AVR460I - $AVRS RRDS: *index#* STATUS: *status* IN USE: ###% *dsname*

Explanation: This message results from the F $avrstaskname,STATUS,RRDS Console command. Each RRDS defined in the $AVRS data base is displayed with RRDS Index Number, the Status [ACTIVE|DRAINED|REMOVE], the amount of RRDS blocks being used as a percentage and the DSNAME of the RRDS.

System Action: Informational.

User Response: None required.

$AVR461C - The maximum number of KSDS/RRDS abnormal/backups has occurred

Explanation: The $AVRS KSDS or RRDS Data File has filled while accumulating a job. The initialization parameters, BUMXKSD#, and or BUMXRDS# value(s) have been exceeded. Restarting $AVRS will re-initialize to the values.

System Action: $AVRS terminates.

User Response: If your KSDS fills frequently, read Section A Chapter 9, Maintaining your KSDS. Also the $AVR441I message explanation. Then restart $AVRS.
If your RRDS fills frequently, add an additional RRDS Data File, decrease online retention, or take more frequent backups. Then restart $AVRS. The BUMXKD# and BUMXRD# counters are reset each time the $AVRS accumulator is restarted.

$AVR463I - NUMBER OF LINES RESTORED TO DATA BASE ####

Explanation: Details the fact that a number of lines has been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored lines is displayed along with this message.

User Response: None required.

$AVR464I - NUMBER OF PAGES RESTORED TO DATA BASE ####

Explanation: Details the fact that a number of pages has been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored pages is displayed on the line with this message.

User Response: None required.

$AVR465I - NUMBER OF FILES RESTORED TO DATA BASE ####

Explanation: Details the fact that a number of files has been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored files is displayed on the line with this message.

User Response: None required.

$AVR466I - NUMBER OF JOB KEYS RESTORED ####

Explanation: Details the fact that a number of JOB keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored JOB keys is displayed on the line with this message.

User Response: None required.

$AVR467I - NUMBER OF STC KEYS RESTORED ####

Explanation: Details the fact that a number of STC keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored STC keys is displayed on the line with this message.

User Response: None required.

$AVR468I - NUMBER OF TSU KEYS RESTORED ####

Explanation: Details the fact that a number of TSU keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored TSU keys is displayed on the line with this message.

User Response: None required.

$AVR469I - NUMBER OF SYSLOG KEYS RESTORED ####

Explanation: Details the fact that a number of SYSLOG keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored LOG keys is displayed on the line with this message.

User Response: None required.

$AVR470I - NUMBER OF STATUS RPT KEYS RESTORED ####

Explanation: Details the fact that a number of STATUS RPT keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored ST keys is displayed on the line with this message.

User Response: None required.

$AVR471I - NUMBER OF MESSAGE KEYS RESTORED ####

Explanation: Details the fact that a number of MESSAGE keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored MSG keys is displayed on the line with this message. Messages that are identified by the ABRVMSG initialization parameter that were issued for this job are retained in this type of key.

User Response: None required.

$AVR472I - NUMBER OF CONTROL KEYS RESTORED ####

Explanation: Details the fact that a number of CONTROL keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored CONTROL keys is displayed on the line with this message.

User Response: None required.

$AVR473I - NUMBER OF ANNOTATION KEYS RESTORED ####

Explanation: Details the fact that a number of ANNOTATION keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored ANNOTATION/ACKNOWLEDGMENT keys is displayed on the line with this message.

User Response: None required.

$AVR474W - NO KEYS FOUND FOR PROCESSING IN *xxxxxxxxx*

Explanation: During $AVRS Key Maintenance processing, no file or base key record was found in the spool communication file. xxxxxxxx in the message is the name of the empty KSDS key transaction file - $AVRDEL, $AVRKUPD, $AVRSELR, $AVRCMNT.

System Action: Processing continues.

User Response: Contact $AVRS technical support to determine the cause of this error.

$AVR475I - NO [MAINTENANCE | CUSTOMIZATION] ZAPS ARE IN EFFECT FOR NUMBERS ### THRU ###

Explanation: A request has been submitted to display the maintenance (TYPE=DZ) or customization (TYPE=DC) zap status. However, no maintenance or customization zaps have been applied.

System Action: None.

User Response: Press the Down PFkey to view zap numbers 261 through 520. Press the Up PFkey to view zap numbers 1 through 260.

$AVR476E - [LOAD | GETMAIN] *function* ERROR AT OFFSET *offset*

Explanation: An error has occurred in the specified function at the specified offset.

System Action: None.

User Response: If the function is GETMAIN, increase the region size and retry the function. If the function is LOAD, contact **SEA** for assistance.

$AVR478I - NUMBER OF FICHE RPT KEYS RESTORED

Explanation: Details the fact that a number of MICROFICHE REPORT keys have been restored to the database for a given execution of $AVRSELR.

System Action: The total number of restored FI keys is displayed on the line with this message.

User Response: None required.

$AVR479W - ANNOTATION UPDATE FROM USER *userid* IGNORED. NO RECORD FOR JOB *job name.*

Explanation: User **userid** has input an annotation update/add for job **jobname**. $AVRS was unable to update/add the annotation because the JOB, STC, TSU or SYSLOG base record was not on the data base. More than likely it was deleted by normal $AVRS processing or another user.

System Action: The annotation update/add is ignored. Processing continues.

User Response: Recreate the $AVRS directory you are working in by refreshing it or back up a screen and coming back. If the job no longer appears on the directory, it was deleted before your annotation update/add took place. You may restore the job to the database and re-create your annotation update/add if it was deleted.

$AVR481W - ANNOTATION UPDATE FROM USER *userid* IGNORED. UPDATE OLDER THAN RECORD FOR JOB *job name.*

Explanation: User **userid** has input an annotation update/add for job **jobname**. $AVRS was unable to update/add the annotation because the current timestamp of the annotation record is greater than the timestamp of the update.

System Action: The annotation update/add is ignored. Processing continues.

User Response: Retry the update. The error is caused by JES passing annotation updates to $AVRS out of sequence.

$AVR482W - ANNOTATION UPDATE FROM USER *userid* IGNORED. RECORD FULL FOR JOB *job name*.

Explanation: User **userid** has input an annotation add for job **jobname**. $AVRS was unable to add the annotation because there is no more room in the annotation record for this sysout.

System Action: The annotation update/add is ignored. Processing continues.

User Response: None. The error is caused by trying to add lines to a full annotation record. This record can only process updates now.

$AVR483W - ANNOTATION UPDATE FROM USER *userid* IGNORED. INVALID LINE LENGTH FOR JOB *job name.*

Explanation: User **userid** has input a annotation add for job **jobname**. $AVRS was unable to update/add the annotation because of an invalid line length for the text.

System Action: The annotation update/add is ignored. Processing continues.

User Response: Contact Technical Support for assistance.

$AVR484W - ANNOTATION UPDATE FROM USER *userid* IGNORED. NO ENTRIES TO ADD/UPDATE FOR JOB *job name*.

Explanation: User **userid** has input a annotation update/add for job **jobname**. $AVRS was unable to update/add the annotation because no entries were present in the annotation update/add.

System Action: The annotation update/add is ignored. Processing continues.

User Response: Contact Technical Support for assistance.

$AVR486I - ALLOCATION FOR "SELRDDN" IS USING EXISTING JCL STATEMENT

Explanation: Dynamic Allocation of the "SELRDDN" file was not successful. The Selective Restore JCL contained a DD Statement with the file name defined.

System Action: Processing continues.

User Response: Insure that the statement is intended to be used to enable a delay in processing of this special $AVRS transaction file. It is not proper to override the Dynamic Allocation for this file for normal restores. If the "SELRDDN" file is not processed as the special transaction file, dead blocks will be created in the database.

$AVR497C - TERMINATING DUE TO AN INVALID JES DSNAME: *dsname*

Explanation: Upon a request to JES for a dataset, JES returned a dataset with an invalid dsname. The dsname is displayed.

System Action: The $AVRS accumulator terminates with a U0497 ABEND. $AVRS must terminate since it would receive the same (bad) dataset after each request for the 'next' file from JES.

User Response: The user must either place this file on 'HOLD' or delete the file from JES. $AVRS accumulation may then be restarted. Use of the WRKINFO=YES parameter in the initialization parms will display the current job $AVRS is accumulating. ($AVRS message $AVR455I)

$AVR498C - STATUS REPORT/FICHE REPORT TABLE FULL

Explanation: The number of status/fiche entries selected has exceeded 15000.

System Action: Abend U0498.

User Response: Reduce the number of these entries by making the selection criteria more specific.

$AVR499I - END OF SELECTIVE RESTORE PROCESSING

Explanation: Processing has completed for a given execution of $AVRSELR.

System Action: Informational.

User Response: None required.

$AVR500W - INVALID OR PRIOR RELEASE RESTORED KEY
$AVR500I - X ' ***xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*****'**
$AVR500I - X ' ***xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*****'**
$AVR500I - X ' ***xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*****'**
$AVR500I - X ' ***xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*****'**
$AVR500I - X ' ***xxxxxxxxx*****'**

Explanation: $AVRS found restore keys on the spool but was unable to accumulate them because they do not conform to current record layouts.

System Action: The records are deleted from the spool, but are otherwise ignored by $AVRS.

User Response: Verify that the restore job that put these keys on the spool is using the correct load library. If not, change the job to use the load library for this release of $AVRS and try it again.

$AVR501I - BACKUP HAS SUCCESSFULLY TERMINATED

Explanation: All backup and reporting phases have successfully completed.

System Action: Informational.

User Response: None required.

$AVR503C - $AVRDUPL *ddname* FAILED TO OPEN

Explanation: Open was unsuccessful for the DUPLEX archive file at the time of an archive/backup of the $AVRS' data base.

System Action: $AVRS may issue abend U0503.

User Response: Correct the cause of the open failure and resubmit.

$AVR504C - $AVRBKUP *ddname* FAILED TO OPEN

Explanation: Open was unsuccessful for the archive file at the time of a archive/backup of the $AVRS' data base.

System Action: $AVRS may issue abend U0504.

User Response: Correct the cause of the open failure and resubmit.

$AVR505C - *ddname* DDNAME FAILED TO OPEN

Explanation: Open was unsuccessful for the file *ddname*.

System Action: $AVRS issues Abend U0505 or $AVRS issues message $AVR680C and abend U0680.

User Response: Correct the cause of the open failure and try the operation again. Insure that the DD card is present and that the DDNAME and DSNAMEs are spelled correctly.

$AVR506C - $AVRASUM DDNAME FAILED TO OPEN

Explanation: Open was unsuccessful for the summary report file at the time of an archive of the $AVRS' data base.

System Action: $AVRS issues abend U0504.

User Response: Correct the cause of the open failure and restart $AVRS. If this was a normal run, the backup will proceed once initialization is complete; if backup was a batch job, verify that the "$AVRASUM" DD is specified correctly in the JCL.

$AVR507C - $AVRSTAT DDNAME FAILED TO OPEN

Explanation: Open was unsuccessful for the status report file at the time of an archive/backup of the $AVRS' data base.

System Action: $AVRS issues abend U0504.

User Response: Correct the cause of the open failure and restart $AVRS. If this was a normal run, the backup will proceed once initialization is complete; if backup was a batch job, verify that the "$AVRSTAT" DD is specified correctly in the JCL.

$AVR514W - NO ABEND SUMMARY BY [JOBNAME|DATE/TIME] PRODUCED DUE TO SORT FAILURE

Explanation: No abend summary report by Jobname or Date/Time was produced due to a sort failure.

System Action: This report is not created and processing continues.

User Response: Review the sort's output message log or the system SYSLOG for the reason for the sort failure.

$AVR516C - ABNORMAL BACKUP REQUIRED BUT NO WORK ON FILE AVAILABLE FOR DELETION

Explanation: The work that is being accumulated will not complete due to insufficient space in the $AVRS' data base.

System Action: $AVRS issues abend U0516.

User Response: If **ONLINE STATUS REPORTS** and/or **MICROFICHE ONLINE INDICES** are the only work that remains in the data base, the data base must be expanded or a reduction of days for the FICHONL and/or HISTORY= must be made in order to free additional space in the data base. Or execute $AVRCLEN with a parameter of "VALIDATE PRINT=ERROR" or $AVRDBRP with DETAILS=NO to be sure the database is healthy. $AVRS accumulation may be restarted after the condition is corrected.

$AVR517I - REPORT PROCESSING PHASE COMPLETE

Explanation: The report processing has completed for this backup.

System Action: Complete.

User Response: None required.

$AVR518C - [$AVROLST|$AVRKUPD] DDNAME FAILED TO OPEN

Explanation: Open was unsuccessful for the $AVROLST or $AVRKUPD file at the time of a backup/archive of the $AVRS data base.

System Action: $AVRS issues abend U0518.

User Response: Correct the cause for the open failure and restart $AVRS. If this was a normal backup/archive the archive will proceed once re-initialization is complete. If this was a batch job, verify that the "$AVROLST" or "$AVRKUPD" DD is specified correctly in the JCL.

$AVR521I - *job name (job number)*,FILE= *jes file,* DATE/TIMESTAMP MISMATCH

Explanation: The work identified by **job name (job number)** was being processed when the error occurred. The file being processed is identified by the '**FILE=jes file**" parameter.

System Action: The work that was previously read for this job is retained and the end of data for the job is adjusted to the point of error. Processing continues with the next work.

User Response: None required.

$AVR522E - *job name*(*#####*), FILE(*ddname*), *text* DTE/TIM MISMATCH. EXPECTED X'*xxxxxxxx*", found *X'xxxxxxxx* '

Explanation: There is a Date/Time Stamp Mismatch for either a Hiper Block or a Data Block (*text* defines which type). The Date/Time Stamp for the job that owns this RRDS Block does not match the Date/Time Stamp found on the actual block.

System Action: If this is issued by the $AVRCLEN job, the job will continue to run but will set a non-zero return code.

User Response: None.

$AVR523I - "SELRONL" REDUCED TO "###" DUE TO FULL DATASET

Explanation: The retention periods for restored work have been reduced to the number shown for "###" due to an emergency archive condition.

System Action: Processing continues.

User Response: None, however if enough emergency archives happen between normal backups, all restored work will be deleted from the data base. Frequent emergency backups are a sign to reconsider the size of the data base and/or online retention of work.
#1 If the RRDS fills frequently, add another RRDSfile, decrease online retention, or take more frequent backups. See $AVR436I.
#2 If the KSDS fills frequently, read Section A Chapter , Maintaining Your KSDS. See $AVR411I.

$AVR524C - CATALOG VERIFICATION FAILED DUPLEX DSN=

Explanation: The verification process that determines if the DUPLEX archive file was successfully cataloged, found that it was not the most recent generation.

System Action: $AVRS issues abend U0524.

User Response: After the condition that prevented the file from being cataloged is corrected, $AVRS may be restarted.

$AVR525C - CATALOG VERIFICATION FAILED BACKUP DSN=

Explanation: The verification process that determines if the archive file was successfully cataloged found that it was not the most recent generation.

System Action: $AVRS issues abend U0525.

User Response: After the condition that prevented the file from being cataloged is corrected, $AVRS may be restarted.

$AVR526I - *job name* (*job number*), FILE= *file name*, KEY RECORD NOT FOUND

Explanation: The key for this job "jobname"and job number could not be found on the KSDS file.

System Action: $AVRS issues the message and continues.

User Response: None.

$AVR527I - *job name* (*job number*) HAS NO RECORDS - JOB NOT RESTORED

Explanation: During the reload of this **job name and jobnumber,** $AVRS= found no data records for the job.

System Action: The reload for this **jobname and jobnumber** is skipped and processing continues.

User Response: None, informational only.

$AVR528I - *job name* (*job number*), FILE (*file name*), FILE HAS NO RECORDS WRITTEN TO DATA BASE

Explanation: During the reload of this **job name** and **job number**, $AVRS found no data records for file **file name**.

System Action: The reload for this **job name** and **job number** file **file name** is skipped and processing continues. The return code is set to "4".

User Response: None, informational only.

$AVR529I - INVALID HIPER BLOCK POINTER FOR JOB=*jobname, jobnumber, FILE=ddname,* HIPER BLK=X'xxxxxxxx'

Explanation: Informational message.

System Action: The job continues to run.

User Response: None.

$AVR531C - JOB NAME (*job number*),FILE KEY READ ERROR, JOB HAS FILES SKIPPED

Explanation: A read operation for a $AVRS file record (TYPE 101) has been attempted on the $AVRS KSDS file. The operation has failed.

System Action: Program continues, skipping this file of this job.

User Response: This job is counted as one dropped. (i.e. it is NOT unloaded) There is possible damage to the $AVRS data base. Call for technical support.

$AVR539I - ALLOCATION IN MODULE program name GAVE RC=X'###1,EC=x###2'KEY=X'###3',NO.=X'###4',LNG= X'###5',VALUE=TEXT

Explanation: A dynamic allocation or deallocation, SVC 99, has failed.

System Action: Function terminates.

User Response: Locate the reason in the OS/390 MVS Authorized Assembler Services Guide for resolution to the problem. Ensure that a screen print or copy of the **exact text** is available.

###1 = SVC 99 A two-byte hexadecimal return code.

###2 = SVC 99 A two-byte reason code.

###3= Key (S99TUKEY): A two-byte hexadecimal number that identifies the function that the DYNALLOC macro performed.

###4= Number (S99TUNUM): A two-byte hexadecimal number that identifies the length and number of parameter combinations in the text unit.

###5 = Length (S99TULNG): A two-byte hexadecimal number specifying the length of the parameter field (S99TUPAR) that follows. For information retrieval text units, the system uses this field to indicate the length of the

information it is returning, or to indicate that the information was not available, by setting it to 0.

TEXT = Parm (S99TUPAR): A variable-length field which shows the function identified by the value in the KEY field. See “SVC 99 Parameter List Verb Codes and Text Units, By Function” in OS/390 MVS Authorized Assembler Services Guide for a description of the values.

$AVR576C - THE RRDS FILE(S) ARE FULL!! REDEFINE WITH MORE SPACE OR ADD ADDITIONAL RRDS FILE(S) AND RERUN.

Explanation: The RRDS Data file(s) has filled while being loaded by $AVRFMRR.

System Action: $AVRFMRR issues abend U0576.

User Response: To increase the size of the RRDS Data file that filled, delete and redefine ALL files in the $AVRS' data base after increasing the space for that component or adding additional RRDS Data files.

$AVR577C - KSDS FILE FULL. REDEFINE ALL FILES AND INCREASE THE SIZE OF THE KSDS AND RERUN FORMAT/RELOAD

Explanation: The KSDS file has filled while being loaded by $AVRFMRR.

System Action: $AVRFMRR issues abend U0577.

User Response: Check available DASD space, then delete and re-defined ALL $AVRS data base files, increasing the size of the KSDS file and re-run $AVRFMRR, the format/reload job.

$AVR579W - INVALID OLD KEY TYPE PASSED TO $AVRKYCV

Explanation: The key passed to the key conversion program is invalid or no longer supported.

System Action: Processing continues with the next record from the file being loaded.

User Response: Ensure that the file being processed is a file created by a previous release of $AVRS. Releases prior to R3.0A are no longer supported.

$AVR580W - INVALID RECORD RDW PASSED TO $AVRCVKY

Explanation: The RDW of the record passed to the key conversion program is invalid for conversion.

System Action: Processing continues with the next record from the file being loaded.

User Response: Ensure that the file being processed is a file created by a previous release of $AVRS. Releases prior to R3.0A are no longer supported.

$AVR581C - LOGIC ERROR DURING RESTORE - NO MAIN KEY AVAILABLE

Explanation: A processing error has been detected when the $AVRS' data base was being reloaded at format time.

System Action: $AVRFMRR issues abend U0581.

User Response: Call vendor for support.

$AVR582C - KEY SORT FAILED - ABENDING WITH THE SORT RETURN CODE

Explanation: An error has been detected when sorting the keys for the $AVRS data base reloaded at format time.

System Action: $AVRFMRR issues a user abend that is the failing return code from the sort.

User Response: Correct the cause of the sort failure and rerun $AVRFMRR. Check the sort files. It may be possible to correct the problem without rerunning $AVRFMRR. Read the recovery technique in the $AVRFMRR JCL member in the INSTLIB.

$AVR583I - $AVRS FORMAT COMPLETED WITH SYSUT1 MISSING OR DUMMIED

Explanation: The $AVRS Database format job was run with the SYSUT1 DD statement either missing or dummied. Therefore no data was loaded on the database during the format.

System Action: Informational message is issued.

User Response: Verify that the format job was run as desired.

$AVR584C - KSDS INCORRECTLY DEFINED, KEY DEFINITION MUST BE KEYS(64 0)

Explanation: The KSDS was checked to determine if it was correctly defined for key length and location. If was determined that the define for the key as not KEYS(64 0) for the KSDS file.

System Action: This message will cause a U0584 abend.

User Response: Correct the definition of the KSDS and re-execute the job being run.

$AVR591C - A $AVRSELR IS CURRENTLY RUNNING, THIS JOB MUST NOT RUN WHEN A SELECTIVE RESTORE IS RUNNING

Explanation: This message appears on the $AVRCLEN-01 report to indicate that there is a $AVRSELR program executing. This is determined by an ENQ done for major name.
"$ELRCLEN" minor name
"The LOCK File DSNAME"

System Action: This message is accompanied by an abend U0591.

User Response: After all $AVRSELR executions are complete, the $AVRS accumulation task must be started to accumulate the keys for the restores that were running prior to re-executing this job. If this is not done or the $AVRSELR maintenance keys are held or deleted, the work that was restored will receive message $AVR930I when referenced.

$AVR592C - $AVRSELR KEYS ARE CURRENTLY ON THE JES SPOOL, ALL $AVRSELR KEYS MUST BE ACCUMULATED PRIOR TO RUNNING $AVRCLEN

Explanation: This message appears on the $AVRCLEN-01 report to indicate that there are $AVRSELR keys on the JES spool.

System Action: This message is accompanied by an abend U0592.

User Response: After all $AVRSELR executions are complete, the $AVRS accumulation task must be started to accumulate the keys for the restores that were running prior to re-executing the $AVRCLEN program. If this is not done or the $AVRSELR maintenance keys are held or deleted, the work that was restored will receive message $AVR930I when referenced.

$AVR593C - INVALID LOCK CNTL DATA, UNABLE TO CHECK FOR DATA BASE SPACE

Explanation: During $AVRSELR processing the program attempted to verify that enough free space was available in the data base to hold sysout being restored. The program was unable to access the $AVRLOCK file to verify the available space.

System Action: Processing continues without space checking.

User Response: Verify that a $AVRLOCK statement exists in the $AVRSELR JCL and try the operation again.

$AVR594W - INSUFFICIENT DATABASE SPACE FOR RESTORE OF *Job name* (*Job number*)

Explanation: During $AVRSELR processing the program detected that enough space was not available in the data base to hold job Job name(Job number).

System Action: Processing continues by skipping this sysout.

User Response: Free up more database space by deleting work or expand the size of your database and try the restore again.

$AVR595C - DATA BASE FULL DURING RESTORE OF *Job name* (*Job number*) - DELETES *xxxxxxxxxx*

Explanation: During $AVRSELR processing the program detected an out of space condition while restoring **Job name(Job number).** The program attempted to delete any partial restoration of this job and informs the user via **xxxxxxxxxx**.

System Action: Processing terminates.

User Response: Free up more data base space by deleting work or expand the size of your data base and try the restore again. Also check your data base via $AVRDBRP and $AVRCLEN for any "dead" blocks.

$AVR596C - CLASS "*class*" IS DUPLICATED IN THE "ACCMTBL" AND "ACCMCLS" PARMS FOR DEST "*destination*"

Explanation: During initialization, $AVRINIT detected a sysout class/destination combination that was duplicated between three different initialization statements; ACCMTBL and ACCMCLS/ACCDEST.

System Action: Processing terminates.

User Response: Remove the duplicated sysout class from either parameter and restart $AVRS.

$AVR597C - ACCM. LOOP FOR CLASS "*class*", DEST "*destination*"
[AND FROM "MAXLCLS/MAXLDEST" PARMS MAY EXIST]
[AND FROM "MAXLCLS/DUMPDEST" PARMS MAY EXIST]
[AND FROM MAXLINE TABLE PARMS PARMS MAY EXIST]
[AND FROM "REQCLS/LCLDEST" PARMS MAY EXIST]
[AND FROM "REQCLAS/ROUTTBL" MAY EXIST]

Explanation: $AVRS has detected a condition where $AVRS would or could produce sysout in its accumulation queues. If any of the optional message text is produced, $AVRS is telling the user what parameters are causing the potential problem.

System Action: If any of the optional message text is produced, $AVRS terminates with a U1002. Otherwise, $AVRS resets the output destination to the "LCLDEST" initialization parm value and respins the dataset.

User Response: If $AVRS has abended, alter the initialization parms or the maxline table parms so that an accumulation loop can not occur and restart $AVRS. Otherwise, change any potential accumulation loops in $AVRUX7 or $AVRUX8 and recycle $AVRS.

$AVR598C - INVALID RESPIN FORM *form* REQUESTED. INSTALLATION DEFAULT USED

Explanation: A respin request for an invalid form number (‘form’ in the message) was encountered. $AVRS uses the installation default form number for the respin.

System Action: Processing continues.

User Response: Verify that any forms numbers used in dynamic respin requests (such as $AVRUX8) are syntactically correct.

$AVR601C - OPEN FAILURE ON DDNAME (RSREMOTE)

Explanation: The OPEN failed for DDNAME RSREMOTE when a respin for a job was required.

System Action: $AVRS issues abend U0601.

User Response: Correct the cause of the open failure and restart $AVRS.

$AVR602C - OPEN FAILURE ON DDNAME(RSLOCAL)

Explanation: The OPEN failed for DDNAME RSLOCAL when a respin for a job was required.

System Action: $AVRS issues abend U0601.

User Response: Correct the cause of the open failure and restart $AVRS.

$AVR622I - function FUNCTION COMPLETED RC X'xxxxxxxx'

Explanation: The requested function ended with a Return Code of xxxxxxxx. Any return code other than zero (0) indicates an error has occurred. A previous message provides the reason for any error conditions.

System Action: Processing continues.

User Response: None required.

$AVR650C - OPEN FAILURE - $AVRUNLD UNDEFINED

Explanation: Open failed for $AVRUNLD in the $AVRS data base unload program.

System Action: $AVRUNLD issues abend U0650.

User Response: Ensure that $AVRUNLD DDNAME is properly defined and then rerun $AVRUNLD.

$AVR651C - FOLLOWING DDNAME(S) NOT SUPPLIED:*xxxxxxxx*

Explanation: The DDNAMEs listed were not supplied in the $AVRUNLD procedure.

System Action: Lists the missing DDNAMEs. *xxxxxxxx*

User Response: Add the correct DDNAMEs and datasets to the JCL being executed and rerun the job.

$AVR663I - NUMBER OF LINES UNLOADED *########*

Explanation: Details the fact that a number of print lines have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of unloaded lines is shown at the end of this message.

User Response: None required.

$AVR664I - NUMBER OF PAGES UNLOADED ########

Explanation: Details the fact that a number of pages have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of pages is shown at the end of this message.

User Response: None required.

$AVR665I - NUMBER OF FILES UNLOADED ########

Explanation: Details the fact that a number of files have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of files is shown at the end of this message.

User Response: None required.

$AVR666I - NUMBER OF JOB KEYS UNLOADED ########

Explanation: Details the fact that a number of JOB keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR667I - NUMBER OF STC KEYS UNLOADED ########

Explanation: Details the fact that a number of STC keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR668I - NUMBER OF TSU KEYS UNLOADED #######

Explanation: Details the fact that a number of TSU keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR669I - NUMBER OF SYSLOG KEYS UNLOADED #######

Explanation: Details the fact that a number of SYSLOG keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR670I - NUMBER OF STATUS RPT KEYS UNLOADED ########

Explanation: Details the fact that a number of STATUS RPT keys unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR671I - NUMBER OF MESSAGE KEYS UNLOADED ########

Explanation: Details the fact that a number of MESSAGE keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR672I - NUMBER OF CONTROL KEYS UNLOADED ########

Explanation: Details the fact that a number of CONTROL keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR673I - NUMBER OF ANNOTATION KEYS UNLOADED ########

Explanation: Details the fact that a number of ANNOTATION keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR678I - NUMBER OF FICHE RPT KEYS UNLOADED ########

Explanation: Details the fact that a number of FICHE RPT keys have been unloaded from the data base by $AVRUNLD.

System Action: The actual number of keys is shown at the end of this message.

User Response: None required.

$AVR679I - NUMBER OF JOBS DROPPED - #

Explanation: This is the count of jobs dropped during the unload process. Please refer to message(s) $AVR071I.

System Action: None, informational.

User Response: None required.

$AVR680C - ABOVE ERRORS HAVE OCCURRED CORRECT AND RERUN THE JOB

Explanation: Errors have occurred during processing. The errors are detailed in the preceding messages.

System Action: Abend U0680.

User Response: Correct the errors and rerun the job.

$AVR681C - ddname NOT FOUND OR IS A NULLFILE

Explanation: **ddname** is not found or dummied out in the JCL.

System Action: Abend U0680 may be issued.

User Response: **ddname** is required to run the job. Correct the JCL and rerun the job.

$AVR682C - READ JFCB FAILED FOR *ddname*

Explanation: $AVRS attempted to check that all data base files were present in the JCL. The internal check failed.

System Action: Abend U0680.

User Response: Ensure that all $AVRS data base files, usually just the LOCK File, are in the JCL with the correct DDNAME(S). Correct any errors and rerun the job.

$AVR683C - TIOT EXTRACT FAILED - CORRECT AND RERUN THE JOB

Explanation: $AVRS attempted to check that all data base files were present in the JCL. The internal check failed. Usually followed by message $AVR680C.

System Action: Abend U0680.

User Response: Ensure that all $AVRS data base files, usually just the LOCK File, are in the JCL with the correct DDNAME(S). Correct any errors and rerun the job.

$AVR699I - END OF UNLOAD PROCESSING

Explanation: Processing has completed for an execution of $AVRUNLD.

System Action: Complete.

User Response: None required.

$AVR700C - INVALID PARM VALUE GIVEN

Explanation: An invalid PARM value has been specified for this program. The only valid options are: NOBANNER and WTOR.

System Action: $AVRS issues abend U0700.

User Response: Correct the erroneous parm and rerun the failing program.

$AVR701A - ENTER PASSWORD FOR PULL-OFF

Explanation: The user option for the operator password is in effect.

System Action: Three attempts to supply the correct password are allowed. If the third attempt is incorrect, message $AVR702C and abend U0702 is issued.

User Response: Reply with the operator password as specified in the current initialization options.

$AVR702C - INVALID PASSWORD REPLIED TO $AVR701A

Explanation: The user did not reply with the correct password to message $AVR701A.

System Action: If the third attempt is incorrect, message $AVR702C and abend U0702 is issued.

User Response: Enter a correct password.

$AVR703A - ENTER JOB/SYSLOG SELECTION CRITERIA OR END

Explanation: This message is issued when $AVRPULL or $AVRTAPE is being run with "**PARM=WTOR**" or being run from TSO.

System Action: Response is acted upon.

User Response: Enter the appropriate selection criteria or end.

$AVR705C - SYSPRINT DCB FAILED TO OPEN

Explanation: This message is issued when $AVRPULL/$AVRTAPE/$AVRFICH/$AVRFDEL is executed as a batch job without any parm options and the open failed for SYSPRINT.

System Action: Abend U0705 is issued.

User Response: Ensure that SYSPRINT is properly defined and rerun the program.

$AVR707W - INVALID VALUE FOR *parameter*

Explanation: This message is issued when the value for the parameter specified in the message does not conform to the requirements of the parameter.

System Action: Request is bypassed.

User Response: Correctly specify the value according to the parameters' requirements and resubmit the request.

$AVR715I - PULL COMPLETED

Explanation: This message is issued when the requested argument(s) are successfully passed.

System Action: Complete.

User Response: None required.

$AVR718W - UNKNOWN PULL SELECTION CRITERIA

Explanation: This message is issued when the pull request contained keywords other than those defined by message $AVR703A.

System Action: Request is bypassed.

User Response: Correctly specify the keywords and their values and resubmit the request to $AVRPULL/$AVRTAPE.

$AVR719I - *work request* <==REQUEST SELECTED *job name job number*

Explanation: This message is issued when the pull request found the data as per the selection criteria. The selection number portion of the message may not appear for history selections - AH, ST, FI, FH.

System Action: The jobname and number is shown along with the particular work request that was responsible for the selection.

User Response: Informational.

$AVR720W - REQUESTED ARGUMENT(S) NOT ON FILE FOR SELECTION #

Explanation: This message is issued when the file did not contain the data according to the selection criteria. Additional message text may be "or matched by another selection".

System Action: Request is bypassed.

User Response: Review the keywords and their values. Correct them as necessary, resubmit the request to $AVRPULL/$AVRTAPE/$AVRSELR.

$AVR721W - NO WORK MATCHED ANY SELECTION CRITERIA

Explanation: The specified selection criteria matched no sysout from the input files.

System Action: Processing is terminated with a warning condition code. The $AVRLOST and $AVRDEL files are NOT produced.

User Response: Re-examine the selection criteria and make any changes needed to cause them to select work. $AVRTPRP can be run on the input files to verify the names of the sysout the input files contain.

$AVR730W - SELECTION DENIED BY USER EXIT

Explanation: This message is issued when the selection was denied by the user exit ($AVRUX2).

System Action: Request is bypassed.

User Response: Respecify your selection criteria and resubmit the request to $AVRPULL/$AVRTAPE.

If the selection continues to be denied, contact the systems programmer who is responsible for the content of the user exit.

$AVR741C - FIRST RECORD ON "*ddname*" FILE NOT A $AVRS CONTROL RECORD

Explanation: The first record on the sequential input file was not a $AVRS control record.

System Action: Abends with a U0741.

User Response: Ensure that the correct file is being used for input and that if it is a multi-volume file the first volume is read first.

$AVR742C - AN UNKNOWN CONTROL KEY WAS ENCOUNTERED AND WAS IGNORED. KEY=*"generic key"*

Explanation: A record, thought to be a Control Key, could not be validated.

System Action: The message is presented with the generic key and the job continues.

User Response: None, informational.

$AVR743I - DATABASE MAINTENANCE IN PROGRESS #### JOBS DELETED

Explanation: During key maintenance, $AVRS indicates the number of maintenance keys processed. This message is issued every 200 keys. The number indicated in the message includes deletes AND updates.

System Action: None required.

User Response: Informational.

$AVR744I - DATA BASE MAINTENANCE PROCESSING (*key maint writer name*) FOR JOB= *job name,job number* IS COMPLETE.

Explanation: $AVRS Key Maintenance for a $AVRBKUP or $AVRFICH has completed. Both the jobname and job number of the job that created the key maintenance is presented for informational purposes.

System Action: $AVRS accumulation will proceed.

User Response: None required.

$AVR749C - INVALID RECORD SIZE FOR FICHE OUTPUT FILE

Explanation: Issued by $AVRFICH. Indicates that the supplied LRECL for DDNAME $FICHOUT is less than 81 or greater than 260 bytes.

System Action: $AVRFICH terminates with a return code of 4.

User Response: Verify that the LRECL is within these limits and resubmit the microfiche creation job.

$AVR750I - FICHE DIMENSIONS FOR RUN--> ### ROWS BY ### COLUMNS -- LINECOUNT= ### STARTROW=*X* CONTROL=*Y* INDEXFRAME=YES BANNERS=YES

Explanation: Issued by $AVRFICH. Indicates the dimensions of the microfiche in effect for this fiche file creation. The values listed are either defaults or are your declared processing options.

System Action: Informational.

User Response: Verify that the values shown match the desired values.

$AVR751W - FOLLOWING SYSIN CARDS CONTAINED ERRORS

Explanation: Issued by $AVRFICH. Specifies that some processing options were found to contain errors.

System Action: Precedes message $AVR752C.

User Response: Correct the errors and rerun the job.

$AVR752C - SYSIN *card image*

Explanation: Issued by $AVRFICH. Lists the image of the request for work or the control option that is in error, followed by an error description. Issued once for each work request or control option that is found to contain an error.

System Action: $AVRFICH terminates with a return code of 4.

User Response: Correct the error condition(s) and resubmit.

$AVR753C - FILE OPEN FAILED : *file name*

Explanation: Indicates an open failure for the file indicated by filename.

System Action: Terminates with a return code of 4.

User Response: Determine the reason for the open failure, correct and resubmit.

$AVR754I - THE FOLLOWING SELECTION ARGUMENTS PROCESSED SUCCESSFULLY

Explanation: Issued by $AVRFICH.

System Action: Precedes message(s) $AVR755I.

User Response: None required.

$AVR755I - work selection criteria *card image*

Explanation: Issued by $AVRFICH. Lists the card image of correctly specified work selection criteria cards.

System Action: Informational.

User Response: None required.

$AVR756I - #### SYSIN CARDS PROCESSED ** ## CARDS IN ERROR ** ## SEARCH ARGUMENTS BUILT

Explanation: Issued by $AVRFICH. Summarizes SYSIN card processing as indicated.

System Action: If there are any card errors, processing will terminate at this point.

User Response: If there were any errors, correct the problems and resubmit the job.

$AVR757I - SORT CRITERIA SELECTED - *criteria1, criteria2, criteria3, etc.*

Explanation: Issued by $AVRFICH. Lists the sort criteria selected by the 'SORT=' option.

System Action: Informational.

User Response: Verify that the options specified are as desired.

$AVR758C - PROCESSING TERMINATED DUE TO ERRORS

Explanation: Issued by $AVRFICH or $AVRFDEL.

System Action: Processing cannot continue due to errors encountered.

User Response: See the message identifying the specific error for action.

$AVR759C - SORT FAILED

Explanation: Issued when an external sort call has failed.

System Action: Processing is terminated at this point.

User Response: See the log and message files to determine the specific reason for the failure. Correct and resubmit.

$AVR760I - DELETIONS TO BE PROCESSED FOR FICHE RUN DATE AND TIME--> *yyyyddd,hhmmss*

Explanation: Issued by $AVRFDEL. Lists the timestamp specified by the sysin card or PARM field of the original fiche run for which delayed delete are to be processed.

System Action: Informational.

User Response: Verify that this timestamp matches that of the originals run (see message $AVR767I).

$AVR762C - PROCESSING TERMINATED - NO FILES MARKED FOR DELETION FOUND

Explanation: Issued by $AVRFDEL. No files which had been output to microfiche on the date and time specified were found to be marked for delayed delete. Either the timestamp was specified incorrectly, or no files were marked for delayed deletion, or these files have already been deleted from the $AVRS data base.

System Action: $AVRFDEL terminates with a return code of 4.

User Response: Verify timestamp, and if in error, correct and resubmit.

$AVR763I - ###### FILES READ ** ###### FILES SELECTED FOR DELETION

Explanation: Issued by $AVRFDEL. '######' specifies the number of files from the $AVRS data base which were scanned, and '######' is the number of files which were found to match the selected criteria and which will be deleted.

System Action: Informational.

User Response: None required.

$AVR765C - THE LRECL OF CONTROL FILE IS TOO LARGE

Explanation: Issued by $AVRFICH. A control file has been designated by the 'CONTROL=DDNAME' initialization parameter, but the LRECL of the file specified exceeds the LRECL of the fiche output file (ddname=$FICHOUT). The LRECL of this file must be no larger than that of the output file.

System Action: $AVRFICH terminates with a return code of 4.

User Response: Correct and resubmit.

$AVR766I - ###### FILES READ ** ###### FILES SELECTED FOR OUTPUT

Explanation: Issued by $AVRFICH. Processing of work selection criteria against the $AVRS data base has been completed. '######' indicates the number of reads from the data base during processing. '######' indicates the number of files selected.

System Action: Issued in conjunction with messages $AVR768I and $AVR769I.

User Response: None required.

$AVR767I - PROCESSING COMPLETE -- NORMAL END OF JOB -- FILE CREATION DATE/TIME= *yyyyddd,hhmmss*

Explanation: Issued by $AVRFICH. Processing has completed normally and a output fiche file has been created. The date and time indicated by '**yyyyddd,hhmmss**' is the time stamp required as a parameter for recreating the file using the RERUN option.

System Action: Complete.

User Response: None required.

$AVR768I - *job name type jobnane ###* SELECTED BY--> selection *card image*

Explanation: Issued by $AVRFICH. Issued for each job selected for output to fiche.

jobname is the 1-8 character job name identifier of the work.
type indicates the type of job, either STC, JOB, TSU or LOG.
jobname is the five digit job number.
gives the number of sysout datasets within the job that were selected for output.
selection card image lists the 80 byte image of the work selection request which selected this work. If this is a recreate, 'selection card image' will be 'RERUN'.

System Action: Informational.

User Response: None required.

$AVR769I - FILE SELECTION COMPLETE

Explanation: Issued by $AVRFICH. Search of the $AVRS data base for work matching the selection criteria has been completed.

System Action: Complete.

User Response: None required.

$AVR770C - OPEN OF A MICROFICHE CONTROL FILE FAILED

Explanation: Issued by $AVRFICH. An open failure for one of the output control files used internally by $AVRFICH has failed.

System Action: Processing is terminated. The fiche output file will have already been created. However, no delete processing will be done, no files will be marked as having been created and no fiche online status report for this run will be produced. *It is recommended that this file not be used.*

User Response: Correct the errors and resubmit. The writer names used by $AVRFICH as control files are $AVRKUPD and $AVRFICH.

$AVR771C - DYNAMIC ALLOCATION OF A MICROFICHE CONTROL FILE FAILED

Explanation: Issued by $AVRFICH or $AVRFDEL. An attempt to allocate one of the microfiche control files used internally has failed.

System Action: Processing is terminated. The fiche output file will have already been created. However, no delete processing will be done, no files will be marked as having been output, and no fiche online status report for this run will be produced. *It is recommended that this file not be used.*

User Response: Correct the errors and resubmit.

The writer names used by $AVRFICH as control files are $AVRKUPD and $AVRFICH.

$AVR772I - FICHE FILE PRODUCED - ###### RECORDS PROCESSED - #### FICHE CREATED

Explanation: Issued by $AVRFICH. Creation of the fiche output file is complete. '######' is the total number of records from the data base written to output. '####' is the number of fiche required for this file.

System Action: Informational.

User Response: None required.

$AVR774W - NO SELECTION REQUESTS SPECIFIED

Explanation: Issued by $AVRFICH. Work selection criteria cards were not included in the sysin, and RERUN was not specified. Therefore, the program has no criteria for selecting work for output from the data base.

System Action: Processing is terminated.

User Response: Include valid work selection criteria cards in the sysin and resubmit.

$AVR775W - PROCESSING TERMINATED - NO FILES SELECTED FOR OUTPUT

Explanation: Issued by $AVRFICH. No work was found to match the specified selection criteria, or it was already marked as fiched in a previous run, and therefore no output file can be created.

System Action: Processing is terminated.

User Response: If this is not a RERUN, either the type of work you wish to output is not contained in the data base, or you have specified your selection criteria incorrectly. If this is a RERUN, you have either specified the timestamp from the original run incorrectly or no work from the original run remains on the data base. Check the value you specified against message $AVR767I from the original run. Correct and resubmit.

$AVR776I - RERUN REQUESTED FOR PREVIOUS RUN DATE AND TIME--> *yyyyddd,hhmmss*

Explanation: Issued by $AVRFICH. The program has been requested to re-create the fiche output file previously produced on the date and time indicated by '**yyyyddd,hhmmss**'. This timestamp should match the value given in message $AVR767I from the original run.

System Action: Informational.

User Response: None required.

$AVR777C - NO CARRIAGE CONTROL SPECIFIED FOR FICHE OUTPUT FILE

Explanation: Issued by $AVRFICH. The microfiche output file to be created by the program did not specify any carriage control. (RECFM=xxA, or RECFM=xxM). Carriage control must be included for this file (DDNAME=$FICHOUT).

System Action: Processing is terminated.

User Response: Correct the problem by specifying RECFM= A or M on the specified file and resubmit.

$AVR780C - STARTROW= EXCEEDS ROWS= ##

Explanation: Issued by $AVRFICH. The STARTROW= or the CONTROW= parameter exceeds the maximum value allowed for the field. The default maximum value is 12.

System Action: The control record is rejected.

User Response: Correct the problem with the control record and resubmit it to $AVRFICH.

$AVR781I - ROWINDEX=

Explanation: Issued by $AVRFICH. The message is a part of the normal processing of the program. It is issued after all control records are read.

System Action: Processing continues.

User Response: Informational.

$AVR782C - ROW VALUE NOT FOUND IN ROWINDEX FOR *parameter*

Explanation: Issued by $AVRFICH. The STARTROW= or the CONTROW= parameter could not be found.

System Action: The control record is rejected.

User Response: Correct the problem with the control record and resubmit it to $AVRFICH.

$AVR806I - INVALID SCROLL VALUE ENTERED

Explanation: Scroll == > value field in upper right of screen has been completely or partially overwritten.

System Action: Cursor is placed in scroll value field.

User Response: Re-initialize scroll value with P/PAGE, C/CUR, H/HALF, or blank the entire field for the default value, usually page.

$AVR807C - JOB SUBMISSION DENIED BY USER EXIT

Explanation: Issued by $AVRUXA. The JCL User Exit has cancelled the job submission.

System Action: The job is canceled and all output is purged

User Response: Contact the group or individual responsible for $AVRS within your organization.

$AVR807U - *xxxxxxxx*

Explanation: Issued by $AVRUXA. The JCL User Exit has canceled the job submission and issued this user provided message.

System Action: The job is canceled and all output is purged.

User Response: Contact the group or individual responsible for $AVRS within your organization.

$AVR808C - NO INPUT RECORDS FOR DDNAME - *ddname*

Explanation: The extended scan ($AVRSXPR) or maxline ($AVRMXPR) parsing program input file is dummied out.

System Action: Processing is terminated.

User Response: Point the input file DDNAME to valid control records and try the operation again.

$AVR815C - SECURITY PROCESSOR DENIED ANNOTATION *xxxxx*

Explanation: The external security module $AVRSECU has denied your access and/or update to Annotations

System Action: The directory screen is re-displayed.

User Response: Obtain proper security level.

$AVR816I - SCREEN PRINTED

Explanation: A PRINT SCREEN REQUEST has completed for the desired panel.

System Action: Complete.

User Response: None required.

$AVR825C - ENVIRONMENT AREA MODULE $AVR*xx*EA IS NOT AVAILABLE TO $AVRS

Explanation: The module $AVRXXEA, where xx is the selection environment area module characters, was not found in a library accessible by the $AVRS.

System Action: An error panel is displayed and the task is terminated.

User Response: If the correct environment area characters were used, ensure the TSO session has access to the library and retry. If not, re-enter and use the correct characters.

$AVR826I - *xxxxxxxx* SIGNON ID NOT DEFINED TO $AVRS

Explanation: The ID entered from the $AVRS VTAM signon panel is not known to $AVRS.

System Action: Signon is terminated.

User Response: Enter the correct userid or, if the ID is not in the field created by the $AVRCRSE program, add the ID and password and recreate the file.

$AVR827A - INVALID PASSWORD GIVEN RE-ENTER

Explanation: The password entered does not match the one defined for the userid.

System Action: Signon is retried three times.

User Response: Re-enter password.

$AVR828C - LOAD FAILURE FOR MODULE:*module*

Explanation: This message is displayed when $AVRS fails to load the module properly. Normally, this is a system 806 abend which has been trapped.

System Action: This task is terminated.

User Response: Place the missing module into the proper load module library and restart. If the problem persists, contact SEA Technical Support.

$AVR829A - VALIDATION OF NEW PASSWORD IS REQUIRED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user attempted to change the password: A NEW PASSWORD was entered without the current PASSWORD being entered.

System Action: The current PASSWORD field is blanked out and the LOGON panel is redisplayed.

User Response: Retry, entering the current PASSWORD data.

$AVR830I - VALIDATION OF NEW PASSWORD FAILED.

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user attempted to change the password, but entered an invalid value for NEW PASSWORD.

System Action: The current PASSWORD and the NEW PASSWORD fields are blanked out and the LOGON panel is redisplayed.

User Response: Retry using the current PASSWORD and a valid value for the NEW PASSWORD.

$AVR831E - $AVRS $AVRUX# NOT 31=BIT, USER EXIT NOT INVOKED

Explanation: $AVRS determined that the User Exit mentioned in the message was not linked as AMODE=31 as required by $AVRS. $AVRS requires 31-Bit address mode when there is addressability to keys or modules that reside above the 16-Meg line.

System Action: The call to the User Exit was ignored. The requested function is NOT performed.

User Response: Re-Link the User Exit with the correct AMODE=31 parameter. RMODE may be allowed to default, be set to RMODE=24 or RMODE =ANY.

$AVR839C - PSW=*psw* TRN=*vaddr* ICL=*X'#"* ICD=*X'#"* SERR=*f1f2f3f4*

Explanation: This message is part of the $AVRS abend diagnostic display.

psw (hexadecimal, 8 bytes) The Program Status Word at time of abend.

vaddr (hexadecimal, 4 bytes) The virtual address causing the translation exception.

X"*#*" (hexadecimal, 1 byte) The instruction length code for the psw. Values can be: 01,02,03.

X"*#*" (hexadecimal, 1 byte) The interrupt code if program check. Values can be one of the following program exceptions:

01 - Operation	02 - Privileged Operation
03 - Execute	04 - Protection
05 - Addressing	06 - Specification
07 - Data	08 - Fixed-point Overflow
09 - Fixed-point Divide	0A - Decimal Overflow
0B - Decimal Divide	0C - Exponent Overflow
0D - Exponent Underflow	0E - Significance
0F - Floating-point Divide	10 - Segment Translation
11 - Page Translation	12 - Translation Specification

f1f2f3f4 (hexadecimal, 4 bytes) The SDWA input flags describing the reasons and conditions for entering a recovery exit routine.

f1 : Error Type Flag
- X'80' - Machine Check
- X'40' - Program Check
- X'20' - Console Restart Key Depressed
- X'10' - Task Issued SVC 13 (Abterm)
- X'08' - System Forced SVC 13 (Abterm)
- X'04' - SVC Issued by a Locked or SRB Routine
- X'02' - Unrecoverable Translation Failure
- X'01' - Page I/O Error;

RTM1 Service Routine Scheduled RTM1 to Continue Processing

f2 : Additional Error Entry Information Flag
- X'80' - Task was Parallel Detached;

X'08' Type 1 SVC in Control at Time of Error
X'04' - Enabled RB in Control at Time of Error
X'02' - Logically or Physically Disabled Routine in Control at Time of Error
X'01' - System in SRB Mode at Time of Error

f3 : Additional Error Entry Information Flag
X'80' - Previous ESTA or FRR Exit Failed
X'40' - ESTAI Exit Previously Received Control
X'20' - IRB Preceded the RB Associated With Exit
X'10' - Recovery Routine Being Percolated To
X'08' - Lower Level Exit Has Recognized an Error and Has Provided Serviceability Information
X'04' - FRRS were Skipped
X'02' - Entry is a Local Resource Manager
X'01' - Entry is a Global Resource Manager

f4 : Additional Error Entry Information Flag
X'80' - Recovery Exit Only to Cleanup and Not Retry
X'40' - RB Associated with ESTA Exit Not in Control at Time of Error
X'20' - ESTA Exit Has Been Entered for a Previous Abend
X'10' - Task Not in Control at Time of Error
X'08' - Task Not in Control at Time of Error But Task Ancestor has Abended
X'04' - Registers, PSW, Control Registers at Time of Error are Unavailable
X'02' - Machine Check Error Information Unavailable
X'01' - Errorid Information Available

System Action: Task is terminated.

User Response: Collect the register displays for error correction.

$AVR840C - CSECT=*csect* OFFSET=*offset* ADDR=*addr1* PERCOLATE=*aaaa*

Explanation: This message is part of the $AVRS abend diagnostic display.

csect (alphanumeric, 8 bytes) The Program CSECT in control at time of abend.

offset (hexadecimal, 4 bytes) The offset of the instruction within the Program CSECT at time of abend.

addr1 (hexadecimal, 4 bytes) The starting address of the Program CSECT.

aaaa (alphameric, 3 bytes) The indication whether the $AVRS Estae recovery routine was percolated to (another Estae recovery routine handled the initial abend). Values can be YES or NO.

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR841C - LOAD/MODULE=*module* OFFSET=*offset* ADDR=*addr1*

Explanation: This message is part of the $AVRS abend diagnostic display.

module (alphanumeric, 8 bytes) The Program Load Module name containing the Program CSECT at time of abend.

offset (hexadecimal, 4 bytes) The offset of the instruction within the Load Module at time of abend.

addr1 (hexadecimal, 4 bytes) The starting address of the Program Load Module.

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR842C - INSTR ADDR=*addr1* STORAGE=*X'##' X'##' X'##' X'##'*

Explanation: This message is part of the $AVRS abend diagnostic display.

addr1 (hexadecimal, 4 bytes) The address of the instruction causing the abend.

X'##' (hexadecimal, 4 bytes) The memory display of the instruction causing the abend. Messages $AVR839C through $AVR842C and $AVR901C are issued by the $AVRS Estae exit when a $AVRS sub-task has abended. They list information which can be used by the SEA support staff in diagnosing the problem causing the abend.

System Action: Task is terminated.

User Response: Collect the register displays for error correction.

$AVR843C - ACCESS DENIED BY THE SECURITY MODULE; USER PROFILE NOT DEFINED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user is attempting to **LOGON** with a user id that is not defined to the external security system, i.e: RACF, etc.

System Action: Task is terminated.

User Response: If the user id used was not correct retry with the correct user id.
If the user id used was correct, have your security administrator define your user id to the external security system with the proper access levels before attempting to **LOGON** to the $AVRS system again.

$AVR844C - ACCESS DENIED BY THE SECURITY MODULE; PASSWORD NOT AUTHORIZED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user entered an invalid password.

System Action: Task is terminated.

User Response: Retry using a valid password for this user.

$AVR845C - ACCESS DENIED BY THE SECURITY MODULE; PASSWORD HAS EXPIRED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user's password has expired.

System Action: Task is terminated.

User Response: Change the user's password using the NEW PASSWORD field and retry.

$AVR846C - ACCESS DENIED BY THE SECURITY MODULE; NEW PASSWORD INVALID

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The NEW PASSWORD field contained an invalid password.

System Action: Task is terminated.

User Response: Check with your security administrator to determine what constitutes a valid password and retry.

$AVR847C - ACCESS DENIED BY THE SECURITY MODULE; USER NOT DEFINED TO GROUP

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

$AVRS access is defined at the group level and the user is not a member of that group.

System Action: Task is terminated.

User Response: Have the security administrator join the user to the proper group before retrying.

$AVR848C - ACCESS DENIED BY THE SECURITY MODULE; FAILED BY INSTALLATION EXIT

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The requested access has been denied by an installation security exit module not the external security system.

System Action: Task is terminated.

User Response: Have the $AVRS security administrator set up the proper access to allow the user to access the required $AVRS resource.

$AVR849C - ACCESS DENIED BY THE SECURITY MODULE; USER ACCESS HAS BEEN REVOKED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user's access to the requested $AVRS resource has been revoked.

System Action: Task is terminated.

User Response: Have the security administrator re-instate the user's access to the requested $AVRS resource before retrying.

$AVR850C - ACCESS DENIED BY THE SECURITY MODULE; RACF IS NOT ACTIVE

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The external security system is not active on the system at this time, consequently it cannot authorize access to the $AVRS resources.

System Action: Task is terminated.

User Response: Activate the external security system before retrying.

$AVR851C - ACCESS DENIED BY THE SECURITY MODULE; ACCESS TO GROUP REVOKED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

$AVRS access is at group level. The user's access to the group is revoked.

System Action: Task is terminated.

User Response: Have the security administrator re-establish the user's access to the proper group before retrying.

$AVR852C - ACCESS DENIED BY THE SECURITY MODULE; OIDCARD PARAMETER NOT SUPPLIED

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

An OIDCARD parameter was required but was not supplied in the call to the external security system. This is entirely within the scope of the modifications made to the installation's version of the $AVRS security exit by a customization applied by the installation's $AVRS system security programmer.

System Action: Task is terminated.

User Response: Refer to your $AVRS system security programmer and/or your security administrator to resolve this.

$AVR853C - ACCESS DENIED BY THE SECURITY MODULE; OIDCARD PARAMETER INVALID

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The OIDCARD parameter that was supplied in the call to the external security system was invalid. This is entirely within the scope of the modifications made to the installation's version of the $AVRS security exit by a customization applied by the installation's $AVRS system security programmer.

System Action: Task is terminated.

User Response: Refer to your $AVRS system security programmer and/or your security administrator to resolve this.

$AVR854C - ACCESS DENIED BY THE SECURITY MODULE; NOT AUTHORIZED FOR TERMINAL

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user is not authorized to access the $AVRS system resources from the terminal used.

System Action: Task is terminated.

User Response: Have the user sign on from the correct terminal and retry.

$AVR855C - ACCESS DENIED BY THE SECURITY MODULE; NOT AUTHORIZED FOR APPLICATION

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The user is not authorized to access this application.

System Action: Task is terminated.

User Response: Have the security administrator set up access to the application by this user before retrying.

$AVR856C - ACCESS DENIED BY THE SECURITY MODULE; UNIDENTIFIED RACF RETURN CODE

Explanation: $AVRS messages 829, 830, and 843 - 856 are due to the $AVRS security exit module, $AVRSECU, passing a return code back to the $AVRLOGN program indicating a security error:

The Code passed back to $AVRLOGN is an unidentified RACF return code.

System Action: Task is terminated.

User Response: Call SEA Product Support.

$AVR857W - VTAM PRINT DDNAME NOT FOUND. PRINT REQUEST IGNORED

Explanation: A VTAM Print request was detected by $AVRPULL but the VTAM Print DD statement was missing or invalid.

System Action: Processing continues by skipping this print request.

User Response: Verify a $AVRVPRT DD statement in coded in the $AVRPULL JCL and try the operation again.

$AVR859W - PRTQIO ERROR - OFFSET = X'####', FDBK = X'####', RC=X'####'

Explanation: A VTAM Print request from $AVRS has failed. This message contains the location in the program, the feedback and return code of the failing operation.

System Action: The Print request is terminated.

User Response: Contact $AVRS technical support with the information in this message for problem resolution.

$AVR860A - COMMAND OR SELECTION ENTERED IS INVALID

Explanation: The command entered is either unknown or invalid from the panel from which it was entered.

System Action: Panel is re-displayed.

User Response: Enter a valid command or selection.

$AVR869I - REQUEST HAS BEEN QUEUED TO VTAM PRINT

Explanation: A $AVRS print request has been successfully sent to VTAM Print.

System Action: Processing continues.

User Response: Informational.

$AVR870W - SOME OR ALL FILES FOR OUTPUT DENIED BY SECURITY

Explanation: An output to dataset request has been submitted for a job, but the user-defined security exit has denied output authority for one or more files.

System Action: The unsecured files are outputted.

User Response: Obtain security clearance.

$AVR871I - DELETION DENIED BY SECURITY PROCESSOR

Explanation: A delete request for work on the data base has been denied by the installation defined security exit.

System Action: Delete will not take place.

User Response: Obtain security level required.

$AVR872W - SECURITY PROCESSOR PARAMETER ERROR

Explanation: An invalid parameter has been passed to the security exit.

System Action: Security checking will not be done for the requested function.

User Response: Contact vendor support.

$AVR873I - SOME OR ALL FILES FOR PRINT DENIED BY SECURITY

Explanation: A print request has been submitted for a job, but the installation defined security exit has denied print authority for one or more files.

System Action: Print request will not take place.

User Response: Obtain security level required.

$AVR875I - SELECTION DENIED BY YOUR INSTALLATION'S SECURITY

Explanation: A browse or file selection request from a user terminal has been denied by the installation defined security exit.

System Action: The selection is not displayed.

User Response: Obtain security level required.

$AVR876I - SOME SELECTIONS DENIED BY YOUR INSTALLATION'S SECURITY

Explanation: A user has entered a request for a directory of work from the $AVRS Primary Control Panel. A directory of work has been produced, but some of the work requested is not listed due to denial by the security exit.

System Action: The work available for display is displayed.

User Response: Obtain security level required.

$AVR877I - RESTORE DENIED BY SECURITY PROCESSOR

Explanation: A request to restore archived work has been entered from a user terminal, but the installation-defined security exit has denied restore authority for the work.

System Action: The requested work to be restored will NOT take place.

User Response: Obtain security level required.

$AVR878I - DENIED BY INSTALLATION'S SECURITY

Explanation: A request from a user has been denied by the security processor.

System Action: This message is issued for each file denied. The request is denied, but processing continues.

User Response: Obtain security level required.

$AVR879A - "ENTER" KEY TO UPDATE,"CAN" TO ABORT, "END" KEY TO CONTINUE

Explanation: Issued from the Annotation/Acknowledgment Panel when the user has updated information and pressed the END Key to request update of changes.

System Action: Waits for response.

User Response: Press Enter if update is desired. Press the END PF Key to return to the Annotation/Acknowledgment Panel for further input. Enter "CAN" from the command line to Cancel Update.

$AVR901C - REGS (*nn-nn*) X'##' X'##' X'##' X'##'

Explanation: This message is a part of the $AVRS abend diagnostic display.
nn-nn (decimal, 4 bytes) The register range for the following register content display. Values can be: 00-03, 04-07, 08-11, 12-15.
X'##' (hexidecimal, 4 bytes) The register contents of four registers at time of abend.

System Action: The task is terminated.

User Response: Collect the register displays for error correction.

$AVR902C - PHYSICAL ERROR WHILE TRYING TO LOAD PASSWORD MODULE

Explanation: A physical I/O error has occurred loading the user's VSAM password module.

System Action: The system terminates.

User Response: Correct the problem with the LOADLIB where the module $AVR resides.

$AVR902E - UNABLE TO ALLOCATE DSNPDSP=dataset name

Explanation: A Demand Print was attempted and $AVRS was unable to allocate the PDS where the print control member(s) are located. The **DSNPDSP=** parm is defined in the **$AVRSFSEA**.

System Action: $AVRS returns to the Demand Print Panel.

User Response: Contact your $AVRS Administrator requesting that the PDS pointed to by the DNSPDSP parameter be validated for spelling and insure that it is cataloged.

$AVR903C - DATA BASE NOT FORMATTED OR UNUSABLE

Explanation: This message is normally one of two problems:

1. The $AVRS' format program has not formatted the data base.
2. An error exists that prevents this processing to complete.

System Action: $AVRS terminates.

User Response: Consult the systems programmer responsible for $AVRS to ensure that the requested function should be operative at your installation.

$AVR904I - INDEX FILE EMPTY OR UNUSABLE AT THIS TIME

Explanation: The VSAM data base files are empty or currently being used exclusively by another user.

System Action: Allocation was impossible.

User Response: Retry at a later time. The $AVRFMRR program is probably executing.

$AVR905A - INVALID VALUE, MUST BE "Y" OR "N" OR NUMERIC 1 THROUGH 9

Explanation: The value entered at the field indicated must be one of the above.

System Action: The panel is re-displayed and the cursor is placed under the field.

User Response: Correct the entry in error.

$AVR906I - SORT NOT VALID ON HISTORY DIRECTORIES

Explanation: A sort command has been entered from the command line of a Fiche History Selection Panel. The directory cannot be sorted.

System Action: Informational.

User Response: None required.

$AVR907I - SORT NOT VALID ON STATUS DIRECTORIES

Explanation: A sort command has been entered from the command line of an Online Status or Fiche Index Selection Panel. These directories cannot be sorted.

System Action: Informational.

User Response: None required.

$AVR908A - STEP INFORMATION NOT AVAILABLE

Explanation: An "I" selection character has been entered for an entry on the $AVRS data base, but one or more of the system datasets required to extract the information is not present in the output for the work.

System Action: The selection directory is re-displayed.

User Response: The selected entry does not contain the JES datasets. Select another entry.

$AVR909I - FUNCTION NOT VALID FOR THIS TYPE

Explanation: A selection character entered from a user's terminal indicates a selection that is not valid for the type of work on the current directory.

System Action: Informational.

User Response: None required.

$AVR910I - *job name, job number, step name, proc step name, ddname,* JLCNT *#####,* FLCNT *######*

Explanation: A LO= type of scan in $AVRPULL or $AVRTAPE successfully found it's search argument. This informational message shows the name of the work, its number, the *ddname* of the file, the line count from the beginning of the job (JLCNT) and the line count from the beginning of the file (FLCNT). The *stepname* and proc *stepname* are also shown if present. The actual line containing the search argument is shown following this line.

System Action: Informational.

User Response: None required.

$AVR911A - START DATE REQUIRED FOR "AH" SELECTIONS

Explanation: Due to installation specifications, (AHDATE=YES in the $AVRFSEA) you must enter a starting date for your Archive History selections.

System Action: The Primary Control Panel is re-displayed.

User Response: Enter a starting date for the selection and try the operation again.

$AVR912C - ERROR READING PANEL *panel* - PANEL NOT FOUND

Explanation: $AVRS attempted to read PANEL *panel* and a "not found" error was returned.

System Action: Task returns to calling function.

User Response: A common cause for this type of error is using the $AVRS ISPPLIB dataset from a prior release. Ensure that the ISPPLIB DDNAME of your logon procedure is using the current $AVRS ISPPLIB dataset.

$AVR913C - I/O ERROR READING PANEL *panel*

Explanation: An I/O error has occurred reading *PANEL panel* .

System Action: Task returns to calling function.

User Response: Ensure that the ISPPLIB DDNAME of your logon procedure is using the current $AVRS ISPPLIB dataset. Try to browse panel xxxxxxxx using non-$AVRS procedures (i.e. ISPF). Check file record size and block size and dsorg. Correct any discrepancies.

$AVR915C - OPEN FAILURE FOR DDNAME *ddname*

Explanation: The open for the DDNAME *ddname* failed.

System Action: Task is terminated.

User Response: Ensure that the DDNAME ddname is in and accessible to your logon procedure. ROSCOE users should have an "allocate" statement in the RPF.

$AVR916W - I/O ERROR READING MEMBER, RC=x###

Explanation: A read error with return code *x###* has occurred while reading the $AVRS Print Control PDS dataset. The member is the one you specified on the $AVRS Print Panel.

System Action: The function is terminated.

User Response: Determine the cause of the error and correct it. Then retry the operation.

$AVR917I - DSNPDSP IN $AVR*xx*EA IS NOT A PDS

Explanation: The dataset in the Environment area contains a dataset name for the Print Control dataset that is not a PDS dataset.

System Action: The function is terminated.

User Response: Reassemble the $AVRFSEA if the DSNPDSP=name was incorrect, sign onto $AVRS again, and retry the function.

$AVR918I - OUTPUT NAME ALLOCATION FAILED/RC=X'###1,REASON=x###2'KEY=X'###3'

Explanation: An allocation attempt to the MVS OUTPUT statement specified by the OUTPUT NAME field on the Print Panel or a Dynamic System Output (SYSOUT) dataset has failed - OR - an invalid PAGEDEF or FORMDEF was specified.

System Action: Requested function is terminated.

User Response: The name specified on the Print Panels OUTPUT NAME field must be the name of an EXISTING MVS OUTPUT statement in your logon proc. Ensure the name is correctly spelled and exists in your current logon procedure.

If a PAGEDEF or FORMDEF name was specified, ensure the name is correctly spelled and exists in the AFP overlay library.

If you did not specify an OUTPUT NAME, locate the reason in the OS/390 MVS Authorized Assembler Services Guide in the OUTADD Section - Create an Output Descriptor for resolution to the problem. Ensure that a screen print or copy of the **exact text** is available.

RC=OUTADD two-byte hexadecimal return code.
REASON=OUTADD two-byte hexadecimal reason code.
KEY=OUTADD Key: A two-byte hexadecimal number that identifies the function that the OUTADD macro performed.

$AVR919I - MAIN KEY CURRENTLY HAS KYWORKIN FLAG ON

Explanation: Processing is currently active by the accumulator for this work.

System Action: Accumulating or deleting.

User Response: Reenter your command at a later time.

$AVR920I - REQUESTED ARGUMENT(S) NOT ON FILE

Explanation: The desired selection criteria was not found on the online data base.

System Action: Primary Control Panel is re-displayed.

User Response: Enter alternate selection criteria.

$AVR921I - AT LEAST ONE OR ALL SELECTIONS DENIED BY SECURITY EXIT

Explanation: The user defined security exit has denied one or more selections from being accessed. This message may have been issued by $AVRUX2.

System Action: Display of all selections which were allowed.

User Response: Obtain security level required.

$AVR922I - RECORD NOT FOUND OR EOF REACHED BEFORE REQUEST COMPLETED

Explanation: An END OF FILE has been detected on the VSAM data base prior to completing the request.

System Action: Panel is re-displayed.

User Response: Retry the request.

$AVR923C - INSUFFICIENT REGION SIZE, USE A LARGER REGION AND RETRY

Explanation: The selection criteria has generated a selection too large to handle in the current region.

System Action: Panel is re-displayed.

User Response: Either limit the selection argument with a date or LOGON with a larger region.

$AVR924A - ENTER REQUIRED VALUE FOR THIS FIELD, MUST BE "Y" OR "N"

Explanation: Either the required value was not supplied or the value was not "Y" or "N".

System Action: Message is displayed and the cursor is placed in the field in error.

User Response: Enter one of the required values to proceed.

$AVR925C - DIAG INFO##: *X'##' X'##' X'##' X'##'*
$AVR925C - DIAG INFO3: *f1f2f3f4 f5f6f7f8 X'##' X'##'*

Explanation: This message is part of the $AVRS abend diagnostic display.

## (decimal, 1 byte)	The diagnostic information indicator for the data that follows. Values can be: 1, 2 or 3.
X'##' (hexadecimal, 4 bytes)	For INFO1 and INFO2, the diagnostic information associated with the abending program.

For INFO3, the first eight bytes are as follows:

f1f2f3f4f5f6f7f8 (hexadecimal, 8 bytes)	The ISP$AVRS initialization and termination flags.

f1 : ISP$AVRS, $AVRINCP, $AVROEOJ Status Flags
- X'80' - ISP$AVRS Init Active
- X'40' - ISP$AVRS Init Completed
- X'20' - $AVRBRO1 Process Active
- X'10' - ISP$AVRS Termination Active
- X'08' - $AVRINCP Init Active
- X'04' - $AVRINCP Init Completed
- X'02' - ISP$AVRS CP Mode
- X'01' - ISP$AVRS Init/Term Error

f2 : ISP$AVRS UPFL Flags
- X'80' - UPFLINT Completed
- X'40' - UPFLUSR Completed
- X'20' - UPFLEXT/UPFLOFF Completed
- X'10' - UPFL ISPF Display Error

X'08' - UPFLINT Error
X'04' - UPFLUSR Error
X'02' - UPFL ISPF Define Error
X'01' - UPFL ISPF VGET/VPUT/VRESET Error

f3 : ISP$AVRS Error Reason Code
Values can be 00-FF.
Refer to source code for explanation.

f4 : ISP$AVRS Mainline Status
Indicates current checkpoint.
Values can be 00-FF.
Refer to source code for explanation.

f5 : $AVRINCP Mainline Status
Indicates current checkpoint.
Values can be 00-FF.
Refer to source code for explanation.

f6 : $AVROEOJ Mainline Status
Indicates current checkpoint.
Values can be 00-FF.
Refer to source code for explanation.

f7 : ISP$AVRS Mainline Error Status
Indicates current error checkpoint.
Values can be 00-FF.
Refer to source code for explanation.

f8 : $AVRINCP Mainline Error Status
Indicates current error checkpoint.
Values can be 00-FF.
Refer to source code for explanation.

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR926C - (5.1B-## aaaa f1f2f3f4-f5f6f7f8) TCB=addr1 RB=addr2

Explanation: This message is part of the $AVRS abend diagnostic display. This message may be repeated depending upon the number of RB addresses at time of error.

5.1B-## (decimal, 2 bytes) Indicates the current $AVRS release (5.1B) and the current maintenance level (##). Values can be from 00-99.

aaaa (alphanumeric, 8 bytes) The Initial Program Load Module name.

f1f2f3f4 (hexadecimal, 4 bytes) The module Estae status flags.

f1 : MEWA Control Block Version Level. Value: 01

f2 : MEWA Status Flags
X'80' - SDWA Passed
X'40' - Estaex Error: Estae Redriven
X'10' - Esatex in Control
X'08' - Percolated To
X'04' - Forced Cancel

X'02' - Terminate $AVRS
X'01' - Buildcell DSA Unavailable

f3 : MEWA Status Flags
X'80' - Do Not Activate Esatex
X'40' - Do Not Activate Estaex in $AVRSBF21
X'20' - No Clean Up
X'10' - Take SVC Dump
X'08' - Display No Messages

f4 : MEWA Status Flags
X'80' - Current File Open
X'08' - Estaex Activated
X'04' - Estaex Activated in $AVRSBF21

f5f6f7f8 (hexadecimal, 4 bytes) The module Estae status flags.

f5 : MEWA Status Flags
X'80' - Abend Detected: S0C4
X'40' - PSW2 Matches TRAN Address
X'20' - REG15 Matches PSW2 & TRAN Address
X'08' - Abend Detected: S0C1

f6 : MEWA Status Flags
X'80' - Percolated RB Address Valid
X'40' - Percolated RB Reg-14 Valid
X'20' - Percolated RB Csect Valid
X'08' - RB Valid
X'04' - SDWANAME Valid
X'02' - RB PSW Valid

f7 : MEWA Status Flags
X'80' - Online Display Facilities Available
X'40' - Online Display Buffer Available

f8 : MEWA Status Flags
Currently unused

addr1 (hexadecimal, 4 bytes) The TCB address of the abending task.

addr2 (hexadecimal, 4 bytes) The RB address of the abending task. (May be multiples)

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR927C - GBLS=*addr1 addr2* $AM=*addr3* $CB=*addr4*

Explanation: This message is part of the $AVRS abend diagnostic display.

addr1	(hexadecimal, 4 bytes)	The address of the global control block ($AVRFSGB for online and $AVRGBLS-part 1 for batch)
addr2	(hexadecimal, 4 bytes)	The address of the global control block ($AVRFSG2 for online and $AVRGBLS-part 2 for batch)
addr3	(hexadecimal, 4 bytes)	The address of the Access Method Work control block ($AMWORK)
addr4	(hexadecimal, 4 bytes)	T he address of the $AVRS control block ($AVRSCB)

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR928I - JOBNAME LESS THAN INSTALLATION REQUIRED LENGTH

Explanation: The name entered for requested sysout is too generic according to the installation's standards. The phrase "too generic" means not enough non-blank, non-masking consecutive characters from the beginning of the name field.

System Action: Selection Panel is re-displayed.

User Response: Re-enter the name of the work in a more specific way by increasing the number of non-blank, non-masking characters starting from the beginning of the name.

$AVR929I - WORK IS ACTIVE BY $AVRS OR HAS BEEN DELETED

Explanation: The selected work is being updated, another user may have deleted the work, or normal purge processing may have deleted the work.

System Action: Work is not selected.

User Response: Re-enter the primary selection to determine if the updates are now complete for the work.

$AVR930I - PREMATURE END OF FILE REACHED ON SELECTED FILE

Explanation: The selected work was deleted by another user or by normal purge processing and the space that the original work occupied has been reused by the accumulation of new work or the $AVRCLEN program was executed while $AVRSELR was running or before $AVRSELR maintenance keys were accumulated (see messages $AVR591C and $AVR592C), or without the proper ENQ safeguards.

This condition exists either as a safeguard to prevent unauthorized access of work or an incomplete selective restore may have occurred.

System Action: Work is not displayed.

User Response: Refresh the directory and determine if the work is still in the data base. If the problem persists, call vendor for support.

$AVR932A - MODTRC MUST BE BLANK, 0, 1, 2 or 3

Explanation: A value other than one of the above was supplied for MODTRC.

System Action: Cursor is placed in the field and the message is displayed.

User Response: Supply an acceptable value and press enter.

$AVR933A - *field* IS INVALID OR A REQUIRED PRINT SPECIFICATION

Explanation: A value for '***field***' is invalid or required.

System Action: Cursor is placed in the field and the message is displayed.

User Response: Supply an acceptable value and press enter.

$AVR934I - STARTING PRINT VALUE OUT OF RANGE

Explanation: The relative starting page or line number is beyond the end of file for the file that was requested.

System Action: Cursor is placed in the field and the message is displayed.

User Response: Supply an acceptable value and press enter.

$AVR937I - NO RESTORES/PRINT QUEUED FOR SUBMISSION

Explanation: A "SUBMIT RESTORE" or "SUBMIT PRINT" request has been entered from a user terminal, but no Archive History Restore requests are pending.

System Action: Restore or Print job is not submitted.

User Response: If restore was desired, re-enter the selections.

$AVR940A - PROCNAME MUST BE SPECIFIED

Explanation: Issued from the Restore or Print Job Submit Panel. The field indicated by $AVRS Procedure cannot be left blank.

System Action: The cursor is placed in the field in error.

User Response: Enter a procedure name.

$AVR941I - SUBMIT AUTHORITY REQUIRED FOR RESTORE OR PRINT PROCESSING

Explanation: A user has attempted to restore or print Archive History work, but does not have submit authority required to submit the Restore or print job.

System Action: The submit is not done.

User Response: Obtain UPDATE authority to submit work.

$AVR942I - ALLOCATION FOR INTRDR FAILED - SUBMIT DENIED

Explanation: $AVRS was unable to allocate the internal reader necessary to submit a user request.

System Action: The request is denied.

User Response: Ensure that enough INTERNAL READERS are available on the system for $AVRS to allocate one.

$AVR943C - SUBMIT *aaa* OPEN FAILURE

Explanation: An open failure has occurred during job submission processing.

System Action: The submit is terminated.

User Response: Determine and correct the problem.

$AVR944I - JOB NAME SUBMITTED

or

- JOB NAME SUBMITTED AS ALTERED

Explanation: Job submit processing for the indicated request has completed successfully. "As Altered" implies the JCL was altered by $AVRUXA or by another User Exit.

System Action: Complete.

User Response: None required.

$AVR945A - INVALID JOB RANGE SPECIFIED

Explanation: The user has entered values in the "NAME" and "THRU" fields on the Primary Control Panel, but the "THRU" value does not exceed the "NAME" value as required.

System Action: The cursor is placed in the field.

User Response: Re-enter a correct range.

$AVR946C - DSNPDSP DATASET NOT DEFINED IN $AVR*xx*EA

Explanation: The user has an member name on the $AVRS Print Panel, but no Print Control dataset has been specified through this Environment Area.

System Action: The function is terminated.

User Response: Enter into $AVRS with different environment area that has a Print Control dataset. You may also specify a Print Control dataset (DSNPDSP=) in this environment area ($AVR??EA), then reassemble and link it. Logoff of $AVRS and try again.

$AVR947C - DSNPDSP DATASET OPEN ERROR, RC=X'##'

Explanation: The current $AVRS environment area specifies a Print Control dataset. $AVRS has allocated to it, but has received an open error with reason code X'##'.

System Action: The function is terminated.

User Response: Determine the cause of the open error. Reassemble the $AVRFSEA if the name was incorrect, sign onto $AVRS again, and retry the function. Correct the error and retry the operation.

$AVR948I - MEMBER(PRE) (or MEMBER(POST)) MEMBER (pds member) NOT FOUND

Explanation: The user has entered a PDS member name on a $AVRS Demand Print Panel that is not in the print control PDS identified by the DSNPDSP=parameter.

System Action: The function is terminated and $AVRS returns to the Demand Print Panel.

User Response: Re-enter a valid member name that currently exists in the PDS dataset or add the member name into the Print Control PDS dataset.

$AVR949W - DSDSNP ALLOCATE ERROR RC=X'##', EC=X'####', IC=X'####'

Explanation: $AVRS has attempted to allocate to the Print Control datasets specified in this environment area and received a dynamic allocation error. The explanation of the error codes listed in this message is the same as for message **$AVR030C**.

System Action: $AVRS continues, but no Print Control processing is allowed.

User Response: Check that the name for the Print Control dataset is correctly specified in the environment area module. This dataset must both exist and be cataloged. Reassemble the $AVRFSEA if the name was incorrect, sign onto $AVRS again, and retry the function.

$AVR950A - SYSOUT CLASS SPECIFICATION IS INVALID

Explanation: During a PRINT RESET request, the SYSOUT CLASS was found not to be A-Z, 0-9.

System Action: Cursor is placed in the field.

User Response: Enter a valid response.

$AVR951A - FCB SPECIFICATION IS INVALID

Explanation: During a PRINT RESET request, the SYSOUT FCB was found to contain invalid characters. A period may be invalid for dynamic allocation.

System Action: Cursor is placed in the field.

User Response: Enter a valid response.

$AVR952A - FORMS SPECIFICATION IS INVALID

Explanation: During a PRINT RESET request, the SYSOUT FORM was found to contain invalid characters or has a length >4. A period may be invalid for dynamic allocation.

System Action: Cursor is placed in the field.

User Response: Enter a valid response.

$AVR953A - REMOTE DESTINATION IS INVALID

Explanation: During a PRINT RESET request, the SYSOUT DESTINATION was found to contain invalid characters.

System Action: Cursor is placed in the field.

User Response: Ensure the validity of all SYSOUT DESTINATION characters and resubmit.

$AVR954I - OPEN FAILURE ON PRINT FILE DDNAME

Explanation: During a PRINT RESET request, the SYSOUT file failed to open correctly.

System Action: The SYSOUT dataset is not allocated and print requests cannot be honored.

User Response: Determine the reason for failure and reissue the PRINT RESET request.

$AVR955I - PRINT RESET REQUIRED BEFORE PRINTING

Explanation: During a PRINT request, the SYSOUT file was found to be closed.

System Action: Print requests cannot be honored.

User Response: Determine the reason for the failure and issue the PRINT RESET request before printing.

$AVR956A - field MUST BE SPECIFIED
$AVR956I - field MUST BE SPECIFIED

Explanation: Issued from the Restore data set not catalogued panel. The *field* indicated cannot be left blank.

System Action: The cursor is placed in the field in error.

User Response: Enter the requested field.

$AVR957A - INVALID SPECIFICATION — "-", "*" and "." NOT ALLOWED

Explanation: Issued when a user attempts to utilize one of these character in defining an acknowledgment code under $AVRS Annotation/Acknowledgment feature. These characters are not allowed, as they are $AVRS "mask" selection characters.

System Action: Waits for response.

User Response: Respecify ACK code.

$AVR958W - *jobname, job number,* DSN = *dataset name (1st 30 characters)* - NOT CATLGED

Explanation: A restore request for an Archive History entry has failed because the Archive dataset indicated in the History entry is not catalogued.

System Action: The selected restore request was not on cataloged files. The restore request will be honored if the unit and volser are specified by the user on the $AVRS Archive History Not Catalogued dataset panel.

User Response: Find another dataset where the work exists catalogued and submit a batch restore. Create an online or batch restore job by specifying the VOLSER number(s) for the uncatalogued archive/backup files.

$AVR960A - SELECTION "TYPE" IS INVALID

Explanation: The command entered is either unknown or invalid from the panel from which it was entered.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR961A - INVALID SYSLOG ID WAS SPECIFIED

Explanation: The SYSLOG ID value specified is not a defined as a valid ID to $AVRS.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR962A - INVALID START TIME WAS SPECIFIED

Explanation: The START TIME value specified is invalid. The format is HHMM of the START TIME desired.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR963A - INVALID END TIME WAS SPECIFIED

Explanation: The END TIME value specified is invalid. The format is HHMM of the END TIME desired.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR964A - INVALID START DATE WAS SPECIFIED

Explanation: The START DATE value specified is invalid. The format is YYDDD or YYYYDDD for the START DATE desired.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR965A - INVALID END DATE WAS SPECIFIED

Explanation: The END DATE value specified is invalid. The format is YYDDD or YYYYDDD for the END DATE desired.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR966A - NO NAME VALUE WAS SPECIFIED

Explanation: The NAME value specified is an invalid format. The name may be from 1 to 8 characters in length.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR967A - INVALID NUMBER WAS SPECIFIED

Explanation: The NUMBER value specified is invalid format. The number may be from 1 to 5 digits.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR968C - END DATE MUST BE GREATER THAN OR EQUAL TO START DATE

Explanation: A request was made to extract work from the online data base based on inconsistent start and end date criteria. The request cannot be satisfied.

System Action: Request is ignored.

User Response: Specify consistent start and end dates.

$AVR969A - UCS SPECIFICATION IS INVALID

Explanation: During a PRINT RESET request, the UCS field was found to contain invalid characters. A period may be invalid for dynamic allocation.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR970C - $AVRS RELEASE *rel#* REQUIRES OS/390

Explanation: $AVRS Release R5.1B requires OS/390 to function. The test for the release of the operating system has indicated that this release does not meet this minimum.

System Action: The system stops with Abend U0970.

User Response: Contact Tech Support at **SEA**

$AVR971A - INVALID SORT PARAMETER SPECIFIED

Explanation: The value entered in the SORT FIELD input field must be the name of a directory field or a shortened form of a field name. The value entered did not match any known directory field or was ambiguous.

System Action: The cursor is placed in the field.

User Response: Enter a correct response.

$AVR972C - TASK ABEND - CODE=*code* REASON=*reason* OS=*system/fmid*

Explanation: This message is part of the $AVRS abend diagnostic display.

Code (alphameric, 1 byte hexadecimal, 2 bytes)	The abend type, either S (system) or U (user) and code
reason (hexadecimal, 4 bytes)	The abend reason code
system (alphanumeric, 8 bytes)	The current operating system designation.
fmid (alphanumeric, 8 bytes)	The current operating system FMID.

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR973I - DELETION OF ARCHIVE STATUS REPORTS IS PROHIBITED BY $AVRS

Explanation: The user does not have the authority to delete STATUS HISTORY (ST) reports from $AVRS.

System Action: The status report is not deleted.

User Response: Allow deletion of status reports through $AVRUX4 and the overriding parameters in $AVRFSEA.

$AVR974C - ON *yyyy-mn-dd@ttt* ASID=*asid aaaa (job- id job number)*

Explanation: This message is part of the $AVRS abend diagnostic display.

yyyy (numeric, 4 bytes)	The year at time of error
mn (numeric, 2 bytes)	The month at time of error
dd (numeric, 2 bytes)	The day at time of error
ttt (numeric, 2 bytes)	The hour, minute and second at time of error
asid (hexadecimal, 2 bytes)	The Address Space Identifier
aaaa (alphanumeric, 8 bytes)	The TSO User, the Started Task or the Batch Job at time of error
job-id(alphameric, 3 bytes)	The identifier for the User (TSU), Task (STC) or Job (JOB) at time of error
job number (numeric, 5 bytes)	The associated JES2 job number

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR975I - DELETION DENIED BY INSTALLATION SECURITY EXIT

Explanation: The installation's defined user exit has denied the deleting of the desired work. This message may have been issued by $AVRUX4.

System Action: The deletion is not allowed.

User Response: Determine your installation's requirements for deleting work and change the exit to meet those requirements.

$AVR976C - *file type* FILE FOUND *file status*: DDN=*ddname* ACB=*addr1* FLGS=*f1f2f3f4*

Explanation: This message is part of the $AVRS abend diagnostic display.

file type (alphameric, 4 bytes)	The file type being reported. Values can be: LOCK or KSDS
file status (alphameric, 4 bytes)	The file status being reported. Values can be: OPEN or CLOSED
ddname (alphameric, 8 bytes)	The DDName of the file being reported
addr1 (hexadecimal, 4 bytes)	The ACB address of the file being reported
f1f2f3f4 (hexadecimal, 4 bytes)	The file status flags of the file being reported

f1 : $AVRS LOCK File Status Flags #1

X'80' - ACB is Open
X'40' - Open for Input
X'20' - Open for Output
X'10' - Access of Data
X'08' - Reading RRDS for Blocks

X'04' - Resetting Requested Lock
X'02' - Forcing Change to Lock
X'01' - Inquiring About Lock Status

f2 : $AVRS LOCK File Status Flags #2
X'80' - Lock for $AVRUNLD
X'40' - Lock for $AVRBKUP
X'20' - Lock for Accumulator
X'10' - Lock for $AVRFORM
X'08' - Reset for DB Id
X'04' - Maint Active/Prevent Online Access

f1 : $AVRS KSDS File Status Flags #1
X'80' - ACB is Open
X'40' - Open for Input
X'20' - Open for Output
X'04' - RPL 0 To Be Used
X'02' - RPL 1 To Be Used

f2 : $AVRS KSDS File Status Flags #2
X'02' - Add Index Buffers
X'01' - Add Data Buffers

f3 : $AVRS ACB File Open/Close Flags
X'80' - 31-bit Buffer Request
X'40' - 31-bit Control Block Request
X'20' - EOV Concatenation
X'10' - ACB is Open
X'08' - Further Requests Impossible
X'04' - Allow Open of Sphere
X'02' - User Exit Flag
X'01' - ACB being Processed by I/O Support Function

f4 : $AVRS ACB File Error Flags
X'04' - ACB Already Open/Not Open

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR977C - *file type* FILE: DDN=*ddname* ACTION=*file status* FLGS=*f1f2*

Explanation: This message is part of the $AVRS abend diagnostic display.

file type (alphameric, 4 bytes)	The file type being reported. Values can be: LOCK or KSDS
ddname (alphameric, 8 bytes)	The DDName of the file being reported
file status (alphameric, 6 bytes)	The file status after closure. Value is: CLOSED
f1f2f (hexadecimal, 2 bytes)	The file status flags of the file being closed

f1 : $AVRS ACB File Open/Close Flags

- X'80' - 31-bit Buffer Request
- X'40' - 31-bit Control Block Request
- X'20' - EOV Concatenation
- X'10' - ACB is Open
- X'08' - Further Requests Impossible
- X'04' - Allow Open of Sphere
- X'02' - User Exit Flag
- X'01' - ACB being Processed by I/O Support Function

f2 : $AVRS ACB File Error Flags

- X'04' - ACB Already Open/Not Open

System Action: The task is terminated.

User Response: Collect the diagnostic displays for error correction.

$AVR978I - PRINT REQUEST HAS BEEN DENIED BY USER EXIT

Explanation: The installation-defined user exit has denied the printing of the desired work.

System Action: Print request not honored.

User Response: Determine your installation's requirements for printing work and set the output allocation specifications to meet those requirements. This message may have been issued by $AVRUX5.

$AVR980I - CLOSE/FREE DSN= *dataset name*

Explanation: The output dataset has been closed and freed.

System Action: Informational.

User Response: None required.

$AVR981A - *field* SPECIFICATION IS REQUIRED OR INVALID

Explanation: An entry has been incorrectly specified on the DATASET ALLOCATION PANEL. The message indicates the field in error.

DSNAME	Dataset name error during allocation
DISP	Disposition can be only "NEW", "MOD" or "OLD"
NRM DISP	Normal DISP.must be "CATLG" or "KEEP"
SPC TYPE	Space units can be only "CYLS" or "TRKS"
PRI. SPC	Primary QUAN is required or non-numeric
SEC. SPC	Secondary must be numeric
DIR BLKS	Directory blocks must be numeric
RECFM	Must be V,VB,VBA,VBM, or F,FB,FBA,FBM
LRECL	LRECLrequired or non-numeric (RECFM=F, MAX LRECL=255 , RECFM=V, MAX LRECL=259)
BLKSIZE	BLKSIZE required or non-numeric

System Action: Cursor is placed in the field in error.

User Response: Enter a correct response.

$AVR982I - DSORG OF DATASET NAME REQUESTED IS NOT "PO" OR "PS"

Explanation: The DSORG for the dataset requested is not PO or PS.

System Action: Dataset is not allocated.

User Response: Use a different dataset.

$AVR983W - *code* ABEND OCCURRED DURING OUTPUT PROCESSING

Explanation: The ABEND CODE given occurred while processing the output request. An abend can also occur during close processing when there is insufficient space to write the buffers and the EOF. If the abend occurs while exiting from $AVRS no indication of the abend is given.

System Action: Output is terminated.

User Response: Correct the problem and re-execute the command.

$AVR984C - $AVRFSEA MODULE RELEASE IS NOT THE SAME AS THE INTERFACE RELEASE

Explanation: The release of the load module $AVRFSEA or the requested ENVIRONMENT AREA MODULE for an alternate $AVRS' data base does not match the current release of the Online interface module.

System Action: Interface is terminated.

User Response: Re-assemble your $AVRFSEA module ensuring that you use the current release of $AVRS' MACLIB, then re-linkedit the newly assembled $AVRFSEA replacing the older $AVRFSEA.

$AVR985I - THE SYSOUT PRINT FILE HAS BEEN CLOSED BY USER REQUEST

Explanation: The current print dataset has been closed and freed. If this was non-held sysout dataset, it has been queued for output processing by JES.

System Action: If creation of the print dataset was in a TSO session, the printed output will be queued with your logon ID as the name and the TSU number for your session as the JES number. If you created the print dataset from $AVRSCP and you were not using TSO, the name for the print will be queued for output using the task name of the executing monitor.

User Response: None required.

$AVR986 - $AVRFSEA PARAMETER UPFLDSN IS IGNORED.

Explanation: The $AVRS environment area did not find a UPFL dataset name in the UPFL environment.

System Action: $AVRS continues to process without UPFL.

User Response: If UPFL environment is desired, update your environment area ($AVRFSEA) with your UPFL dataset name. Assemble and link..

$AVR987A - PRINT CC MUST BE "A", "M", OR "N"

Explanation: The PRINT CC field on a demand print panel is not a valid value. The possible valid values are: A for ansi, M for machine or N for no carriage control.

System Action: The cursor is positioned at the PRINT CC field.

User Response: Respecify the value for this field and retry the demand print operation.

$AVR988I - PRINT FILE NOT OPEN, NO CLOSE NEEDED

Explanation: A request to close the current print dataset has been issued, but the current print file is not open. This indicates that no prints have been done since the session started or since the last time the file was closed.

System Action: Processing continues.

User Response: Informational.

$AVR995C - CALL SEA AT (516) 328-7000<==========

Explanation: A critical error, which requires intervention by **SEA**, has occurred.

System Action: The task is terminated.

User Response: See the message which precedes or follows this one for additional information. Call SEA with all available information.

$AVR996C - $AVRS' #.## GRACE PERIOD HAS EXPIRED<===
$AVR996C - $AVRS' #.## TRIAL PERIOD HAS EXPIRED<====
$AVR996C - $AVRS' #.## LICENSE HAS EXPIRED<=========

Explanation: The grace period, trial period or software license for release #.## has expired.

System Action: The task is terminated.

User Response: Contact SEA for assistance.

$AVR997C - $AVRS #.## PRODUCT NOT LICENSED
$AVR997C - $AVRS #.## RELEASE NOT LICENSED
$AVR997C - $AVRS #.## COMPONENT NOT LICENSED

Explanation: The product, release or component check in the $AVRS CPU zap has failed for release #.##.

System Action: $AVRS terminates.

User Response: Contact SEA for assistance.

$AVR998W - $AVRS #.## GRACE PERIOD WILL EXPIRE IN *xx* DAYS
$AVR998W - $AVRS #.## TRIAL PERIOD WILL EXPIRE IN *xx* DAYS
$AVR998W - $AVRS #.## LICENSE WILL EXPIRE IN *xx* DAYS
$AVR998W - RELEASE##.# WILL EXPIRE IN *xx* DAYS

Explanation: The $AVRS release #.##, grace period, trial period or software license is about to expire. This message is produced once a day or at task initialization commencing 45 days before a software license expiration, 5 days before a trial period expiration and 30 days for a grace period expiration.

System Action: Processing continues.

User Response: Contact SEA for a new release or for a new password for this release before the current one expires for un-interrupted operation of $AVRS.

$AVR999C - $AVR999C INVALID CALL TO *module* MODULE

Explanation: A call was issued from another program other than a $AVRS program.

System Action: Abend U0999.

User Response: Call SEA for a resolution.

The $AVRFM## messages are issued by the $AVRFMRG program. The User Abend numbers in Appendix A CANNOT be used to determine abend resolutions from $AVRFMRG.

$AVRFM01 - *ddname* FAILED TO OPEN SUCCESSFULLY

Explanation: Open failed for the specified *ddname*.

System Action: Abend U001.

User Response: Check the JCL to insure that all required JCL statements are present and that the DDNAMEs are spelled correctly and re-run the program.

$AVRFM02 - MAXIMUM OF 500 SYSIN INPUTS EXCEEDED

Explanation: The maximum number of sysin statements has been reached.

System Action: Abend U002.

User Response: Increase the value of the field MAXSYSIN in the $AVRFMRG source code. Then reassemble the Program and re-run the program.

$AVRFM03 - NO SYSIN INPUT GIVEN FOR COPY SELECTION

Explanation: No SYSIN data was detected by the program.

System Action: Abend U003.

User Response: Supply input control statements and re-run the program.

$AVRFM04 - RDJFCB FAILED FOR CURRENT INPUT FILE

Explanation: The read JFCB for the SYSUT1 failed.

System Action: Abend U004 with a dump.

User Response: Re-run the program in an initiator the does not use above the line SWA.

$AVRFM05 - STATUS REPORT TABLE EXHAUSTED

Explanation: The Maximum number of status report delete keys, 300, have been created.

System Action: Abend U005 with a dump.

User Response: Increase the value of the field MAXDEL in the $AVRFMRG source code. Then reassemble the Program and re-run the program. An increase to the region size may also be needed.

$AVRFM06 - DYNAMIC ALLOCATION FAILED FOR $AVROLST

Explanation: A dynamic allocation error has occurred.

System Action: Abend U005 with a dump.

User Response: Check the $AVRS initialization parameters for valid values for class, destination, form, etc. Correct as necessary and re-run the program.

$AVRFM07 - UNIT NAME NOT DEFINED OR IS INVALID

Explanation: Unit name missing or invalid.

System Action: Abend U007.

User Response: Correct the JCL or the SYSIN UNIT= value. Re-run the program.

$AVRFM08 - HISTORY RETENTION MISSING OR VALUE INVALID

Explanation: The HISTORY= value in the SYSIN is missing, non numeric, or too long.

System Action: Abend U008.

User Response: Correct the value and re-run the program.

$AVRFM09 - TIOT EXTRACT REQUEST FAILED

Explanation: The request failed.

System Action: Abend U004 with a dump.

User Response: Re-run the program in an initiator the does not use above the line UCB's.

$AVRFM10 - TIOT SEARCH FOR SYSUT1 DDNAME FAILED

Explanation: The request for SYSUT1 failed.

System Action: Abend U004 with a dump.

User Response: Correct the JCL by adding the DD statement and re-run the program.

$AVRFM11 - STAT/HIST TABLE LIMIT EXCEEDED'

Explanation: The maximum number of input history records (ST and FH), 15,000, has been reached.

System Action: Abend U011 with a dump.

User Response: Increase the MAXBUF# value. Reassemble the program and re-run the program. An increase in the region size may also be needed.

The $CRD### messages are issued by the $AVRS accumulation task. They are written to the $AVRERR file. At the end of the file is a list of all possible $CRD### messages.

$CRD000I - INITIALIZATION PARAMETER ACCEPTED

Explanation: This parameter is correct.

System Action: None required.

User Response: None required.

$CRD001C - SYSIN PARAMETER PREVIOUSLY DEFINED

Explanation: This parameter is a duplicate of a parameter that can only appear once.

System Action: Initialization may stop with a U102 abend or use the last value encountered.

User Response: Delete or correct one of the parameters and restart $AVRS.

$CRD001I - CONSOLE MODIFICATION ALLOWED BY DEFAULT

Explanation: The PASSWORD parameter is missing or blank.

System Action: None required.

User Response: None required if this is acceptable. If you require passwords for $AVRS modify commands add the PASSWORD= with the desired password and restart $AVRS.

$CRD002C - UNIDENTIFIABLE INITIALIZATION PARAMETER

Explanation: This parameter is unknown.

System Action: Initialization stops with a U102 abend.

User Response: Correct or remove the parameter involved.

$CRD002I - "ERRSRTN" FORCED TO "YES" BY THIS PARAMETER

Explanation: This parameter is correct and has forced ERRSRTN=YES even if the ERRSRTN=NO parameter was specified.

System Action: None required.

User Response: None required if yes is acceptable. If yes is not acceptable change this parameter accordingly or place the ERRSRTN=NO last in your initialization parameters.

$CRD003C - CLASS DEFINITION INVALID

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Validate the Initialization Parameter “ACCMCLS” and restart $AVRS.

$CRD003I - ROUTE TABLE PARAMETER FORCED "ERRSRTN" TO "YES"

Explanation: The ROUTTBL parameter has forced ERRSRTN=YES even if the ERRSRTN=NO parameter was specified.

System Action: None required.

User Response: None required if yes is acceptable. If yes is not acceptable change this parameter accordingly or place the ERRSRTN=NO at the end of the initialization parameters.

$CRD004C - CLASS PREVIOUSLY DEFINED FOR ANOTHER "LOGCLS#"

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Each LOGCLS# must have a unique class that matches one of the ACCMCLS parameters. The same class cannot be used twice. Correct the value and restart $AVRS. Consider using $AVRC178 and the $AVRLOG1 table to allow multiple systems syslogs to be accumulated from the same accumulation class.

$CRD004I - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD005C - SYSLOG IDENTIFIER NOT DEFINED ON "LOGCLS#" CARD

Explanation: This parameter is missing or invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct or add a SID value to the LOGCLS# parameter and restart $AVRS.

$CRD005I - PASSWORD PROTECTED MODIFY HAS BEEN TURNED OFF

Explanation: This parameter is correct.

System Action: None required.

User Response: None required if no password is desired for use of the modify commands. If you require passwords for $AVRS modify commands add the PASSWORD= with the desired password and restart $AVRS.

$CRD006C - SYSLOG ID PREVIOUSLY ASSIGNED TO A SYSLOG CLASS

Explanation: This parameter is a duplicate SID value.

System Action: Initialization stops with a U102 abend.

User Response: Correct or create a unique SID for each LOGCLS# parameter and restart $AVRS.

$CRD006I - NULL VALUE OR "OFF" SPECIFIED, PARAMETER HAS BEEN RESET TO OFF STATUS

Explanation: This parameter is correct.

System Action: None required.

User Response: None required if the off status is acceptable. Otherwise add the desired parameter value and restart $AVRS.

$CRD007C - PARAMETER VALUE NOT EQUAL TO "YES" OR "NO"

Explanation: This parameter value is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Change the value to yes or no and restart $AVRS.

$CRD007I - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD008C - ROUTING PARAMETER VALUE NOT "B" (BOTH) "L" (LOCAL) "R" (REMOTE) OR "N" (NOT RETURNED)

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Change the value to a B, L, or N and restart $AVRS.

$CRD008I - ABRVMSG TABLE HAS REACHED CAPACITY OF 31 ENTRIES

Explanation: This parameter is not considered for use.

System Action: Only the first 31 ABRVMSG parameters will be used.

User Response: Reconsider the messages that you wish to use. *To disable an old message you must set it off.* Just removing the parameter will NOT delete an old message. You must set off each message no longer needed using ABRVMSG=OFF=MSGID. New messages can then be added when $AVRS is restarted.

$CRD009C - REMOTE ID NOT INCLUSIVE OF "0-999"

Explanation: This parameter value is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Change the value of the parameter to be zero through nine hundred ninety- nine or correct the syntax of the parameter, ROUTBBL=###=dest.

$CRD009I - GLOBAL HIGHEST CONDITION CODE SET - NO EXCEPTIONAL CONDITION CODE TABLE IN USE

Explanation: This parameter is valid.

System Action: None required.

User Response: Consider using a $AVREXCC table to handle specific jobs with nonzero condition codes instead of 9999 in HIGHACC.

$CRD010C - NULL VALUE OR INVALID CHARACTERS SPECIFIED FOR "SORTDEV" PARAMETER

Explanation: This parameter is correct.

System Action: Initialization stops with a U102 abend.

User Response: Validate the Initialization Parameter “SORTDEV” and restart $AVRS.

$CRD010I - ALL DAYS HAVE BEEN EXCLUDED FROM BACKUP/ARCHIVE - NO NORMAL AUTOMATIC BACKUP WILL BE TAKEN

Explanation: This parameter is correct, but probably not desirable.

System Action: None required.

User Response: This is a warning that no task initiated backups, except emergencies will be done. Batch backups will need to be scheduled for the $AVRS database to function properly.

$CRD011I - SORTPGM= VALUE INVALID OR MISSING, DEFAULT OF "SORT" USED.

Explanation: This parameter is invalid or missing, but not a critical error.

System Action: Use 'SORT' as the sort program name for $AVRS and continue initialization.

User Response: None required if 'SORT' is a valid sort program name on your operating system. Otherwise, correct or add the parameter and restart $AVRS.

$CRD011C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD012C - FORMS NUMBER NOT ALPHANUMERIC AND/OR NATIONAL CHARACTERS

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: The RESFORM or RPTFORM value may be any combination of up to 4 characters consisting of A-Z, 0-9, @, #, or $. Correct the values, remove or comment out the parameter. To disable a previous value and use the system default make the parameter equal to blanks, "RESFORM=" .

$CRD012I - MAXLJES VALUES "YES" OR "NO" NO LONGER VALID. "YES" CONVERTED TO NOJESRSP AND "NO" CONVERTED TO NOCOUNT

Explanation: This parameter value is invalid, but not a critical error.

System Action: MAXLJES=YES converted to NOJESRSP, MAXLJES=NO converted to NOCOUNT, Initialization continues.

User Response: None required if this conversion is acceptable. Otherwise correct the parameter value and restart $AVRS.

$CRD013C - FCB NAME IS INVALID

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: The RESPFCB or RPTSFCB value may be any combination of up to 4 alphanumeric characters. Correct the values, remove or comment out the parameter. To disable a previous value and use the system default make the parameter equal to blanks, “RESPFCB=” .

$CRD014C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD015C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD017C - "BKUPTIM" INVALID - TIME MUST BE IN HHMM 0000 - 2359 ** (A 24 HOUR CLOCK)

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct the numeric values in the parameter or remove the parameter if no task scheduled backup or the default time is desired. The default is 2300 - 11:00 PM.

$CRD018C - SMF RECORD TYPE NOT "128 TO 255"

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct the value to be 128 through 255 or OFF. A value of OFF will disable the writing of SMF records. The system default is OFF.

$CRD019C - "HIGHACC" IS NOT FOUR NUMERICAL DIGITS

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct the numeric value. It must be a four position numeric from 0000 to 9999. A value of blank or OFF will disable HIGHACC checking and reset its value.

$CRD020C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD021C - VALUE GIVEN IS GREATER THAN MAXIMUM LENGTH FOR THE PARAMETER

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Validate the specified Initialization Parameter and restart $AVRS.

$CRD022C - VALUE NOT A NUMBER INCLUSIVE OF 1-255

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 ABEND.

User Response: Correct value to be one through two hundred and fifty-five or remove the parameter.

$CRD023C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD024C - "BKUPDAY=" FIELD(S) DO NOT CONTAIN "B" OR "X"
- MUST CONTAIN "B" OR "X" FOR EACH DAY OF WEEK - SUNDAY THROUGH SATURDAY.

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 ABEND.

User Response: A combination of seven “B” or “X”s are required. Add the necessary values to make seven. No commas or spaces can separate the values.

$CRD025C - INVALID VALUE OR RANGE SPECIFIED FOR THIS PARAMETER

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Check the value specified for the parameter in the $AVRS manual.

$CRD026C - VALUE SPECIFIED ON JESTYPE= PARAMETER NOT "JES2" OR "JES3"

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: The only valid value is JES2 or JES3. Correct the value or remove the parameter to get the default value of JES2.

$CRD027C - VALUE SPECIFIED ON WEEKEND= PARAMETER NOT "NORMAL" OR "EXTRA"

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct the value to be NORMAL or EXTRA. Check the spelling of the value or remove the parameter to get the default value of NORMAL.

$CRD028C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD038C - MAXLEXC= VALUES CAN ONLY BE JOB ,STC OR TSU OR COMBINATION SEPARATED BY COMMAS. IE; MAXLEXC=TSU,JOB

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Check the spelling of the parameter values - TSU,JOB,STC. Any combination of three may be used and if multiples are used they must be separated with commas. No parenthesis is necessary. OFF is a valid value to reset previous values or disable the parameter.

$CRD039C - INVALID WARNING PERCENT NUMERIC

Explanation: This value is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct the value to be zero through ninety-nine. There is no default for this parameter. “WARN=” will not reset or override the previous value. Setting WARN=0 resets and disables the $AVR029I RRDS percentage warning message function.

$CRD040C - INVALID WARNING INTERVAL NUMERIC

Explanation: This value is invalid.

System Action: Initialization stops with a U102 abend.

User Response: The valid values are one through 99. Zero or blank is not valid. The default value is one. This time parameter is only valid if WARN has a valid value.

$CRD041C - ** OBSOLETE MESSAGE **

Explanation: ***

System Action: ***

User Response: ***

$CRD042C - MAXLNES PARAMETER IS INVALID

Explanation: This value is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct this value to be a numeric up to ninety-nine million, nine hundred thousand ninety-nine and nine hundred ninety-nine - 99,999,999. A ‘,D’ can also be appended after the numerics to indicate DATASET level consideration. The default value is 7000.

$CRD043C - "ACCMTBL" STATEMENT CONTAINS AN UNKNOWN KEYWORD

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: A unknown keyword was encountered. Valid keywords/sub parameters are CLAS=, DEST=, FORM=, and REQC= each followed by an appropriate value and separated by commas. All the desired sub parameters must be on the same line as the ACCMTBL parameter. The first blank encountered is the end of the line. Multiple ACCMTBL parameters are permitted. CLAS has no default and is a required sub parameter. The default for DEST is any destination. The default for FORM is any form. The default for REQC is class X.

$CRD044C - DUPLICATED ACCUMULATION CLASS FOR THIS DESTINATION

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Change one of the accumulation class and destinations combinations, duplicates are not permitted. Check ACCMCLS/ACCDEST and ACCMTBL sub parameter combinations for the duplicates.

$CRD045C - INVALID VALUE FOR "REQC" KEYWORD

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct this value to be a valid respin sysout class or an ‘*’ to indicate suppression of respins. The default class is X.

$CRD046C - INVALID VALUE FOR "DEST" KEYWORD

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct this value to be a valid JES DESTINATION for your system. The default is any destination if the sub parameter is not included.

$CRD047C - INVALID VALUE FOR "CLAS" KEYWORD

Explanation: This parameter is invalid.

System Action: Initialization stops with a U102 abend.

User Response: Correct this value to a valid JES NON-HELD SYSOUT class. This is a required sub parameter for ACCMTBL, there is no default.

$CRD048C - MAXLJES PARAMETER MUST BE "COUNT", "NOCOUNT" OR "NOJESRSP"

Explanation: This parameter has an invalid value.

System Action: Initialization stops with a U102 abend.

User Response: Correct the value indicating how JES DATASET lines are counted toward MAX LINES processing. The three valid values are COUNT, NOCOUNT, and NOJESRSP. COUNT - all lines regardless of DATASET type as counted toward MAX LINES processing. NOCOUNT - do not count JES DATASET lines toward MAX LINES processing. NOJESRSP - JES DATASET lines are counted toward MAX LINES but JES DATASETS are never part of MAX LINES processing. The default is NOJESRSP.

APPENDIX C

$AVRS ACCESS METHOD

RETURN AND FEEDBACK CODES

Access Method Return & Feedback Codes

Appendix C provides a description of return codes and feedback codes returned from the $AVRS ACCESS METHOD.

Return codes 4, 8, and 12 are IBM VSAM codes with associated feedback codes. Reference the OS/VS Virtual Storage Access Method (VSAM) Programmer's Guide for a description of the function's feedback code.

Any return code 124 X(7C) and above is from the $AVRS' ACCESS METHOD and has an associated feedback code. These $AVRS' ACCESS METHOD codes are described in the following tables.

IBM Return Codes

IBM Access Method Return Codes

RETURN CODE	IBM ERROR DESCRIPTION
4 X(04)	VSAM processing error -- Reference OS/VS Virtual Storage Access Method (VSAM) Programmers Guide for description of feedback code.
8 X(08)	VSAM logical error -- Reference the OS/VS Virtual Storage Access Method (VSAM) Programmer’s Guide for a description of the function’s feedback code.
12 X(0C)	VSAM physical error -- Reference the OS/VS Virtual Storage Access Method (VSAM) Programmer’s Guide for a description of the function's feedback code.

Figure 58 IBM Access Method Return Codes

$AVRS Access Method Return Codes

RETURN CODE	$AVRS ERROR DESCRIPTION
124 X(7C)	Error occurred in $AVRS ACCESS METHOD main module ($AVRSAM)
128 X(80)	Error occurred in $AVRS ACCESS METHOD open module ($AVRAMOP)
132 X(84)	Error occurred in $AVRS ACCESS METHOD close module ($AVRAMCL)
136 X(88)	Error occurred in $AVRS ACCESS METHOD put module ($AVRAMPT)
140 X(8C)	Error occurred in $AVRS ACCESS METHOD get RRDS module ($AVRAMGR)
144 X(90)	Error occurred in $AVRS ACCESS METHOD get KSDS module ($AVRAMGK)
148 X(94)	Error occurred in $AVRS ACCESS METHOD get RRDS block module ($AVRAMGB)
152 X(98)	Error occurred in $AVRS ACCESS METHOD free RRDS block module ($AVRAMFB)
156 X(9C)	Error occurred in $AVRS ACCESS METHOD compress decompress module ($AVRAMCD)
160 X(A0)	Error occurred in $AVRS ACCESS METHOD modify control block module ($AVRAMMD)
164 X(A4)	Error occurred in $AVRS ACCESS METHOD end request module ($AVRAMEQ)
168 X(A8)	Error occurred in $AVRS ACCESS METHOD set LOCK module ($AVRAMLK)
172 X(AC)	Error occurred in $AVRS ACCESS METHOD delete work module ($AVRAMDL)
176 X(B0)	Error occurred in $AVRS ACCESS METHOD synchronization module ($AVRDBID)
180 X(B4)	Error occurred in $AVRS ACCESS METHOD allocates module ($AVRAMAL)
204 X(CC)	Error occurred in $AVRS ACCESS METHOD open/close module ($AVRAMOC)

Figure 59 $AVRS Access Method Return Codes

$AVRS Feedback Codes

The following codes are valid when the RC= value is 120 or greater.

$AVRS FEEDBACK CODES

FEEDBACK CODE	ERROR DESCRIPTION
4 X(04)	An invalid or unknown function was requested. Correct the requested function.
8 X(08)	Getmain failed for $AMWORK control block. Increase the amount of virtual storage available for this task and re-execute the task.
12 X(0C)	Action was requested against an invalid or unknown file. If this error is from a user written program, ensure the macro file specification is correct. Ensure that the correct data set name was specified on the DD card for this file. Ensure the data set name specified is defined in the VSAM catalog.
16 X(10)	No valid pointer could be found for the LOCK file UCB. Enqueue/Reserve processing failed. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
20 X(14)	No valid pointer could be found for the LOCK file UCB. Dequeue processing failed. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
24 X(18)	An attempt was made to update a KEY type record but the KSDS was not open for output. If this is a user written program, no KEY updates are allowed. If this is a $AVRS program the there was an internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.

Figure 60 $AVRS Access Method Feedback Codes

The following codes are valid when the RC= value is 120 or greater.

$AVRS FEEDBACK CODES	
FEEDBACK CODE	ERROR DESCRIPTION
28 X(1C)	Database is full; either KSDS or RRDS Check the return code or look at the percent full statistics using the full screen interface to determine if the KSDS or the RRDS is full. Delete and reformat the files. Instructions for formatting may be found in the installation guide.
32 X(20)	No record pointer found when issuing a put to the RRDS DATA file. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
36 X(24)	No record length specified when issuing a put to the RRDS DATA file. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
40 X(28)	An attempt was made to update a LOCK record but the LOCK file was not open for output. If this is a user written program, no LOCK updates are allowed. If this is a $AVRS program the there was an internal $AVRSAM error. Try re executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
44 X(2C)	No record pointer found when issuing a put to the KSDS KEY file. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.

Figure 61 $AVRS Access Method Feedback Codes (continued)

The following codes are valid when the RC= value is 120 or greater.

$AVRS FEEDBACK CODES	
FEEDBACK CODE	ERROR DESCRIPTION
48 X(30)	A Dequeue request has been issued without a preceding Enqueue/Reserve request Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
52 X(34)	A call to the $AVRAMCD did not specify the type of processing desired (compress or decompress). Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
56 X(38)	An attempt was made to access a file that had not been opened. If this is a user written program, ensure that you have coded open macros for all of the files you are trying to access. Correct any errors and reexecute the task. If this is a $AVRS program the there was an internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
60 X(3C)	An RPL number other than 0 or 1 was specified on a macro attempting to access the KSDS file. If this is a user written program, ensure that you have coded you macros properly for the files you are trying to access. Correct any errors and re-execute the task. If this is a $AVRS program the there was an internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.

Figure 62 $AVRS Access Method Feedback Codes (continued)

The following codes are valid when the RC= value is 120 or greater.

$AVRS FEEDBACK CODES	
FEEDBACK CODE	ERROR DESCRIPTION
64 X(40)	No key length specified on a macro attempting to access the KSDS file. If this is a user written program, ensure that you have coded you macros properly for the files you are trying to access. Correct any errors and re-execute the task. If this is a $AVRS program the there was an internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
68 X(44)	No block number was specified on a macro attempting to access the RRDS DATA file. If this is a user written program, ensure that you have coded your macros properly for the files you are trying to access. Correct any errors and re-execute the task. If this is a $AVRS program the there was an internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
72 X(48)	While processing a file from the RRDS DATA file a miss matched date and time from the file key and the data header has occurred. Processing for this file cannot continue. This condition is normally encountered when the file being processed by your job has been deleted and reused by another task. You may treat this as a premature end of file condition in any user written program and continue processing with the next file.
76 X(4C)	An attempt to lock the $AVRS DATABASE was made while an UNLOAD job was running. This task must be wait until the UNLOAD process completes.

Figure 63 $AVRS Access Method Feedback Codes (continued)

The following codes are valid when the RC= value is 120 or greater.

$AVRS FEEDBACK CODES	
FEEDBACK CODE	ERROR DESCRIPTION
80 X(50)	An attempt to lock the $AVRS DATABASE was made while a FORMAT job was running. This task must wait until the FORMAT process completes.
84 X(54)	An attempt to lock the $AVRS DATABASE was made while an ACCUMULATION task was executing. This task must be wait until the ACCUMULATION process completes. If the ACCUMULATION task is not active, the $AVRLOCK program may be run to release a lock left by an aborted termination of the ACCUMULATION task.
88 X(58)	An attempt to lock the $AVRS DATABASE was made while an BACKUP task was executing. This task must be wait until the BACKUP process completes.
92 X(5C)	A delete was requested but the key type was not for a job or file key. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
96 X(60)	An attempted GETMAIN failed during a file deletion process. The file was not deleted Increase the amount of virtual storage available for this task and re-execute the task.
100 X(64)	Delete not valid when BACKUP/PHASE1 or UNLOAD running of JOB KEY active. Internal $AVRSAM error. Try re-executing the task; if the error persists please have all available documentation -- including core dump -- at hand and call for technical support.
104 X(68)	An invalid $AVR1ST module has been place into the load library. Assemble and link edit a valid $AVR1ST module.
108 X(6C)	A synchronization error has occurred in the $AVRS database. Ensure that all data base DDNAMES are present and accounted for and that the dataset names are spelled correctly. If necessary, run the $AVRLOCK program with a parm of SYNCLOCK.

Figure 64 $AVRS Access Method Feedback Codes

INDEX

INDEX

$AVRBCTL parameter 11, 14, 92, 393
$AVRBKUP parameter
EXPDT 92
$AVRECOV 126
Replaced by $AVRCLEN in 5.1A. 126
$AVRPULL 131
ACKNOWLEDGMENT= 143
AHTYPE= 148
AHWORK= 148
ANNOTATIONS= 147
CCODE= 144
CLASS= 147
DDNAME= 145
EDATE= 141
ETIME= 142
FA= or SCAN= 146
FCB= 148
FORM= 147
ID= 144
JO= 144
LOCATE= 145
LUNAME= 149
NAME= 141
NUMBER= 144
PROCSTEP= 145
RANGE= 143
REP 150
Return Codes 131, 167, 185
Sample Clist 138
Sample JCL 135, 136
Sample Procedure 133
Sample Work Selection List and Error Report 152
SDATE= 141
Security 131
Selection Criteria 139, 151, 182
Selection Examples 151
SRANGE= 146
SSTART= 146
STEPNAME= 144
STIME= 141
SYSIN Parameters 139
TSO foreground 137
TYPE= 140
UCS= 148
VPCLASS= 149
VPFCB= 149
VPFORM= 149
VPHOLD= 150
$AVRS
//*$AVRS JCL Comment Statement .. xxv, 47
//*$AVRS JCL Destination Statement ... xxv
Abends 201
Accessing 5
Browse Facility 4
Control Cards 74
Description xv
Features xvii
How $AVRS Works With Your System xxiii
Introduction xv
Messages 213
Reports xviii
Security xvii
Welcome to $AVRS v
What's New With $AVRS vii
$AVRS programs 125
$$DELDMP 125
$$DELKEY 125
$AVRBANR 125
$AVRBKPR 125
$AVRBKUP 126
$AVRCLEN 126
$AVRCOND 126
$AVRCRSE 126
$AVRDBRP 126
$AVRFDEL 126
$AVRFICH 126
$AVRFMRR 126
$AVRFOLD 126
$AVRGRPT 126
$AVRINIT 126
$AVRLIDR 126
$AVRLOCK 126
$AVRMRGF 126
$AVRMXPR 126
$AVRPULL 127
$AVRSCP 127
$AVRSELR 127
$AVRSTMP 127
$AVRSXPR 127
$AVRTAPE 127
$AVRTPRP 127
$AVRUNLD 127

$AVRUX1 127
$AVRUX2 127
$AVRUX3 127
$AVRUX4 127
$AVRUX5 127
$AVRUX7 127
$AVRUX8 127
$AVRUX9 127
$AVRUXA 127
$AVUMTRR 127
$AVURMRR 127
DISPLYGBL 128
DSPLYLCK 128
EQQYX010 128
FRONTEND 128
ISP$AVRS 128
USERPGM 128
$AVRSECU
with $AVRPULL 132
with $AVRTAPE 167
$AVRSELR 153, 157
$AVRREST 156
ACKNOWLEDGMENT= 162
Backup/Archive Files as Input 153
CCODE= 163
EDATE= 160
ETIME= 160
ID= 162
JO= 162
NAME= 159
NUMBER= 163
ONEMATCH= 158
RANGE= 162
Return Codes 153
Sample JCL 155
Sample Selective Restore Status Report 164
SDATE= 159
Selection criteria 153, 157
Selection Examples 164, 192
STIME= 160
TYPE= 158
$AVRTAPE
ACKNOWLEDGMENT= 176
AHTYPE= 181
AHWORK= 180
ANNOTATIONS= 179
Backup/archive files as input . . . 167
Batch 168
CCODE= 177
CLASS= 179
DDNAME= 178
EDATE= 174
ETIME= 175
FA= or SCAN= 178
FCB= 180
FORM= 180
ID= 177
JO= 176
LOCATE= 178
NAME= 173
NUMBER= 177
ONEMATCH= 172
PARM WTOR 168
PROCSTEP= 178
RANGE= 175
Sample JCL 170, 171
Sample Procedure 169
Sample Work Selection List and Error Report 183
SDATE= 174
Security 167
Selection Criteria 172
Selection Examples 182
SRANGE= 179
SSTART= 179
STEPNAME= 177
STIME= 174
TYPE= 173
UCS= 180
$AVRTPRP 185
ACKNOWLEDGMENT= 190
Archive Tape Index Report 195
EDATE= 188
ERROR= 190
ETIME= 189
ID= 192
LINES= 191
MESSAGES= 191
NAME= 188
NUMBER= 191
Program Description 185
RANGE= 190
Return Codes 185
Sample $AVRTPRP Parse Report 193
Sample Archive Tape Index Report 194
Sample JCL 186
Sample Reports 193
SDATE= 188
Selection Criteria 186
STIME= 189

TYPE= 187
$AVRUX2
with $AVRPULL 131
with $AVRTAPE 167
A/D (Ascending/Descending)
with SORT Field 30
Abends 201
Access Method 393
$AVRS Feedback Codes 396
IBM Return Codes 394, 395
Return & Feedback Codes 393
Accessing $AVRS 5
ACCUMULATE
definition xxi
ACK CODE
Primary Selection Panel 27
Acknowledgment 40, 43, 75
Primary Selection Panel 27
Acknowledgment and Annotation
Introduction xvi
ACKNOWLEDGMENT=
$AVRPULL 143
$AVRSELR 162
$AVRTAPE 176
$AVRTPRP 190
ACT
Action Columns 39
ACT field 38
AH (Archive History)
Selection Field 92
with TYPE Field 18
AHTYPE=
$AVRPULL 148
$AVRTAPE 181
AHWORK=
$AVRPULL 148
$AVRTAPE 180
ALL
Subparameter of FIND 66, 67
Allocation Panel 116
Annotation 40, 43, 75
Introduction xvi
ANNOTATIONS=
$AVRPULL 147
$AVRTAPE 179
ARCHIVE
definition xxi
Archive Function or Backup
Introduction xvii
Multiple archive levels xvii
Archive History
Detail Panel 92
Directory Fields 44
Displaying 92
with TYPE Field 18
Archive History (AH). 95, 98
B (Browse)
Selection Character 57
BACKUP
definition xxi
Banner Pages
Print Attributes Panel 109
with $AVRPULL 132
with $AVRTAPE 168
BASE KEY
definition xxi
Batch Facilities
Introduction xvii
BROWSE Command 78
Browse Feature 56, 57
"S"electing Individual Files 85
Browsing Concatenated Files 78
CC Command 57, 64
COLS or COLUMNS Command 57
Column Command 64
Find Command 65
Finding HEX Characters 71
Finding Non-Display Characters 72
Finding Text Strings 71
Hex Command 57, 61
Introduction . . . . . xvi, 4, 56
ISPF 4
JOBs, STCs and TSUs 77, 78
Limit Command 69
Limiting Search by Columns 69
Locate Command 59
Lock Command 57, 59
Moving Data 57
Positioning By Files 85
Sample, concatenated browse 79
Scrolling 57
Selection Character 56
Subparmeters of FIND 65
Viewing Data 57
VSAM xvi
CAPS command 69
CC Command
Displaying Carriage Control 57

in Browse 64
CCODE
values 144, 163, 177, 190
CCODE 42
Values 42
CCODE=
$AVRPULL 144
$AVRSELR 163
$AVRTAPE 177
Changing functions 78
CLASS=
$AVRPULL 147
$AVRTAPE 179
CLOSE DS command 117
CLOSE PRT Command 110, 114
COLS command 64
COLUMNS Command 64
in Browse 57
Command Processor 6
Commands
(AC)k 78
(BR)OWSE 78
(FI)LES 78
(IN)FO 78
(PR)INT 78
BROWSE 78
CAPS 69
CC (Carriage Control) 57, 64
CLOSE DS 117
CLOSE PRT 110, 114
COLS 57, 64
COLUMNS 64
DC 121
DC 122
DZ 121
FILES 78
FIND 57, 65
HEX 57, 62
INFO 78
LIMIT 69
LOCATE 51, 59
LOCK 57, 59
MODIFY xxv
N (Next) 78, 85
P (Prev) 78, 85
PRINT 114
REFRESH 52
Shortcuts 78
SORT 29, 51
SUBREST (Submit Restore) . . . xxvi, 93
TF (Top of File) 78
COMMENT
$AVRS Special JCL Statement . . . . . xxv, 46
Directory Field 42
COMPLETION CODE
Information Panel 73
CONFIRM DELETIONS
Primary Selection Panel 33
Control Cards
$AVRS 74
CPU TIME
Information Panel 73
Customizations
Online Display 121
D (Delete)
Selection Character 118
Data Base
USAGE 34
Database
VSAM xv
Work retention by days or cycles xxiii
Dataset Allocation Panel 115
fields 116
Dates
END DATE Selection Field 21
Formats 20
START DATE Selection Field 19
DBBLKS
Directory field 42
DC 19
DDNAME=
$AVRPULL 145
$AVRTAPE 178
Definition of Terms xxi
DELETE 107, 118
Selection Character 56
Deleting Work 118
Confirm Deletions Field 33, 118
Delete Confirmation Panel 118
Selection Character 118
Description of $AVRS xv
DEST Statement
//*$AVRS Special JCL Statement 46
//*$AVRS Special Statement xxv
Directories 3
Archive History 44
Archive History Directory 36
Archive History(AH) 44
Fiche History 45
Fiche History Directory 36
Fiche History(FI) 45

Fiche Index Reports 99
Fields 39
File Selection 82, 83
Formats 38
Header Lines 38
Heading fields 39
JOB 40
JOB, STC, TSU Directory 36
JOB(JN) 40
Locate Command 51
Primary Selection Directory 3
Print, output or delete 36
Processing Entries 56
Refresh Command 52
SCROLL variable 48
Scrolling 3, 47
Selecting work for processing . . . 55
Sort Command 51
Sorting 52
Status Report 46
Status Report Directory 36
Status Report(ST) 46
Status Reports 90
STC 40
STC(SN) 40
Syslog 42, 87
Syslog Directory 36
Syslog Directory(SL) 36
TSU 40
TSU(TN) 40
Tutorial 7
Work Selection 36
Display Customization Zap Status
Display Customization Zap Status 19
Display Customization Zaps 19
DC 19
Displaying
Archive History (AH) 92
Fiche History 101
Fiche History (FH) 101
Fiche Index Reports 99
Fiche Index Reports (FI) 99
JOBs 77
Status Reports 89
Status Reports (ST) 89
Status Reports(ST) 90
STC(SN) 78
STCs 77
Syslog 87
SYSLOG(SL) 88
TSU(TN) 78
TSUs 77
DZ 19
EDATE=
$AVRPULL 141
$AVRSELR 160
$AVRTAPE 174
$AVRTPRP 188
END DATE
Primary Selection Panel 21
END TIME
Primary Selection Panel 22
ERROR=
$AVRTPRP 190
ETIME=
$AVRPULL 142
$AVRSELR 160
$AVRTAPE 175
$AVRTPRP 189
ETSO
Screen Printing 114
Exact name specification 15
EXPDT
Type 1 Sub-parameter 92
FA=
$AVRPULL 146
$AVRTAPE 178
FCB=
$AVRPULL 148
$AVRTAPE 180
Features xvii
Acknowledgment 40, 43
Acknowledgment Code 75
Annotation 40, 43
Annotations 75
Archive Function or Backup xvii
Batch Facilities xvii
Microfiche Facility xvii
Security xvii
FH (Fiche History)
Selection Field 101
with TYPE Field 18
FI (Fiche Index Reports)
with TYPE Field 19
FI (Fiche Index)
Selection Field 99
FI# (Fiche Number)

Directory Field 46
Fiche History
Detail Panel 101
Directory Fields 45
Displaying 101
with TYPE Field 18
Fiche Index Reports
Displaying 99
Sample 100
with TYPE Field 19
File Selection
Directory Fields 83
Panel 82
Selection Character 56
File Selection Directory
Informational Fields 83
JOBs, STCs, and TSUs 82
Sample 82
File Selection Panel
JOBs, STCs and TSUs 77
Files
Positioning By in Browse 85
Printing 111, 115
Selecting for browse 85
FILES Command 78
FIND 57, 65
Altering FIND Search Criteria Parameters 68
CAPS Command 69
Entering FIND Strings 68
HEX Characters 71
Limit Command 69
Non-display Characters 72
Subparameters 66
Text Strings 71
FIND Command
in Browse 57, 65
FIND subparameters 66
ALL 67
FIRST 67
LAST 67
PRE 67
PREFIX 67
PREV 67
SUF 67
SUFFIX 67
WORD 67
FIRST
Subparameter of FIND 66
FORM=
$AVRPULL 147
$AVRTAPE 180
Fullscreen
Reentrant Interfaces 3
Hardware/Software Requirements xxiv
HC= 40, 195
Help and Tutorial Facility
Examples 8
Help 4, 8
Help Panel in Browse 8
Introduction xvi
Tutorial 7
HEX 62
Finding HEX Characters 71
HEX Command 62
in Browse 57
non-HEX mode 64
Syntax 64
HOW $AVRS WORKS WITH YOUR SYSTEM
Hardware/Software Requirements xxiv
Output Media Considerations xxiv
IBM Return Codes 394
IC= 37, 40, 195
ID=
$AVRPULL 144
$AVRSELR 162
$AVRTAPE 177
$AVRTPRP 192
IEF Messages
Information Panel 73
INCMPLT 40, 196
INFO Command 78
Information Feature
Information Panel 72
Introduction xvi
JOBs, STCs and TSUs 77
Selection Character 56
Information Panel
COMPLETION CODE 73
CPU TIME 73
General 72
IEF Messages 73
JCL Errors 72
PROCSTEP NAME 73
Sample 72, 74
Step Completion 72
STEPNAME 73
Interfaces
Online . . . . . xv, 3
Introduction xv
ARCHIVE xxv
Introduction to $AVRS xv

Restore xxvi
ISPF
$AVRS interface 5
Interface 5
ISPF Browse 4
J# (Job Number)
with TYPE Field 18
JCL ERR
..... 195
JCL ERROR 296
JCL Errors
Displayed on Information Panel
..... 72
Information Panel 74
JO (Job Owner)
with TYPE Field 18
JO=
$AVRPULL 144
$AVRSELR 162
$AVRTAPE 176
Job Owner
with NAME Field 17
Job Selection
Directory fields 40
JOBs
Directory 40
Displaying 77
Displaying JCL Errors 73
Displaying STEP Information ... 73
File Selection 82
Outputting 115
Printing 107, 111
Selecting by TYPE 18
LAST
Subparameter of FIND 66
LIMIT Command
in Browse 69
Limiting a Search by Columns ... 70
LINES=
$AVRTPRP 191
LOCATE Command 51, 59
in Browse 57
LOCATE=
$AVRPULL 145
$AVRTAPE 178
LOCK Command 59
in Browse 57
LUNAME=
$AVRPULL 149
Maintenance
Online Display 121
Masking Names
Defination 15
Primary Selection Panel 17
Messages 213
APPENDIX B 211
MESSAGES=
$AVRTPRP 191
Microfiche Facility
Introduction xvii
MINLENG 17, 18
MODIFY command
General Information xxv
MSGCLASS 23
Multiple entries
Printing 110
Processing 56
N (Next) Command 78
Files Selection Directory 85
NAME
Masking 15
NAME Range
Primary Selection Panel 16
NAME=
$AVRPULL 141
$AVRSELR 159
$AVRTAPE 173
$AVRTPRP 188
NEXT Command
File Selection 85
in Browse 78
Syntax for Browse 86
NO-EXEC 40
NOTCTLG 40
Notify
Job Owner Selection 17
NUMBER
Selection field 19
NUMBER=
$AVRPULL 144
$AVRSELR 163
$AVRTAPE 177
$AVRTPRP 191
O (Output)
Selection Character 115
ONEMATCH=
$AVRPULL 172
$AVRSELR 158

Online Interfaces xv
ACTion Field 39
Browse Feature xvi
Command Processor 6
ETSO 6
Help and Tutorial Facility xvi
Information Feature xvi
ISPF 5
Overview 3
ROSCOE ETSO 6
TSO 6
VCSI . . . . . xv, xx, 3
OUTPUT 107, 115
Closing the output dataset 117
Defining dataset 116
Selection Character 56
Output Media Considerations xxiv
P (Prev) Command 78
Files Selection Directory 85
P (Print)
Selection Character 107
Panels
Archive History Detail 92
Archive History Detail (AH) 92
Confirm Deletions 118, 119
Customizations 121
Fiche History Detail (FH) 101
File PRINT 111, 115
File Selection 82
File VTAM Print 113
Help 8
Maintenance Zaps 121
Output Dataset Allocation 115
Primary Selection 13
PRINT attributes 108
Status Report(ST) 91
Submit Restore 94, 97
SYSLOG Selection Directory(SL) 87
PARTIAL 41, 196
PGMER NAME
Primary Selection Panel 29
PREFIX
Subparameter of FIND 66
PREV
Subparameter of FIND 66
PREVIOUS Command
File Selection 85
in Browse 78
Syntax for Browse 86
Primary Control Panel 13
Primary Selection Directory 3
Primary Selection Panel 13
ACK CODE 27
AH (Archive History) 18
CONFIRM DELETIONS field 33
Data Base Usage 34
DC (Display Customization Zaps) 19
DZ (Display Maint. Zaps) 19
END DATE Selection Field 21
END TIME 22
Example 13
FH (Fiche History) 18
FI (Fiche Index Reports) 19
Fields 15
J# (Job Number) 18
JO (Job Owner) 18
NAME field 15
NUMBER Field 19
PGMER NAME 29
RANGE Field 32
RETURN CODE Field . . 23, 24, 142, 161, 175, 189
SL (SYSLOG) 18
SORT Field 29
ST (Status Reports) 19
START DATE Field 19
START TIME 20
SYSLOG ID Field 17, 19
THRU field 15
TYPE Field 17
PRINT 107
Attributes Panel 110, 113
Command 114
Multiple Screens 114
Printing
Closing the Print File 110
Files 111, 115
FROM ARCHIVE HISTORY 96
Multiple entries 110
Multiple screens 114
Screen Print 114
Selection Character 56
Specifying characteristics 108
Process Function Designators
A (Ann/Ack) 56
B (Browse) 56
D (Delete) 56
I (Step Info) 56
O (Output) 56
P (Print) 56
R (Restore) 56

S (Select) 56
V (VTAM Print) 56
Processing Entries
General Information 56
PROCSTEP NAME
Information Panel 73
PROCSTEP=
$AVRPULL 145
$AVRTAPE 178
Programs
$AVRTPRP 185
Quick Path 103
DC 105
J# 104
JN 104
QX 105
SD 105
SN 104
ST 105
TN 104
BR 105
FI 105
IN 105
QUICKREF
Chicago-Soft, LTD 213
QUIKHELP
available in the Installation Library 213
R/C (Row/Col)
Directory Field 46
RANGE=
$AVRPULL 143
$AVRSELR 162
$AVRTAPE 175
$AVRTPRP 190
Primary Selection Panel 32
RC= 40, 224
RECAP Period
Directory Field 46
REFRESH Command 52
REP
$AVRPULL 150
Reports xviii
$AVRPULL Sample Work Selection List and Error Report . . 152
$AVRTAPE Sample Work Selection List and Error Report . . 183
Archive History Status, TYPE field 19
Archive Tape Index Report 195
Error Summary by Name Within Date/Time xviii
Error Summary by Name Within Type . . xviii
Fiche Index 119
Fiche Index, Displaying 99
Fiche Index, TYPE field 19
File Contents xviii
Miscellaneous Error Messages and Abend Messages xviii
Recap by Jobname . . . . . xviii, 185
Sample $AVRTPRP Parse Report 193
Sample Selective Restore Status Report 164
Status 119
Status, Displaying 89
Unload/Format xviii
RESPIN
definition xxi
Respun
Selecting Work 25
Restore Archived Work 153
$AVRSELR 153
FROM ARCHIVE HISTORY 95, 97
Restoring Work
$AVRTPRP Report Program 185
Selection Character 56
Submit Panel 94, 97
RETURN CODE
Primary Selection Panel 23, 24, 142, 161, 175, 189
Return codes
access method 393
IBM 394
ROSCOE
$AVRS interface to 6
Screen Printing 114
RQ 40, 195
Sample Archive Tape Index Report 194
Sample JCL
$AVRSELR 155
$AVRTPRP 186
SCAN=
$AVRPULL 146
$AVRTAPE 178
SCANERR 41, 196
Screen Navigation
Shortcuts 78
Screen Printing - (Non ISPF) 114

Scrolling 47
Directories 47
File Selection Directories 83
in Browse 57
Setting the SCROLL variable 48
SDATE=
$AVRPULL 141
$AVRSELR 159
$AVRTAPE 174
$AVRTPRP 188
Security xxvi
Introduction xvii
SEL# field 38
Selecting Work
For Processing 55
General 14
General Information 56
NAME field 15
Respun 25
THRU field 15
TYPE Field 17
SELECTION RULE 23, 142, 161, 175, 189
Selection Functions
Listed on Directories 38
SELECTION RULE . . . 23, 142, 161, 175, 189
SELECTION RULE 13, 14
Selective Restore Facility
Introduction xxvi
Shortcuts For Screen Navigation 78
(AC)k 78
(BR)OWSE 78
(FI)LES 78
(IN)FO 78
(PR)INT 78
SI/O 40, 196
SL (SYSLOG)
Selection Field 87
with TYPE Field 18
SORT
from Directories 51
Primary Selection Panel 29
SORT command 51
Sorting
From the Primary Selection Panel 29
Special Control Statements 46
SRANGE=
$AVRPULL 146
$AVRTAPE 179
SSTART=
$AVRPULL 146
$AVRTAPE 179
ST (Status Reports)
Selection Field 89
with TYPE Field 19
START DATE
Primary Selection Panel 19
START TIME
Primary Selection Panel 20
Status Report
Directory 90
Displaying 89
Sample 91
Status Reports
Directory fields 46
with TYPE Field 19
STCs
Directory 40
Directory fields 40
Displaying JCL errors 73
Displaying STEP information 73
Selecting by TYPE 18
STCs
Displaying 77
Step Completion
Displayed on Information Panel 72
STEPNAME
Information Panel 73
STEPNAME=
$AVRPULL 144
$AVRTAPE 177
STIME=
$AVRPULL 141
$AVRSELR 160
$AVRTAPE 174
$AVRTPRP 189
Submit Restore Panel 94, 97
SUBREST command xxvi
Submit Restore Command 93
SUFFIX
Subparameter of FIND 66
Switching functions 78
SYSLOG
Directory 42
Displaying 87
with TYPE Field 18
SYSLOG
Selection Directory 87
SYSLOG ID 17
Primary Selection Panel 17, 19
SYSTEM ID 19

SYSTEM ID 19
Text string xxvi
Finding 71
TF (Top of File) Command 78
THRU
selection field 15
Time
Also see RANGE= 32
as Selection Criteria 20
TSO
$AVRS interface to 6
Screen Printing 114
TSUs
Directory 40
Directory fields 40
Displaying 77
Selecting by TYPE 18
Tutorial 7
TX 40, 195
Type 1
EXPDT= 92
TYPE Selection Field
Primary Panel 17
TYPE=
$AVRPULL 140
$AVRSELR 158
$AVRTAPE 173
$AVRTPRP 187
UCS=
$AVRPULL 148
$AVRTAPE 180
Unnn 196
USER
Job Owner Selection 17
USER PROFILE (UPFL 6
V (VTAM Print)
Selection Character 111
VCSI . . . . . xv, xx, 3, 6
Screen Printing 114
VPCLASS=
$AVRPULL 149
VPFCB=
$AVRPULL 149
VPFORM=
$AVRPULL 149
VPHOLD=
$AVRPULL 150
VSAM
Browse xvi
database xv
VTAM PRINT 107
VTAM Printing
Files 113
Selection Character 56
Specifying Characteristics 112
Welcome to $AVRS v
WORD
Subparameter of FIND 66
Work Selection
Directories 3, 36
Directory Sample 37
Guidelines 35
Scenarios 35
with TYPE Field 17
Work Selection Directories 40
AH 44
FH 45
JN, ... 40
SL 42
SN 40
ST 46
TN 40
Zap
customizations applied 121
Display Maintenance Zap Status 19
identification numbers 121
maintenance applied 121