E-SRF

**EKC Security
Event Reporting Facility**

Release 2.1

Command Reference

E-SRF™ is a proprietary product
developed and maintained by

EKC Inc.
10400 West Higgins Road
Rosemont, Illinois 60018
USA

(847) 296-8010

Technical Support:
(847) 296-8035

Contents

Name	Contents
Installation Guide	E-SRF installation including: installation and maintenance steps, startup and shutdown considerations, and backup and recovery procedures.
Change Summary Guide	Contains all new features and system function changes.
General Overview	An overview of E-SRF and its components.
Resource Grouping Facility Guide	Brief overview of the Resource Grouping Facility, its relationship to E-SRF, language command syntax, TSO commands and JCL.
Access Analysis Reports Guide for ACF2 *Access Analysis Reports Guide for RACF*	Brief overview of Access Analysis reports, explanation of the DataOwner and Userid/LogonidOwner reports, command syntax, utilities necessary for creating input to reports, and JCL.
Event Reporting User Guide	A "How To" guide for users of E-SRF Event Reporting.
Event Reporting Facility - Command Reference	Explains the Event Reporting Facility command processor, command syntax, and JCL.
Event Reporting Facility - Masterfile and Data Dictionary Reference	Explains the structure of the E-SRF Masterfile and describes all Masterfile fields.
Event Reporting Facility - Messages and Codes	Lists Event Reporting Facility messages and codes.
Event Reporting Facility - Report Overlays Guide	An overview of the report overlays provided with the Event Reporting Facility.

Chapter 1:   E-SRF Command Function Overview

Introduction

The E‑SRF Event System is a batch processing system that is command driven. This publication explains the E‑SRF Command Processor and its commands. Anyone executing E‑SRF may use this publication as a reference manual.

The Command Processor

In this release of E‑SRF, the only way to run E‑SRF Event System functions is to execute the off‑line Command Processor) through a batch program execution of ESRFCMD. E‑SRF will minimally, open its Masterfile which is stored on a VSAM Cluster (MASTER), a general print data set for logging E‑SRF events (SYSPRINT), and the input parameter file (SYSIN) which contains the commands listed in this reference manual

The E‑SRF Command Processor uses the SYSIN dataset to acquire parameters in the form of "commands" for execution. All parameters are presented and executed through the Command Processor. This includes setting E‑SRF system options, initiating updates, and producing reports.

The program execution PARM normally supplied in JCL on the EXEC statement is stored within the current E‑SRF execution. This parameter may be required for certain operations. This parameter will be referred to as the *"entry parameter"* when required.

E-SRF Control Report (SYSPRINT)

The Command Processor provides a control report to list all activity performed by the submitted job. The Command Processor, as well as many other E‑SRF Event Reporting components posts messages and statistical information on this report for your evaluation.

Statistics are provided at the end of the report that may be used to further tune the system or to improve the effectiveness of the commands presented for processing.

SYSPRINT is a required print dataset and must be specified in your executing JCL. Failure to do so will result in an E‑SRF ABEND.

It is strongly recommended that the E‑SRF Control Report be reviewed during any UPDATE operation. Information relating to EXCLUDED events, rolled-off events, and other information is provided for your review and may be helpful in verifying that E‑SRF is processing your journalized security data in accordance with your requirements.

Grouping and Grouping Rules

How grouping is used in the E-SRF Event System:

Grouping and report distribution are very useful and powerful components of this product. However to make use of these capabilities, grouping schemes for your organization must be provided. This task is not trivial, as it will consume time and resource.

You **do not** have to have a grouping scheme to run the E-SRF Event System.

When you are first starting out, it is not necessary to concern yourself with grouping. When (and if) it is desired, you can start implementing your grouping structure.

The E-SRF Event System will make use of the "*EKC Integrated Grouping facility*" (if grouping is desired in any form).

- Grouping is *mandatory* if you are attempting to run reports in a distributed mode.

- Grouping is *optional* if you are running reports that display group names or owner names. If grouping is omitted, the group name columns will be suppressed.

- Grouping is *optional* if you are using group names or owner names in report selection criteria. If grouping is omitted, a warning will appear, and the selection will be ignored.

- Grouping is *optional* if the Update Function Exclude facility is being utilized.

- Grouping is *not required* and may be omitted if group or owner references are not present in the reports, or if running an update, the exclude facility is not being utilized.

The grouping information used to build the grouping structures is maintained by the EKC Integrated Grouping Facility. Data maintained by this facility is stored in a dataset referred to as the RULES dataset.

Please NOTE: Grouping rules and their dataset are NOT the same as ACF2 rules and the ACF2 "rules" dataset. Please do not attempt to use the ACF2 rules for this processing.

Event Reporting uses the EKC Integrated Grouping Facility to assist in the task of building grouping structures suitable for the data contained on your Masterfile, using the E-SRF Event Reporting grouping options you have specified.

As of release 2.1, group structures are stored on the Masterfile. When necessary, the group structure contained on the Masterfile is loaded to afford the capability of grouping functions for your processing requirements.

The E-SRF Event System will make sure the group structures are loaded and in place when the Command Processor encounters a need for them (*such as UPDATE or RUN command*).

Grouping structures are loaded either COLD or HOT. A COLD build means the entire grouping process must be performed. New grouping structures must be constructed and stored on the Masterfile. This can take considerable time. A HOT build means the structures contained on the Masterfile from a previous COLD build will be used which completes almost immediately.

A COLD build will occur if any of the following occur

- If the name of the grouping rules dataset has changed (RULES DD in your JCL).

- If the rules were recompiled.

- If any of your grouping options that could have impact on your grouping structures have changed.

More information of grouping is presented in subsequent topics in this publication, and in other publications you will be referencing. To learn how to actually define your grouping within your organization, please refer to the *Resource Grouping Facility Guide*.

How to Turn Grouping Off:

If you are setting up a job that does not require the use of grouping, you may *TURN GROUPING OFF*.

This was important in releases prior to release 2.1 because the grouping structures were build and loaded "COLD" on every execution. In recent releases, unless you change your grouping rules or options, grouping structures will be loaded "HOT", which makes all of this not really necessary from a performance point of view.

There still may be reasons to turn grouping off, and if grouping is not needed, turning it off does not hurt anything.

This may be accomplished by one of the following methods:

Omitting the //RULES DD statement in the Command Processor's JCL.

This will cause the grouping function to fail and present to the Command Processor a *non-zero* return code. The Event System grouping structure will NOT be built. A brief description of the cause and effect of the non-zero return code is displayed to ensure you are doing exactly what you want. If grouping was not desired anyway, ignore the return code and its narrative.

Specify the following statement in the beginning of the Command Processor's input:

OPTION GROUPING(NONE)

This will allow the Integrated Grouping Facility to initialize itself, but the Event System will not build its Grouping Structures. This will save processing time.

If you want to make the execution run as fast as possible, do BOTH of the items mentioned above.

Masterfile Conversion Upgrade Issues

Your E-SRF Masterfile is created and maintained using software programs contained in the E-SRF Event System program product. From time to time, the product is enhanced. These enhancements could have an effect on the format of the data contained on the Masterfile.

The Masterfile, (like other software components of the product), has its own release level, which could change as a result of either a new product release or maintenance applied to an existing release.

When product enhancements and other maintenance force changes on the Masterfile, the Masterfile release level is upgraded. E-SRF will automatically upgrade the Masterfile for you the first time the new programs are executed against whatever Masterfile version you are currently using.

Conversion is *automatic*. E-SRF determines what type(s) of conversion(s) are required and performs them for you.

This means if you want to run old reports from a previous Masterfile (perhaps archived on tape), you can restore the previous Masterfile to VSAM, and run the request using the new product version.

You cannot however, run prior versions of E-SRF that support Masterfile levels greater than the version of the product was designed to support.

This means the E-SRF Event System is "*upward*" compatible. It is NOT "*downward*" compatible.

The conversion will be dynamically attempted.

Please note: As of release 1.4, in order for Dynamic Upgrade to be carried out, you must "*authenticate*" the automatic request by specifying: **"UPGRADE"** as an entry parameter on the job. This is accomplished by using the "PARM=" specification in your JCL. This was required to keep installations from converting live production Masterfiles while testing new releases of the product, and then later attempting to use the converted Masterfile in production with the prior release of the product.

```
EXEC PGM=ESRFCMD,PARM=UPGRADE
```

Failure to specify the UPGRADE parameter when required will result in the message E490, and the system will be abended. If the parameter is specified when not required, the execution will continue after the E491 warning message is posted on the control report.

The purpose of the UPGRADE entry parameter, as explained, is to prevent you from "*shooting yourself in the foot*" after you apply maintenance to the product that changes structures contained on the Masterfile. Executing this product with the UPGRADE entry parameter specified does not *cause* the UPGRADE to take place; it "*authenticates*" the process. This means it assumes YOU are aware the conversion will occur, and you agree with it by specification of the UPGRADE parameter. If the upgrade is not required, the parameter is not inspected, even if it was specified. A warning may be posted informing you that you requested an UPDRADE when in fact no upgrade was required. You should remove the UPGRADE parameter when it is not required.

Failure to remove the UPGRADE parameter when not required defeats the purpose of requiring.

Operational problems may occur if you specify this parameter when not necessary. Because you may not know until after it is too late that you are upgrading your production Masterfile to a higher Masterfile level, it is possible to get yourself in a situation where you are no longer able to process the Masterfile using the current software in production. When the Masterfile is subsequently used with the prior version of the software, the execution will fail. You will be forced to either restore your Masterfile to a level prior level, or release the new version of the software immediately.

Command Syntax

E-SRF commands are presented to the E-SRF Command Processor (ESRFCMD) in the sequence you wish to process them.

E-SRF commands consist of MAJOR and MINOR commands:

> *Major commands* initiate a particular E-SRF function.

> *Minor commands* provide additional information to augment the major command.

The collection of a single major command, and its related minor commands, comprise a "command-set", that is a single function for E-SRF to carry out. An E-SRF session may consist of one or more command-sets. Command-sets may be specified on a single statement, or be continued on multiple statements. A dash (-) preceded with a blank indicates that the next line is a continuation of the current command-set.

The command input file is SYSIN. The boundaries are positions one through seventy-one. Seventy-two through eighty are ignored. E-SRF reads each SYSIN statement, processes it, and prints the statement and other related data on the E-SRF control report (SYSPRINT). E-SRF reads and scans each input command statement for the first non-blank character, indicating the beginning of the command.

Comment Statements

If the entire statement (that is, positions one through seventy-one) is blank, the statement is printed on the control report and is ignored. Blank statements may be placed in the middle of command-sets. The continuation flow will not be affected.

If the first non-blank character is an asterisk (*), the entire statement is treated as a comment.

If a slash/asterisk (/*) combination is detected, all subsequent data will be treated as comments until an asterisk/slash (*/) is detected.

Comments have no effect on E-SRF processing. The comment statement(s) will be printed on the control report. Comments may be embedded in the middle of command-sets. Comment statements do not have to contain continuation characters.

Command Statements

Commands are specified by either their full keyword text, or enough text to make them unique from other commands within their competing keywords. If a major command were being specified, the only other competing command keyword would be another major command. If a minor command is being specified, the only other competing keywords are other minor commands associated with the major command.

For example, the major commands: UPDATE and UPLOAD, could be abbreviated down to UPD and UPL. Specifying UP would result in an error since UP is not precise enough. Specifying UPDAX would also result in an error because it does not match any command.

Date Formatting, Processing, and Command Specifications

Date formatting is controlled by the DATEFORMAT and CENTURY keywords.

Dates contained in E-SRF are stored independent of any external format and are completely year 2000 compliant.

Years may still be represented in their two-character format if desired. The CENTURY keyword will cause years to appear in their four-character format.

Dates may be formatted on reports in either of two formats:

USA   MM/DD/YY format where:

- MM is the "two character" month (with leading zero).
- DD is the "two character" day (with leading zero).
- YY is the "two character" year (with leading zero). All years from 50 to 99 are considered dates in the twentieth century, and are converted and stored as such. All years from 00 to 49 are considered twenty first century dates and are stored as such. This means the 98, 99, 00, 01, 02,…order is maintained.

INTERNATIONAL   DDMMM/YY format where:

- DD is the "two character" day (with leading zero).
- MMM is the "three character" month name. (JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC).
- YY is the "two character" year (with leading zero). All years from 50 to 99 are considered dates in the twentieth century, and are converted and stored as such. All years from 00 to 49 are considered twenty first century dates and are stored as such. Again, the 98,99,00.01.02…order is maintained.

The date format on reports may be controlled three ways:

1) As a system option using the SET command which provides the DEFAULT setting.
2) Overriding by a specification in a report's RUN command.
3) Overriding by a specification in the OWNER definition.

The determination of proper date formatting starts with the OWNER definition if owners are part of the reporting. If no OWNER specification is found, the RUN specification will be used if provided. If not specified in the RUN command, the system option as specified by the SET command will be used.

Years are normally displayed in abbreviated "two character" format. You may override this format by specifying the CENTURY keyword. This keyword is available as a system wide specification (SET) or may be specified in a report's RUN command. When CENTURY is specified, the date format field is extended from eight to ten characters.

All dates displayed as *CONTROL INFORMATION* are formatted in the selected format, with *FULL FOUR CHARACTER YEARS*.

When entering dates in commands, either format is acceptable, despite what format E-SRF is set to display dates in.

Time of Day Formatting, Processing, and Command Specifications

Time formatting is controlled by the TIMEFORMAT keyword.

Times contained in E-SRF are stored independent of any external format,

Times may be formatted on reports in either of two formats:

STANDARD   HH:MM XM format where:

- HH is the "two character" hour, displayed relative to a twelve-hour clock with no zero hour. Zero hour is treated as 12 belonging to the pervious meridian.
- MM is the minutes relative to the current hour, from zero to fifty-nine.
- XM is the meridian period. AM representing morning, and PM representing evening. Fifteen minutes after midnight would appear as 12:15 AM.

MILITARY   HH:MM:SS format where:

- HH is the "two character" hour, displayed relative to a twenty-four hour clock with a zero hour. Fifteen minutes and thirty seconds after midnight would appear as 00:15:30
- MM is the minutes relative to the current hour, from zero to fifty-nine.
- SS is the seconds relative to the current hour, from zero to fifty-nine.

The format of time of day information on reports may be controlled three ways:

1) As a system option using the SET command which provides the DEFAULT setting.

2) Overriding by a specification in the OWNER definition.

3) Overriding by a specification in a report's RUN command.

The determination of proper time of day formatting starts with the OWNER definition if owners are part of the reporting. If no OWNER specification is found, the RUN specification will be used if provided. If it is not specified in the RUN command, the system option as specified by the SET command will be used.

Major Commands

As mentioned, major commands initiate a unit of work to be performed by E-SRF. A major command initiates a command-set. Major commands stand alone, that is they do not contain variable data directly associated with them. Variable data is provided by minor commands.

The following Major Commands are supported by E-SRF and are documented in this publication in alphabetical order:

ALTER or REPLACE

ASSIGN

CACHE

CONFIGURE

DEBUG

DELETE or REMOVE

INITIALIZE

INSERT or ADD

OPTION

PURGE

RENAME

RETAIN

RUN

SET

SYNCHRONIZE

UPDATE

USERDELETE

Minor Commands

Minor command(s) augment major commands with additional information provided to execute the major command. Minor commands are documented in this publication under the major commands they apply to.

Command-set example

The following command-set consists of the major command "RUN" (run a report), followed by two of its minor commands "REPORT" and "PARM". REPORT identifies the name of the specific E-SRF report overlay to execute. PARM specifies report overlay-related parameter information. The three commands together, a major command plus its two minor commands, comprise the command-set to be processed.

```
RUN        REPORT(ESRFLIST)          PARM(USER)
```

The major command, RUN, and its two minor commands, REPORT and PARM, could either be entered all on a single statement the way shown above, or each on their own statement, with the continuation character (-) anywhere after the command text.

If a second major command is supplied within a command-set, it will be treated as a minor command. This would more than likely result in an error.

Sample Command Processor JCL:

```
1)    //ESRFJOB JOB(accounting),CLASS=A
2)    //*
3)    //ESRF    EXEC PGM=ESRFCMD,REGION=0M
4)    //*
5)    //STEPLIB DD   DSN=ESRF.ESRFLOAD,DISP=SHR
6)    //*
7)    //MASTER  DD   DSN=ESRF.MASTER,DISP=SHR
8)    //*
9)    //RULES   DD   DSN=ESRF.RULEOBJ,DISP=SHR
10)   //*
11)   //SYSPRINT    DD   SYSOUT=*  E-SRF CONTROL REPORT.
12)   //REPORTS DD   SYSOUT=*  E-SRF REPORT DEFAULT OUTPUT.
13)   //SYSUDUMP    DD   SYSOUT=*  PRINT DUMP IF ABEND
14)   //*
15)   //SYSIN   DD   *        COMMAND INPUT STREAM

16)   … E-SRF command input …

17)   /*
```

What Each Control Statement Means

1)   Standard MVS "JOB" statement. Code this to your installation's specifications.

2)   Standard MVS "comment statement".

3)   Standard MVS "EXECUTE" statement. It indicates the name of the program that you wish to execute. In this case, we want to execute the E-SRF Command Processor (ESRFCMD). This program will interpret your commands and interact with the E-SRF system to provide the results. Notice the REGION specification of 0M. E-SRF uses megabytes of storage to run. By specifying 0M, you are telling MVS to unconditionally provide the storage that is needed. If an optional entry parameter is required, place the PARM='*parameter text*' on this statement.

4)   Standard MVS "comment statement".

5)   Standard MVS "DD" statement defining the name of the program library which contains your E-SRF programs and report overlays. This is where the MVS loader looks for programs that are required to be loaded and executed.

6)   Standard MVS "comment statement".

7)   Standard MVS "DD" statement defining the name of the E-SRF Masterfile VSAM cluster. This "DD" is required.

8)   Standard MVS "comment statement".

9)   Standard MVS "DD" statement defining the name of the E-SRF Rule Object dataset. If you <u>do not</u> have "grouping" implemented or are not using the Rule Object dataset, omit this statement.   Expect an error and a warning during E-SRF initialization.   Most grouping functions are not available without this "DD".

10)   Standard MVS "comment statement".

11)   Standard MVS "DD" statement defining where the E-SRF "control report" will go. SYSPRINT is a standard 133-character FBM dataset (or JES output).   This report will contain a chronological log of what E-SRF executed on behalf of your command input and other related matters. The intention of this report is to provide a means for you to review what happened during the E-SRF execution and verify that everything was accomplished the way you expected. This "DD" must be supplied. If it is omitted, an error message followed by an 0C1 ABEND will occur.

12)   Standard MVS "DD" statement defining where E-SRF reports will go by default if not directed elsewhere. 'REPORTS' is a standard 133 character FBM dataset (or JES output). If this DD statement is omitted, it will be dynamically allocated as a SYSOUT dataset.

            See RUN CTRCHAR for more information on other characteristics of this file.

            You may have additional report output "DD" statements for the following reasons:

            You are running report distribution and want to allocate the report dataset yourself.

            You have specifically coded the DDNAME parameter on the RUN command for some of your reports.

13)   Standard MVS "DD" statement defining where MVS writes a storage dump in the event of a program ABEND. This is optional and is rarely used.

14)   Standard MVS "comment statement".

15)   Standard MVS "DD" statement defining where E-SRF will read its command input.

16)   Data statements containing your command input which you want E-SRF to process during the current execution. This is where the major and minor commands will be listed according to the specifications in later sections of this *Command Reference*.

17)   Standard MVS "end of data statement".

Chapter 2:   E-SRF Commands

The Command Processor controls E-SRF function execution. The command syntax is free form with optional values enclosed in parentheses.

A major command directs E-SRF to perform a specific function. That command alone may not provide enough information to properly carry out the function. Additional information is provided in the form of minor commands. These sub-commands have the same format as major commands, but their keywords and values only apply to the major command they are associated with. The collection of a single major command and its optional minor commands form a command-set. A single E-SRF session consists of one or more command-sets.

The Command Processor treats all commands following a major command as minor commands associated with the major command.

To end the current command-set, do not specify the continuation character "-" at the end of the last statement. When the command processor detects the end of the current command-set, it attempts to execute it. No other commands will be processed until the current command-set's function has been completed.

When entering commands, they may be abbreviated down to the point where they are still unique within the processing level. This means if we are specifying a major command, you only have to specify enough characters to make it unique within all other *MAJOR COMMANDS*. Once a major command is determined, any minor commands may be abbreviated enough to make then unique to other *minor commands within the current major command*. In some cases, all that is needed is the first character of the command keyword.

The following pages describe each major command and its subsequent minor commands. Note that minor commands relate to major commands. A specific minor command may have a completely different meaning with each major command that uses it.

To understand the descriptions on the following pages, you should have a basic understanding of the major E-SRF functions and components. For more information about the major functions of E-SRF, consult the section, "*Introduction and Overview*" in the *User Guide*.

For more information about the components of E-SRF including Objects and Segments, refer to the *Data Dictionary Reference*.

This page intentionally left blank

Chapter 3:   ALTER/REPLACE   Object Field Elements(s)

ALTER **or REPLACE**	**GROUP(***group name***)** **OWNER(***owner name***)**

Description:

This command may be used to alter the contents of specific information on the E-SRF Masterfile.

As this product manages security information that must be trusted, most fields cannot be altered. The current implementation of this facility only relates to the GROUP and OWNER segments of the E-SRF Masterfile.

In E-SRF, there are two classifications of field elements. *SINGULAR* elements, meaning each field occurs once within the object, and *ARRAY* elements, meaning a series of fields are repeated multiple times (such as a list of all violations that occurred for a given resource).   This ALTER/REPLACE command affects only SINGULAR field elements.

The minor commands shown below are used to locate particular target E-SRF objects. To actually alter data contained on the object, additional minor command(s) consisting of the E-SRF Data Dictionary "dataname", followed by the desired data value for the particular dataname must also be supplied.

For more information on which fields you can use for GROUP and OWNER, see the sections, "*Group Segment Datanames*" and "*Owner Segment Datanames*", in the *Data Dictionary Reference*.

Minor Commands:

GROUP(*group name***)**   **identify "GROUP" segment group name**

Supply the one to sixteen character target group name. Specification of GROUP indicates that the named GROUP will be the target of this request.

OWNER(*owner name***)**   **identify "OWNER" segment group name**

Supply the one to eight character target owner name.

During report distribution, owner names are used to relate to a report output dataset. The names chosen become the Operating System DD names and must be coded as such. If the owner "ACCTG" were specified, E-SRF will reference and possibly allocate an output report data set called "ACCTG".

Command Example:

```
ALTER      GROUP(PAYROLL) DESC(PAYROLL RESOURCES) OWNER(BETTY)
```

This page intentionally left blank

Chapter 4:   ASSIGN   Assign Domains to Security IMAGES

ASSIGN	**DOMAIN(***Domain ID mask***)** **IMAGE(***Security Image ID***)** **NAME(***assignment name***)** **REMOVE** **SYSTEM(***Domain ID mask***)**

Description:

This command is used to assign one or more Domains (sysids) to a particular Security Image.

The E-SRF Event System may warehouse and process data relative to more than one Resident Security System (RSS) and may do this for multiple "images" of the same RSS.]

First example: You may have several MVS systems (LPARS) that use the same security databases. This means they are logically the same system, the userid "JOHN" is the same user on all of these LPARS, and John's activity is updated on the same data records whether or not John signs on to one or more of these LPARS. In this case, all of these LPARS would be assigned to the same Security Image.

Second example: You have more than one security database, whether the same RSS or not, and John may or may not be the same John across the security databases. In this case, even if they are the same RSS, and they are defined exactly the same, you still should separate them because the information registered within E-SRF will not match the RSS if E-SRF is randomly updated based on the activity across these security databases by John, it will reflect the particular RSS which hosted the last event recorded for John.

All detail activity is stored in E-SRF on data elements belonging to objects. These data elements include the Domain ID. This means activity is and always was separated based on the supplied Domain ID. Users, however, must be separated further. That is, a user in one Image must be independent of a user with the same userid in another image.

The proper use of this command will cause a separate Userid Header (UA) and all of its associated objects to be created for each user for each security image.

Again, using John as an example, assuming John was defined in two separate security databases, on E-SRF, the User segment would contain two sets of objects, a set for John in one image, with activity represented in the domains assigned to that image, and in the second image, with all of that image's assigned domain activity. John is basically two users.

Please note that the use of security images only affects the User Segment of the Masterfile. The other segments remain as they were without this facility. Also, note that it is possible to treat any E-SRF object that has DOMAIN as if they were separate based on IMAGE selecting instead of DOMAIN selecting.

If the E-SRF Update Function detects transactions for a Domain that is not assigned to an IMAGE, the unmatched transactions are NOT applied to the Masterfile. There are statistics on the Control Report that provide information about this condition. E-SRF does not consider this an error, because it is possible that the same update input data set may be presented to multiple update runs across multiple E-SRF Masterfiles.

You may have more than one ASSIGN command for a single Security Image assigning separate domains to a single image.

For more information on DOMAINS and Security IMAGES, please see the *User Guide.*

Minor Commands:

DOMAIN(*Domain ID mask***)**   **Domains (sysids) to be assigned**

Supply the one to eight characters Domain name to be assigned to this security Image. This specification may or may not be masked. Assignments are made from most specific to least specific.

This is the same specification as the SYSTEM specification described below.

IMAGE(*Image ID***)**   **Identifies name of Security Image**

Supply the one to eight character target Image ID.

This is the name of the Security Image that you wish to assign the specified Domain(s) to. The Image must already be Configured on the Masterfile.

NAME(*name text***)**   **Assignment name information**

Supply the one to twenty character assignment name information.

This specification is optional, if omitted the Assignment name will be blank.

This information is stored on the Masterfile and is used for display purposes. Name information is recommended so individuals reviewing security Image and Domain information may have a better understanding of what each assignment represents.

For this reason, it is recommended that Domain ID masking not be used.

REMOVE   **Remove named assignment**

There are no parameters associated with this minor command.

The specified domain name (or mask) will be removed from the IMAGE structures. If remove is issued against an active domain, the ESRF will respond by associating the domain(s) involved with the next logically matched IMAGE. If there are no domain names (or masks) that match, the domain will be considered unmatched and will be process as such.

Care must be exercised when removing domain assignments, and is normally performed to correct errors in previous assignments.

SYSTEM(*Domain ID mask***)**   **SYSids (Domains) to be assigned**

This specification is identical to DOMAIN and is present for compatibility purposes.

Domains and SYSids mean the same thing in E-SRF. This is the same specification as the DOMAIN specification described above.

Command Example:

```
ASSIGN      DOMAIN(SYSX) IMAGE(CHICAGO) NAME(DEVELOPMENT SYSTEM)
ASSIGN      DOMAIN(SYSY) IMAGE(CHICAGO) NAME(TECH SUPPORT SYSTEM)
ASSIGN      DOMAIN(CHG*) IMAGE(CHICAGO) NAME(PRODUCTION)
ASSIGN      DOMAIN(NY01) IMAGE(NEWYORK) NAME(NEWYORK ONLINE)
ASSIGN      DOMAIN(NY02) IMAGE(NEWYORK) NAME(NEWYORK PRODUCTION)
ASSIGN      DOMAIN(SYSP) IMAGE(CHICAGO) NAME(NEWYORK TEST)
```

In this example, we have two separate processing centers, one in Chicago and one in New York. These centers may or may not be using the same RSS, but the same RSS is being used for all LPARs confined to the center. Additionally, all LPARs are using the SAME SECURITY DATABASES for their processing, meaning as far as the security system is concerned, the LPARS are as one.

We want to make sure all user activity from SYSX, SYSY and all LPARs with a SYSID that start with CHG (CHG1, CHG2, etc.) gets applied to users in the CHICAGO IMAGE.

Additionally, we want all user activity from NY01, NY02 and SYSP to be applied to users in the NEWYORK IMAGE.

The E-SRF Update Function will reject any other activity that enters the system that does not match these DOMAINS named.

The ASSIGN commands will provide the necessary routing of user activity as described above. Remember, everything is broken down at the DOMAIN level. This just separates user activity to preserve user identity and allow multiple views or users based on the RSS as well as the actual security database associated to the target IMAGE.

Note: Target IMAGES must have already been defined (configured) on the E-SRF Masterfile.

Removing a single domain assignment:

It may be desirable to remove a previously defined domain assignment defined on the Masterfile. If this is desired, specify the targeted DOMAIN ID and IMAGE ID as for the *remove* request as shown below.

```
ASSIGN      DOMAIN(SYSX)      IMAGE(CHICAGO)   REMOVE
```


Removing all domain assignments for a specific IMAGE:

It may be desirable to remove all domain assignments targeting a specific image defined on the Masterfile. If this is desired, specify the targeted IMAGE ID and the constant **!ALL** as the requested domain ID for the *remove* request as shown below.

```
ASSIGN      IMAGE(CHICAGO)   DOMAIN(!ALL)      REMOVE
```


Removing all domain assignments for all images:

It may be desirable to remove all domain assignments for all images defined on the Masterfile.. If this is desired, specify !ALL as the image ID for the *remove* request as shown below.

```
ASSIGN      IMAGE(!ALL)                REMOVE
```

This page intentionally left blank

Chapter 5: CACHE   Establish E-SRF Caching Options

CACHE	**OFF** **ON** **UPGRADE**

Description:

This command specifies what (if any) cache options are desired. The use of caching may have dramatic positive effect on execution performance, especially during the execution of an UPDATE command and in the report distribution environment.

Caching is the act of loading the entire contents of the E-SRF Masterfile's VSAM cluster into main storage. A high performance table management routine will be used to directly access or browse any object, just as if it were accessing the VSAM Cluster, but much faster.

A cache is built by logically loading the entire data component of the Masterfile into storage. This cache may be referred to as the "level-two" cache.

Additionally, due to the overhead associated in the data *"compress-decompress"* process, fifteen high performance look-aside buffers are allocated which will be used to "quick-cell" the most recent objects which were directly read. These buffers will maintain objects <u>decompressed</u> and ready for processing. In an update situation, up to fifty percent, and sometimes much more of all direct reads and writes are accessed from these buffers. These buffers are referred to as the "*level-one*" cache.

The level-one cache only exists if the level-two cache is active.

Storage for the Level-two cache resides on data-only address spaces.

Normally, the term 'cache' refers to both caching schemes collectively.

The Decision to Use Cache:

The cache uses quite a lot of main storage most of which is in data-only address spaces. The index to the level-two cache, as well as the level-1 cache is located in the PRIMARY address space.

There are several issues to consider. If it takes fifteen minutes to update E-SRF using the cache, and several hours without the cache, using the storage for fifteen minutes (or less) may be a good choice. In recent releases of MVS, memory considerations have become secondary to real-time performance.

The decision is based on what you intend to do in your E-SRF session. If you are executing an UPDATE command, the cache should be turned on. An exception would be you are only updating a small percentage (1-5%) of the records on the Masterfile.

For report processing, it depends on how many "trips" through the Masterfile may be required to produce the required reports. Assuming the worst, a full trip for each report, several reports would have to be run to "break even" using the cache. It takes one trip through the Masterfile to build the cache, plus the overhead of managing the storage. If you are running reports under report distribution, a single report may run many times to produce the compliment of reports required for a single report execution. In this situation, using the cache may prove very beneficial.

When you are running reports, especially when doing report distribution, we recommend you run as many reports as you can in a single batch job with the cache turned on.

Notes about the CACHE:

The cache is only built when requested. The default is NOT TO BUILD it.

The level-one cache will always reside in the primary address space (that is the address space hosting the E-SRF execution). The size of the level-one cache is dependent on how big the largest object uncompressed object that resides on the Masterfile is. If you have an object that is three million bytes long, than it would be the product of three million times fifteen (forty-five million bytes).

As of release 1.6, the level 2 cache is normally maintained in a Data-Only Address Space (unless overridden by 'OPTION POOL(LOCAL)'. The level-two cache's index is always maintained in the primary address space. In release 2.1, the level-two cache may reside in multiple data-only address spaces. The level-two cache is allocated out of E-SRF POOLED storage. It will default to Data-Only Address Space mode because that is where POOLED storage normally resides

You may have some problems with possible site restrictions that may be in place for the use of data-Only Address Spaces in application programs. E-SRF Event Reporting runs as a standard application program that is not APF authorized. (*APF authorized programs normally ignore site restrictions and execute as such*). Non-APF authorized programs normally must adhere to the restrictions. If this is the case, you should contact your systems programmer and see what you have to do to "condition" your job to run without these restrictions.

The use of the Data-Only Address Space is only slightly more efficient that using the primary hosting address space. It was provided mainly to deal with memory constraints that may exist with very large Masterfiles are maintained. If you cannot use a Data-Only Address Space, and your jobs seem to fit in the standard MVS 2.2 GB address space, you can run that way without penalty.

To use local storage for your cache, see the discussion relating to: `OPTION POOL(LOCAL)`

VSAM PERFORMANCE

When using the cache, if ten percent of the total objects contained on the Masterfile, or 1000 objects (which ever is smaller), the VSAM cluster is REBUILT instead of just updated. This means the VSAM cluster may never have to be rebuilt by you because of Control Area and Control Interval splits. This also means you may allocate your cluster with ZERO FREESPACE. This will reduce the amount of physical DISK space required to contain your cluster.

This *does not* however remove the necessity to backup your VSAM cluster, or review the amount of DISK space used by the cluster to prevent the cluster from filling up and failing when objects are added, extended or upgraded by a cache reload.

Remember: **The cache is a performance improvement, you do not get any more or less functionality whether it is used or not.**

Default Cache Operation:

When the E-SRF command processor is entered for executing, the caching facility is off. If you wish to use it, you must turn it on in each job with the commands shown below.

Minor Commands:

AUTOEXPAND   **Turn on cache "*autoexpand*" feature**

When cache memory is allocated to a particular Masterfile Object, the exact storage requirement is obtained and provisioned for its use.

If an object is shortened, the residual storage is retained. If the object is lengthened, a new storage area must be acquired large enough to accommodate the entire object.

If AUTOEXPAND is set, additional storage is added to the base additional requirement in the anticipation that the object will be lengthened during subsequent processing, and added to the new storage length requirement.

The use of this option will reduce Central Processing Unit (CPU) time, but increase storage consumption. This option should only be used if you are concerned about CPU time, and storage consumption is not an issue.

Note that unless you are updating your Masterfile, the use of this option would render no benefit, and at the same time, consume more level-two cache storage.

OFF   Turn off Masterfile caching

The existing cache (if present) will be used to upgrade the E-SRF VSAM cluster with any object that was altered during the life of the cache.

From this point on, all database requests will be satisfied by directly accessing the VSAM Cluster.

Please note that when the E-SRF session terminates and the cache is on, this function will automatically be performed.

ON   Turn on Masterfile caching

A cache will be built consisting of all E-SRF objects contained on the E-SRF VSAM cluster. From this point on, all requests to read or write to E-SRF objects will use the Cache instead of accessing the Masterfile.

UPGRADE   Upgrade Masterfile VSAM cluster with current cache

The existing cache will be used to upgrade the E-SRF VSAM cluster with any object that was altered during the life of the cache.

After the upgrade, the cache will still remain in effect. Subsequent UPGRADE commands will result in upgrading only new changes.

UPGRADE will "update" the existing VSAM cluster. If more than ten percent of the E-SRF objects were altered, the entire VSAM cluster will automatically be rebuilt.

Command Example:

```
CACHE      ON
CACHE      UPGRADE
CACHE      OFF
```

The first example will initiate the Masterfile cache.

The second will cause the current Masterfile cache to be applied to the Masterfile without shutting down the cache. The cache will still be present and be continued to used to access the objects contained on the Masterfile

The third example will do the same processing as the second example, but additionally, will disable the use of the cache and release all resources currently being used by the cache.

Please note that is not necessary to shut down the cache. When the Command Processor terminates normally, a cache upgrade is automatically performed.

Possible storage "GETMAIN" ABENDS

Using the CACHE requires a considerable amount of virtual processor storage. Most MVS systems still, with good reason track virtual storage usage, and limit the amount of virtual storage allotted to any given address space. This can cause E‑SRF, when using the CACHE option, to be abended with a storage related ABEND code.

To correct this situation, you will have to find out what conventions are required to permit a particular job to be able to acquire whatever virtual storage is needed. The only other option would be to refrain from using the cache, which will cause enormous performance issues during the Update and possibly during Report Distribution processing.

The act of limiting virtual storage originally was done due to the expense of the actual memory user in mainframe systems. Today, it is used to control the mix of jobs running in a given processor workload. The objective of limiting virtual storage would allow a good mix of jobs sharing the real processor storage attached to the computer system. Once your storage usage exceeds the real storage attached to your system, you begin to use auxiliary storage (possibly disk drives) to contain your immediate storage demand. When this occurs, it is referred to as "paging" and it will slow down your system.

System programmers have solved this problem by attaching enough processor storage to their systems what would accommodate a "typical" workload, and making sure that workload is seldom exceeded. Part of this workload management is keeping the aggregate virtual storage requirement in line with what was planned. This is the reason your E‑SRF job (or any other job) will ABEND for storage related reasons.

To fix this, you have to find out what type of parameters you need to allow your job to run. Once this is established, alter your JCL statements as required and re-run the job. There may only be a single execution "class" for it to run in, with a queue of jobs waiting to execute, or you may have to schedule it to run at a certain time, but it will run. You may have to contact your system programmer to find out exactly what you have to do. This procedure is different in every installation, and it must be determined individually. The use of "REGION=0M" on the job or execute JCL statement usually fixes it, but sometimes it is overridden by installation defined overrides and exits.

Should you, or your systems programmer have any further questions, or want to discuss more on this topic, you should contact EKC Technical Support for assistance.

Possible ABENDS while the cache is on

One of the nice things about using the cache is that even though the Masterfile's VSAM Cluster may be open for UPDATE, it is not physically updated until the cache is UPGRADED to the Masterfile itself. If the job should ABEND for any reason, the VSAM Cluster itself is still in tact; at the state it was in when the cache was last built.

Please note that the cache is only automatically upgraded to the Masterfile if data contained in the cache was changed. If the cache was on for a job that only produced reports, there is no consideration required. The Masterfile will always be as it was before the ABEND.

The VSAM Cluster will only be bad if the ABEND took place while upgrading the cache.

The cache upgrade will automatically set the high used RBA (Relative Byte Address) to zero and begin "loading" the cluster with data (in the same fashion that a IDCAMS REPRO would be used to rebuild the cluster). If an ABEND occurs at this point, the cluster will contain all the E-SRF objects in sequence up to the point where the ABEND occurred. ***THE REMAINDER OF THE MASTERFILE IS LOST.***

The normal reason for an ABEND at this point is *RUNNING OUT OF VSAM CLUSTER DISK SPACE*. Other reasons include operator cancel or elapsed time has exceeded limits imposed on the job.

The cache UPGRADE does not take much time to process, but is a serious window for VASM file corruption. If the cache were not used, that window would be extended to the entire update execution.

The issue with disk storage access methods, VSAM and abnormal job termination, or insufficient disk storage causing end user problems exists in almost any system. E-SRF does not impose any additional exposures, and in some cases actually reduces your exposures by shortening down this window and automatically reorganizing the VSAM cluster after each update to make sure your disk space is properly utilized with little or no fragmentation.

This page intentionally left blank

Chapter 6:   CONFIGURE   Define a Security Image to E-SRF

CONFIGURE	DDNAME(*RSS parameter dataset*)
	IMAGE(*security IMAGE ID*)
	LOADLIB(*Load Module library ddname*)
	NAME(*Image name*)
	RESET/<u>NORESET</u>
	PARAMETERS(*parameter data*)
	RSS(Resident Security System ID)

Description:

E-SRF will process event data only if it knows which security Image it relates to.

The ASSIGN command relates DOMAINS (sysids) to IMAGES.

This command actually defines the Security Images to E-SRF. A Security Image must know exactly how E-SRF is to process event data. This information is *learned* by this command. Information is supplied by the specifications in this command plus additional information that is acquired from the named RSS itself is stored on the Masterfile.

Note that the specifications needed to run configure a specific image will be Resident Security System (RSS) specific and will vary according to the particular RSS being configured.

The examples and explanations will be for ACF2. If you are configuring an image for ACF2 or another supported RSS, please consult the E-SRF event system documentation for the specific RSS being configured.

Minor Commands:

DDNAME(*ddname*)   **RSS parameter external dataset**

DDNAME of external dataset produced by running ESRFxxxP against a specific RSS to obtain information E-SRF needs about the particular RSS.

The DDNAME must be defined to a dataset in the JCL executing this command.

In the case of ACF2, if this command is omitted, the ACF2 configuration will look at the ACF2 control block structure that exists in the MVS system hosting the execution of this parameter. For ACF2, this is the first choice if multiple images are to be configured.

In the case of RACF, you must have this dataset to perform the configuration.

Please refer to the documentation specific to the particular RSS you are attempting to configure to determine how this dataset is created.

Information on how to run ESRFxxxP is presented at the end of this command section.

IMAGE(*Image ID***)**   **Security Image ID**

Supply the one to eight character Security Image ID. This ID must be unique. This is the ID that is specified in the Domain ASSIGN command to relate Domains to Images.

LOADLIB(*ddname***)**   **RSS parameter module load library**

DDNAME of the RSS load library containing information E-SRF will look at to configure the RSS to the target Image.

n the case of ACF2, if the DDNAME specification was omitted, you have the ability point to a load library specified by this parameter, that contains the RSS information. This means you can configure an IMAGE that is that is NOT being used on the MVS system hosting our execution. In a multiple Image environment, this is the only alternative if the RSS definitions are different (UID string or LIDREC format, etc.),

In the case of RACF, this specification is ignored.

NAME(*image text***)**   **Name of image to be configured**

Supply the one to twenty characters Security Image name information.

This specification is optional, if omitted the Security Image name will be blank.

This information is stored on the Masterfile and is used for display purposes. Name information is recommended so individuals reviewing Security Image and Domain information may have a better understanding of what each assignment represents.

RESET / NORESET   **Existing Image reset option**

This specification determines whether or not you are defining a new Security Image or redefining an existing one.

This will keep you from shooting yourself in the foot if you are attempting to define a new Security Image and in error name it one that you have already defined.

If you wish to redefine a particular Security Image either because the RSS has changed or you have converted to a different RSS, special care must be assumed. ***If this is the case, please contact EKC Technical Support for the proper information and sequence of events.***

PARAMETERS(*data***)**   **Configuration parameters**

Supply the one to thirty two character Security Image parameters that may be required. These parameters are stored on the Masterfile associated with the IMAGE you are configuring.

As of this release, there are no required update parameters associated with any supported RSS.

Please refer to the documentation specific to the particular RSS you are attempting to configure to determine what parameters (if any) may be required for proper configuration.

RSS(*RSS ID***)**   **Resident Security System ID**

Supply the identification of the RSS you are configuring this Security Image for. In this release of E-SRF, the only valid RSS ID are ACF2 and RACF.

Command Example:

```
CONFIGURE IMAGE(CHICAGO)       -
        RSS(ACF2)              -
        DDNAME(CHGO)           -
        NAME(CHICAGO OPERATIONS)


CONFIGURE IMAGE(NEWYORK)       -
        RSS(ACF2)              -
        DDNAME(NY)             -
        NAME(NEW YORK OPERATIONS)


CONFIGURE IMAGE(DALLAS)        -
        RSS(RACF)              -
        DDNAME(DALLAS)         -
        NAME(DALLAS OPERATIONS)
```

You will notice we have three individual IMAGES, one in New York, one in Chicago and one in Dallas. New York and Chicago both use ACF2, with different Logonid database formats. The system in Dallas is running RACF. All three systems will use the same Masterfile, and reporting will reflect all three systems.

The first command assumes the RSS information on the LPAR running this configure command is appropriate for the Security Image being defined (E-SRF itself is being run in Chicago). The ACF2 configuration program will look at the running ACF2 system and obtain its parameters directly from it.

The second configuration is going to use the output from the execution of ESRFACFP on the machine in New York. The data was transmitted from New York to Chicago, and will be used to configure the New York Security Image (NEWYORK) on the machine in Chicago.

The third configuration is going to use the output from the execution of ESRFRACP on the machine in Dallas. The data was transmitted from Dallas to Chicago, and will be used to configure the Dallas Security Image (DALLAS) on the machine in Chicago.

Note: If both NEWYORK and CHICAGO were running the same RSS with the same UID and Logonid record formats; both configurations could have been run using the RSS in Chicago (omit the DDNAME and the execution of in New York).

Running the ESRF*xxx*P program (obtain RSS operational parameters).

E-SRF supplies an off-line batch utility to create a file suitable for input to the IMAGE CONFIGURE command. This program is called the RSS parameter file program. The name of the program is ESRF *xxx* P. The *xxx* represents the name of the RSS being configured.

In the case of ACF2, it would be ACF. In the case of RACF, it would be RAC.

The "P" indicates the parameter acquisition program. The name of the parameter acquisition program is ESRFACFP for ACF2, and ESRFRACP for RACF.

This output of this program is a file that contains all the information relative to a specific RSS as it exists on the system running the program. CONFIGURE will use this data to build a Security IMAGE on the E-SRF Masterfile.

If you are not able to CONFIGURE a particular IMAGE from the system running the E-SRF Command Processor, or the particular RSS is not dynamically configurable, YOU MUST RUN THIS PROGRAM before running your CONFIGURE command.

The output file contains parameter information in a syntax that is not intended to be user modified, so therefore is not documented in this guide. An exception to this would be the ability to edit the RSS dictionary entries to assign more meaningful names to specific data elements. The keyword is DESC and the text is from one to thirty-six characters. Do not alter other any other data.

Sample JCL to run ESRFACFP (ACF2) parameter acquisition program)

Sample JCL to run ESRFRACP (RACF) parameter acquisition program)

```
1) //ESRFJOB    JOB  (),CLASS=A,MSGCLASS=A,MSGLEVEL(1,1)
2) //ESRF       EXEC PGM=ESRFACFP or PGM=ESRFRACP
3) //STEPLIB    DD   DSN=SYS1.ESRF.ESRFLOAD,DISP=SHR
4) //PARMOUT    DD   DSN=ESRF.PARMOUT,DISP=SHR
5) //SYSPRINT   DD   SYSOUT=*
```

What each control statement means

1)   Standard MVS "JOB" statement. Code this to your installation's specifications.

2)   Standard MVS "EXECUTE" statement. It indicates the name of the program that you wish to execute. (ESRFACFP is for ACF2, ESRFRACP is for RACF).

3)   Step library that contains the E-SRF product load modules.

4)   PARMOUT is a 255 FIXED length dataset that will contain the parameter information extracted from the RSS running on the system hosting this execution. This file will become input to the E-SRF IMAGE CONFIGURATION for the IMAGE being configured.

5)   SYSPRINT is a status and error report that is produced by the program. It should be looked over before continuing with the configuration.

Chapter 7: DEBUG   Activate and Control E-SRF Debugging

THIS COMMAND SHOULD ONLY BE ACTIVATED UNDER DIRECTION OF EKC TECHNICAL SUPPORT.

DEBUG	**ABEND(***argument***) 0C1(***argument***)** **EXTERNAL(***ddname***)** **GCP** **CTLCHAR** **LIMIT(***number***)** **MCP** **MESSAGES** **OFF** **RSP** **SCP** **TOR** **TRACE(***trace controls***)** **UASINGLE** **UCP** **UPDATE** **USERID(***userid***)** **XGP**

Description:

E-SRF consists of many assembler language management modules that are necessary to support a product of this type. Because of their complexity, an internal debugging facility was incorporated in the product's design. This facility may be activated to help diagnose problems that cannot be analyzed any other way.

It may be unpredictable what will happen if you turn this facility on without direction. Different components of E-SRF do different things when in DEBUG mode.

EKC hopes users will never have to initiate any of these commands, but the need to document this facility exists in the event that problems occur in which no other alternative exists for proper problem determination and resolution.

Minor Commands:

ABEND(*argument***)**
 or 0C1(*argument***)**   **Establish program abend at E-SRF trace point**

This command will trap all calls made to the trace control program. The calling routine names, as well as other trace particulars are examined for a match to the eight-character argument.

If the argument matches, an 0C1 ABEND is forced.

At this point, a dump could be taken, or if running under a debugging environment such as TSO test, an interactive dialog may commence.

CACHE(*options***)**   **Turn on CACHE debugging options**

This minor command turns on the Masterfile caching facility debugging options.

    RESIDUAL   **Display CACHE residual storage statistics**

Post number of bytes for each object that was left unused when the cache is deactivated. A line will be present which identifies the object, and the number of remaining unused bytes.

CTLCHAR / <u>NOCTLCHAR</u>   **Control Report Output CTLCHAR debugging facility**

This command is used to debug final report print output processing controlled by the CTLCHAR RUN specification.

EXTERNAL(*ddname of external trace data set***)**   **Turn on EXTERNAL trace**

This minor command turns on EXTERNAL trace, and assigns a DDNAME for trace data collection. Trace output consists of eighty byte records.

For this option to take effect, E-SRF INTERNAL TRACE MUST BE ON. You can turn trace on and set this option in the same command.

All events traced by the trace control program will be written to the external trace data set. This could generate thousands of records. The number or records depends on when trace was turned on, what subsequent functions were executed and how long it was left on.

GCP / <u>NOGCP</u>   **Control Group Control bugging facility**

This command is used to debug E-SRF Group Control Program (GCP).

LIMIT(*number***)**   **Debugging event limit**

This command is used to establish a limit on how many debugging events take place for a particular event. The numeric value specified will act as a threshold before the debugging event is suppressed.

If this specification is omitted, there will be no threshold on the particular debugging event.

LINECOUNTER   **Turn on "***linecount***" display**

Report line positioning as managed by tracking the current line number relative to the start of the current page. Decisions as to where lines are to be placed, as well as the beginning of the next report page are controlled by the report's line counter.

This specification will cause Report Services to place the current line counter followed by a blank character on the first four print positions of every printed report line.

MCP / <u>NOMCP</u>   **Control Masterfile Control Program debugging facility**

This option is used to debug E-SRF Masterfile control problems.

MESSAGES   **Display all E-SRF messages**

This minor command calls the Message Formatting Program (MFP) to produce a list of all messages produced by E-SRF.

OFF   **Turn off debugging facility**

This command turns off all debugging options that may have been previously set.

RSP / <u>NORSP</u>   **Control report services debugging facility**

This option is used to debug E-SRF report services control problems.

SCP / <u>NOSCP</u>   **Control storage control debugging facility**

This option is used to debug E-SRF storage control problems.

TOR / <u>NOTOR</u>   **Control Token Object Resolution (TOR) debugging facility**

This option is used to debug E-SRF Masterfile tokenization processing facility.

TRACE(*trace controls*)   **Establish E-SRF tracing controls**

This specification is used to establish the presence of the E_SRF tracing facility.

This specification in conjunction with the EXTERNAL specification make it possible to trace the internal programming events that take place within the E-SRF Event system.

The trace facility is only intended to be used for debugging problems under the direction of the EKC Technical Support staff.

    OFF   **Turn off the E-SRF Event System trace facility**

This command turns E-SRF trace facility off (default trace operations).

The current trace table will be left in tact, but will no longer be updated. In the event of a controlled ABEND, the trace table will not be printed out. It will exist, however, for dump analysis and reflect the system as it was just before trace was turned off.

    ON   **Turn on the E-SRF Event System trace facility**

The above command turns on the trace facility. If the EXTERNAL specification was made, the events will also go to the dataset indicated by the specified file DDNAME.

Tracing will continue until it is turned off.

UCP / <u>NOUCP</u>   **Control update control debugging facility**

This option is used to debug E-SRF update control problems.

UASINGLE   **Turn on User Header (UA) single field update debugging**

This option is used to debug E-SRF update control problems dealing with single User Header field updating. This only applies to RSS updating that treats each individual field as a separate data object and the various problems and issues relating to that level of processing.

UPDATE / <u>NOUPDATE</u>   **Control Update Overlay debugging facility**

This minor command is used to debug E-SRF RSS UPDATE overlay problems.

USERID(*userid*)   **Debug specific userid**

This minor command is used to augment existing E-SRF RSS UPDATE debug specifications in an effort to debug UPDATE overlay problems.

XGP / <u>NOXGP</u>   **Control External Grouping interface debugging facility**

This minor command is used to debug E-SRF External Grouping interface UPDATE problems.

Chapter 8:   INITIALIZE   E-SRF Masterfile control options

INITIALIZE	NORESET
	RESET

Description:

This command verifies that the E-SRF Masterfile control segment is in order. Its scope is limited to possible adjustments to the control segment only. All other data on the Masterfile is left untouched.

This function should be executed when:

- A new E-SRF database VSAM cluster (E-SRF Masterfile) is being established.

- You have a need to rebuild the E-SRF "System" Data Dictionary.

- If EKC technical support recommends you do so.

The following functions are performed:

- The necessary E-SRF control objects are placed in the control segment if not already there.

- An E-SRF minimum data dictionary is established if there was not one already in existence. This dictionary will map all current E-SRF data fields contained on the Masterfile.

The following functions are NOT performed:

- The existing non-control (data) objects are *NOT* removed or altered. Redefining a new cluster and then running this command accomplish this.

- There is no dictionary mapping of any Resident Security System (RSS) specific fields. Adding RSS fields is performed with the CONFIGURE command for the specific RSS required for processing. Each IMAGE contains its own image wide data dictionary, which was placed on the Masterfile during the CONFIGURATION that was executed for a particular IMAGE.

Minor Commands:

NORESET   **DO NOT modify the E-SRF "system" Data Dictionary**

If the E-SRF Data Dictionary already exists, it will not be reset (*default*)

RESET   **RESET the existing E-SRF "system" Data Dictionary**

The E-SRF System Data Dictionary will be created and will either be added, if not already there, or upgrade the one that currently exists.

This page intentionally left blank

Chapter 9: INSERT/ADD Object Field Elements(s)

INSERT **or ADD**	**GROUP(group name)** **MODEL(model name)** **OWNER(owner name)**

Description:

This command may be used to alter the contents of specific information on the E-SRF Masterfile. As this product manages security information that must be trusted, most fields cannot be altered. This function relates only to the GROUP and OWNER segments of the E-SRF Masterfile.

In E-SRF, there are two classifications of field elements. *SINGULAR* elements, meaning each field occurs once within the object, and *ARRAY* elements, meaning a series of fields are repeated multiple times (such as a list of all violations that occurred for a given resource). This INSERT/ADD command only affects SINGULAR field elements.

The minor commands below identify a particular E-SRF object to be inserted. To actually insert data, additional minor command(s) consisting of the E-SRF Data Dictionary "dataname" and a value are used. Alternately, specify the INSERT command without any datanames, followed by a REPLACE command with the datanames. If the data already exists, the INSERT will fail, but the REPLACE will apply the desired data elements on the object.

Minor Commands:

GROUP(*group name*)   **Identify "GROUP" name**

Supply the one to sixteen character target group name. Specification of GROUP indicates that the named GROUP will be the target of this request.

MODEL(*model name*)   **Specify model for cloning**

Supply the model name of the object that will be used as a TEMPLATE to facilitate cloning of other objects. The model name must exist within the same SEGMENT, and be of the same TYPE. The model data is applied PRIOR to any of the data specified explicitly by this command.

OWNER(*owner name*)   **Identify "OWNER" owner name**

Supply the one to eight character target owner name. Specification of OWNER indicates that the named OWNER will be the target of this request. Note that owner names become the DDNAME of the owner's distributed report.

Command Example:

```
INS  GROUP(ACCTPAY) MODEL(PAYROLL) DESC(ACCOUNTS PAYABLE) -
                    OWNER(JAMES) -
                    PARTY1(BETTY)
```

The group header ACCTPAY would be added to the E-SRF Masterfile, modeled after the existing group PAYROLL overridden by the group OWNER "JAMES" with the OWNER "BETTY" as an interested party (1).

Chapter 10: OPTION   Current Execution Options

OPTION	GROUPING(*grouping option*)
	LINEMODE
	POOL(*pooled storage override*)
	RELATIVEDATE(*relative date override*)
	SYSTEMDATE(*system date override)*

Description:

This command sets the processing options for the current E-SRF execution. This information is not saved for any subsequent runs.

Minor Commands:

GROUPING (*grouping option*) Specify "*GROUPING*" options

This option controls the way group structures are dynamically built for the current E-SRF execution.

As mentioned before, all grouping in E-SRF is done dynamically upon each execution. This consumes time and computer resources. This option allows you to reduce the overhead by limiting the scope of group structures for the current execution.

CLASS   Provide grouping by classname

This option builds all group structures, but assigns all group names to E-SRF.*classname*' where '*classname*' is the resource class name.

COLD   Force grouping Structure COLD start

Specify COLD if you want to force a cold start on your grouping structures regardless of whether the system deems this necessary or not.

Normally, the determination to do a COLD start is automatically performed based on certain criteria being met.

Default operation is to let the system decide.

EXTERNAL   Use external grouping   (Default option)

All group structures will be built meaning all required resources are grouped according an external grouping facility. E-SRF currently uses the EKC Resource Grouping facility as its external grouping facility.

For more information on the EKC Resource Grouping Facility, and how to use it, see the *Resource Grouping Facility Guide*.

NONE   Turn off all grouping

This option suppresses all grouping. If any reference to group names are made, the group name $ESRF.NOGROUPING will be presented as the group name.

NOSOURCE	**Turn off SOURCE grouping**

This option will cause SOURCE grouping to be suppressed. No SOURCE grouping structures will be built. The end effect is the same as NONE for SOURCE grouping.

NOUSER	**Turn off USER grouping**

This option will cause USERID grouping to be suppressed. No USERID grouping structures will be built. The end effect is the same as NONE for USERID grouping.

SOURCE	**Turn SOURCE grouping back on**

This option will cause SOURCE grouping to be re-established for a particular execution. This option is not used unless it is determined that source grouping is required, after it was previously turned off by the OPTION GROUPING(NOSOURCE) command.

USER	**Turn USER grouping back on**

This option will cause USERID grouping to be re-established for a particular execution. This option is not used unless it is determined that userid grouping is required, after it was previously turned off by the OPTION GROUPING(NOUSER) command.

LINEMODE(*linemode specification*) Establish reporting LINEMODE for current execution

This option overrides the current LINEMODE setting controlled by system SET options.

All reports, unless specifically specified, will inherit this LINEMODE setting.

For further information how and when to use this specification, please refer to the LINEMODE description provided in the chapter discussing the RUN statement.

POOL(*pool storage override*) Establish where "pooled storage" will reside.

Control where "Pooled Storage" will reside during an E-SRF execution.

Pooled storage is a storage management facility that was developed to enhance performance and is unique to this product.

The Masterfile's level-two CACHE facility is the primary user of pooled storage.

EXTERNAL	**Use external (secondary data only) address space**

This option (*which is the default specification*) will cause E-SRF to create a Data-Only Address Space the first time pooled storage is required, and process subsequent pooled storage requests from the created Data-Only Address Space.

This option frees up storage in the primary address space by satisfying all pooled storage requests from a dedicated address space.

This is the option of choice and should always be used unless your site has restrictions in place that prohibit (or restrict) the use of a Data-Only Address Space by standard (non APF authorized) application programs that cannot be overridden.

LOCAL   **Use local address space.**

This option forces all pooled storage to be allocated on demand from the local (home) address space (which was the only processing option for pooled storage in previous releases).

This option is not recommended unless site restrictions make it the only choice available. It will restrict memory usage to 2.2 GB.

Most applications that make use of Data-Only Address Space processing are APF authorized and therefore bypass all restrictions. Restrictions are normally in place to control application programs from causing problems by abusing the facility.

BEFORE USING "LOCAL", please consult your system programmer, who may have information as to how you can condition your job so you can avoid this situation.

RELATIVEDATE(*relative date override***)**   **Alter RELATIVE base date**

This option overrides the current date used as a "relative" base point when running reports that have relative date specifications in their RUN commands.

This command is considered a special use command.

One very common usage is a rerun situation. Someone lost reports, or they were not run for a particular reason.

Another may be during testing, when you want a controlled environment. A special "test" Masterfile is created, possibly copied from your "production" Masterfile.

You want to set up some reports, perhaps testing your grouping for report distribution. You have a set of reports with dates specified via WHEN statements, which may be relative to the "current date". This command will set that "current date" to what ever you want it to be. This allows you to test the same data over multiple days, with the exact same data available for reporting.

Review the 'RUN" command to determine how and where relative dates are used.

DAYAFTER   **Use the day after the most recent update**

The date of the last update plus one day would be used as the base date for relative date determination.

For example, if on June 10, the date of the last update was May 15, the resulting relative base date would be May 16 instead of June 10.

LASTRANS   **Use the date of the last transaction**

The date of the last transaction to be applied to the Masterfile (most recent transaction date) would be used as the base date for relative date determination.

For example, if on June 10, the date of the last transaction was May 15, the resulting relative base date would be May 15 instead of June 10.

SYSTEM   **Use the current system date**

The current system date would be used as the base date for relative date determination. This is the normal way E-SRF processes relative dates.

Please note: it is possible to override the SYSTEM date. If this were done, the override would be used as the system date.

For example, if the current system date were June 10, the resulting relative base date would be June 10.

Actual date   **May be specified as MM/DD/YY or DDMMM/YY**

The month day and year specified would be used as the base date for relative date determination. This date may be specified in either the USA (12/31/98) format, or the International (31DEC/98) format

For example, if the current system date is June 10, and the specified date is 01/24/98 (or 24JAN/98), the resulting relative base date would be the 24th. Day of January 1998.

SYSTEMDATE(*system date override*) Alter SYSTEM date

This option overrides the current date used as the SYSTEM date. E-SRF by default uses whatever date the hosting operating system has established as the SYSTEM date. This specification will allow you to override this date with any valid date desired.

WARNING: This is a dangerous function and could have negative effects if used improperly. This specification should only be used if there are no other means of correcting a situation.

For example, if you set a future date and run an update, you could end up with a totally empty Masterfile because your current elements may be expired and purged and the new update would not be applied because it too could be expired.

This specification would normally be made under the direction of EKC Technical Support.

If report reruns on later dates are required and your RUN commands contain relative dates, please either change the commands or use the RELATIVEDATE specification.

RESET   **Reset system date**

The system date would be reset to what ever date is current as far as the operating system is concerned.

For example, if on June 10 you have overridden the system date to June 6, this specification would change it back to June 10.

Actual date	**May be specified as MM/DD/YY or DDMMM/YY**

The month, day and year specified would be used as the current SYSTEM date as far as E-SRF is concerned. This date may be specified in either the USA (12/31/98) format, or the International (31DEC/98) format

For example, if the current system date is June 10, and the specified date is 01/24/98 (or 24JAN/98), the system date as far as E-SRF is concerned would be overridden to the 24 Th. day of January, 1998.

FREEZE	**Freeze system Date and Time information.**

This option allows you to obtain the current Operating System date and time information and freezes this information to their current values.

All subsequent attempts to automatically upgrade this information will be set with the now current values.

This option is very useful when running an E-SRF execution around midnight, which has the potential to cause the execution to straddle the beginning of a new day. When this occurs, all update and report selection information becomes relative to the NEW day. This may or may not be what you want; depending on when how you planned the job was going to be run.

With this option set, the job will be consistent (*relative to dates*) when straddling the midnight hour. You still must take due care to make sure the job was provisioned to run relative to the current day (before midnight) or the next day (after midnight).

Please note that this option is only referenced when it is required to obtain the Operating Systems date and time information. In the event that date information is being obtained from other means (such as *SYSTEMDATE* and *RELATIVEDATE* overrides). These specifications will prevail; time information only will be effected.

UPGRADE	**Allow system Date and Time information to be upgraded.**

This option allows you to obtain the current Operating System date and time, and permits E-SRF to automatically upgrade this information at the start of each command executed by the Command Processor.

This is the DEFAULT mode of operation.

The ability to post the exact date and time a report is produced is a desirable feature for knowing exactly when a report was produced. However, this may cause problems when date sensitive commands are executed around the midnight hour. With rare exception, date and time refresh is performed at the start of each command. If commands executed after midnight are date sensitive, the results will be relative to the <u>new</u> <u>day</u>. If this is a problem, consider using OPTION SYSTEMDATE(FREEZE) to freeze this information to the current date and time.

Performance tip:

Turning off grouping for a particular execution:

You should run OPTION GROUPING(NONE) and leave out the "RULES" DD statement on all E‑SRF executions unless you desire some sort of grouping for any part of the execution.

By leaving out the "RULES" DD statement, the external grouping facility will not build its grouping rule sets in the address space, saving both main storage and time.

By specifying "OPTION GROUPING(NONE) at the start of the command list, E‑SRF will not build its internal grouping structure tables in memory. This saves the time necessary to accomplish this, as well as the virtual storage needed to contain the grouping structure.

Partially turning off grouping for a particular execution:

You can save a considerable amount of time by specifying partial grouping under certain situations. If you know you are not going to use a particular grouping structure, you can turn it off from a particular execution. For instance, you are running a report using the RESOURCE segment, or you are running an UPDATE that needs grouping for EXCLUDE processing. You are grouping your users, but those groups are not needed for the current execution. If you specify OPTION GROUPING(NOUSERS), the userid grouping process will not be performed. This is OK, because you have determined that userid groups are not required form this execution.

Chapter 11: REMOVE/DELETE   Existing E-SRF Objects

REMOVE **or DELETE**	**DOMAIN(***domain ID***)** **GROUP(***group name***)** **IMAGE(***image ID***)** **OWNER(***owner name***)**

Description:

This command may be used to alter the contents of specific information on the E-SRF Masterfile. As this product manages security information that must be trusted, most fields cannot be altered.

In E-SRF, there are two classifications of field elements. *SINGULAR* elements, meaning each field occurs once within the object, and *ARRAY* elements, meaning a series of fields are repeated multiple times (such as a list of all violations that occurred for a given resource).

This command may be used to remove entire E-SRF objects from the E-SRF database. This command should only be used in specialized circumstances.

Minor Commands:

DOMAIN(*domain ID***)**   **identify a specific DOMAIN ID**

Supply the one to eight character target DOMAIN Id you wish to remove from E-SRF.

GROUP(*group name***)**   **identify a specific "GROUP" ID**

Supply the one to sixteen character target group object you wish to remove from E-SRF.

IMAGE(*image ID***)**   **identify a specific IMAGE ID**

Supply the one to eight character target IMAGE Id you wish to remove from E-SRF.

OWNER(*owner name***)**   **identify a specific "OWNER" ID**

Supply the one to eight character target owner object you wish to remove from E-SRF.

Command Example:

The following examples should give you a base to start with. Be very careful. Make sure you have backed up your E-SRF Masterfile before attempting these operations.

Removing Group Headers from the Masterfile:

The following example will remove (delete) the GROUP Header previously defined for the group "ACCTPAY". Please note that only the E-SRF Event Reporting System Masterfile definition is removed. No attempt is made to alter grouping rules, or any other definition that may target the particular group name.

    REM   GROUP(ACCTPAY)

To remove *all defined groups*, specify the constant !ALL as the target group ID. All groups (except the DEFAULT group) will be removed from the Masterfile.

    REM   GROUP(!ALL)

Removing Owner Headers from the Masterfile:

The following example will remove (delete) the OWNER Header previously defined for the owner "BETTY". Please note that only the E-SRF Event Reporting System Masterfile definition is removed. No attempt is made to alter grouping rules, or any other definition that may target the particular group name. In the case of report distribution, if group headers currently target the removed owner, the default owner will receive the data previously routed to the removed owner. This will continue until either the group headers are adjusted to point to their proper owners, or the owner is reinstated back on the Masterfile.

    REM   OWNER(BETTY)

To remove *all defined owners*, specify the constant !ALL as the target owner ID. All owners (except the DEFAULT owner) will be removed from the Masterfile.

    REM   GROUP(!ALL)

Removing a DOMAIN from the Masterfile:

The REMOVE command provides a means to totally remove all event data with the exception of the Maintenance Chronological data from the Masterfile.

Upon completion of this function, no other event data will exist for the named DOMAIN.

Due to masking considerations, the DOMAIN ASSIGNments will remain on the Masterfile. If another update were to be executed, the event data related to the domains removed would be applied to the Masterfile in the exact same manner that it was before the purge, except there would be no previous data for the removed domain.

The following statement may be used to remove the DOMAIN called SYST:

    REMOVE   DOMAIN(SYST)

Removing an entire IMAGE from the Masterfile:

The REMOVE command provides a means to totally remove all event data with the exception of the Maintenance Chronological data from the Masterfile.

Upon completion of this function, the following components will be removed from the Masterfile:

- All Resource Segment events for the DOMAINS assigned to the removed image.
- All Source Segment events for the DOMAINS assigned to the removed image.
- All User Segment objects related to the removed image.
- All domain assignments that target the removed IMAGE
- The RSS configuration data for the removed IMAGE.

What this means is the resulting Masterfile will be as if the IMAGE never existed. The only data remaining will be maintenance events journalized in the Maintenance Chronological object. This data is maintained for historical reference as to who updated what security definitions on the particular RSS that managed the removed IMAGE.

If another update were to be executed, the data would be assigned to what ever assignments that remain on the Masterfile. After executing this function, it is recommended that you RUN the ESRFSHOW report to review what is left relative to system definitions. Make sure you are not routing data previously routed to the removed IMAGE to an incorrect IMAGE due to your masking on other domain assignments, unless this is the intended goal of the removal operation.

This is a very big hammer, use it carefully.

The following statement may be used to completely remove an IMAGE called "CHICAGO":

```
REMOVE      IMAGE(CHICAGO)
```

This page intentionally left blank

Chapter 12: RENAME   Renaming Existing E-SRF Objects

RENAME	**IMAGE(***image ID***)** **NEWNAME(***new name***)**

Description:

This command may be used to rename the object ID (KEY) of specific objects or groups of objects that exist on the Masterfile. As this product manages security information that must be trusted, most objects cannot be renamed. Currently, this function relates only to the renaming of entire IMAGES.

Minor Commands:

IMAGE(*image ID***)**   **identify a specific IMAGE ID**

Supply the one to eight character target IMAGE Id you wish to rename.

NEWNAME(*new name***)**   **specify the new name (ID) for the requested rename function.**

Supply the new name (or identifier) you wish to have is the new name.

Command Example:

The following examples should give you a base to start with. Be very careful. Make sure you have backed up your E-SRF Masterfile before attempting these operations.

Renaming an entire IMAGE on the Masterfile

The RENAME command provides a means to associate all data for a specific Image to another non-existing Image. In effect, this is "renaming" the entire Image. The Domains existing on the Image remain unchanged.

For this function to be carried out, the following requirements must be met:

*   The existing Image must exist on the Masterfile.
*   The target "_new_" Image must not exist on the Masterfile.

Upon completion of this function, the following components will be removed from the Masterfile:

*   All User Segment objects related to the current image are renamed to the new target Image.
*   The Image control objects, including the RSS configuration data are renamed to the new target Image ID
*   All Domain assignments that target the renamed image are adjusted to reflect the new Image ID. This includes all masked Domain assignments. You may want o review the effects of any masked Domain assignment and its effect on the remaining Domains and Images existing on your Masterfile.
*   The Image and Domain structures are rebuilt. All subsequent references to it will reflect the new Image ID.

What this means is the resulting Masterfile will be as if the IMAGE always had the new ID.

```
    RENAME     IMAGE(CHICAGO) NEWNAME(DALLAS)
```

Chapter 13: RETAIN   Establish Data Retention and Control

RETAIN	**COMPRESS(***compression algorithm***)**
	DAYS(*number of days***)**
	OBJECT(*object ID)*
	RESET

Description:

This command controls how long in days event-related data will be stored on particular E-SRF Masterfile objects.

The event date is considered the base day; the retention day is considered to be the size of the data window. For example, if an event had a date of March 6 and a particular object has a retention value of 15, the active data window will be from March 6 through March 19.

If you specify retention of *zero*, any existing data (if present) will be purged from the object the next time dynamic Masterfile purge is invoked. Any new data will be considered expired and will not be applied to the Masterfile.

Please note that if an object contains no data elements, the object is automatically purged from the Masterfile. If subsequent activity for the object is encountered, the object will automatically be re-instated.

There are two limiting factors that govern E-SRF event-related data retention:

- the SET ELEMENTS (*nnn*)   (described later)

- this RETAIN command

E-SRF event data will be stored on the Masterfile during the UPDATE process in a FIFO (First In First Out) structure. There is a limit to the number of events that can be stored in an object.

- Data may be dropped from E-SRF due to excessive activity. This is considered an "unexpired rolloff". The SET ELEMENTS (*nnn*) command governs the amount of activity for a given object.

- Events that are presented to E-SRF older than the retention day value set in this parameter are considered outside of the active data window and are never applied to the Masterfile. They are considered expired.

- Existing events outside of the data window are dynamically purged from the Masterfile during the dynamic Masterfile purge operation. Dynamic purging of the Masterfile occurs automatically during the first execution of the update function for any particular day, BEFORE any new update is applied to the Masterfile.

The update process Control Report displays the number of events dropped for either reason.

Each object has its own data window retention value. Specify a separate command for each object you wish to control.

BE CAREFUL WHEN SPECIFYING THIS PARAMETER. Security data by nature may consume great amounts of space on the Masterfile. The larger the value, the more space is required to contain the elements within the window. If the CACHE is used, this amounts to greater consumption of virtual storage.

Know which objects you need and for how long. Unneeded objects should be set to zero.

To find out what your current object RETAIN specifications are set to, RUN the ESRFSHOW report overlay.

The following is a list of objects and their system defaults:

Object Identifier	Segment	Type	Retention days	Compress
FC	Console	Chronological	14	STANDARD
RC	Resource	Chronological	3	NONE
RM	Resource	Maintenance	30	STANDARD
RR	Resource	Recap	14	NONE
RS	Resource	Statistical	3	NONE
SR	Source	Recap	3	NONE
SU	Source	User Chrono	0	NONE
UA	User	Header	30	STANDARD
UB	User	Administration	14	NONE
UC	User	Chronological	3	NONE
UF	User	Firecall	30	STANDARD
UM	User	Maintenance	30	STANDARD
UR	User	Recap	14	NONE
US	User	Statistical	3	NONE
UT	User	Trace	0	NONE

Retain value on UA object note:

The UA (User Header) object is only purged if the user is marked for **DELETE**, (UA.DELETED contains a delete date) within a specific Security Image. Once a user is marked for delete, E-SRF will retain the user in this state for the number of days:

EQUAL TO THE GREATEST VALUE SET FOR ANY USER OBJECT

The purpose of setting a value for the UA object is as follows:

1)   If **ZERO** is specified, the UA object will be deleted immediately.

2)   If a non-zero value is specified, it becomes another number used to determine how long a UA object is retained. This allows you to retain deleted users longer than other user objects. If the UA RETAIN value is lower than other user RETAIN values, it is ignored.

About the UP (User Profile) Object:

The UP (User Profile) object is only purged when the UA header is removed. The contents of the UP's array elements are NOT DATE SENSITIVE and are maintained by specific routines that update the object.

There is no automatic purging of these objects and therefore there is no RETAIN specification for this object.

Minor Commands:

COMPRESS(*compression algorithm*)   object compression option

This specification should not be made. Let the system decide for you. The ESRFSHOW report overlay will display the status of this specification

Otherwise, specify the name of the compression algorithm for the named object.

The choice to compress data is a compromise between main storage and disk space against Central Processing Unit cycles needed to do compression.

If compression is set to 'NONE', you will use less CPU time, but at the same time, the disk space needed to store the object and the main storage required to CACHE the object may increase.

The 'UA" object should always be compressed.

When RESET is specified, certain objects that do not compress well are automatically set to none.

Additional compression routines will be developed and used to enhance this product. Each will be good with a particular type of data structure.

As of release 2.1, the need to compress many of the objects has been eliminated with the implementation of "tokenization".

DAYS(*number of days*)   Array element retention days

Supply the number of days that you wish to retain event data for the particular named object. The acceptable range is from 1 to 4095 days.

OBJECT(*object ID*)   Object identification

Supply the two-character object ID from the list above that applies to the target object you wish to control.

RESET   Reset to system defaults

Set retention days for all segments to system defaults specified in the table above.

This page intentionally left blank

Chapter 14: RUN   Execute an E-SRF Report Overlay

RUN	**ALL**
	AND(*dataname operator text***)**
	ANY
	BACK(*back information box text***)**
	BANNERS / NOBANNERS
	CENTURY / NOCENTURY
	CLASS(*JES output class***)**
	CONTROLTEXT(*control information text***)**
	COUNT(*number***)**
	CTLCHAR(*specification***)**
	DATADD(*ddname***)**
	DATEFORMAT (USA/INTERNATIONAL)
	DDNAME(*ddname***)**
	DESTINATION(*output destination***)**
	DISTRIBUTION(*distribution type***)**
	FIELDS(*field1, field2, ... fieldn***)**
	FIFO
	FRONT(*front information box text***)***t*
	FOOTERS / NOFOOTERS
	FOOTER1(*footer line 1 text***)**
	FOOTER2(*footer line 2 text***)**
	FOOTER3(*footer line 3 text***)**
	FOOTER4(*footer line 4 text***)**
	GROUPS(*group1, group2, groupn***)**
	ID(*ID text***)**
	IEBUPDTE (*membername***)**
	IF(*field1, field2, ... fieldn***)**
	INFO / NOINFO
	LIFO
	LIMIT(*number***)**
	LAF(D/*code***)**
	LINEMODE(*linemode specification***
	LINES(*number of lines***)**
	MAXIMUMLINES(*number of lines***)**
	MAXLINES(*number of lines***)**
	MEMBER(*membername***)**
	OR(*dataname operator operands***)**
	OWNERS(*owner1, owner2,... ownern***)**
	PARAMETER(*parameter data text***)**
	PARMS (*parameter data text***)**
	USERPARAMETER(*parameter data text***)**
	PDS(*membername***)**
	REPORTNAME(*overlay name***)**
	NAME(*overlay name***)**
	SHOWALL
	SORTBY(*field1, field2, ... fieldn***)**
	SUBTITLE(*report sub-title text***)**
	SUMMARY
	TERSE
	TIMEFORMAT (STANDARD/MILITARY)
	TITLE(*report title text***)**
	TRIM / NOTRIM
	WHEN(*dataname operator operands***)**
	WIDTH(*report format width***)**
	WRAPPER / NOWRAPPER

Description:

This command is used to execute a specific report overlay. Report overlay execution results in E-SRF report production output. Each report overlay is slightly different to meet varied needs. Report overlays may require specific data in order to provide the desired results.

The minor commands below describe all RUN minor functions.

They may not all apply to the desired report overlay you are executing. (Some overlays only require the name of the overlay). See the *Report Overlays Guide* to determine which minor commands may be required.

Many RUN minor commands are report execution specific overrides of system specific specifications provided by the SET command. The default in these cases is what may be specified in the SET command.

Please note that RUN commands only apply to the current report being executed, and is reverted back to system specifications at the conclusion of the current report being executed.

Minor Commands:

ALL   **"IF" statement control option**

This is part of automatic report data selection services.

This specification indicates that all IF conditions must be true for the particular selection to take place. If one or more are false, the data will not be selected. This command only applies to a preceding "IF" command.

For example, A is true, B is false, C is false and D is true.   The selection would NOT be made.

AND   **Specifies a selection criteria**

This is part of automatic report data selection services.

This command operates the same as the "WHEN" statement.   This command only applies to a preceding WHEN or AND command.

For example, A is true, B is false. The selection would NOT be made.

See "USING SELECTION CRITERIA OPERANDS" discussed at the end of the RUN command discussion.

ANY   **"IF" statement control option**

This is part of automatic report data selection services.

This statement combines with an "IF" selection. If ANY of the conditions is true, the selection is made. This command only applies to a preceding "IF" command.

For example, A is true, B is false, C is false and D is true. The selection would be made.

BACK(*information text***)**   **Specifies information text to appear at end of report**

This specification will cause the specified sixty four-character text to be displayed in the report's BACK "flower box", which will be displayed as part of the report wrapper at the end of the report.

This flower box will only appear if one or more of these specifications are made.

The flower box will appear regardless of the WRAPPER specification.

You may specify as many of these lines as desired. They will appear in the order specified. They do not have to be grouped together.

The text is formatted in the flower box left justified, padded with blanks.

If you desire the text to be centered, place a % (percent sign) in the first position. The percent sign will not appear, and the text will be centered within the flower box.

BANNERS/<u>NOBANNERS</u>   Specifies usage of banner pages

Produce E-SRF report separator banner pages for this report. This is an additional set of banner pages, not the one described in the *Report Overlays Guide*. We recommend you do not use this option unless you have a specific need to.

CENTURY/<u>NOCENTURY</u>   Display dates in full century format

This option controls the formatting of report line dates relative to the way the century appears.

If NOCENTURY (default) is specified, the century is abbreviated to the last two characters. The date will occupy eight characters on the report.

If CENTURY is specified, the date is formatted to include the entire four-character century. This will cause the date field to be extended to ten characters.

Please note this specification applies only to report data lines. Any date information shown in heading lines and control lines are formatted in full century format. The use of NOCENTURY is provided for compatibility to previous releases, to allow you to conserve report line space and for personal preference.

CLASS(*JES output class*)   Specifies JES SYSOUT CLASS output DDNAME

This parameter is optional, and is only used by E-SRF when the target report output must be opened and is not already allocated by JCL. Once the report output DD is opened, it remains so for the duration of the E-SRF execution.

If this parameter is omitted, and the report output DD must be dynamically allocated, the output CLASS will default to the current job's MSGCLASS specification.

CTLCHAR(*specification*) Print control character specification.

This parameter is optional, and is only used by E-SRF when the requester does not desire the use of the default printer control characters, which are standard print CCW machine codes (REFCM=FBM).

This specification is examined only when E-SRF determines that a particular report output file must be opened. This occurs when the first report accesses the output *DDNAME*, as controlled by the DDNAME specification, defaulted to, or is the target OWNER report bundle report output file.

Once set, all subsequent reports written to the particular report output file will inherit the CTLCHAR characteristics specified when the file was opened.

This specification is ignored if the target report output file was previously opened.

If the CTLCHAR specification is omitted. The system will force RECFM=FBM processing, regardless of the DCB settings, *and processing will be as it always was from the inception of the product.*

Possible impact on existing systems:

If the specification is not used, there will be no impact on existing systems.

ASA ASA print control characters will be used

If CTLCHAR(ASA) is specified, it is assumed that control characters are desired and RECFM=FBA will be used. Each print line will contain an ASA control character, followed by the text to be printed. If dataset already exists, RECFM=FBA will be forced.

ASCII ASCII print file will be created

If CTLCHAR(ASCII) is specified, it is assumed that the output file will be created encoded in ASCII instead of the standard mainframe EBCDIC character representation.

The output must be a variable length file with a LRECL of 512 bytes.

The data will be encapsulated with ASCII print control characters that will provide standard ASCII form feed character in front of each page, and CR-LF control characters at the end of each line.

This file is suitable to be BINARY transferred from your mainframe to a PC (*filename*.txt) and subsequently printed on a standard PCL printer with no data alteration required.

You may examine the data using a standard word processor.

CCW CCW (machine) print control characters will be used

If CTLCHAR(CCW) is specified, it is assumed that control characters are desired and RECFM=FBM will be used. Each print line will contain a CCW control character, followed by the text to be printed. If dataset already exists, RECFM=FBM will be forced.

This is the default mode of operation and is the same as CTLCHAR(YES)

HTML   **HTML print file will be created**

If CTLCHAR(HTML) is specified, it is assumed that the output file will be created in standard HTML format. The data itself will encoded in the standard mainframe EBCDIC character representation.

The output must be a variable length file with a LRECL of 512 bytes.

The data will be encapsulated with HTML print control characters that will provide all necessary protocols to allow the file to be viewed on a standard WEB browser such as Firefox or Microsoft's Internet Explorer.

The resulting data is optimized for a WEB browser and therefore may not be suitable to print on a printer.

This file can be used as-is on mainframe web hosting facilities and browsers.

The file may be downloaded to the PC platform using the standard CR/LF download protocols or FTP. This file may then be viewed on a PC using standard PC browser programs or published on PC based WEB hosting facilities.

NO   **Print control characters are NOT to be provided**

If CTLCHAR(NO) is specified, control characters will not be provided and E-SRF will attempt to use RECFM=FB. Each print line will only contain the text to be printed. If dataset already exists and RECFM=FBM.FBA is present, the RECFM *will be preserved*, and the control character byte of the print line will contain blanks.

YES   **Standard machine control characters are to be provided**

If CTLCHAR(YES) is specified, it is assumed that control characters are desired and RECFM=FBM will be used. Each print line will contain a CCW control character, followed by the text to be printed. If dataset already exists, RECFM=FBM will be forced.

This is the default mode of operation and is the same as CTLCHAR(CCW)

CONTROLTEXT(*information text*)   **Specifies information text to appear at the in front of the report wrapper.**

This specification will cause the specified sixty-four-character text to be displayed in front of the report wrapper.

This information may be used to provide control information to be used by various report distribution sub-systems that may be in place to post-process report output from E-SRF.

This information, if provided, will appear regardless of the WRAPPER specification.

You may specify as many lines as desired. They will appear in the order specified. They do not have to be grouped together.

The text is formatted left justified relative to the first print position.

COUNT(*number*)   **Set the "COUNT" report specification control**

Specifies a numeric value that particular report overlays *may* use. Although report overlays may be written to use this value for anything, it is normally used to limit some sort of activity.

For example, there are reports that RANK events, such as violations. The top of the list shows the user or resource with the highest violation count, ranked 1, the end of the report shows the user or resource with the least number of violations, ranked last. This parameter may be used to control the minimum number of events that qualify for displaying on the report. Specifying COUNT(3) may cause a "top 40" report to be less than the top 40 because the individual rank level (say at rank 28) has three or less events. The report overlay was programmed to compare the items on the line to the "count" value, skipping lines with less than the value set. Because of the way the overlay was designed, the remaining ranks (if any) would contain less than the count value and therefore also be dropped. In this case, the report would only include the top 27 levels.

DATADD(<u>REPORTS</u>/*ddname*) Report overlay data output file DDNAME

This specification defines the DDNAME of the dataset which the report overlay will create a specific file into.

DO NOT CONFUSE THIS WITH THE ACTUAL REPORT DDNAME, as specified by the DDNAME specification.

This specification is used in specific report overlays and its use is limited to the particular report overlay. This specification is ignored in all other cases. Do not include this specification unless the report overlay requires it.

The characteristics of the dataset are dictated by the report overlay requiring it, and its usage.

You may have only one of these specifications in a particular RUN command, but you may have other RUN commands specifying the same or different datasets.

The maximum number of these datasets is dictated by the maximum number of files that may be opened by a single executing job running on the operating system hosting the execution.

Please note that once a particular DATADD is opened, it remains open until the termination of the Command Processor session. This means the characteristics remain with the dataset despite subsequent references to this DD by other RUN commands. If different characteristics are desired, use a different DATADD for each specific requirement.

DATEFORMAT(USA/INTERNATIONAL)   Specifies date format

Sets the appearance of dates displayed by E-SRF for the particular report execution.

This specification is also available as a system wide option specified by the SET DATEFORMAT(..) command. The SET specification is used unless overridden in the RUN command for a specific report execution.

This specification may also be made for a specific OWNER by maintaining the OA.DATE data field. This specification is ignored unless the report is being executed under Automatic Report Distribution. In this case, if OA.DATE was specified for specific owners, the OA.DATE specification will override any other DATEFORMAT specification for the specific owners provisioned as such.

Note that both formats use the same number of characters.

USA   United States date format

Dates will be formatted in the month, day and year format. For example, April 2, 2003 will appear as 04/02/03 (or 04/02/2003 in CENTURY mode).

To specify this format for a particular owner, alter the owner header with the following: OA.DATE(U)

INTERNATIONAL   International date format

Dates will be formatted in the day, month year format. For example, April 2, 2003 will appear as 02APR//03 (or 02APR/2003 in CENTURY mode).

To specify this format for a particular owner, alter the owner header with the following: OA.DATE(I)

DESTINATION(*output destination*) Specifies the JES SYSOUT DESTINATION of the report output DDNAME

This parameter is optional, and is only used by E-SRF when the target report output DDNAME must be opened and is not already allocated by JCL. Once the report output DD is opened, it remains so for the duration of the E-SRF execution.

If this parameter is omitted, and the report output DD must be dynamically allocated, no destination will be considered for the allocation.

DDNAME(<u>REPORTS</u>/*ddname*) **Report overlay output file DDNAME**

This specification defines the DDNAME of the dataset which the report overlay will create the report file into.

Do not confuse this with the DATADD specification that is optionally used by some report overlays for writing out data files.

This specification is used to override the use of REPORTS as the output DDNAME for the report data file. This allows you to place reports on specific output report datasets.

This specification is ignored if the report is running under Automatic Report Distribution, where the report is written to the OWNER's output dataset.

As with the REPORTS dataset, the file is <u>opened on demand</u>. If the file was pre-allocated, the maximum valid record length is 255 characters.

If the dataset was not pre-allocated, what ever WIDTH was in effect for the report causing the open will dictate the file' logical record length.

If your report is writing to a report output file with a width smaller than the specified logical record, the report will obey the WIDTH setting and pad the unused characters to the right with blanks. This is true even if the report is terminated due to excessive width. The report always conforms to its width, whether you specify it, of it is fixed within the report overlay itself.

If the file's record length is less than the width, the right side report data will be truncated, with warning messages posted.

You may have only one of these specifications in a particular RUN command, but you may have other RUN commands specifying the same or different datasets.

The maximum number of these datasets is dictated by the maximum number of files that may be opened by a single executing job running on the operating system hosting the execution.

Please note that once a particular DDNAME is opened, it remains open until the termination of the Command Processor session. This means the characteristics remain with the dataset despite subsequent references to this DD by other RUN commands. If different characteristics are desired, use a different DDNAME for each specific requirement.

DISTRIBUTION(<u>NONE</u>/OWNER/GROUP)   Controls automatic report distribution

This is part of automatic report distribution services.

Controls report distribution mode

NONE   (Undistributed report - default option)

The report produced by the current RUN command will be a single undistributed report, produced in report sequence placed on the DDNAME optionally specified or on the REPORTS output.

The report will be produced under the full control of the report overlay and augmented by the original RUN minor commands. No attempt to distribute the report or group the report is made.

OWNER   (Distributed report - single reports by owners)

The RUN command will execute the current report overlay for <u>EACH OWNER</u> that has been defined to E-SRF (E-SRF Owner Masterfile object) and place each report on the OWNER'S specific report DDNAME.

Each report will contain data that was grouped within groups that the OWNER was associated with, either as the owner itself, or perhaps as an interested party.

Any report information, which was not accounted for by E-SRF Event System, defined GROUP and OWNER objects in the Masterfile will be placed on the undistributed report and treated as if it was undistributed.

If you have ten owners defined to E-SRF, you will have **11 reports,** one for each owner plus a report containing all data that belonged to groups that were NOT DEFINED to the E-SRF Event System Masterfile.

This is the recommended group distribution method.

For more information on report distribution, see the *User Guide*.

GROUP   (Distributed report - separate for each group by owner)

The RUN command will execute the current report overlay for <u>EACH GROUP</u> defined to E-SRF separately and place each report on the Group's OWNER'S specific report DDNAME

This is a very intensive distribution mode. If you have 200 groups defined to E-SRF across 10 owners, the report overlay will produce 200 reports and distribute them to their various owners. Additionally, another single report will be produced with all the data that belonged to groups that were not defined to E-SRF. In total, you will have 201 reports in eleven bundles, each referenced owner plus the UNDEFINED Owner's report.

This mode of distribution should be avoided if possible. The recommendation if you are running report distribution is to use DISTRIBUTION(OWNER).

For more information on report distribution, see the *User Guide*.

FIELDS(*field1,field2...,fieldn*) Specifies list of fields to appear on report

This is part of automatic report formatting services.

The field names specified in the minor command must be represented in the E-SRF dictionary. Columns will be displayed in the order you specify the field names. The minimum format width is eight characters regardless of the actual data requirement. This provides space necessary to construct report heading lines. The header for the column will be the field name. There will be two spaces between each field. An error will occur if the space required to format all fields in the list exceeds the maximum report width.

FIFO "First In First Out" report presentation

This is part of automatic report data selection services.

This specification indicates that the line items printed on the report will appear in First In First Out sequence.

Normally, this means date/time order, the oldest event on top, concluding with the most current event at the bottom within a specific report object.

This is the default mode of operation.

FOOTERS / NOFOOTERS Specifies whether to use report footers

This specification is optional, and is only used by E-SRF when report footers have been established globally for all reports.

If this specification is omitted, and global or local report footers have been defined, the defined footers will appear on the report rendered by this particular RUN command.

The default for this specification is: FOOTERS

FOOTER1(*footer line 1 text*)
FOOTER2(*footer line 2 text*)
FOOTER3(*footer line 3 text*)
FOOTER4(*footer line 4 text*) Specifies footer line text override

Supply the one to sixty four character text string that will appear as footer information for the desired footer line. This text will be centered relative to the WIDTH specification on bottom of each report page.

If a blank line is required, you must specify the line with a blank character.

Only specify as many footers as required. If you only need two lines, there is no need to specify three and four as being blank. No additional room will be provided for unused footer lines.

If footers are specified in the RUN command, they will be merged with the ones specified in the SET command, with the ones specified in the RUN command overriding the ones specified by the SET command.

This means if you only specify footer line *one* in the RUN command, you will receive footer line *one* from the RUN command, and footer lines *two*, *three* and *four* will come from the original SET specifications.

To "clear" individual footer lines specified in the SET command, specify the line with a "null" text specification, for example FOOTER4() will null out the FOOTER4(…*text*…) SET command specification.

FRONT(*information text***)** **Specifies information text to appear at the beginning of the report**

This specification will cause the specified sixty four-character text to be displayed at the BEGINNING report "flower box", which will be displayed as part of the report wrapper at the beginning of the report.

This flower box will only appear if one or more of these specifications are made.

The flower box will appear regardless of the WRAPPER specification.

You may specify as many of these lines as desired. They will appear in the order specified. They do not have to be grouped together.

The text is formatted in the flower box left justified, padded with blanks.

If you desire the text to be centered, place a % (percent sign) in the first position. The percent sign will not appear, and the text will be centered within the flower box.

GROUPS(*group1, ..., groupn***)** **Specifies list of groups included in report generation process**

This is part of automatic report selection services.

This option specifies a list of groups to determine if the current data will be selected. You may specify a list of group name(s) that you wish to be processed by the executing report overlay. There is no limit on the number of group names you may specify. Additionally, the sequence has no effect on the outcome of the report.

NOTE: If you specify a GROUP list when using report distribution "DIST(OWNER)", the list will be ignored.

The group name(s) specified in this parameter are processed in the same sequence as they are entered.

Group names may be masked, that is wild card characters may be used to represent characters. The asterisk ('*') character indicates any character in that one position will match. A dash ('-') indicates any number of characters (if any) from the dash position to the end of the group name will match. For example GROUP(SYS**21-) would match the following groups:

  SYSAB21   SYSCD21CONTROL   SYSXX21C

If you code GROUP(-), all groups would match.

If this command is omitted, all information is selected.

ID(*ID text*)   Specifies an ID character string

This option specifies a one to eight character identification string that is made available to report overlays for specific processing.

Its use is considered specialized based on the particular report overlay's intended use.

For example, the report overlay ESRFDXD uses it to identify each specific record produced for a particular run statement. This would allow a post-processing program to separate out the downloaded data and process it according to its specific needs.

IEBUPDTE(*membername*)   Specifies IEBUPDTE control statements

This option will establish the IEBUPDTE format for this DD if this command is issued when the report output file is opened.

The *membername* is optional.

If *membername* is present, this particular report will contain the specified member name for the current report.

If *membername* is omitted, E-SRF will assign a member name for you in the following format: RPT*nnnnn*. The first unnamed report will be RPT00001, the second unnamed report will be RPT00002.

Generated report member names are relative to each output file.

IEBUPDTE is the name of an IBM utility that may be used to update members of a PDS. The intent is not to duplicate this functionality. The IBM utility would function, provided the report output width was 80 characters. The intent is to give you a widely used means of locating and identifying reports contained on the report output flat file for subsequent processing.

Please note: Once a report file is opened, its attributes will be used for ALL REPORTS written to it. This means all other RUN commands that reference the file, including report distribution, will have their individual reports written out with the IEBUPDTE statement placed in front of the report.

IF(*field1,field2...,fieldn*)   Specifies conditions for inclusion in report generation process

This is part of automatic report data selection services.

Specifies a list of fields to be used to determine if the current data will be selected (used) or skipped over. "IF" selections are based on "YES/NO", zero or greater than zero, present or not present conditions. The intent is a "shorthand" implementation for some "WHEN" processing, primarily used to identify bit privileges or flags. This command is augmented by the "ALL" and "ANY" command.

If a "-" (minus sign) appears in front of the field name, reverse logic is used, that is "no means yes", If IF(-UA.DELETED,ACF2.ACC-DATE) is specified, the user would be selected if the user is **not deleted** and an **access date is present** (non-zero).

<u>INFO</u> / NOINFO   Display report information on wrapper

This option controls whether or not report control information appears on the report WRAPPER. It is recommended that INFO (default setting) be used, as it would be difficult to determine the criteria used to create the report without this information.

LAF(<u>D</u>/*code*)   Look and Feel report specification

Attempts were made to normalize the appearance of reports to a unique set of terminology that could cross over various RSS systems. Some terms were taken from individual RSS where it was thought to make sense. Although this seems like the ideal approach for an enterprise-wide product, customers still have a preference to particular terminology and formatting. This specification provides a means to address this.

You may specify a "look and feel" code that will alter the reports accordingly.

The following Look and Feel codes are available:

A   make the report look like ACF2

D   default report "look and feel"

R   makes the report look like RACF

This specification is also available as a SET command. The SET specification is used unless overridden in the RUN command for a specific report execution.

This specification may also be made for a specific OWNER by maintaining the OA.LAF data field. This specification is ignored unless the report is being executed under Automatic Report Distribution. In this case, if OA.LAF was specified for specific owners, the OA.LAF specification will override any other LAF specification for the specific owners provisioned as such.

LIFO   "Last In First Out" report presentation

This is part of automatic report data selection services.

This specification indicates that the line items printed on the report will appear in Last In First Out sequence.

Normally, this means date/time order, the latest (newest) event on top, concluding with the oldest at the bottom within a specific report object.

This type of report presentation was the way E-SRF worked in prior releases. The default is **FIFO**.

LIMIT(*number*)   Set the "LIMIT" report option control

Specifies a numeric value that particular report overlays <u>*may*</u> use. Although report overlays may be written to use this value for anything, it is normally used to limit some sort of activity.

For example, there is a set of overlays that RANK events, such as violations. The top of the list shows the user with the highest violation, ranked 1; the end of the report shows the user with the least number of violations, ranked last. This parameter may be used to control the depth of the ranking. Specifying LIMIT(10) causes the report to become a "Top 10" report.

LINEMODE(<u>STANDARD</u> / *linemode specification*) Establish reporting LINEMODE system default

This option provides the system wide base setting for LINEMODE report functions. LINEMODE alters the appearance of reports in the event that there is not enough room to contain the required text across a single print line width.

The current report will be formatted under the influence of this specification.

STANDARD   Standard LINEMODE operation

Specification of this parameter will cause Report Services to format reports using normal E-SRF formatting, coupled with specific report overlay formatting characteristics. If the report line width is too small, attempts will be made to make the report fit.

Inter column spacing will be reduced.

Shrinkable fields will be reduced in size. The minimum size a shrinkable field may be reduced to is eight characters.

If after these considerations have been mad, and the report still does not fit, the report execution will be terminated.

WRAP   Wrap text to a second line

Specification of this parameter will cause Report Services to attempt to use normal E-SRF formatting, coupled with specific report overlay formatting characteristics. If the report line width is too small, attempts will be made to make the report fit.

Inter column spacing will be reduced.

Shrinkable fields will be reduced in size. The minimum size a shrinkable field may be reduced to is eight characters.

If after these considerations are made, the report still does not fit, an attempt to wrap the line to a "secondary" continuation line. This is accomplished in the following manner.

The maximum report width is limited to 256 characters, including the print control character.

The *WIDTH* specification still maintains control over a single line; however, it is not considered to be the report "*window*" width. This means if the current width is set to 133, the report *WINDOW* will be 133 characters, but the report data capacity will be considered twice that width or 256, *which ever is smaller*. In this case the primary line will be 133 characters wide (the report window width), the secondary line will be 124 characters, 123 report text characters plus a control character for the secondary line).

The split point, which divides the line into a "primary" line and a "secondary" line is determined by looking for the first non-blank character starting at the end of the window width, working backwards.

If the split point is less than the window width, any shrinkable fields in the primary line will be restored back until the window width is reached, of no more shrinkable characters are detected.

The primary line will be left justified. The headings will appear, just as they would had the report been run using the STANDARD specification.

The secondary line is the residual of the source line text that did not fit on the primary line. This text will be processed in a manner similar to the primary line, except it will be formatted after the primary line, right justified. Heading lines will also appear right justified.

Detail heading lines will be formatted as two sets of detail headings, the primary first, followed by the secondary lines. This means the detail heading lines will be together, requiring twice the required number of lines to print.

Detail lines are formatted and printed in a similar fashion, except if the detail line is blank, it is not printed.

All line control is dynamically adjusted according to action taken by report services to provide this type of output.

Normally, this output is acceptable, but sometimes the report may become difficult to read. This mode of operation should be experimented in some cases, the report output actually looks better run this way.

MAXIMUMLINES(lines) or MAXLINES(*lines*) or LINES(*lines*)
Set report print maximum lines

Sets the maximum number of lines that may be printed on a report page before E-SRF automatically skips the printer to the next page. If this value is specified, it overrides the system setting in the "SET" command.

MEMBER(*membername*)
Specifies a member name for your report output

This option provides a member name (if one is required). If no member name is required, this command is ignored.

The use of this command will allow you to control the name of the member for this report if you are using the IEBUPDTE or PDS options. If you are using either of these options, and you do not specify a member name, E-SRF will create a member name for you.

Please see IEBUPDTE or PDS for more information.

OR
Specifies a selection criteria

This is part of automatic report data selection services.

This command is the same as the "WHEN" statement, except either the condition specified here or a previous condition must be TRUE. This command only applies to a proceeding WHEN command.

For example, A is true, B is false. If either were true, the selection would be made. If neither were true, the data would NOT be selected.

See "USING SELECTION CRITERIA OPERANDS" discussed at the end of the RUN command discussion.

OWNERS(*owner1, ..., ownern***)** **Specifies list of owners included in report generation process**

This is part of automatic report selection services.

This option specifies a list of owners to determine if the current data will be selected. You may specify a list of owner name(s) that you wish to be processed by the executing report overlay. There is no limit on the number of owner names you may specify. Additionally, the sequence has no effect on the outcome of the report.

Owner names may be masked, that is wild card characters may be used to represent characters. The asterisk ('*') character indicates any character in that one position will match. A dash ('-') indicates any number of characters (if any) from the dash position to the end of the group name will match. For example OWNER(ACCT*X-) would match the following owners:

   ACCTQX1   ACCTDXD   ACCTDXSYS

If you code OWNER(-), all owners would be equal.

If the owner list is omitted, all information is selected.

In an undistributed report:

The owner list is reviewed and a list of groups related to the specified owner(s) *augments* an existing group list (GROUPS parm) if applicable. The resulting group list will be used to select records to be processed by this RUN command.

In a distributed report:

If one or more owners are specified by this parameter, only the owners named in the list will be processed. This is useful if you want to provide owner distributed reports for only one or more owners for a particular report run.

PARAMETER(*text***)**
or PARMS(*text***)**
or USERPARAMETER(*text***)**   **Supply report overlay parameter information**

This command provides the optional one through sixty-four-character parameter character string that may be required for the current report overlay. Each report overlay can use this option differently. See the *Report Overlays Guide* for more information.

PDS(*membername***)** **Specifies PDS output dataset format**

This option indicates the output file will be a PDS if this command is issued when the report output file is opened.

The *membername* is optional.

If *membername* is present, this particular report will be stored in the PDS using the supplied member name.

If *membername* is omitted, E-SRF will assign a member name for your PDS in the following format: RPT*nnnnn*. The first unnamed report will be stored as RPT00001, the second unnamed report will be stored as RPT00002.

Generated report member names are relative to each output file.

Once a report file is opened, its attributes will be used for ALL REPORTS written to it. This means all other RUN commands that reference the file, including report distribution, will have their individual reports written out as a PDS members.

REPORTNAME(*overlay name***)** **Specifies report overlay to execute**
or NAME(*overlay name***)**
or ID(*overlay name***)**

The report overlay will be loaded into storage (if not already there) and executed until completion, where control will return to the command processor. The report overlay will remain in storage for subsequent RUN command executions.

SHOWALL **Set the "SHOWALL" report option control**

This is part of automatic report formatting services.

Specifies whether a particular report overlay is to operate in a "showall" mode.

This option may be used by some report overlays. Although report overlays may be written to use this flag for anything, it is normally used to consolidate information making the report more compact.

If this facility is being used by the report overlay, this control decides whether or not to print the formatted line, based on the composite field configuration. This option will override suppression and cause the line to print as if it was intended to be printed.

This specification used in this mode is normally made for debugging purposes.

SORTBY(*field1,field2...,fieldn***)** **Specifies list of fields used to sort data**

> *This is part of automatic report formatting services.*
>
> The field names specified will be used to create a sort key, which will be used to determine the sequence of the report data presented on the report.
>
> In this release, the specification relates to ESRFLIST only.
>
> You may specify from one to eight Data Dictionary field names.
>
> The names you specify do not have to appear on the report.
>
> The original sequence of the particular data will be preserved, with the adding of your sequence as a major sequence. This means you only need to specify the order you want your report in, minor sequencing will be performed for you automatically.
>
> Page breaks are controlled by placing a + (plus sign) in front of a particular field names you wish to break on. E-SRF will skip to a new page when one or more of these fields are determined to have changed.
>
> If page breaks are used, an additional heading line is provided that will display the status of the control break. The entire key is shown, with the actual data shown on page break fields.

SUBTITLE(*report sub-title text***)** **Sub-title to placed on report**

> This optional specification provides an additional one through sixty-four character report *sub-title* text that will be centered on the report under the "*official*" report title line.
>
> This allows for yet a third title specification. The first is the title produced by the Report Overlay. Second is the data from the "TITLE" specification that is concatenated to the Report Overlay title. This specification is the third report title provision, which would appear under the first two.

SUMMARY **Set the "SUMMARY" report option control**

> *This is part of automatic report formatting services.*
>
> Specifies whether or not particular report overlays are to operate in a "summary" mode.
>
> This option may be used by some report overlays. Although report overlays may be written to use this flag for anything, it is normally used to suppress the body of a report and produce summary information only (first page parameters, last page statistics).

TERSE **Set the "TERSE" report option control**

> *This is part of automatic report formatting services.*
>
> Specifies whether or not particular report overlays are to operate in a "terse" mode.
>
> This option _may_ be used by some report overlays. Although report overlays may be written to use this flag for anything, it is normally used to suppress verbose detail, making the report more compact.

TIMEFORMAT(STANDARD/MILITARY**)** **Specifies time format**

> Sets the appearance of time of day information displayed by E-SRF for the particular report execution.

This specification is also available as a system wide option specified by the **SET TIMEFORMAT (..)** command. The SET specification is used unless overridden in the RUN command for a specific report execution.

This specification may also be made for a specific OWNER by maintaining the OA.TIME data field. This specification is ignored unless the report is being executed under Automatic Report Distribution. In this case, if OA.TIME was specified for specific owners, the OA.TIME specification will override any other TIMEFORMAT specification for the specific owners provisioned as such.

Note that both formats use the same number of characters.

STANDARD

Format time in 12-hour AM/PM format

Specification of this parameter will cause time of day information to be formatted using a twelve-hour clock. If the time were 11:30 in the evening, the time would appear as: 11:30 PM on the report. (The seconds are dropped)

To specify this format for a particular owner, alter the owner header with the following: OA.TIME(S)

MILITARY

Format time in 24-hour military notation

Specification of this parameter will cause time of day information to be formatted using the military notation, which is a twenty-four hour clock with seconds. If the time was 11:30 (and 29 seconds) in the evening, the time would appear as 23:30:29 on the report. (The seconds are carried out to hundredths)

To specify this format for a particular owner, alter the owner header with the following: OA.TIME(M)

TITLE(*report title text***)**

Title to be used to augment report overlay title text

This command provides the optional one through sixty-four-character report title text that will be concatenated with the title information supplied by the report overlay program.

TRIM / NOTRIM	**Download data format option**

THIS SPECIFICATION APPLIES ONLY TO REPORT OVERLAYS THAT OUTPUT DELIMITED TEXT RECORDS.

This command provides a means to control for format of the data that is outputted in a "delimited text" format. This format is produced by specific Report Overlays such as ESRFDXD to enable E=SRF Masterfile data to be downloaded into Personal Computers (PCs) and subsequently processed by their powerful spreadsheet, database or graphical applications.

If "**TRIM**" (default) is specified, all blank or null characters removed from the formatted output. If downloading into normal PC applications, this is the format of choice.

If "**NOTRIM**" is specified, all columns are fixed length with no attempt to remove blanks or nulls. *The data is still in delimited text format.* If downloading into applications that you have developed and you have not supplied the necessary programming logic to deal with variable length field within variable length records, this is the format you would need. All data locations for a specific record type would map into a standard programming language record declarative and your programming would be simplified.

The default is TRIM

WHEN	**Specifies a selection criteria**

This is part of automatic report data selection services.

This command supplies criteria that will be used to determine if a particular dataname on an E-SRF object matches your requirements for selection. Selection of the object is based on results dictated by the "WHEN", "OR" and "AND" statements. When all selection criteria commands are processed, the decision to select is made based on a selection indicator's status. If a particular selection criteria command does not apply to a particular object, it is ignored and the selection indicator is not altered for that particular command.

- A dataname supplied identifies the data to examine.
- An operator specified identifies the type of test to perform.
- At least one operand providing the "argument".

See "USING SELECTION CRITERIA OPERANDS" discussed at the end of the RUN command discussion.

WIDTH(*report format width*)   Report data format width

This command allows you to override the global report width setting for the execution of this RUN command.

If this report is opening the target report output dataset, and it has not already been set with a record length, this specification will control the record length instead of the global value.

This may give you the ability to have reports wider than the standard 133 printer limitation that has become the standard from the onset of the IBM 1403 impact printer.

Most EKC supplied reports will fit in the 133 character de facto standard limitation. ESRFLIST however is totally user defined and could go on forever if there was no width limit provision.

If this specification is used, whenever E-SRF references the global report width specification, this value will be used instead. All rules governing the global report width specification adhere to this specification.

There are several considerations that are associated with this specification.

Please note this specification includes the character required for the printer control character.

Please refer to the discussion about this parameter in the "SET" command.

WRAPPER / NOWRAPPER   Report "wrapper" control

This command allows you to suppress the production of the "report wrapper" for any reports being produced by the current RUN command.

The "report wrapper" is the first page of the report (block letters containing the Report Overlay name) all specified report parameters and options and the very last page that provides some report overlay execution statistics.

The report wrapper should always be produced. It contains information about how the report was produced, where the data came from and other important information. Without it you really have no way to know how the report was produced.

The recommendation is to leave this specification out and let it default to WRAPPER, and you will continue to receive a report wrapper on every report.

Please note, if NOWRAPPER is specified, and FRONT and/or BACK specifications are made, the "flower boxes" will appear despite what this specification has in effect.

Data Dictionary Datanames

Data what you wish to use for selection and reporting are stored on the E-SRF Masterfile. You reference this data by using *DATANAMES.* Datanames are contained in the E-SRF Event Reporting Data Dictionary.

There are many data formats contained on the Masterfile that may have particular requirements. The following examples illustrate this:

You present NUMRIC data when specifying selections on fields that are numeric.

When selecting multi-valued fields, the selection examines all fields in the list until the condition is met.

When displaying multi-valued fields, the column will be extended to contain the maximum number of fields that are possible, (separated by a single space.

To learn more about the data contained on the Masterfile and the Data Dictionary, please refer to the *E-SRF Event Reporting Masterfile and Data Dictionary Reference*.

Using Selection Criteria Operands: "IF" statements

"IF" statements contain a list of fields. If you have more than one IF statement, the field names are concatenated together to form a single list of field names.

The IF selection processor will run down the list of fields and determine if the field is true or false.

The entire IF field list is considered a single selection criterion.

Normally, ***if any of the fields are false, the selection is not made***. You can change this by specifying "ANY" anywhere in the RUN command-set. If ANY is set, ***the selection will be true if ANY individual field test is true***.

You can also reverse the true/false setting of a test. If you place a - sign in front of the particular field in the list, than if the field test result is no, it will be treated as true, conversely, if the result is yes, it will be treated as false.

IF statements work on any field defined in any E-SRF dictionary.

The following fields will set the test condition to YES if any character data exists in the field:

```
Access Type Code                Hexadecimal Nibbles
Action Code                     Image ID
Character format                Journalizing Reason Code
Encrypted data                  Maintenance Request Type
E-SRF UID Character Format      Maintenance Type
ETF/x Status Byte               Owner ID
ETF/x FIRECALL Request          Rectype Code
Group ID                        Resident Security System (RSS) Code.
Group ID Related Comment
```

The following fields will set the test condition to YES if the numeric value is greater than zero:

```
Binary Value                    Packed Numeric Value
Date in Absolute Format         TOD in STCK DOUBLEWORD Format
Date in Packed Format           TOD in Relative .01 Second Resolution
```

The following fields will set the test condition to YES if the particular flag is on within the field.

```
BIT-BYTE FLAG
```

Using Selection Criteria Operands: "WHEN", "AND" and "OR" statements

"WHEN" and "AND" are the exact same statement. The command name "AND" just makes it easy for you when setting up selection logic. When "WHEN" is mentioned, AND may also be assumed.

Each selection criteria starts with a WHEN (or AND) statement and may contain only one.

A WHEN statement requires the test to be **true** in order to make the selection.

"OR" statements may be used to augment a WHEN statement. As many OR statements may be specified as are required to make up a particular selection criteria.

Once a selection criterion fails, the selection process is terminated and the current item test is considered to be untrue (fail).

If you have multiple WHEN statements, you actually have multiple selection criteria.

You may have as much selection criteria as needed to build your report selection.

Boolean Consideration while Attempting to Exclude Data

A selection criterion is designed to *select* data. If the selection criteria deem the data not appropriate, the data is NOT selected, regardless of any additional selection criteria. In most cases, this is straight forward except, when attempting to EXCLUDE data from a report.

Consider the following example: You want to select all maintenance requests from the Resource Maintenance (RM) segment, except for "DEFINE" and "DELETE" requests, as indicated by the RM.REQUEST field. This implies all data except any that contain "DEFINE" and "DELETE" for the current selection criteria must pass. This is the opposite of selecting all RM.REQUEST types except the ones to be excluded, and is a valid function to perform.

One would assume a *not equal* WHEN for the "DEFINE" and a *not equal* OR for the "DELETE", would accomplish this. THIS WILL NOT WORK. The OR will supersede the WHEN's results, and you will pass the selection criteria with "DEFINE" requests being considered selected. You are saying WHEN RM.REQUEST *not equal* "DEFINE", OR RM.REQUEST *not equal* "DELETE". Assuming no additional selection criteria present, you will receive the unwanted "DEFINE" requests along with the desired data.

To properly accomplish the above example, you must issue a WHEN (*not equal*) against RM.REQUEST for the "DEFINE" request, followed by another WHEN or AND (*not equal*) against RM.REQUEST for the "DELETE" request. The desired result is to exclude *both* conditions. Each exclude must be presented as a separate selection criteria.

Constructing "WHEN", "AND" or "OR" statements

For this discussion, we will consider all three statements one in the same.

This statement consists of the following:

Statement framing: the actual bracketing of the statement is as follows:

WHEN(*dataname operator operand1 operand2*)

Dataname: Name of a particular data field contained in the E-SRF Data Dictionary. To determine the specific dataname, please consult the Data Dictionary Reference.

Operator: Operation to be performed against the specified dataname. The table below briefly describes operators that may be used:

Operand 1:   Specify the argument to be used by the operator against the dataname.

Operand 2:   OPTIONAL, only used where required. Specify an additional argument to be used by the operator against the dataname.

Operator	Action	Result
EQ or EQUAL or =	Compare EQUAL	TRUE if field value equals operand value
NE or NOT or NOTEQUAL	Compare NOT EQUAL	TRUE if field value does not equal operand value
GREATER or GT or >	Compare GREATER THAN	TRUE if field value is greater than operand value
GE or GTE or >=	Compare GREATER THAN or EQUAL	TRUE if field value is greater than or equal to operand value
LESS or LT or <	Compare LESS THAN	TRUE if field value is less than operand value
LE or LTE or <=	Compare LESS THAN or EQUAL	TRUE if field value is less than or equal to operand value
SCAN	Textual non-positional generic scan	TRUE if total operand value matches any piece of field value *(character fields only)*
MASK or MATCH	Compare equal UNDER MASK	TRUE if the character field value matches the supplied *mask.* True "SKK style" masking is used for any character field referenced by this operator
NOMASK or NOMATCH	Compare unequal UNDER MASK	TRUE if the character field value does not match the supplied *mask.* True "SKK style" masking is used for any character field referenced by this operator
BETWEEN or RANGE	Compare field value to range match	TRUE if field value is within the grange set by operand values *This is the only operator that requires the use of Operand 2.*

The table above contains a list of all valid operators, and the results of their usage:

Operand(s)   Argument(s) used by operator against the specified dataname. Value(s) are dictated by the field format and operator type.

Operands act as arguments for logical comparisons to be made.

There are a numerous combinations of data and operands. This discussion will attempt to describe this and give you a feel for how to construct these statements.

With the exception of the BETWEEN (RANGE) operator, every statement must contain a single operand.

Checking for the presence of data within a field

You can check for the existence of any data format (the same way as done with the IF statement) by supplying any data dictionary field name, the EQUAL operator, followed by an operand of either YES or NO. If what ever you are testing has a value, and you specified "YES", the test is true. If you look at the way the IF statement works, the testing is basically the same. Refer to the IF discussion to find out each individual field format is processed.

It is possible to use this statement to replace the use of the IF statement, and in many cases will give you more flexibility. The IF statement is a very simple approach to select data and for that reason it was preserved in the E-SRF command-set. It is personal preference on which approach to use for this type of test.

Remember, the entire compliment of IF statements comprise a single selection criterion. Each WHEN statement, (along with its possible OR statements) is a single selection criteria.

The first criteria that fails, stops the particular item from being processed.

Comparing character fields:

When comparing character fields, you supply a character string as the operand. The character string operand is compared to the character string contained in the Data Dictionary field name (target) specified. The condition is set accordingly, based on the operator you chose and the data being compared. Comparison is left to right and continues until a not equal condition occurs, or both arguments end. If the operand length is shorter than the target data, the compare will be considered equal if the remaining characters in the target field are blank.

An example of this type of compare would be you want to select all events for a particular user. The following WHEN statement could be used:

```
WHEN(UA.NAME EQ 'KATHY WILLIMS')
```

Comparing encrypted data fields:

The data, in theory, would be processed just like character data. No attempt is made to decrypt this type of data. Whether or not the product is able to is not the issue. It is encrypted for a reason and it will stay that way. Even attempting to list the data will only yield an indicator stating the data is encrypted.

Attempting to use encrypted data in a WHEN statement will render a false condition, therefore resulting in the item not being selected.

Comparing date fields:

When comparing date fields, you supply a date as an operand. The date may be coded in either the USA or INTERNATIONAL format. Either format may be used despite what you have set for your date formatting style.

For example, if you wanted to select a list of all users who had security events on the fifteenth day of March, 1998. Either of these WHEN statements will accomplish the selection:

```
WHEN(US.DATE GTE 03/15/98)
WHEN(US.TIME GTE 15JUL/98)
```

You may use relative date positioning instead of a specified date.

*	current date (may be altered via the OPTION command)
WEEK	date of first day of the current week
WEEKEND	date of last day of the current week
MONTH	date of first day of the current month
MONTHEND	date of last day of the current month
YEAR	date of first day of the current year
YEAREND	date of the last day of the current year

The date is calculated based on the system date. The system date may be overridden at either the report or system level.

For example, if you wanted to select a list of all users who had security events for the current week, enter the following WHEN statement:

WHEN(US.DATE RANGE week weekend)

You may specify the above keyword alone, or with an offset, indicated by the + (plus) or - (minus) sign after the keyword, followed by the number of days to offset from. The 'base' date will be determined, and the specified offset will be *ADDED* or *SUBTRACTED* from the date, arriving at the actual date that will be used. Normally, the plus sign (adding of days) is not used.

For example, if you wanted to select a list of all users who had security events from yesterday for the past three days, (a running three day report), enter the following WHEN statement:

WHEN(US.DATE RANGE *-1 *-4)

Comparing numeric fields:

When comparing numeric fields, you supply a number as the operand. The numeric operand is compared to the number contained in the Data Dictionary field name specified. The condition is set accordingly, based on the operator you chose and the data being compared.

For instance, if you wanted to select a list of all users who had five or more violations on a given day, you would specify the following when statement:

WHEN(US.VIOS GE 5)

Comparing time fields:

When comparing time fields, you supply a time as an operand. The time may be coded in either the STANDARD or MILITARY format. Either format may be used despite what you have set for your time formatting style.

For example, if you wanted to select a list of all users who had security violations on or after 12:00 noon, either of these WHEN statements will accomplish the selection:

```
WHEN(US.FIRST GTE 12:00 PM)
WHEN(US.FIRST GTE 12:00:00)
```

Comparing "descriptor" fields:

Descriptor fields are single byte data elements that expand into descriptions of items, events, results, etc.

Their physical makeup is dependent on the facility providing the data. Some are "normalized" by E-SRF; others are simply placed on the Masterfile as presented from a particular Resident Security System.

You treat the data as a character string, and it will follow the same rules that character data follow. E-SRF will expand the descriptor to a preset character string that attempts to describe the particular data. Lists of descriptor data character strings are found in the appendix of the Data Dictionary reference.

For example, if you wanted to select a list of all journalized accesses that were granted for 'UPDATE", you could supply the following WHEN statement:

Appendix B of the data dictionary was used to determine the descriptor text for the particular access.

```
WHEN(UC.ACCESS EQ UPDATE)
```

Comparing using masking:

The use of the MATCH (or MASK) operator allows you to compare any character-based data to a particular mask,

The compare is either true or false, based on the matching of the target data contained in the data Dictionary field named to the supplied mask.

E-SRF was not going to introduce yet another set of masking conventions to an already complex world of masking conventions. An extensive review of various masking schemes already in existence was performed when this product was designed. The masking scheme that was the most powerful (at the time of the analysis) was the one used in ACF2. For lack of a better description, we will refer to this as SKK masking. (SKK being the company that invented it). The debate can go on forever, but SKK masking is what is used in this product.

To use masking, you supply a masking operator, a Data Dictionary name of a field that is compatible (character based) and the desired mask. The masking operators are MATCH or MASK.

The following example will show you how to set up a selection criterion that will select all datasets that start with SYS1 *or* SYS2, *and* contain the qualifier "TECHSUPT as the second level qualifier:

```
WHEN(US.RESOURCE MATCH SYS1.-)
  OR(US.RESOURCE MATCH SYS2._)
 AND(US.RESOURCE MATCH *-.TECHSUPT.-)
```

For specific samples, see the section, "Using IF and WHEN Logic for Selection Criteria", in the Report Overlays Guide.

Comparing using masking: making the reverse true

The use of the NOMATCH (or NOMASK) operator allows you to compare any character based data to a particular mask, just like MATCH (or MASK), except the selection will be made if the match is false instead of true.

The specification is identical to MATCH.

The following example will show how to set up a selection criteria that will select all datasets EXCEPT those that start with SYS1 *or* SYS2, *and* contain the qualifier "TECHSUPT as the second level qualifier:

```
    WHEN(US.RESOURCE NOMATCH SYS1.-)
      OR(US.RESOURCE NOMATCH SYS2._)
     AND(US.RESOURCE MATCH *-.TECHSUPT.-)
```

Chapter 15: SET   E-SRF System Options

SET	BANNERS / <u>NOBANNERS</u>
	COMPRESS(*mask1, ... maskn*)
	DATASET(*class name*)
	DATEFORMAT(<u>USA</u>/INTERNATIONAL)
	ELEMENTS(*number*)
	EXCLUDE (<u>*NONE*</u>/*exclude option*)
	FOOTER1(*report footer line 1 text*)
	FOOTER2(*report footer line 2 text*)
	FOOTER3(*report footer line 3 text*)
	FOOTER4(*report footer line 4 text*)
	GDG(*option*)
	HIDE(*dataname*) *option*
	LINEMODE(*linemode specification*)
	LINES(*number*)
	MAXLINES(*number*)
	MAXIMUMLINES(*number*)
	LAF(D/*code*)
	MESSAGES(*message table name*)
	SOURCE(*class name*)
	SYSIDGROUPING (*option*)
	TIMEFORMAT (<u>STANDARD</u>/MILITARY)
	TITLE(*overall report title*)
	USERID(*class name*)
	VOLUME / NOVOLUME
	WIDTH(*report format width*)

Description:

This command is used to establish E-SRF default system options. These options apply to all executions using the target E-SRF database.

Some options may be overridden by other command minor commands during their execution.

To find out what is currently in effect, RUN the ESRFSHOW report overlay.

Minor Commands: :

BANNERS / <u>NOBANNERS</u>   Specifies usage of banner pages

Produce E-SRF report separator banner pages for each undistributed report overlay that is executed. We recommend you do not use this option.

This is an additional banner page, not to be confused with the *report wrapper* described in the *Report Overlays Guide*.

This option is ignored during report distribution.

<u>*We recommend you do not use this option.*</u>

COMPRESS(RESET,*mask1, mask16*)   Maintain data set qualifier compression mask list

Specifies a list of data set qualifier compression masks. These masks may be used to identify data set name qualifiers that have numeric characters and replace the numbers with zeros. This is useful for reducing the number of related data set resource names into single resources.

For example, if you specify G0000V00 as a mask, all generation data group (GDG) qualifiers, no matter what their generation and version is, will be treated as G0000V00, and E-SRF will regard them as the same data set.

What you specify in this list is appended to the current list. The maximum number of masks E-SRF will maintain is sixteen.

If the constant RESET is specified as a mask, all entries in the list will be removed, and a new list will be built which contains all masks (if any) that follow the RESET constant.

DATASET(*class name*)   Override "grouping rule" class name for data sets

The EKC Resource Grouping Facility considers data set resources to be unique from all other resources and uses an entirely different set of processing routines for data sets, and requires a separate "rule" for each High Level Qualifier stored and processed. For more information, see the section, "*Grouping and Ownership*", in the *User Guide*.

The class name "DATASET" can be assigned internally to all data set resources.

If you wish to consider data set resources to be just like any other resource, specify this parameter with the class name you desire. IT MAY EVEN BE 'DATASET'. This will allow you to write a single "grouping rule" for all data sets.

This parameter has no effect of the classification of data set resources as "DATASET" on report output.

NOTE: The *existence* of this parameter will make the EKC Resource Grouping Facility treat data sets just like other resources, *REGARDLESS OF WHAT YOU SPECIFY AS A CLASSNAME*. Do not specify this parameter unless this is what you want.

DATEFORMAT(USA/INTERNATIONAL) Specifies date format

Sets the default appearance of all dates displayed by E-SRF.

This specification is also available as a report specific specification contained in the RUN command. If DATEFORMAT is specified on the RUN command, the RUN specification will override this specification for the duration of the report execution.

This specification may also be made for a specific OWNER by maintaining the OA.DATE data field. This specification is ignored unless the report is being executed under Automatic Report Distribution. In this case, if OA.DATE was specified for specific owners, the OA.DATE specification will override any other DATEFORMAT specification for the specific owners provisioned as such.

Note that both formats use the same number of characters.

INTERNATIONAL International date format

Dates will be formatted in the day, month year format. For example, April 2, 2003 will appear as 02APR//03 (or 02APR/2003 in CENTURY mode).

To specify this format for a particular owner, alter the owner header with the following: OA.DATE(I)

USA United States date format

Dates will be formatted in the month, day and year format. For example, April 2, 2003 will appear as 04/02/03 (or 04/02/2003 in CENTURY mode).

To specify this format for a particular owner, alter the owner header with the following: OA.DATE(U)

ELEMENTS(2000/*number*) Define maximum number of elements stored

Certain E-SRF objects are formatted with "*array*" element types. Array Elements each individually contain a single reporting information block with its related information. For example, the RESOURCE CHRONOLOGICAL object contains a list of array events. Each element will describe a particular element reported by the Resident Security System against the resource by each user who caused the event.

Array elements are stored on the Masterfile as long as the event date falls within current event "window" and the maximum capacity of elements has not been reached for a particular object.

This command sets the maximum number of array elements that may be retained on any single E-SRF Masterfile object. The number ranges from 1024 to 32,767.

This command works with the RETAIN command to help you control how much data is actually stored on your Masterfile.

The RETAIN command sets the number of DAYS data may be stored (which is the event retention "window").

The ELEMENTS command sets the total number of elements (capacity) that may be stored on a single object. Data will be purged when the FIRST of these two limits are encountered.

When elements are purged due to the RETAIN specification, old elements (*whose event dates are considered expired*) are removed from the Masterfile. This action is referred to as an "Expired Rolloff". This occurs constantly and is how E-SRF keeps your Masterfile "clean". This is what you expect to occur throughout the life of the product.

When elements are purged due to the ELEMENTS limit, the action is referred to as an "Unexpired Rolloff", and is statistically reported as such. This action should not be a common occurrence, or if it is, it should be understood why it is occurring.

When an unexpired rolloff situation is detected, TEN PERCENT of the oldest elements are dropped from the object.

Care should be exercised when setting this value above 8,000. Two issues must be considered.

First consider VSAM and storage resources as well as processor resources.

Then consider the usefulness of the data. If there are too many loggings against a particular resource, consideration may want to be made as to why the loggings are being produced. Are we logging a resource that "everyone" has access? Was there an error in an access control specification that resulted in thousands of loggings (and/or violations)?

No attempt is being made to tell you how to determine what is to be logged, allowed or denied (*or how access is to be provided*). What we are stating is thought must be expended in an effort to make any reporting of this type useful.

EXCLUDE(*exclude option*) **Specifies Update Function exclude option**

This specification will control E-SRF's ability to exclude specific resource update activity from being applied to the E-SRF Masterfile.

A function of the EKC External Grouping Facility is to determine if the resources assigned to the group are to be excluded from E-SRF update processing. If the exclude takes place, the security loggings will never be applied to the Masterfile. As far as E-SRF is concerned, the EVENTS NEVER TOOK PLACE.

NONE **Do not honor EXCLUDE specifications (Default option)**

Specify NONE if you do not want to EXCLUDE update from the Masterfile no matter what the EKC External Grouping Facility says about the resource. This is the default option and is recommended unless you specifically want to use the E-SRF exclude processing function during the Masterfile Update.

YES **Honor EXCLUDE generically at the resource level**

Specify YES if you want to EXCLUDE update from the Masterfile generically, that is not by any RECTYPE specification. This is how EXCLUDE worked in releases prior to Release 1.3.

If you were using the EXCLUDE function in release 1.1 or 1.2, this setting will cause E-SRF to process your excludes exactly the same way… dropping ALL RECTYPES for a specific resource.

RECTYPE **Honor EXCLUDE at the RECTYPE level**

Specify RECTYPE if you want to EXCLUDE update from the Masterfile at the RECTYPE level, as documented in other E-SRF publications.

This is the fullest EXCLUDE processing option within E-SRF.

FOOTER1(*footer line 1 text***)**
FOOTER2(*footer line 2 text***)**
FOOTER3(*footer line 3 text***)**
FOOTER4(*footer line 4 text***)**   **Specifies report footer line text**

These specifications provide a means to supply from one to four lines of report footer text that will appear on the bottom of every report.

Supply the one to sixty four character text string that will appear as footer information for the desired footer line. This text will be centered relative to the WIDTH specification on bottom of each report page.

If a blank line <u>is required</u>, you must specify the line with a blank character.

Omitting all four specification implies *system default* footers will not be provided.

You may define any or all lines. If you desire a blank line, you must provide a specification with blank text. You should not specify <u>unused</u> blank footer lines. If you only need footer lines one and two, do not specify three or four.

For example you are formatting two footer lines, but you want a space between them, specify: FOOTER1(*text*) FOOTER2( ). FOOTER3(*text*) Please note that "FOOTER4" was not specified.

The use of footers is governed by the presence of these specifications. If they are omitted, the only way to specify footers on reports is to do so individually using the RUN command.

These specifications, as well as the actual use of footers may be altered by specifications provided in the RUN command.

If NOFOOTERS is specified in the RUN command, these specifications are ignored for the particular report.

If footers are specified in the RUN command, they will be merged with the specifications made here, *with the RUN specifications overriding these*.

This means if you only specify line *one* in the RUN command, you will receive line *one* from the RUN command, and lines *two*, *three* and *four* will come from this command.

GDG(<u>RETAIN</u> **/ TRUNCATE)**   **Establish GDG representation option**

This option controls the way Generation Data Group (GDG) data set names are stored on the E-SRF Masterfile. This option, like the "COMPRESS" option, is used to reduce the number of objects contained on the Masterfile.

      RETAIN   **Retain all GDG generation qualifiers   (Default option)**

Specify GDG(RETAIN) if you want all of your GDGs to be individually represented, or are already masking them with the COMPRESS parameter. (*default*)

      TRUNCATE   **Eliminate all GDG generation qualifiers**

Specify GDG(TRUNCATE) to drop the G0000V00 qualifier from the data set name, and treat all GDG data sets as one.

The recommendation is to leave this mask OUT of the COMPRESS list and specify GDG(TRUNCATE).

HIDE(*dataname*) OPTION   Hide specific data from E-SRF reports

This specification may be used to "hide" specific data contained on the E-SRF Masterfile from being formatted on certain E-SRF reports.

This feature was provided to address certain privacy issues required by specific clients and should be considered a very specific feature and should only be used if the need is present.

To enable this facility, a list of up to sixteen datanames is provided by this command. If any dataname matches a fieldname specified on a FIELDS parameter, it will be rejected, (in a manner similar to specifying an invalid fialdname). The report will be executed, but the specific column containing the matched dataname will be excluded.

If your report references either the *Resource Maintenance* (RM) or *User Maintenance* (UM) objects, specifically the *x*M.OLDDATA and *x*M.NEWDATA fields, the *x*M.DATANAME contents will be matched against the HIDE dataname table. If the data contained in *x*M.DATANAME field matches the HIDE dataname list *and data is present in the target field*, the data will be replaced by the constant "<HIDDEN>".

Please note, this substitution occurs *after* data selection. The ability to match against the constant <HIDDEN> is not available.

To maintain the HIDE dataname table, specify a one to sixteen character *dataname* and an OPTION. If the option is omitted, an ADD is assumed. You may maintain from zero to sixteen datanames.

The following options are available:

ADD   Add a dataname to the dataname table.

REMOVE   Remove an existing dataname from the dataname table.

If you want to empty out the entire HIDE dataname table, specify the following command:

SET HIDE REMOVE

The contents of the HIDE dataname table may be viewed by examining the ESRFSHOW report.

LAF(D/*code*)   **Look and Feel report specification**

In E-SRF attempts were made to normalize the appearance of reports to a unique set of terminology that could cross over various RSS systems. Some terms were taken from individual RSS where it was thought to make sense. Although this seems like the ideal approach for an enterprise-wide product, customers still have a preference to particular terminology and formatting. This specification provides a means to address this.

You may specify a "look and feel" code that will alter the reports accordingly.

The following Look and Feel codes are available:

A make the report feel like ACF2

D   default report "look and feel"

R   makes the report feel like RACF

This specification is also available in a RUN command for a specific report execution. The SET specification is used unless overridden in the RUN command for a specific report execution.

This specification may also be made for a specific OWNER by maintaining the OA.LAF data field. This specification is ignored unless the report is being executed under Automatic Report Distribution. In this case, if OA.LAF was specified for specific owners, the OA.LAF specification will override any other LAF specification for the specific owners provisioned as such.

LINEMODE(STANDARD / *linemode specification*)   Establish reporting LINEMODE system default

This option provides the base setting for LINEMODE report functions.

All reports, unless specifically specified, will inherit this LINEMODE setting. This specification may be overridden by the OPTION command, which alters the default setting for the duration of the E-SRF execution, or in the report execution's RUN command.

For further information how and when to use this specification, please refer to the LINEMODE description provided in the chapter discussing the RUN statement.

LINES(*lines*)   **Set report print maximum lines**
or MAXLINES(*lines*)
or MAXIMUMLINES(*lines*)

Sets the maximum number of lines that may be printed on a report page before E-SRF automatically skips the printer to the next page.

MESSAGES(*message table name*)   **Specify message text table name**

This parameter is used to override the standard message text table (ESRF$MT). This table is used to format E-SRF message text. Specify this parameter only if you have a customized message table you wish to use instead of the table shipped with the E-SRF product. For example, messages can be translated into other languages in an alternate table.

SOURCE(NONE / classname)   **Override "grouping rule" class name for sources**

This parameter is used to establish a class that will be used to provide grouping of the Source Segment of the Masterfile.

NONE   **(Default option)**

Specification of this parameter will suppress all group structures for the SOURCE segment. This is appropriate if your installation has not provided external grouping for SOURCES.

This option suppresses all SOURCE grouping. If any reference to group names are made, the group name $ESRF.NONE will be presented as the group name.

class-name

Identifies the class name to be used by the EKC Resource Grouping Facility to determine the group name of the source. With this parameter specified, group structures for the SOURCE segment will be built.

SYSIDGROUPING(option)   **Controls resource grouping**

The EKC External Grouping Facility will group _resources_ at the SYSID level. SYSID is an eight-character field that as appended to the grouping criteria that may be used to extend the granularity of grouping resources.

Note that grouping is dynamic, meaning it is performed each time E-SRF is executed. The structures are not stored on the Masterfile. This means this parameter may be altered at any time with no ill effects. Additionally, your grouping rules may change at any time and will become effective immediately.

This specification provides a means to _control_ the SYSID provision of the External Grouping facility. External grouping uses CLASS, RESOURCE name, VOLUME and SYSID for its criteria for grouping.

You have to consider your External Grouping rules when making this specification. These considerations are stated for each setting.

Please note that this applies to actual resources, (not userids or sources). When a resource id is to be grouped, the Resource CLASS, Resource NAME and the "sysid" are used to associate the resource's group name. The SYSID grouping criteria in your rules should be omitted for USER and SOURCE rules.

Do not confuse this with the Userid Segment, where the objects within the entire segment have a "_resource_" name that consists of the USERID and the IMAGE concatenated together as one sixteen character "_resource_" name.

NONE   **(default option)**

Do not use any form of "SYSID" grouping

Specification of this parameter will prevent the "SYSID" from becoming part of the grouping criteria. This means all resources with the same name will be grouped under the same group name.

BLANKS would be presented as the "SYSID" to the External Grouping facility.

Your grouping rules should be written in such a way that SYSID is not part of any grouping criteria.

**This is the recommended setting, from both performance and usability perspectives.**

IMAGE　　　　　　　**Use the Security IMAGE as the "SYSID" name.**

Specification of this parameter will cause E-SRF to pass the current Security IMAGE ID as the "SYSID" and it will become part of the grouping criteria

Your grouping rules should be written in such a way that the SYSID grouping rule criteria is either omitted, or contains the IMAGE that the grouping rule applies to. IMAGES are grouping if DOMAINS (systems or LPARS) controlled by the ASSIGN command. See the ASSIGN command for additional information.

This is the next best setting; it will allow you to group all domains (SYSIDs or LPARS) that belong to the same IMAGE to a group name based on the resource's CLASS, NAME and IMAGE.

DOMAIN or SYSID　　　**Use the DOMAIN ID as the "SYSID" name.**

Specification of this parameter will cause E-SRF to pass the current DOMAIN (SYSTEM) ID as the "SYSID" and it will become part of the grouping criteria

Your grouping rules should be written in such a way that the SYSID grouping rule criteria is either omitted, or contains the actual DOMAIN (system or LPAR) that the grouping rule applies to.

This is the most granular, but very expensive in computer resources to specify. Grouping structures are built for every resource, containing an entry for each DOMAIN detected on the Masterfile for the particular resource. It is there if you need it, but make sure you really need it before you specify this mode. Most cases, NONE or IMAGE will provide what you need,

Avoid using the DOMAIN as the grouping SYSID unless it is necessary to do so.

TIMEFORMAT(STANDARD / MILITARY)　　　**Indicate format for time of day fields**

Sets the default appearance of all time of day information displayed by E-SRF.

This specification is also available as a report specific specification contained in the RUN command. If TIMEFORMAT is specified on the RUN command, the RUN specification will override this specification for the duration of the report execution.

This specification may also be made for a specific OWNER by maintaining the OA.TIME data field. This specification is ignored unless the report is being executed under Automatic Report Distribution. In this case, if OA.TIME was specified for specific owners, the OA.TIME specification will override any other TIMEFORMAT specification for the specific owners provisioned as such.

Note that both formats use the same number of characters.

STANDARD　　　　　**Format time in AM/PM format (Default option)**

Specification of this parameter will cause time of day information to be formatted using a twelve-hour clock. If the time were 11:30 in the evening, the time would appear as: 11:30 PM on the report. (The seconds are dropped)

To specify this format for a particular owner, alter the owner header with the following: OA.TIME(S)

MILITARY   **Format time in 24-hour military notation**

Specification of this parameter will cause time of day information to be formatted using the military notation, which is a twenty-four hour clock with seconds. If the time was 11:30 (and 29 seconds) in the evening, the time would appear as 23:30:29 on the report. (The seconds are carried out to hundredths)

To specify this format for a particular owner, alter the owner header with the following: OA.TIME(M)

TITLE(*overall report title*)   **Supply report main "title" text**

This command sets the overall main report title information that is printed on every page of every E-SRF report using the heading line function. This title cannot be modified at the report level, and stands alone when printed on reports.

The suggested information for this title is the name of your organization and any classification information you want specified on every report page. Supply one to sixty-four characters.

USERID(NONE/*class name)*)   **Override "grouping rule" class name for userids**

This parameter is used to establish a class that will be used to provide grouping of the Source Segment of the Masterfile.

NONE   **(Default option)**

This parameter is used to establish a class that will be used to provide grouping of the Source Segment of the Masterfile.

Specification of this parameter will suppress all group structures for the USER segment. This is appropriate if your installation has not provided external grouping for userids.

This option suppresses all userid grouping. If any reference to group names are made, the group name $ESRF.NONE will be presented as the group name.

class-name

Identifies the class name to be used by the EKC Resource Grouping Facility to determine the group name of the userid. With this parameter specified, group structures for USERID will be built.

VOLUME / NOVOLUME   **Include VOLUME as part of resource name**

NOVOLUME   **(Default option)**

This parameter is used to establish a class that will be used to provide grouping of the Source Segment of the Masterfile.

Specification of this parameter will prevent the VOLUME from becoming part of the resource name, both in storing data on the Masterfile and in grouping the resource.

This is the recommended setting, from both performance and usability perspectives.

VOLUME   **VOLUME is part of resource name**

Use the VOLUME as part of the resource name, both in storing the data on the Masterfile, and in grouping the resource.

Because of the way resources are managed on large file systems (i.e., DFSMS), a single resource may be represented multiple times in E-SRF. The additional records will consume more space on the E-SRF Masterfile, make the reports larger and cause confusion to the personnel who review the reports.

EKC recommends not using this setting unless you have a specific need.

WIDTH(133/*report format width*)   Report data format width

Specify a value between 80 and 256

This command allows you to establish a global report width setting for the execution of Report Overlays in the E-SRF Event System.

This specification assumes data will be formatted on a single line and includes the printer control character.

This *may* give you the ability to have reports wider than the standard 133 printer limitation that has become the standard from the onset of the IBM 1403 impact printer.

Most EKC supplied reports will fit in the 133 character de facto standard limitation. ESRFLIST however is totally user defined and could go on forever if there was no width limit provision.

If this specification is used, whenever E-SRF references the global report width specification, this is the value that will be used, unless overridden at the report execution level.

There are several considerations that are associated with this specification.

The report WIDTH value will always be used for planning a dynamically configured report, such as ESRFLIST.

The target report dataset, as well as the Report Overlay programming has ultimate control as to how wide a report can be made.

There are several considerations that are associated with this specification

At no time will E-SRF build any type of report greater than 255 characters wide.

Report width is a component of the target report output dataset. Once it is opened with a particular logical record length (LRECL), that length will stay across the entire execution, despite how this specification was overridden or used when the dataset was originally opened.

E-SRF will examine the report output dataset definitions within the MVS operating system control block structures before attempting to use this value for a particular report output dataset when opening it. ***If there was a LRECL defined with the dataset, the existing one will be used; otherwise this value will be used.***

This means if a particular report dataset has been pre-defined without a LRECL the WIDTH value will be passed to MVS for the actual dataset open. OTHERWISE THE EXISTING LRECL WILL BE USED. If the LRECL used is greater than the WIDTH specification, the report will be formatted left justified right filled with blanks. If the LRECL was less than the WIDTH specification, hen the report will be formatted left justified, with right truncation.

The role of this specification may change during report production, depending on the setting of LINEMODE.

Messages will be posted on the report wrapper indicating discrepancies between WIDTH parameters and physical datasets reports are directed to.

Chapter 16: SYNCHRONIZE   Image to a Particular RSS

SYNCHRONIZE	**DNAME(***RSS userid dataset***)** **IMAGE(***image ID***)**

Description:

This command initiates the SYNCHRONIZE update overlay for the named Security Image. Each Security Image is associated with a particular RSS and has its own unique requirements for synchronization. This command must be executed AFTER a particular Security IMAGE has been defined (CONFIGURED)

Unlike the CONFIGURE command; there may be a need to SYNCHRONIZE more often.

For example, if you have a Security Image configured for ACF2, and you have performed a UID string conversion or rebuilt userid security data off-line, you will have to re-SYNCRONIZE E-SRF with the new userid information for the IMAGE.

If this is not performed, E-SRF will upgrade users only when activity occurs, which will result in a "mixed bag" of inconsistent data for your users in the particular IMAGE.

Minor Commands:

DDNAME(*ddname*)   Specify synchronization input DDNAME

This specification allows you to override the synchronization DDNAME used to acquire synchronization input. To fully understand what (if anything) to specify in this command, consult the E-SRF documentation relative to the specific RSS you are synchronizing.

IMAGE(*imageID*)   Specify target Security IMAGE ID

Specify the Security Image you wish to synchronize. This image must have already been defined to the E-SRF Masterfile via the CONFIGURE command.

This page intentionally left blank

Chapter 17: UPDATE   Perform RSS Journal Update

UPDATE	ACF2
	DDNAME(ddname)
	NOSTEPDOWNS
	RSS(*Resident Security System*)
	SMFRECORD(*record ID*)
	STEPDOWNS

Description:

E-SRF is an EVENT repository and reporting facility. E-SRF data comes from security maintenance, and additionally the access control violations and loggings created by the Resident Security System (RSS) during its day-to-day execution. The data contained on the E-SRF database is "normalized", meaning it is in a format of its own which is independent of any one single RSS.

E-SRF supplies an UPDATE OVERLAY for each RSS supported by the product. The respective overlay is executed by this command. The overlay reads the RSS journal file and creates event transaction updates. These transactions update the E-SRF database with the event. A single RSS journal record may create as many as five E-SRF database update transactions.

It is STRONGLY RECOMMENDED that the CACHE be turned ON.
If the cache is not turned on during an update,
the update may take an enormous amount to time to complete.

Update input data integrity

Normally, RSS activity is journalized on a recording dataset, such as SMF. In an attempt to maintain control over the integrity of the data being updated to E-SRF, the Update function does not read the "raw" SMF files recording files directly. This data must be brought down in a sequential dataset first. They do not however have to be "stripped", E-SRF will read the entire dataset and select what is needed for the Update function. A control summary is provided on the Control Report summarizing the SMF dataset read, and how many of each type was detected.

E-SRF will not allow you to double-update your Masterfile by running the Update Function with the same input file more than once and therefore prevent you from corrupt your Masterfile.

As of Release 1.3, it is now possible to process a file containing multiple executions of the SMF download process on a single dataset. This appears to be a common practice. For example, a fresh file is created on a Monday morning, followed by a subsequent process of appending new data behind the previous data repeatedly for the duration of the week. At the end of the week, you may have many individual SMF downloads, but a single logical file containing one week of security SMF data. *E-SRF will now allow you to process this file as many times as necessary to extract new data that may exist on it.*

Before actually updating the Masterfile, E-SRF Update Function will examine the data to see if it was processed before. If it was not, it will use it to update the Masterfile. However, if it has seen it before, it will relate the data to a previous execution of the Update Function. E-SRF will automatically skip the number of logical records processed during the previous Update and start the verification process over again. This will continue until either new data is located, or end of data is detected. *If end of data is detected before any new update is detected, the update function will be terminated with related messages and a return code.*

Minor Commands:

ACF2   **UPDATE E-SRF using the CA-ACF2 update overlay**

This specification is obsolete and was left in for compatibility only, use the RSS(*RSS ID*) specification instead.

The ACF2 SMF records are selected from the input SMF file and analyzed. Any event that E-SRF determines applicable to security event reporting is processed. Transactions are created and passed to the E-SRF database update program for further processing.

This is an old setting and has been replaced with: RSS (*see below*).

DDNAME(SMFIN / ddname)   **Specifies DD name of input RSS security data**

This parameter is used to identify the DD name pointing to the RSS security journal data set to be presented for update.

It is possible to update from multiple SMF files in one session, as long as each SMF file being processed is specified with a unique DDNAME.

NOTE: E-SRF summarizes much of the raw journal data, making it very difficult to prevent an accidental double updating of the same data until it is too late. Therefore, E-SRF checks for a duplicate SMF UPDATE prior to applying any information to the Masterfile. *If it is determined that the SMF file being processed has been seen before, the UPDATE is aborted.*

NOTE: If the particular RSS acquires its journal data from SMF, you must specify a DDNAME that points to a flat file containing the extracted SMF data. The file cannot be the live SMF clusters that MVS uses to collect the data. The reason for this is to help insure update integrity. The flat file does not have to be a stripped file, and may contain other SMF records that are not required by E-SRF. See: SMFRECORD (*nn*)

RSS(*RSS ID*)   **Specifies the ID of the Resident Security System**

This parameter is used to identify the particular Resident Security System (RSS) the update you are applying came from.

This specification causes the RSS specific Update Overlay program to process the update input data.

As of this release, ACF2 or RACF are the only valid specifications for this parameter.

STEPDOWNS   **Accept update "step-downs"**
NOSTEPDOWNS   **Do not accept update "step-downs"**

STEPDOWNS is the default, and should be over-ridden only for specific cases.

E-SRF stores event data on the database in a specified sequence. Before it applies data, it checks to see if the data already existing on the object is more recent that what is being applied. If it is, it is considered to be a "step-down". This command will not allow step-downs to be applied to the database.

You may want to specify this option in a single IMAGE, single DOMAIN, environment, if you can guarantee that all transaction activity will be updated in sequence, and you want to force this practice if required by your auditors *userid header (UA) update will not accept STEPDOWNS regardless of the setting of this specification.* UA information is user specific, and not event specific and therefore is replaced, not added and merged with other information. E-SRF will maintain only the latest userid information and not step it down with older data that was applied to the Masterfile out of chronological order.

SMFRECORD(<u>80</u>/*record id*)   **Specifies the SMF Record ID to be selected from the Journal input file.**

The update function processor reads SMF input journals from a file that may contain many SMF record types. This parameter indicates which SMF record type contains the RSS journal data that is to be used for the update function.

The update function will call a service function to read SMF data. This routine will post a summary of all SMF record types detected. If you want to see exactly what SMF records are contained on the SMF input file, please refer to the update function control information posted on the E-SRF Control Report

This page intentionally left blank

Chapter 18: USERDELETE   Set users to "delete" status

USERDELETE	IMAGE(*Security Image ID*)
	FORCE/NOFORCE UID(*E-SRF Universal Identification mask*) USERID(*Userid*)

Description:

This command may be used to set users as being DELETED.

Users set to DELETED by this command are treated the same was as if the particular RSS indicated they were deleted.

You may delete users only within a named IMAGE.

You may delete a single user within the named IMAGE.

You may delete a one or more users that match a specified UID mask within the named IMAGE.

Precautions:

This function was never intended to be executed alone. It is a tool, which may be used with other procedures to get rid of unwanted users from your Masterfile. Depending on the scope of your specifications, you may delete anywhere from a single user to all users within the named IMAGE. This command is a "*big hammer*" and should be used with extreme care.

This function may be used to "synchronize" deleted users between E-SRF and the Resident Security System (RSS) being used to secure the named IMAGE. If this is the intended desire, this function is executed with the named IMAGE, without any other specifications, followed up by an security image SYNCHRONIZE using the most recent userid RSS data. Because SYNCHRONIZE will reinstate deleted users, only users that currently "live" on the dataset used for synchronization would be reinstated in E-SRF.

After reinstatement, the user will look the same as if the user were never deleted.

If you run this function without the SYNCHRONIZATION executed afterwards, the only way the deleted users would become reinstated is if the users themselves became active.

In E-SRF, if a userid delete is requested, the UA object is <u>marked</u> for delete. The user is still present on the Masterfile, and will appear in reports, just as if the user were not deleted. The actual removing of the user is performed in accordance with the RETENTION specification for the User header object.

You may exclude deleted users from reports by testing the UA.DELETED field the same way you would treat any other date field.

Minor Commands:

IMAGE(*Image ID*)   Security Image ID

Supply the one to eight-character Security Image ID that this function is to be executed against.

This is a MANDATORY specification, and may not be masked.

FORCE / NOFORCE   **FORCE already deleted users with current delete date**

If this specification is present, and a requested userid is already deleted, the UA.DELETED date will be upgraded with the current date.

This is an OPTIONAL specification, *NOFORCE is the default*.

UID(*UID mask*)   **E-SRF Universal Identification specification**

Supply a one to twenty-four character E-SRF UID specification that may be used to limit the users selected for delete. This data is dependent of the particular RSS associated with the target IMAGE.

The specification may be *masked,* meaning *wildcard* characters may be used. The normal E-SRF masking conventions apply and the entire field is considered a single "*qualifier*".

This is an OPTIONAL specification, if omitted, UA.UID is not evaluated during the delete operation.

USERID(*userid*)   **Individual USERID**

Supply the one to eight-character Userid to be deleted from the named IMAGE.

This is an OPTIONAL specification, and may not be masked. If supplied, E-SRF will perform a direct read against the userid in the named IMAGE. If omitted, the normal browse will be performed selecting users for deletion as specified.

Chapter 19:   **Command Reference Index**

E

F

G

H

I

J

L

M

S