

EMS Reference Summary

Abstract

This document summarizes the Event Management Service (EMS) used to report and retrieve messages. Intended for experienced users, this manual provides a quick reference for information used to create filters for events, to retrieve events either interactively or programmatically, and to report events from subsystems or applications.

Product Version

EMS F40

Supported Releases

This manual supports D30.02 and all subsequent D-series releases, and G01 and all subsequent G-series releases.

Part Number	Edition	Published	Release ID
124742	Fifth	December 1996	G01

Document History

Edition	Part Number	Product Version	Earliest Supported Release	Published
Second	100856	SPI D20, EMS D20	C30.09 and D20	September 1993
Third	106812	SPI D20, EMS D21	C30.09 and D20	May 1994
Fourth	114754	SPI D30, EMS D31	D30.02	July 1995
Fifth	124742	SPI F40, EMS F40	D30.02 and G01	December 1996

New editions incorporate any updates since the previous edition.

A plus sign (+) after a release ID indicates that this manual describes function added to the base release, either by an interim product modification (IPM) or by a new product version on a .99 site update tape (SUT).

Ordering Information

For manual ordering information: domestic U.S. customers, call 1-800-243-6886; international customers, contact your local sales representative.

Document Disclaimer

Information contained in a manual is subject to change without notice. Please check with your authorized Tandem representative to make sure you have the most recent information.

Export Statement

Export of the information contained in this manual may require authorization from the U.S. Department of Commerce.

Examples

Examples and sample programs are for illustration only and may not be suited for your particular purpose. Tandem does not warrant, guarantee, or make any representations regarding the use or the results of the use of any examples or sample programs in any documentation. You should verify the applicability of any example or sample program before placing the software into productive use.

U.S. Government Customers

FOR U.S. GOVERNMENT CUSTOMERS REGARDING THIS DOCUMENTATION AND THE ASSOCIATED SOFTWARE:

These notices shall be marked on any reproduction of this data, in whole or in part.

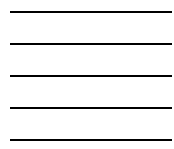
NOTICE: Notwithstanding any other lease or license that may pertain to, or accompany the delivery of, this computer software, the rights of the Government regarding its use, reproduction and disclosure are as set forth in Section 52.227-19 of the FARS Computer Software—Restricted Rights clause.

RESTRICTED RIGHTS NOTICE: Use, duplication, or disclosure by the Government is subject to the restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 52.227-7013.

RESTRICTED RIGHTS LEGEND: Use, duplication or disclosure by the Government is subject to restrictions as set forth in paragraph (b)(3)(B) of the rights in Technical Data and Computer Software clause in DAR 7-104.9(a). This computer software is submitted with “restricted rights.” Use, duplication or disclosure is subject to the restrictions as set forth in NASA FAR SUP 18-52 227-79 (April 1985) “Commercial Computer Software—Restricted Rights (April 1985).” If the contract contains the Clause at 18-52 227-74 “Rights in Data General” then the “Alternate III” clause applies.

U.S. Government Users Restricted Rights — Use, duplication or disclosure restricted by GSA ADP Schedule Contract.

Unpublished — All rights reserved under the Copyright Laws of the United States.

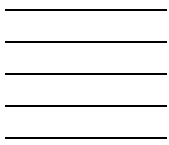


New and Changed Information

This reference summary replaces the *EMS Reference Summary* (PN 114754) released in July 1995.

The following changes have been made to this manual:

- Updated Procedure Summary Tables (Page 3-1).
- Added distributor error and warning messages, and updated tokens (Section 4).
- Updated collector error and warning message tokens (Section 5).



Contents

New and Changed Information iii

About This Manual xi

Notation Conventions xiii

1. Filters

Filter Language for Compiled Filters 1-1

Language Elements 1-1

Reserved Words 1-1

Tokens 1-1

Fields 1-2

Constants 1-3

Bit-Extraction Operator 1-4

EMF Data Types 1-5

Boolean Expressions 1-6

Functions 1-7

Statements 1-8

Filter Declaration 1-12

Parameters 1-13

Filter Compiler 1-13

Compiler Directives 1-13

Compiler Invocation 1-14

Compiler Completion Codes 1-15

Filter Tables 1-15

Filter Table Format 1-16

Key Words 1-16

Loading Filters 1-17

Restrictions 1-17

Burst Filters 1-19

Burst Filter Directives 1-20

Burst Filter Example 1-21

2. Utility Programs

EMSACOLL 2-1

EMSCCTRL 2-3

EMSCINFO 2-5

EMSDINFO 2-5

EMSDIST 2-5

3. EMS Procedures

Procedure Summary Tables	3-1
SPI Error Codes	3-2
EMSADDSUBJECT and EMSADDSUBJECTMAP	3-3
EMSADDTOKENS and EMSADDTOKENMAPS	3-4
EMSGET and EMSGETTKN	3-5
EMSINIT and EMSINITMAP	3-7
EMSTEXT	3-8
EMSTEXT Status Codes	3-9
EMSTEXT Extended Status Codes	3-10

4. Distributor Programmatic Interface

Distributor Commands	4-1
Command Summary	4-1
Extended Programmatic Interface	4-2
Object Support Summary	4-4
Common SPI Tokens	4-5
Distributor Command Errors	4-7
ADD Command (ZCOM-CMD-ADD)	4-8
ALTER Command (ZCOM-CMD-ALTER)	4-8
CONTROL Command (ZEMS-CMD-CONTROL)	4-9
DELETE Command (ZCOM-CMD-DELETE)	4-10
GETEVENT Command (ZEMS-CMD-GETEVENT)	4-11
GETVERSION Command (ZEMS-CMD-GETVERSION)	4-12
REPLACE Command (ZEMS-CMD-REPLACE)	4-12
STATUS Command (ZCOM-CMD-STATUS)	4-13
STATUS Command (ZEMS-CMD-STATUS)	4-16
Distributor Event Messages	4-17
Distributor Event Message Summary	4-17
Header Tokens	4-18
Common Data-Portion Tokens	4-19
538: ZEMS-EVT-BURST-START	4-22
539: ZEMS-EVT-BURST-END	4-23
1000: ZEMS-EVT-LOG-ACCESS	4-24
1001: ZEMS-EVT-COLL-ACCESS	4-25
1002: ZEMS-EVT-DEST-ACCESS	4-26
1003: ZEMS-EVT-LOGFILE-EOF	4-27
1005: ZEMS-EVT-BAD-FILTER	4-28
1006: ZEMS-EVT-COLL-PROTOCOL	4-29

1007: ZEMS-EVT-BAD-EVENT	4-30
1008: ZEMS-EVT-DEVTYPE	4-31
1009: ZEMS-EVT-INTERNAL-ERROR	4-32
1010: ZEMS-EVT-CHECKOPEN-FAILED	4-33
1011: ZEMS-EVT-TAKEOVER	4-34
1012: ZEMS-EVT-CREATEBACKUP-FAILED	4-35
1013: ZEMS-EVT-BACKUP-CREATED	4-36
1014: ZEMS-EVT-BACKUP-ABENDED	4-37
1015: ZEMS-EVT-BACKUP-DELETED	4-37
1016: ZEMS-EVT-CHECKPOINT-FAILED	4-38
1017: ZEMS-EVT-BAD-LOG	4-38
1018: ZEMS-EVT-FILES-LOST	4-39
1019: ZEMS-EVT-COL-DISCONNECT	4-40
1020: ZEMS-EVT-STARTUP-FAILED	4-41
1021: ZEMS-EVT-STARTUP-FAILED	4-42
1022: ZEMS-EVT-STARTUP-FAILED	4-43
Distributor Errors	4-44
Common SPI Tokens	4-44
EMS Token Codes	4-45
Distributor Warning Codes	4-46
Distributor Error Codes	4-47

5. Collector Programmatic Interface

Collector Commands	5-1
Command Summary	5-1
Common SPI Header Tokens	5-2
Common Collector Error Numbers	5-3
ADD Command (ZCOM-CMD-ADD)	5-3
ALTER Command (ZCOM-CMD-ALTER)	5-4
CONTROL Command (ZEMS-CMD-CONTROL)	5-5
DELETE Command (ZCOM-CMD-DELETE)	5-9
GETVERSION Command (ZCOM-CMD-GETVERSION)	5-9
GETVERSION Command (ZEMS-CMD-GETVERSION)	5-10
INFO Command (ZCOM-CMD-INFO)	5-10
LISTOBJECTS Command (ZCOM-CMD-LISTOBJECTS)	5-11
REPLACE Command (ZEMS-CMD-REPLACE)	5-12
STATUS Command (ZCOM-CMD-STATUS)	5-13
STATUS Command (ZEMS-CMD-STATUS)	5-15
STOP Command (ZCOM-CMD-STOP)	5-17

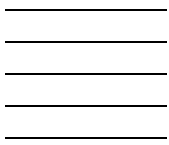
STOP Command (ZEMS-CMD-STOP)	5-17
Collector Event Messages	5-18
Primary Collector Event Message Summary	5-18
Alternate Collector Event Message Summary	5-19
Header Tokens	5-20
Common Data Portion Tokens	5-21
1-511: Tokenized Operator Console Messages	5-22
512: Tokenized Text Message	5-22
513: ZEMS-EVT-COLD-LOAD	5-23
Header Token Values	5-24
514: ZEMS-EVT-FILESWITCH	5-24
515: ZEMS-EVT-COLL-DISC-FAILED	5-25
517: ZEMS-EVT-COMPAT-DISTR-STOPPED	5-26
518: ZEMS-EVT-COL-EVENT-DISCARDS	5-27
519: ZEMS-EVT-MSGR-EVENTS-DISCARDED	5-28
520: ZEMS-EVT-FILE-ROTATE-PURGE	5-29
521: ZEMS-EVT-LOGGING-STOPPED	5-30
522: ZEMS-EVT-COLLECTOR-RUN	5-31
523: ZEMS-EVT-ACOL-EVENT-DISCARDS	5-32
524: ZEMS-EVT-LOGTIME-DECREASE	5-33
525: ZEMS-EVT-INVALIDEVENT	5-34
526: ZEMS-EVT-ACOL-INTERNAL-ERR	5-35
527: ZEMS-EVT-ACOL-SHUTDOWN	5-36
528: ZEMS-EVT-ACOL-ALLOCATESEG-ERR	5-36
529: ZEMS-EVT-ACOL-CHECKOPEN-FAILED	5-37
530: ZEMS-EVT-ACOL-TAKEOVER	5-38
531: ZEMS-EVT-ACOL-CREATEBACKUP-ERR	5-39
532: ZEMS-EVT-ACOL-BACKUP-CREATED	5-41
533: ZEMS-EVT-ACOL-BACKUP-ABENDED	5-42
534: ZEMS-EVT-ACOL-BACKUP-DELETED	5-42
535: ZEMS-EVT-ACOL-CHECKPOINT-ERR	5-43
536: ZEMS-EVT-COL-PURGETABLE-OVRFLO	5-44
537: ZEMS-EVT-COL-CONFIG-WARNING	5-44
538: ZEMS-EVT-COL-BURST-START	5-45
539: ZEMS-EVT-COL-BURST-END	5-46
540: ZEMS-EVT-COL-PLF-ERROR	5-47
Collector Errors	5-48
Common SPI Tokens	5-48
Common EMS Tokens	5-48

Collector Error Codes	5-49
-3: ZCOM-ERR-CMD-INV-IN-SUMSTATE	5-49
-4: ZCOM-ERR-CMD-MISMATCH	5-49
-5: ZCOM-ERR-CMD-NOT-SUPP	5-49
-15: ZCOM-ERR-OBJ-ALRDY-DEF	5-50
-17: ZCOM-ERR-OBJ-NOT-FOUND	5-50
-22: ZCOM-ERR-SECUR-VIOL	5-50
-23: ZCOM-ERR-SPI-ERROR	5-50
-24: ZCOM-ERR-SSID-INV	5-51
-25: ZCOM-ERR-SUB-NOT-FOUND	5-51
-26: ZCOM-ERR-TKN-CODE-INV	5-51
-27: ZCOM-ERR-TKN-DUP	5-51
-28: ZCOM-ERR-TKN-LEN-INV	5-51
-29: ZCOM-ERR-TKN-REQ	5-52
-30: ZCOM-ERR-TKN-VAL-INVAL	5-52
-32: ZCOM-ERR-VSN-INCOMP	5-52
-33: ZCOM-ERR-EMPT-RSP	5-52
-39: ZCOM-ERR-CMD-NOT-SUPP-BY-OBJ	5-53
1001: ZEMS-ERR-VERSION	5-53
1002: ZEMS-ERR-INV-CMD	5-53
1003: ZEMS-ERR-INV-SSID	5-54
1004: ZEMS-ERR-INV-TKN	5-54
1005: ZEMS-ERR-INV-VALUE	5-54
1006: ZEMS-ERR-DUP-TKN	5-54
1008: ZEMS-ERR-INV-OBJECT	5-55
1009: ZEMS-ERR-INV-CPU	5-55
1010: ZEMS-ERR-CPU-RANGE	5-55
1013: ZEMS-ERR-CDIST-CPU	5-55
1015: ZEMS-ERR-REQ-TKN	5-55
1017: ZEMS-ERR-INV-OCCURS	5-56
1019: ZEMS-ERR-FLT-FORM	5-56
1020: ZEMS-ERR-FLT-LOAD	5-56
1031: ZEMS-ERR-LOG-ACCESS	5-57
1034: ZEMS-ERR-OPEN-LOG	5-57
1036: ZEMS-ERR-DEST-ACCESS	5-57
1040: ZEMS-ERR-ZFIL	5-57
1041: ZEMS-ERR-ZSPI	5-58
1048: ZEMS-ERR-ACC-VIOL	5-58
1053: ZEMS-ERR-INV-MODE	5-58

1054: ZEMS-ERR-CDIST-DOWN	5-58
1055: ZEMS-ERR-NO-BACKUP	5-59
1056: ZEMS-ERR-ALLOC-LOG	5-59
1057: ZEMS-ERR-LOGGING-STOPPED	5-59
1059: ZEMS-ERR-MAXFLT	5-59
1060: ZEMS-ERR-FLT-ALLOC	5-60
1067: ZEMS-ERR-ZOPR-SEND	5-60
1068: ZEMS-ERR-ZOPR-SYNC	5-60

Tables

Table 1-1.	Burst Filter Directives	1-19
Table 4-1.	Distributor and Filter Object Support	4-4
Table 4-2.	Event Source and Event Destination Object Support	4-4
Table 4-3.	Forwarding Target Object Support	4-5



About This Manual

This document is a quick-reference summary of information contained in the *EMS Manual*. Like the *EMS Manual*, the *EMS Reference Summary* is intended for programmers, operators, and system managers responsible for the configuration and management of systems and networks.

Manual Contents

This manual contains the following sections:

Section 1, “Filters”	Describes compiled filters, the EMF filter language, the EMF compiler, filter tables, and burst filters.
Section 2, “Utility Programs”	Describes the programs you can run to change the configuration of or display status information about various components of EMS.
Section 3, “EMS Procedures”	Describes the procedures used by management applications to retrieve information from event messages and the procedures used by subsystems to produce event messages.
Section 4, “Distributor Programmatic Interface”	Describes the EMS distributor’s programmatic interface, including commands, event messages, and error messages.
Section 5, “Collector Programmatic Interface”	Describes the EMS collector’s programmatic interface, including commands, event messages, and error messages.

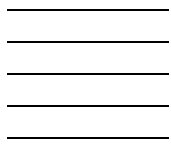
Your Comments Invited

After using this manual, please take a moment to send us your comments. You can do this by returning a Reader Comment Card or by sending an Internet mail message.

A Reader Comment Card is located at the back of printed manuals and as a separate file on the Tandem CD Read disc. You can either FAX or mail the card to us. The FAX number and mailing address are provided on the card.

Also provided on the Reader Comment Card is an Internet mail address. When you send an Internet mail message to us, we immediately acknowledge receipt of your message. A detailed response to your message is sent as soon as possible. Be sure to include your name, company name, address, and phone number in your message. If your comments are specific to a particular manual, also include the part number and title of the manual.

Many of the improvements you see in Tandem manuals are a result of suggestions from our customers. Please take this opportunity to help us improve future manuals.



Notation Conventions

General Syntax Notation

The following list summarizes the notation conventions for syntax presentation in this manual.

UPPERCASE LETTERS. Uppercase letters indicate keywords and reserved words; enter these items exactly as shown. Items not enclosed in brackets are required. For example:

```
MAXATTACH
```

lowercase italic letters. Lowercase italic letters indicate variable items that you supply. Items not enclosed in brackets are required. For example:

```
file-name
```

[] Brackets. Brackets enclose optional syntax items. For example:

```
TERM [ \system-name. ] $terminal-name
```

```
INT[ERRUPTS]
```

A group of items enclosed in brackets is a list from which you can choose one item or none. The items in the list may be arranged either vertically, with aligned brackets on each side of the list, or horizontally, enclosed in a pair of brackets and separated by vertical lines. For example:

```
LIGHTS [ ON           ]  
        [ OFF         ]  
        [ SMOOTH [ num ] ]
```

```
K [ X | D ] address-1
```

{ } Braces. A group of items enclosed in braces is a list from which you are required to choose one item. The items in the list may be arranged either vertically, with aligned braces on each side of the list, or horizontally, enclosed in a pair of braces and separated by vertical lines. For example:

```
LISTOPENS PROCESS { $appl-mgr-name }  
                  { $process-name }
```

```
ALLOWSU { ON | OFF }
```

| Vertical Line. A vertical line separates alternatives in a horizontal list that is enclosed in brackets or braces. For example:

```
INSPECT { OFF | ON | SAVEABEND }
```

... Ellipsis. An ellipsis immediately following a pair of brackets or braces indicates that you can repeat the enclosed sequence of syntax items any number of times. For example:

```
M address-1 [ , new-value ]...
```

```
[ - ] { 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 }...
```

An ellipsis immediately following a single syntax item indicates that you can repeat that syntax item any number of times. For example:

```
"s-char..."
```

Punctuation. Parentheses, commas, semicolons, and other symbols not previously described must be entered as shown. For example:

```
error := NEXTFILENAME ( file-name ) ;
LISTOPENS SU $process-name.#su-name
```

Quotation marks around a symbol such as a bracket or brace indicate the symbol is a required character that you must enter as shown. For example:

```
"[ repetition-constant-list ]"
```

Item Spacing. Spaces shown between items are required unless one of the items is a punctuation symbol such as a parenthesis or a comma. For example:

```
CALL STEPMOM ( process-id ) ;
```

If there is no space between two items, spaces are not permitted. In the following example, there are no spaces permitted between the period and any other items:

```
$process-name.#su-name
```

Line Spacing. If the syntax of a command is too long to fit on a single line, each continuation line is indented three spaces and is separated from the preceding line by a blank line. This spacing distinguishes items in a continuation line from items in a vertical list of selections. For example:

```
ALTER [ / OUT file-spec / ] CONTROLLER
      [ , attribute-spec ]...
```

!i and !o. In procedure calls, the !i notation follows an input parameter (one that passes data to the called procedure); the !o notation follows an output parameter (one that returns data to the calling program). For example:

```
CALL CHECKRESIZESEGMENT ( segment-id           !i
                        , error                 !o
                        ) ;
```

!i,o. In procedure calls, the !i,o notation follows an input/output parameter (one that both passes data to the called procedure and returns data to the calling program). For example:

```
error := COMPRESSEDIT ( filenum ) ;           !i,o
```

!i:i. In procedure calls, the !i:i notation follows an input string parameter that has a corresponding parameter specifying the length of the string in bytes. For example:

```
error := FILENAME_COMPARE_ ( filename1:length  !i:i
                          , filename2:length ) ;  !i:i
```

!o:i. In procedure calls, the !o:i notation follows an output buffer parameter that has a corresponding input parameter specifying the maximum length of the output buffer in bytes. For example:

```
error := FILE_GETINFO_ ( filenum           !i
                       , [ filename:maxlen ] ) ;      !o:i
```

Notation for Messages

The following list summarizes the notation conventions for the presentation of displayed messages in this manual.

Nonitalic text. Nonitalic letters, numbers, and punctuation indicate text that is displayed or returned exactly as shown. For example:

```
Backup Up.
```

lowercase italic letters. Lowercase italic letters indicate variable items whose values are displayed or returned. For example:

```
p-register
process-name
```

[] Brackets. Brackets enclose items that are sometimes, but not always, displayed. For example:

```
Event number = number [ Subject = first-subject-value ]
```

A group of items enclosed in brackets is a list of all possible items that can be displayed, of which one or none might actually be displayed. The items in the list might be arranged either vertically, with aligned brackets on each side of the list, or horizontally, enclosed in a pair of brackets and separated by vertical lines. For example:

```
LDEV ldev [ CU %ccu | CU %... ] UP [ (cpu,chan,%ctrlr,%unit) ]
```

{ } Braces. A group of items enclosed in braces is a list of all possible items that can be displayed, of which one is actually displayed. The items in the list might be arranged either vertically, with aligned braces on each side of the list, or horizontally, enclosed in a pair of braces and separated by vertical lines. For example:

```
LBU { X | Y } POWER FAIL
```

```
process-name State changed from old-objstate to objstate
{ Operator Request. }
{ Unknown.           }
```

| Vertical Line. A vertical line separates alternatives in a horizontal list that is enclosed in brackets or braces. For example:

```
Transfer status: { OK | Failed }
```

% Percent Sign. A percent sign precedes a number that is not in decimal notation. The %*o* notation precedes an octal number. The %*B* notation precedes a binary number. The %*H* notation precedes a hexadecimal number. For example:

```
%005400
```

```
P=%p-register E=%e-register
```

Notation for Management Programming Interfaces

UPPERCASE LETTERS. Uppercase letters indicate names from definition files; enter these names exactly as shown. For example:

```
ZCOM-TKN-SUBJ-SERV
```

lowercase letters. Words in lowercase letters are words that are part of the notation, including Data Definition Language (DDL) keywords. For example:

```
token-type
```

!r. The !*r* notation following a token or field name indicates that the token or field is required. For example:

```
ZCOM-TKN-OBJNAME          token-type ZSPI-TYP-STRING.          !r
```

!o. The !*o* notation following a token or field name indicates that the token or field is optional. For example:

```
ZSPI-TKN-MANAGER          token-type ZSPI-TYP-FNAME32.          !o
```


1 Filters

This is a reference summary. For detailed information about EMS filters, see the *EMS Manual*.

Filter Language for Compiled Filters

Describes the filter language for compiled filters, which is used to select some event messages and screen out others.

Language Elements

Describes the basic components of the filter language.

Element	Description
Comments (--)	Starts a comment line (except inside a string).
Escape Character (~)	Precedes a TACL special character (or character group) in a filter specification.
Names	Contains up to 30 letters, digits, underscores (_), and circumflexes (^) and begin with a letter. Case is not significant.

Reserved Words

Lists reserved words used by the filter language.

AND	EMSTEXTMATCH	LIST	PASS	TRUE
BEGIN	END	LITERALLY	REQUIRED	WHILE
BOOLEAN	FAIL	MATCH	SSID	
DESTINATION	FALSE	NOT	SSIDLIST	
DO	FILTER	OPTIONAL	THEN	
ELSE	IF	OR	TOKENPRESENT	

Tokens

Describes unqualified and qualified tokens.

Unqualified Tokens

```
token-name [ ( index )
```

Parameter	Description
<i>token-name</i>	The token name.
<i>index</i>	If present, an integer, with a value between 1 and 1023 inclusive, that specifies an occurrence of the token in the event message (or in the set of parameter tokens). If you omit <i>index</i> , you select the first occurrence of the token. If you specify a value for <i>index</i> that is less than 1 or greater than 1023, the filter compiler issues the following message: ... Error ...

Qualified Tokens

```
SSID ( ssid , token-name [ ( index ) ] )
```

Parameter	Description
<i>ssid</i>	Subsystem ID that specifies the token owner.
<i>token-name</i>	The token name.
<i>index</i>	An integer, with a value between 1 and 1023 inclusive, that specifies an occurrence of the token in the event message. By omitting <i>index</i> , you select the first occurrence of the token.

Fields

Provides the syntax used to refer to a field and describes each parameter.

```
token-name.struct-name [ : field-name [ ( index ) ] ] ...
    [ : field-name [ ( index ) ] ]
                [ ( range ) ]
```

Parameter	Description
<i>token-name</i>	The name of a structured token. That is, <i>token-name</i> is the name of a TACL variable that contains the token map or token code for the structured token. You can qualify the token name and include an index; see “Tokens” and “Qualified Tokens” above.
<i>struct-name</i>	The name of a TACL variable of type STRUCT that presents the structure of <i>token-name</i> .
<i>field-name</i>	The name of a field or substructure within the structure.
<i>index</i>	An integer that specifies one element of an array of fields. If you omit both <i>index</i> and <i>range</i> , <i>index</i> is zero.
<i>range</i>	A pair of indexes (integers) separated by a colon. This notation represents an array of fields, which includes the field at the first index through the field at the second index.

Constants

Describes the constant types supported by the filter language: integers, strings, subsystem IDs, file names, and constant lists.

Integers

You can use any signed integer that can be expressed internally in 64 bits.

Strings

The characters of a string constant must be enclosed within double quotes (“”).

Subsystem IDs

A subsystem ID can be referred to in either of two ways:

- As the name of a TACL variable that has been appropriately initialized.
- As a series of component parts, as shown in the following syntax diagram:

```
organization . subsystem . version
```

Parameter	Description
<i>organization</i>	The name of the organization associated with the subsystem. It includes from one to eight letters, numbers, or dashes, beginning with a letter.
<i>subsystem</i>	Either an integer or a name. Each <i>organization</i> uses integers to represent its subsystems; you can use an integer to specify <i>subsystem</i> . Names can only be used for Tandem subsystems.
<i>version</i>	Either an integer or a version identifier. An integer represents the <i>subsystem</i> version; this form is defined for all subsystems. A version identifier, such as D20, represents the version of a Tandem subsystem; this form is defined only for Tandem subsystems.

File Names

Provides the syntax for expressing a file name and describes the parameters.

```
[ \ sys-name . ] $ vol-name [ . subvol-name ] . file-name
```

Parameter	Description
<i>sys-name</i>	If present, <i>sys-name</i> is the name of the system (node).
<i>vol-name</i>	The name of the volume.
<i>subvol-name</i>	If present, this is the name of the subvolume. The filter compiler uses your file defaults if you omit <i>subvol-name</i> .
<i>file-name</i>	The file part of the file name.

Constant Lists

Provides the syntax for a constant list and describes the parameter.

```
( constant [ , constant ] ... )
```

Parameter	Description
<i>constant</i>	A number, a quoted string, a subsystem ID, or a file name. Each number in the constant list is assumed to occupy 1 byte and must be in the range 0 to 255.

```
( constant [ , constant ] ... )
```

Bit-Extraction Operator

Use the following syntax to extract bits from a specified operand creating an unsigned integer value.

```
operand . < bit-1 : bit-2 >
```

Parameter	Description
<i>operand</i>	Any token value, field, or constant that occupies a byte, word, double word, or quad word.
<i>bit-1</i>	An integer in the range 0 to 63, inclusive, that does not exceed the range implied by the data type: (0:7) for bytes, (0:15) for words, and so forth.
<i>bit-2</i>	An integer in the range 0 to 63, inclusive, that is greater than or equal to <i>bit-1</i> .

Comparisons

Provides the syntax for comparisons.

```
operand
      =
      >
      >=
      <>
      <
      <=
```

Parameter	Description
<i>operand</i>	A token name (qualified or unqualified), a field name (qualified or unqualified), a constant, or a bit-extraction operator applied to any of the language elements described above.

EMF Data Types

Describes the EMF data types used to represent each token (and field) type, and related information. In the last column, a “Y” or “N” indicates whether or not the compiler issues a warning if the data type is used in a comparison. A dash indicates information that either depends on an individual token or is not applicable.

Token Type (ZSPI-TDT-)	EMF Type	Size of Element (in Bytes)	Number of Elements	Warning?
BOOLEAN	Signed	2	1	N
BYTE	Unsigned	1	1	N
CHAR	String	1	1	N
CRTPID	Unsigned	2	4	Y
DEVICE	Unsigned	2	4	Y
ENUM	Signed	2	1	N
ERROR	Unsigned	2	7	Y
FLT	Unsigned	2	2	Y
FLT2	Unsigned	2	4	Y
FNAME	File name	24	1	N
FNAME32	String	1	32	Y
INT	Signed	2	1	N
INT2	Signed	4	1	N
INT4	Signed	8	1	N
LIST	Illegal	-	-	N
MAP	Unsigned	2	-	Y
SSCTL	Illegal	-	-	N
SSID	Unsigned	2	6	N
SSTBL	Illegal	-	-	N
STRUCT	Unsigned	2	-	Y
SUBVOL	String	1	16	Y
TIMESTAMP	Unsigned	8	1	N
TOKENCODE	Unsigned	4	1	N
TRANSID	Unsigned	8	1	Y
UINT	Unsigned	2	1	N
UNDEF	Illegal	-	-	N
USERNAME	String	1	1	Y

Boolean Expressions

Describes elements that combine to form Boolean expressions.

Element	Description
TRUE	Boolean value of a successful comparison.
FALSE	Boolean value of an unsuccessful comparison.
<i>Boolean-variable</i>	Has the value given to it by an assignment statement or the value FALSE if it is not set.
AND	(A AND B). TRUE if both A and B are TRUE; otherwise, FALSE. (If A is FALSE, B is not evaluated.)
OR	(A OR B). TRUE if A is TRUE, if B is TRUE, or if both A and B are TRUE; otherwise, FALSE. (If A is TRUE, B is not evaluated.)
NOT	(NOT A). TRUE if A is FALSE; otherwise, FALSE.
(A)	Boolean expression surrounded by parentheses.
<i>comparison</i>	See “Comparisons” earlier in this section.
EMSTEXTMATCH function	See “EMSTEXTMATCH” later in this section.
LITERALLY function	See “LITERALLY” later in this section.
MATCH function	See “MATCH” later in this section.
TOKENPRESENT function	See “TOKENPRESENT” later in this section.

Precedence of Operators

The following operators are listed in the order of their evaluation (the compiler evaluates comparisons first, then NOT, and so forth):

- Comparisons (< <= > >= = <>)
- NOT
- AND
- OR
- Assignment (:=)

Functions

Describes the Boolean functions (EMSTEXTMATCH, LITERALLY, MATCH, and TOKENPRESENT) and the SSID function.

EMSTEXTMATCH

Compares the text of an event message, as formatted by EMSTEXT, against a specified template.

```
EMSTEXTMATCH ( value , paramtkn )
```

Parameter	Description
<i>value</i>	Initial template to be supplied to EMSTEXT.
<i>paramtkn</i>	Parameter token passed to the collector or distributor containing the template against which the text generated by EMSTEXT is to be compared. The template can contain optional metacharacters.

LITERALLY

Forces case-sensitive evaluation of string of MATCH-function comparisons within a Boolean expression.

```
LITERALLY ( Boolean-expression )
```

Parameter	Description
<i>Boolean-expression</i>	Any valid Boolean expression.

MATCH

Determines whether the value of a specified token matches the pattern specified by a template.

```
MATCH ( token-specifier , template )
```

Parameter	Description
<i>token-specifier</i>	The name of any token or field whose value type is a string or a file name.
<i>template</i>	A quoted string that contains the characters—and metacharacters asterisk and question mark—to be compared with <i>token-specifier</i> . A question mark (?) in <i>template</i> matches any single character in <i>token-specifier</i> . An asterisk (*) in <i>template</i> matches zero or more consecutive characters in <i>token-specifier</i> .

SSID

Returns the first two parts (organization and subsystem number) of a subsystem ID.

```
SSID ( subsystem-id )
```

Parameter	Description
<i>subsystem-id</i>	A subsystem ID, expressed as a name or a constant.

TOKENPRESENT

Determines whether a specified token is present and returns TRUE or FALSE.

```
TOKENPRESENT ( argument )
```

Parameter	Description
<i>argument</i>	The name of a token or field.

Statements

Describes the filter language statements: assignment, compound, FAIL, IF, and PASS.

Assignment

Stores the value of a Boolean expression in a declared variable.

```
variable := Boolean-expression
```

Parameter	Description
<i>variable</i>	Any variable declared in the filter.
<i>Boolean-expression</i>	Any valid Boolean expression.

Compound

Groups multiple statements together as a single statement.

```
BEGIN [ SSID ( subsystem-id ) ]
    [ statement ; ] ...
END
```

Parameter	Description
<i>subsystem-id</i>	If present, <i>subsystem-id</i> gives the new EMF default subsystem ID.

FAIL

Stops filter execution immediately, rejecting the current event message.

```
FAIL
```


IF

Provides for the conditional execution of a statement following THEN and a statement following ELSE.

```
IF expression THEN statement-1 [ ELSE statement-2 ]
```

Parameter	Description
<i>expression</i>	Any Boolean expression.
<i>statement-1</i>	Any statement, simple or compound. <i>statement-1</i> is executed only if <i>expression</i> is TRUE.
<i>statement-2</i>	Any statement, simple or compound. If present, <i>statement-2</i> is executed only if <i>expression</i> is FALSE.

PASS

Directs the collector or distributor to print, forward, or return the current event message (and pass value, if any), depending on the activity being performed by the collector or distributor.

```
PASS [ n ] [ m ] | PASS [ n ] ( m1, m2, ... mN )
```

Parameter	Description
<i>n</i>	Any signed integer constant whose value is expressible as a 16-bit signed integer. Include this value to tell your application which PASS statement returned a particular event message. The pass value allows you to categorize event messages to prevent redundant computations in your application.
<i>m</i> , <i>m1</i> ... <i>mN</i>	Routing IDs as defined in the DESTINATION statements.

DESTINATION

```

DESTINATION
  RID  n ,                ! Routing ID           : required
  TYPE  type ,           ! Dest type           : required
  NAME  fname ,         ! Dest name           : required
  FORMAT ON | OFF ,     ! Event formatting    : optional
  RECLEN n ,            ! Desired rec len     : optional
  INDENT n ,           ! Indentation         : optional
  OBJECT fname ,       ! Program object      : optional
  IN  fname ,          ! IN file             : optional
  OUT  fname ,         ! OUT file            : optional
  STARTUP "text" ,    ! Startup text        : optional
  PRI  n ,             ! Priority             : optional
  CPU  n               ! Cpu                 : optional
  TIMEOUT sec         ! Timeout             : optional

```

Parameter Description

RID	Routing ID. This can be an integer in the range from 1 to 255. It need not be unique within one filter; that is, more than one destination may have the same routing ID. The RID is used in the PASS statement to indicate to which destination the event is to be routed.
NAME	Name of the routing destination. It can be a local or remote name. Currently, the EMF (EMS filter) language requires file names to begin either with a dollar sign (\$) or a backslash (\) character. If a node name is not given, the distributor's node will later be used. To allow the distributor's home terminal to be specified as a destination, the name \$HOME has been reserved. To avoid hard coded file names, a DEFINE name may be given that is later resolved by the distributor.
TYPE	is PROCESS DISK PRINTER PRIMCOL ALTCOL CONSOLE Supplied for informational purposes only; it makes the filter easier to maintain. No checking is done by either the compiler or the distributor to verify this parameter.
FORMAT	Indicates if events for this destination should be formatted by EMSTEXT or submitted in binary form. The default is set to ON (formatted).
RECLEN	For formatted events, indicates how many columns are to be formatted per line. The largest length accepted is 4096. If not specified, the device's record length is substituted.
INDENT	Specifies by how many columns lines are to be indented, excepting the first line of the event. This number must be smaller than RECLEN. Default is 36.

Parameter	Description
OBJECT	<p>Program file name of the destination process. It must be fully qualified; if the node name is not given, the name of the distributor's node will later be substituted. Currently, the EMF (EMS filter) language requires file names to begin with either a dollar sign (\$) or a backslash (\) character. To avoid hard coded file names, a DEFINE name may be given that is later resolved by the distributor. The destination type must be of type PROCESS, although the compiler does not enforce this.</p> <p>The parameters IN, OUT, STARTUP, PRI, and CPU are optional but may be given only if OBJECT is specified.</p>
IN	<p>Input file name of the destination process. It must begin with either a dollar sign (\$) or a backslash (\) character. If a node name is not given, the distributor's node is used. To avoid hard coded file names, a DEFINE name may be given that is later resolved by the distributor. OBJECT must be specified (see OBJECT parameter description). If omitted, the distributor's IN file will be used.</p>
OUT	<p>Output file name of the destination process. It must begin with either a dollar sign (\$) or a backslash (\) character. If a node name is not given, the distributor's node is used. To avoid hard coded file names, a DEFINE name may be given that is later resolved by the distributor. OBJECT must be specified (see OBJECT parameter description). If omitted, the distributor's OUT file will be used.</p>
STARTUP	<p>Startup sequence used to be sent to the destination process. It must be text imbedded within quotes; current maximum length is set to 128 characters. OBJECT must be specified (see OBJECT parameter description).</p>
PRI	<p>Priority of the destination process. If not specified, the priority of the distributor is inherited. OBJECT must be given (see OBJECT parameter description).</p>
CPU	<p>CPU number for the destination process. If not specified, or not available, the distributor picks the next available CPU, excluding its own. OBJECT must be given (see OBJECT parameter description).</p>
TIMEOUT	<p>Maximum time in seconds that this destination process should be allowed to process an event before sending a response to the distributor. The distributor selects the largest timeout value from all current destinations and waits for that amount of time for any destination to respond to a write request.</p>

Filter Declaration

Provides the syntax for declaring a filter and describes the parameters.

```
FILTER filter-name [ ( param [ , param ] ... ) ];
  BEGIN [ SSID ( subsystem-id ) ]
    [ variable-decls ]
    statement [ ; statement ] ...
  END;
```

Parameter	Description
<i>filter-name</i>	An identifier.
<i>param</i>	The qualified name of a parameter token, optionally followed by the keyword REQUIRED or the keyword OPTIONAL .
<i>SSID (subsystem-id)</i>	If present, gives the subsystem ID that changes the EMF default subsystem ID within the filter. If you omit this parameter, the NULL subsystem ID becomes the EMF default subsystem ID.
<i>variable-decls</i>	If present, is the keyword BOOLEAN , followed by a list of identifiers separated by commas.
<i>statement</i>	Any filter-language statement.

Parameters

Provides the syntax for the **OPTIONAL** and **REQUIRED** parameters.

<i>param</i>	[REQUIRED]
	[OPTIONAL]

Parameter	Description
<i>param</i>	Token name that is qualified by the subsystem ID of the user subsystem.

Filter Compiler

Includes information about compiling a filter specification.

Compiler Directives

Describes the **?LIST**, **?NOLIST**, and **?SOURCE** directives supported by the compiler.

?LIST

Causes current source statements to be included in the compiler output file.

?LIST

?NOLIST

Causes current source statements to be excluded from the compiler output file.

?NOLIST

?SOURCE

Causes the compiler to include language text from the specified file.

```
?SOURCE file-name
```

Parameter	Description
<i>file-name</i>	File name of an EDIT file (file code 101) with language text.

Compiler Invocation

The compiler reads a filter specification in an EDIT file and produces a filter object file.

EMF Command

The EMF command invokes the compiler from TACL.

```
EMF / [ { IN } source ]
      [ , { INV }
        { OUT } listing ]
        { OUTV }
      [ , run-option ] ...
      / [ object-file ]
```

Parameter	Description
<i>source</i>	The filter specification. Use the IN keyword if <i>source</i> is a file name; use the INV keyword if <i>source</i> is a TACL variable name.
<i>listing</i>	The listing destination. Use the OUT keyword if <i>listing</i> is a file name; use the OUTV keyword if <i>listing</i> is a TACL variable name.
<i>run-option</i>	Any of the following:
<i>CPU cpu-number</i>	Integer in the range of 0 through 15. Specifies CPU number of primary process.
<i>LIB [file-name]</i>	Selects user library file that is searched by program.
<i>MEM num-pages</i>	Maximum number of virtual data pages to allocate for new process. Integer in the range of 1 through 64.
<i>PRI [priority]</i>	Run-time priority of alternate collector process. Integer in the range of 1 through 199.
<i>STATUS</i>	Indicates why a process stops.
<i>SWAP</i>	Specifies name of file used to hold virtual data of process.
<i>TERM</i>	Specifies home terminal for new process.
<i>object-file</i>	The file name that contains the compiled filter. The default object file name is FOBJECT, on the current volume and subvolume.

Compiler Completion Codes

Summarizes the codes returned after compilation.

Completion Code	Description
0	Successful compilation
1	Warning messages only; listing and object file complete
2	Errors and warnings; listing complete, no object file
3	Fatal error; premature termination, incomplete listing, no object file
4	Compilation not started; no compiler output
5	Internal compiler error
6	Compiler was stopped by a TACL STOP command or by a STOP procedure call from a process
8	Object file named ZZEFnnnn; listing and object file complete

Filter Tables

EMS collectors and distributors support filter tables that can be used instead of or in addition to compiled filters. All distributor types will accept specification of multiple filters in the startup line.

Filter Table Format

A filter table is represented as an EDIT file. When loaded into a collector or distributor, the table is automatically converted into a filter object and saved as a disk file. The format of the filter table EDIT file is as follows, given here as an example:

```
?COMMENT
EMS FILTER TABLE

?EMPHASIS 1
?USER 165,168

?COMMENT

owner      subsys      event#          passval

?PASS
TANDEM     EMS          1000..1050      10
           TMF          *                20
           TLAM        -3              30
           513          31
           139          32
MYSYS      15           100..500        40
YOURSYS    *            50
```

Key Words

The following key words may be specified as filter table directives, preceding the actual filter table:

- ?PROCESS *name* where *name* is the ID of the process that generated the event. The comparison is not case sensitive, but the match must be exactly for the number of characters given (maximum of six for Cxx based events), except when the wild-card character (*), an asterisk, appears as a last character (for example, \$SP*).
- ?NODE *n* where *n* is the node number of the system that generated the event. *n* must be smaller than 255 for Cxx based events.
- ?CPU *n* where *n* is number of the CPU in which the event generator executed. *n* must be smaller than 16 for Cxx based events.
- ?PIN *n* where *n* is the PIN number of the process that generated the event. *n* must be smaller than 255 for Cxx based events.
- ?EMPHASIS *n* where *n* is either (0) or (1) and signifies the severity of the event. If set to (1), the event is considered critical.

?USER <i>n, m</i>	where <i>n</i> is the group number and <i>m</i> the user number of the user ID associated with the process that generated the event. Either one must be smaller than 256. The user number may be the wild-card character (*), an asterisk (example: 255,*).
?START <i>time</i>	where <i>time</i> is the start time given in hh:mm (hours:minutes, local civil time). If stop time is not given, the following midnight 0:00 is assumed. Any event that has a generation time outside the start/stop interval fails the filter. The date portion of the timestamp is not considered; that is, the filter operates on a daily basis and need not be updated. This feature is currently not available with compiled filters.
?STOP <i>time</i>	where <i>time</i> is the stop time given in hh:mm (hours:minutes, local civil time). Stop time must be larger than start time. If start time is not given, the previous midnight 0:00 is assumed. Any event that has a generation time outside the start/stop interval fails the filter. The date portion of the timestamp is not considered; that is, the filter operates on a daily basis and need not be updated. This feature is currently not available with compiled filters.
?STYPE [<i>n</i>]	where <i>n</i> is an enumeration for the standard event type. The possible range for <i>n</i> is defined in the event standards section of the ZEMSDDL file. Zero is not a valid type. If <i>n</i> is omitted, it is assumed that the event number column in the filter table is to be interpreted as standard event type column.
?UTYPE [<i>n</i>]	where <i>n</i> is an enumeration for the user defined event type. Some possible values for <i>n</i> are defined in the event standards section of the ZEMSDDL file. Zero is not a valid type. Note that ?STYPE and ?UTYPE are mutually exclusive. If <i>n</i> is omitted, it is assumed that the event number column in the filter table is to be interpreted as user defined event type column.

Loading Filters

The SPI command ADD may be used to load multiple filters or a single filter into a collector or distributor. Filter names are submitted as multiple object token values. The DELETE command can be used to remove a filter, or replace a set of filters with a PASSALL default filter. Note that the CONTROL command cannot be used for loading multiple filters because it does not provide object tokens to specify filter files. Only a single filter may be configured when using the CONTROL command.

Restrictions

In a filter table, the maximum number of events per subsystem, when subtracting the lowest event number from the highest, cannot exceed 4096. The maximum number of subsystem IDs per owner, when subtracting the lowest from the highest, cannot exceed 256.

Filter Table Errors

The following errors may be returned when the conversion from filter table text file to filter table object fails:

Required param missing	= -10
Too many subsystems	= -11
Open problem	= -12
Editreadinit problem	= -13
Editread problem	= -14
PASS or FAIL missing	= -15
Invalid SSID	= -16
Unknown SSID	= -17
Invalid event number	= -18
Invalid format for range	= -19
Too many event numbers	= -20
Table overflow	= -21
Duplicate owner	= -22
Duplicate subsystem	= -23
Invalid passval	= -24
Passval not allowed	= -25
Invalid column position	= -26
Wildcard placement error	= -27
Subsystem entry required	= -28
Owner entry required	= -29
Event entry required	= -30
Invalid owner	= -31
No filter table entry	= -32
Invalid start time	= -40
Invalid stop time	= -41
Invalid user ID	= -42
Invalid standard type	= -43
Invalid cpu number	= -44
Invalid pin number	= -45
Invalid process ID	= -46
Invalid standard type	= -47
Invalid user defined type	= -48
Invalid emphasis	= -49
Invalid keyword	= -50
Duplicate keyword	= -51
Event type incompatibility	= -52
Invalid burst parameter N	= -53
Invalid burst parameter T1	= -54
Invalid burst parameter T2	= -55
Invalid burst parameter T3	= -56
Invalid burst parameter S	= -57
Invalid burst parameter L	= -58
Directive SUPPRESS missing	= -59
Incompatible directives	= -60
Object file purge error 48	= -100
Object file create error	= -101
Object file open error	= -102
Object file write error	= -103

Burst Filters

A burst filter is a type of filter table that is used to identify the criteria for event burst detection and suppression (BDS) in a collector or distributor. When a burst filter has been installed in a collector or distributor, BDS prevents the forwarding of any event burst that exceeds the specified number of events per time duration. In collectors, BDS can also be enabled by means of the SUPPRESS parameter of the EMSACOLL and EMSCCTRL programs. However, a burst filter and the SUPPRESS parameter cannot be used simultaneously to implement BDS for a given collector.

Note. “Burst Detection and Suppression” in the *EMS Manual* describes the BDS feature of EMS in detail, including the methods used to configure and implement BDS in collectors and distributors.

Burst Filter Directives

Table 1-1 identifies and describes the directives used to configure a burst filter. Note that all directives must be preceded by a question mark (?).

Table 1-1. Burst Filter Directives

Filter Directive	Meaning
?SUPPRESS	Designates the filter as a burst filter. This directive is required and must precede all other burst filter configuration directives.
?N <i>m</i>	Represents the number of similar events repeated within T1 seconds that constitute a burst condition. The variable <i>must</i> be in the range of from 2 to 32,767 events. The default value is 100.
?T1 <i>time</i>	Represents the time interval during which similar events are counted to determine if a burst has occurred. The variable <i>time</i> is given in seconds and must be in range of from 1 to 3,600 seconds. The default value is 120 seconds.
?T2 <i>time</i>	Represents the time interval during which a bursting event must occur at least once; otherwise, the burst is considered to have ended. The variable <i>time</i> is provided in seconds and must be in the range of from 1 to 3600 seconds. The default value is 120 seconds.
?T3 <i>time</i>	Represents the time interval chosen for periodic checking of burst end conditions. The timer starts when the first burst condition is detected and cancels when there are no more burst conditions to monitor. The variable <i>time</i> is given in seconds and must be in the range of from 15 to 3600 seconds. The default value is 300 seconds.
?S <i>m</i>	Represents the number of simultaneous bursts that can be monitored. This also determines the size of the burst table. The variable <i>m</i> must be in the range of from 2 to 128 bursts. The default value is 6 bursts. The larger the value that is chosen for S, the higher the probability that event bursts will be detected, but at the expense of increased processing time duration and table size.
?L <i>m</i>	Represents the length, in bytes, of the subject value that is used to determine if events are similar. The variable <i>m</i> must be in the range of from -1 to 254 bytes (the default value is 254 bytes). If set to -1, subject token and value are not considered in the determination of similar events; only the event number and subsystem ID are used in the comparison. If set to zero, the subject token ID is compared, but the value is not. Note that the value chosen for L impacts the size of the burst table.

Note. Other table keywords (except ?COMMENT) and filter table entries cannot appear together with these burst filter directives. Only one burst filter can be loaded into a collector or distributor; however, other filter tables and compiled filters can be added. The order of execution is the same as the order in which the filters are added.

Burst Filter Example

The following is an example EDIT file for a burst filter.

```
?COMMENT
Burst Detection/Suppression
?SUPPRESS
! Number of event occurrences for a burst
?N 50
! Burst start detection interval (seconds)
?T1 60
! Burst end detection interval (seconds)
?T2 120
! Periodic check for burst end (seconds)
?T3 30
! Maximum number of simultaneous bursts
?S 20
! Subject length to compare
?L 0
```

See “Burst Detection and Suppression” in the *EMS Manual* for examples of the consequences of different BDS configurations and a summary of the methods that can be used to configure and enable BDS from the primary collector, an alternate collector, and a distributor.

2 Utility Programs

This is a reference summary. For detailed information about EMS programs, see “EMS Programs” in the *EMS Manual*.

EMSACOLL

The object program for the alternate collector.

```
EMSACOLL [ / run-options / ] [ altcol-options ]
```

Parameter/Options

run-options

NAME [*process-name*]

NOWAIT

CPU *pri-cpu*

PRI *priority*

altcol-options

BACKUP *backup-cpu*

BLOCKING { ON | OFF }

LOGSUBVOL *subvol-name*

DEFAULTSUBVOL *subvol-name*

SUPPRESS [(*suppression-parameters*)]

FILTER { *filter-name* |
(*filter-name* [, *filter-name*
] ...) }

Description

Any of the valid run options, separated by commas. Options of particular importance:

Name of alternate collector process.

If specified, returns control to the TACL.

CPU number of primary process.

Run-time priority of alternate collector process. Value must be in the range 1 through 199.

Any of the following alternate collector options, separated by commas:

CPU number for backup process.

Assigns the value of the BLOCKING attribute.

Subvolume to be used for log files.

Subvolume to be used if log subvolume becomes inaccessible.

Allows the default values for the burst detection and suppression (BDS) parameters to be overridden with user-supplied values. Use of the SUPPRESS keyword causes the alternate collector to be started with BDS enabled. The absence of the SUPPRESS keyword caused BDS to be disabled at startup unless a burst filter has been specified in the FILTER keyword.

Selects the filter or filters to be used for pre-log filtration (PLF). Each *filter-name* variable is the name of a filter file. The named filter can be a compiled filter, a filter table, or a burst filter. If a burst filter is named, then burst detection and suppression (BDS) is automatically enabled.

Parameter/OptionsLOGPREFIX *char***Description**

Allows the user to specify a different prefix pattern for an alternate collector's configuration file and log files. The default pattern for the context file is ZZEVCONF, and the default pattern for the log files is ZZEVnnnn. The *char* variable can be any alphanumeric character (0 through 9; A through Z) and modifies the third character of the prefix pattern. For example, if the *char* variable is Y, then the context file is named ZZYVCONF and the log files are name ZZYVnnnn.

ROTATEFILES { ON | OFF }

Determines what action collector takes when MAXFILE files exist in log subvolume and collector needs to create a new one.

ON (TRUE): Primary collector purges oldest file. Alternate collector renames oldest file if attributes are correct; otherwise, the file is purged.

OFF (FALSE): Event 521 (ZEMS-EVT-LOGGING-STOPPED) is issued and logging is stopped. Alternate collector checks the oldest file. If file is not valid, alternate collector creates next file in sequence.

MAXFILE *nnnn*

Maximum number of files alternate collector retains in log subvolume. Integer in the range 2 through 1000.

EXT {*ext*|(*pri,sec*)}

Selects values (in pages) for PRIMARYEXTENT and SECONDARYEXTENT attributes. *ext*, *pri*, and *sec* are even integers in the range 2 through 65534.

BUFFERED { ON | OFF }

ON directs disk process to buffer records in disk cache buffer, rather than writing them immediately to disk.

REFRESH { ON | OFF }

Selects value of EOFREFRESH attribute.

SECURITY *rwep*

Selects value of PROTECTION attribute.

REPLYAFTERWRITE { ON | OFF }

ON directs alternate collector to ensure each event is written to disk before replying to event generator.

ALLOCATE

Causes alternate collector to attempt to create and allocate MAXFILE files in log subvolume in the range ZZEVnnnn through ZZEVnnnn+MAXFILE.

POOLPAGES *pages*

Determines number of 2048-byte pages alternate collector uses for buffering events. Must be in the range 20-128.

EMSCCTRL

Controls the operation of the primary and alternate collectors and the attributes of collector log files. It also controls the compatibility distributor.

```
EMSCCTRL [ / run-options / ] [ HELP ]
[ col-name col-options ]
[ acol-name STOP ]
```

Parameter/Options

Description

run-options

Any of the valid run options, separated by commas.

HELP

Displays brief syntax description of EMSCCTRL; omitting collector name and all collector options is equivalent to selecting HELP.

col-name

Name of the primary collector (\$0) or alternate collector on local system.

acol-name

Name of an alternate collector on local system. *acol-name STOP* initiates shutdown of named alternate collector.

col-options

One or more of the following collector options, separated by commas:

ALLOCATE

(Alternate collector only). Causes alternate collector to attempt to create/allocate MAXFILE files in current log subvolume. See “EMSACOLL” (earlier in this section) for range of log file numbers created/allocated.

BLOCKING { ON | OFF }

Assigns the value of the BLOCKING attribute.

BUFFERED { ON | OFF }

ON directs disk process to buffer records in disk cache buffer, rather than write them immediately to disk.

CDISTMODE
{ *selection-criterion* }

Specifies mode of compatibility distributor and selection criterion used by compatibility distributor to determine which messages to print. Allowable values for *selection-criterion*:
CRITICAL-ONLY ON
CRITICAL-ONLY OFF

CDISTSTOP

Stops compatibility distributor.

CDISTUSER
usergroup.username

Specifies user ID used by compatibility distributor to access CONSOLE. Parameter only used when remote logging device is specified.

EXT { *ext* }
{ (*priext* , *secext*) }

Selects values (in pages) to give primary and secondary extents.

Parameter/Options

FILTER { *filter-name* |
(*filter-name* [, *filter-name*
] ...) }

LOGSUBVOL [*subvol*]

LOGUSERID *usergroup.username*

MAXFILE *nnnn*

NEXTLOGFILE

REFRESH { ON | OFF }

ROTATEFILES { ON | OFF }

SECURITY { *rwep* | "*rwep*" }

Description

Selects the filter or filters to be used for pre-log filtration (PLF). Each *filter-name* variable is the name of a filter file. The named filter can be a compiled filter, a filter table, or a burst filter. If a burst filter is named, then burst detection and suppression (BDS) is automatically enabled.

If *subvol* is present, gives name of volume and subvolume in which a collector will create log files. If *subvol* is omitted, option directs collector to use user's current volume and subvolume.

User ID that the primary collector uses when it accesses its log files.

Limits number of log files that can exist at one time in the volume and subvolume specified by LOGSUBVOL. *nnnn* must be in the range 2 to 1000 (decimal).

Directs collector to close the current log file, create a new one, and open it.

ON tells a primary or alternate collector to update the end-of-file pointer on disk for each block written.

Determines action collector will take when MAXFILE files exist in log subvolume and collector needs to create a new one.
ON (TRUE): Primary collector purges oldest file. Alternate collector renames oldest file if attributes are correct; otherwise, file is purged.
OFF (FALSE): Event 521 (ZEMS-EVT-LOGGING-STOPPED) is issued and logging is stopped. Alternate collector checks the oldest file. If file is not valid, alternate collector creates next file in sequence.

Specifies file security that collector is to use when creating a log file. System supplies "COOO" if SECURITY is not specified.

Parameter/Options

SUPPRESS [(*suppression-parameters*)]

SWITCH { COLL *cpu* }
 { CDIST *cpu* }

TEXTOUT [*console*]

Description

Allows the default values for the burst detection and suppression (BDS) parameters to be overridden with user-supplied values. Use of the SUPPRESS keyword causes the alternate collector to be started with BDS enabled. The absence of the SUPPRESS keyword caused BDS to be disabled at startup unless a burst filter has been specified in the FILTER keyword.

Designates which process of a process pair is to become the primary process. Process is specified by CPU number.

Specifies a console device (CONSOLE), in place of the default console selected at cold load. \$Z0 routes event messages to specified console.

Note. The CDISTMODE, CDISTSTOP, CDISTUSER, SWITCH CDIST, and TEXTOUT options can be used only with the primary collector.

EMSCINFO

Displays up-to-date collector operational information and event-message statistics. The DETAIL option can be specified to display detailed BDS and PLF information. Refer to the *EMS Manual* for detailed descriptions of EMSCINFO display format and terminology.

```
EMSCINFO [collector name] [, DETAIL ]
```

EMSDINFO

Displays up-to-date distributor information and filter statistics. Refer to the *EMS Manual* for a detailed description of the EMSDINFO display format and terminology.

```
EMSDINFO distributor-name
```

EMSDIST

The object program for a consumer, forwarding, or printing distributor process.

```
EMSDIST [ / run-options / ] dist-options ]
```

Parameter/Options*run-options*NAME [*process-name*]CPU *cpu*

NOWAIT

PRI *priority**dist-options*BACKUP *cpu*

TYPE	{ CONSUMER	C }
	{ FORWARDING	F }
	{ PRINTING	P }

```
COLLECTOR
{ name
  ( name [ ,name ] ... ) }
```

LOGFILE *name*

```
FILTER
{ name
  ( name [ ,name ] ... )
}
```

TARGET *collector-name*

```
TEXTOUT { name
         { ( name [ ,name ] ... )
       }
```

Description

Any of the valid run options, separated by commas. Options of particular importance:

Names the distributor process. For the D-series release of EMS or later, you must supply the NAME [*process-name*] information. For the C-series release of EMS, if you omit *process-name*, the system assigns a process name of the form \$Znnn.

Processor in which primary process of compatibility distributor is to execute.

Returns TACL prompt while distributor runs.

Execution priority of primary distributor process.

Any one of the following distributor options, separated by commas:

Processor in which backup process of distributor is to execute.

Type of distributor to be started. Required option.

Name (or list of names) of collectors. Parentheses are required if multiple collectors are specified. *name* may be:

```
$0 \system.$0
alternate-collector-name
\system.alternate-collector-
name.
```

Maximum of ten COLLECTOR names allowed. COLLECTOR option excludes LOGFILE option.

Name of log file that is the source of event messages for this distributor.

Name or list of names of filters. Multiple filters are executed in sequential order for each event. The maximum allowed is ten filters. An EDIT file is accepted for a filter table or burst filter. When a burst filter is specified, BDS is activated when the distributor starts. If you omit the FILTER parameter when starting up a printing or consumer distributor, a default PASSALL filter is used.

(For a forwarding distributor only.) Name of primary or alternate collector to which event messages will be forwarded.

(For printing distributor only.) Name (or a list of names) of devices, processes, and disk files that receive display-formatted event messages. Parentheses are required in list alternative. Maximum of ten TEXTOUT names allowed.

Parameter/Options

WAIT *n*

TIME { *date* }
 { [*date*] *time* }

STOP { *date* }
 { [*date*] *time* | }
 { EOF }

DELAY { *time* }

SBUF { ON | OFF }

AUTOSTOP { *time* }

Description

Overrides the two minute default time-out for all textout destinations specified to a printing distributor. *n* is given in seconds. If WAIT is specified, the distributor retries every *n* seconds, even if there is only one destination. If the textout is a routing destination, the event is skipped. If a file system error occurs, and WAIT is specified, the distributor does not retry and skips to the next line. This situation can occur with certain unprintable characters. If WAIT is not specified, the error is retried indefinitely.

Specifies generation time of first event message to be examined by the distributor.

date can take the following forms:

year - *month*# - *day*
month *day* *year*
day *month* *year*

year is a four-digit integer.

month# is the number of the month, one or two digits.

month is a three-letter abbreviation for the month (JAN, FEB, MAR... and so on)

day is a one-digit or two-digit integer.

Hyphens are used to separate the parts of the date in the first example. Spaces are used as separators in the second and third examples.

time takes the following form:
hour: *minute* [: *second*]

hour is a one-digit or two-digit integer.

minute is a one-digit or two-digit integer.

second is a two-digit integer.

Used with TIME option to specify range of values (ending date and time) to be examined by distributor. See TIME for a description of values.

Delays by *time* after detection of EOF before reading log again. If EOF is received again, a message is sent to the collector asking for the next event. *time* is given in sec/100.

Selects or deselects sequential block buffering for all log files except the current log file associated with a collector.

Specifies how long a consumer distributor stays open after the last opener has closed it. *time* is given in seconds.

Parameter/Options**Description**INDENT *n*

Overrides the default indentation (36) for a printing distributor. *n* is the number of indentation spaces. *n* must not be larger than the record size of the destination.

GMT ON

Converts the local time back to GMT (Greenwich Mean Time) for printing distributors. Time is displayed in the original GMT instead of LCT (Local Civil Time).

DUMP ON

Causes the printing distributor output to be in a labeled token dump format instead of EMSTEXT format. Numerical values are displayed in hexadecimal.

3 EMS Procedures

This is a reference summary. For detailed information about EMS procedures, see “EMS Procedures” in the *EMS Manual*.

Procedure Summary Tables

Describes EMS procedures that retrieve event information.

The following tables summarize the procedures used to process event messages. The second column uses the following codes: Y means a procedure with reference token parameters, N means a procedure with value token parameters, and a dash (-) means neither Y or N apply.

Name	Token-Parameter Type	Description
EMSGET	Reference	Retrieves tokens and their values or related information from an event message.
EMSGETTKN	Value	
EMSTEXT	(does not apply)	Produces displayable text derived from an event message.

Describes EMS procedures that report events.

Name	Token-Parameter Type	Description
EMSINIT	Value	Initializes an event-message buffer: first step in event message creation.
EMSINITMAP	Reference	
EMSADDTOKENS	Value	Adds one to four tokens to a buffer that has already been initialized by EMSINIT or EMSINITMAP.
EMSADDTOKENMAPS	Reference	
EMSADDSUBJECT	Value	Adds an additional subject to an event-message buffer.
EMSADDSUBJECTMAP	Reference	

Describes additional procedures that process event messages.

Name	Token-Parameter Types	Description
OPEN	(does not apply)	Opens \$0.#ZSPI, <i>altcol</i> .#ZSPI, or a distributor for command messages. Opens \$0 or an alternate collector for event messages.
CLOSE	(does not apply)	Ends communication begun by the OPEN procedure.
WRITEREAD	(does not apply)	Transports command messages to and from the distributor and collectors for the STATUS, GETEVENT, GETVERSION, and CONTROL commands. Reports standard event messages to \$0 and any alternate collectors.

Name	Token-Parameter Types	Description
WRITE	(does not apply)	Transport TEXT messages to collectors. The text after 104 bytes is truncated.
SSPUT SSPUTTKN	Reference Value	Used only for special operations such as positioning.
SSMOVE SSMOVETKN	Reference Value	Moves tokens from one message buffer to another.
DEVICEINFO2	(does not apply)	Determines whether a named process is an EMS collector.

SPI Error Codes

Lists and describes the SPI error codes returned by the EMS procedures.

Error Code	Description
0	No error
-1	Invalid buffer format
-2	Illegal parameter value
-3	Missing parameter
-4	Illegal parameter address
-5	Token too large for buffer
-6	Invalid checksum
-7	Internal error
-8	Token not found
-9	Illegal token code or map
-10	Invalid subsystem ID
-11	Operation not supported
-12	Insufficient stack space
-30	Buffer length larger than ZEMS-VAL-EVT-BUFLLEN

EMSADDSUBJECT and EMSADDSUBJECTMAP

Adds a subject to an event-message buffer.

```

{ status := } { EMSADDSUBJECT } ( buffer ! i/o
{ CALL      } { EMSADDSUBJECTMAP } , subject-token-id ! i
                                     , [ subject-value ] ! i
                                     , [ subject-length ] ! i
                                     , [ ssid ] ..! i
                                     );
    
```

Parameter	Value / Ref	Description
<i>status</i>	value	INT. Zero or SPI error code.
<i>buffer</i>	.EXT:ref:*	INT. Event-message buffer, which caller must allocate and initialize.
<i>subject-token-id</i>	value .EXT:ref:*	INT(32). (EMSADDSUBJECT) INT. (EMSADDSUBJECTMAP) Token code or token map of subject token to be added to event message.
<i>subject-value</i>	EXT:ref:*	STRING. Value of new subject token. Include if subject token has associated value.
<i>subject-length</i>	value	INT. Length, in bytes, of <i>subject-value</i> field. Ignored unless the <i>subject-token-code</i> is defined as variable-length token.
<i>ssid</i>	.EXT:ref:6	INT. Subsystem ID that qualifies token code. If not supplied or equal to zero (6*[0]), defaults to one of the following: If current position is in a list = subsystem ID of current list. If current position is not in a list = subsystem ID in event-message header (ZSPI-TKN-SSID).

EMSADDTOKENS and EMSADDTOKENMAPS

Adds one to four tokens to a buffer previously initialized by EMSINIT or EMSINITMAP.

```

{  status  := }  { EMSADDTOKENS }  (  buffer          ! i/o
{  CALL    }  { EMSADDTOKENMAPS }  , [  ssid ]       ! i
                                           ,  tkn-triplet   ! i
                                           [ , tkn-triplet ] ! i
                                           [ , tkn-triplet ] ! i
                                           [ , tkn-triplet ] ! i
                                           );
    
```

Parameter	Value / Ref	Description
<i>status</i>	value	INT. Zero or SPI error code.
<i>status</i>	value	INT. Zero or SPI error code.
<i>buffer</i>	.EXT:ref:*	INT. Event-message buffer, which caller must allocate and initialize.
<i>ssid</i>	.EXT:ref:6	INT. Subsystem ID that qualifies token code.
<i>tkn-triplet</i>		Token code or token map of token to be added to buffer followed by token value and length of value. Syntax: <i>token-id</i> , [<i>data-buf</i>], [<i>data-len</i>]
<i>tkn-triplet</i>		Token code or token map of token to be added to buffer followed by token value and length of value. Syntax: <i>token-id</i> , [<i>data-buf</i>], [<i>data-len</i>]
<i>token-id</i>	value .EXT:ref:*	INT(32). (EMSADDTOKENS) INT. (EMSADDTOKENMAPS) Token code or token map of token to be added to event message.
<i>data-buf</i>	EXT:ref:*	STRING. Value of token to be added to event message. Parameter required if token takes a value.
<i>data-len</i>	value	INT. Length, in bytes, of value in <i>data-buf</i> . Parameter ignored unless <i>token-id</i> is defined as variable-length token. If not present and token is variable length, then the length must be in the first 2 bytes of <i>data-buf</i> .

EMSGET and EMSGETTKN

Provides tokens and their values or related information from an event message.

```

{ status := } { EMSGET } ( buffer ! i/o
{ CALL } { EMSGETTKN } , token-id ! i
, [ token-value ] ! i/o
, [ index ] ! i
, [ count ] ! i/o
, [ ssid ] ); ! i/o
    
```

Parameter	Value / Ref	Description
<i>status</i>		INT. Zero or SPI error code.
<i>buffer</i>	.EXT:ref:6	INT. SPI buffer from which information is to be extracted.
<i>token-id</i>	.EXT:ref:* value	INT. (EMSGET) INT (32). EMSGETTKN) Token code (EMSGETTKN), or pointer (EMSGET) to token code or token map. Identifies token to be retrieved. If <i>token-id</i> is ZSPI-TKN-DATALIST, ZSPI-TKN-ERRLIST, or ZSPI-TKN-LIST, EMSGET or EMSGETTKN selects list so that subsequent calls can retrieve tokens within list.
<i>token-value</i>	.EXT:ref:*	STRING. Variable in which requested token value is to be returned. Can be used as input parameter for control and positioning operations.
<i>index</i>	value	INT. >0 = absolute index for <i>token-id</i> , starting from beginning of the buffer or list. If 0 or not supplied, EMSGET or EMSGETTKN returns next occurrence of token code after current position in buffer.
<i>count</i>	.EXT:ref:1	INT. Input and output count parameter. On call, specifies maximum number of token values to return. On return, specifies actual number of token values returned.
<i>ssid</i>	.EXT:ref:6	INT. Subsystem ID that qualifies token code. If not supplied or equal to zero (6*[0]), <i>ssid</i> defaults to one of the following: If current position is in list = subsystem ID of current list. If current position is not in list = subsystem ID in message header (ZSPI-TKN-SSID).

EMSINIT and EMSINITMAP

Initializes an event-message buffer: first step in event-message creation.

```

{ status := } { EMSINIT } ( buffer ! i/o
{ CALL } { EMSINITMAP } , buffer-length ! i
, generating-ssid ! i
, event number ! i
, subject-token-id ! i
, [ subject-value ] ! i
, [ subject-length ] ! i
, [ subject-ssid ] ! i
, [ time-stamp ] ! i
;
    
```

Parameter	Value / Ref	Description
<i>status</i>	value	INT. Zero or SPI error code.
<i>buffer</i>	.EXT:ref:*	INT. Event-message buffer, which caller must allocate and EMSINIT (or EMSINITMAP) will initialize.
<i>buffer-length</i>	value	INT. Length, in bytes, NOT TO EXCEED 4024, of buffer—the length to be initialized.
<i>generating-ssid</i>	.EXT:ref:6	INT. Subsystem ID of subsystem originating event message. The version field of this parameter is stored in the event message.
<i>event-number</i>	value	INT. Number, specific to subsystem, identifying event message. The version is stored in the header.
<i>subject-token-id</i>	value .EXT:ref:*	INT(32). (EMSINIT). INT. (EMSINITMAP). Token code or token map of event message subject.
<i>subject-value</i>	.EXT:ref:*	STRING. Value of subject to be added to event message.
<i>subject-length</i>	value	INT. Length, in bytes, of <i>subject-value</i> field. Ignored unless <i>subject-token-code</i> is defined as variable-length token. If not present and token is variable length, then the length must be in the first 2 bytes of <i>subject-value</i> .
<i>subject-ssid</i>	.EXT:ref:6	INT. Subsystem ID of subsystem to which subject token belongs. If omitted or all zeros (6*[0]) supplied, default is <i>generating-ssid</i> . The version field of this parameter is not stored in the event message.
<i>timestamp</i>	value	FIXED. Time stamp in Julian GMT (64-bit) format for event message. If omitted, current time is used.

EMSTEXT

Produces displayable text derived from an event message.

```

{ status := } EMSTEXT ( event-message-buffer      ! i
{ CALL      }      ( displayable-text-buffer     ! o
                  , display-line-length        ! i
                  , number-display-lines      ! i
                  , actual-lengths           ! o
                  , [ header-template-key ]   ! i
                  , [ indent ]                ! i
                  , - RESERVED -              -
                  , [ extended-status ]       ! o
                  ) ;

```

Parameter	Value / Ref	Description
<i>status</i>	value	INT(32). Pair of 16-bit-integer status codes.
<i>event-message-buffer</i>	.EXT:ref:*	INT. Buffer containing event message from which displayable text will be derived.
<i>displayable-text-buffer</i>	.EXT:ref:*	STRING. Buffer in which EMSTEXT stores displayable text.
<i>display-line-length</i>	value	INT. Number of character spaces (bytes) in each logical line in <i>displayable-text-buffer</i> .
<i>number-display-lines</i>	value	INT. Number of logical lines needed in <i>displayable-text-buffer</i> to display message text.
<i>actual-lengths</i>	.EXT:ref:*	INT. Array of <i>number-display-lines</i> integers.
<i>header-template-key</i>	.EXT:ref:10	INT. Key of a format template. Used by EMSTEXT to format <i>headers</i> of all event messages for display as text.
<i>indent</i>	value	INT. Hanging indent value; defines how many blanks will precede text in every logical line of <i>displayable-text-buffer</i> except first.
RESERVED	value, int	This parameter reserved for future use.
<i>extended-status</i>	EXT:ref:1	INT(32). Pair of 16-bit-integer status codes. Value is undefined if returned value of EMSTEXT (the simple status) is (0, <i>x</i>) and <i>x</i> > 0.

EMSTEXT Status Codes

Describes the status code pairs returned by a function call to EMSTEXT.

Status Codes		Description
0	0	Normal return.
0	22	Bad parameter (address or value); no displayable text.
0	29	Required parameter missing; no displayable text.
0	632	Insufficient stack space; no displayable text.
10	0	Template file problem, or no template and no TEXT token for event; or EMSTEXT error.
11	0	Bad event buffer.

EMSTEXT Extended Status Codes

Describes the extended status code pairs returned by a function call to EMSTEXT.

Status Codes		Description
0	0	Normal return.
1	<i>x</i>	ALLOCATESEGMENT returned error <i>x</i> because it did not get private segment for EMSTEXT.
2	<i>x</i>	Problem with the template file. <i>x</i> > 0: file-management error from the OPEN procedure <i>x</i> = -1: the file code is not 839 or 844 <i>x</i> = -2: the file is not a disk file <i>x</i> = -3: the file is not key-sequenced <i>x</i> = -4: the file has the wrong record size <i>x</i> = -5: the file has the wrong primary-key definition
3	<i>x</i>	File-management error <i>x</i> when reading template file.

Status Codes	Description
4 <i>x</i>	<p><i>event-message-buffer</i> is bad.</p> <ul style="list-style-type: none"> x = 0: First word in buffer is not - 28, or buffer length is wrong x = 1: SSPUT failed to position to front of buffer x = 2: EMSGET for ZEMS-TKN-CRTPID failed x = 3: EMSGET for ZEMS-TKN-SYSTEM failed x = 4: EMSGET for ZEMS-TKN-GENTIME failed x = 5: EMSGET for ZEMS-TKN-EVENTNUMBER failed x = 6: EMSGET for ZSPI-TKN-SSID failed x = 7: EMSGET for ZEMS-TKN-XSYSPID failed with status code other than ZSPI-ERR-MISTKN (possible only for an EMS event that should contain the XSYSPID token) x = 8: EMSGET for ZEMS-TKN-OPMSG failed with status code other than ZSPI-ERR-MISTKN x = 9: EMSGET for ZEMS-TKN-TEXT failed with status code other than ZSPI-ERR-MISTKN x = 10: EMSGET for ZSPI-TKN-DEFAULT-SSID failed x = 11: EMSGET failed for a token referenced in the template, with status code ZSPI-ERR-INVBUF, ZSPI-ERR-NOSPACE, or ZSPI-ERR-XSUMERR x = 12: EMSGET or SSPUT of ZSPI-TKN-POSITION failed (trying to save or restore the position within the buffer) x = 13: EMSGET for ZEMS-MAP-EXIOADDR failed with status code other than ZSPI-ERR-MISTKN x = 14: EMSGET for ZEMS-TKN-EMPHASIS failed
5 <i>x</i>	<p>File-management error <i>x</i> while reading the nonresident template file for the initial template.</p>
6 <i>x</i>	<p>Problem with the format template</p> <ul style="list-style-type: none"> x = 0: unrecognized version number for the template structure x = 1: unrecognized item-type code in the template x = 2: no edit-descriptor item to match an edit-marker item in the template x = 3: unrecognized edit code in the template x = 4: unrecognized token type or structure-field type in an edit-descriptor item x = 5: bad DATE edit-string x = 6: bad TIME edit-string x = 7: bad item length (extends past the end of the template) x = 8: EMSGET failed for a token referenced in the template, with status code other than ZSPI-ERR-MISTKN and other than one of those listed for error (4,11) above. x = 9: cannot find the end of the true branch of an *IF x = 10: cannot find the conditional-expression definition for an *IF x = 11: unrecognized operation code in a conditional-expression item
7 <i>x</i>	<p>Private-segment access-error <i>x</i></p>
8 <i>x</i>	<p>Other error</p> <ul style="list-style-type: none"> x = 0: more than 10 levels of nesting of MSG edit codes x = 1: the format template referenced a token that was not found x = 3: the buffer is too small for the format template
9 <i>x</i>	<p>File-management error <i>x</i> while reading a nonresident template file for a template referenced by a MSG format code</p>

Status Codes		Description
12	<i>x</i>	PROCESS_HANDLE_TO_FILENAME error <i>x</i>
13	<i>x</i>	PROCESS_HANDLE_DECOMPOSE error <i>x</i>
14	<i>x</i>	FILENAME_DECOMPOSE error <i>x</i>

4

Distributor Programmatic Interface

This is a reference summary. For detailed information, see the *EMS Manual*.

Distributor Commands

Describes the command and response messages for the consumer, printing, and forwarding distributors. For more detailed information, see “Distributor Commands and Responses” in the *EMS Manual*.

Command Summary

Summarizes each distributor command.

Command Name	Description
ZCOM-CMD-ADD	Adds filters, event sources, and event destinations (textouts). This command supports the distributor’s extended programmatic interface and is described next under “Extended Programmatic Interface.”
ZCOM-CMD-ALTER	Alters distributor attributes, replaces filter parameters, and replaces the forwarding target collector. This command supports the distributor’s extended programmatic interface and is described under “Extended Programmatic Interface.”
ZEMS-CMD-CONTROL	Changes operational environment of distributor. CONTROL can change the source of event messages (and the position within the source at which the distributor begins examining event messages), the filter and filter parameters, and the destinations of event messages that pass the filter. The CONTROL command may be replaced by the ADD, ALTER, and DELETE commands, which support the distributor’s extended programmatic interface.
ZCOM-CMD-DELETE	Deletes filters, event sources, and event destinations (textouts). This command supports the extended programmatic interface and is described under “Extended Programmatic Interface.”
ZEMS-CMD-GETEVENT	Provides the next event message that passes the current filter. GETEVENT applies only to consumer distributors.
ZEMS-CMD-GETVERSION	Provides the version number of the programmatic interface.
ZEMS-CMD-REPLACE	Replaces one configured filter with another filter.
ZEMS-CMD-STATUS	Gets information about the distributor, its associated collectors, and event-message traffic. This command supports SPI Common Extensions and is described under “Programmatic Interface.”

Extended Programmatic Interface

An extended SPI interface has been added to the distributor to more fully comply with the SPI common extensions standards (refer to the *SPI Common Extensions Manual* for more information on common extensions standards). However, the distributor's extended, object-oriented command set was developed before the SPI common extensions standard was released. This command set does not fully comply with the SPI common extensions (extended SPI) standard.

The object-oriented distributor commands provide additional functionality relative to the distributor commands that still use the basic SPI interface. The extended programmatic interface for the distributor supports multiple or field-level wild-card objects in these commands and, where appropriate, returns multiple-object information. The objects supported are the EMS distributor (ZCOM-OBJ-DIST), EMS filter (ZCOM-OBJ-FILTER), event source (ZCOM-OBJ-SOURCE), forwarding target (ZCOM-OBJ-TARGET), and event destination (ZCOM-OBJ-TEXTOUT).

For all of these object types, an object hierarchy exists. In these cases, ZCOM-OBJ-FILTER, -OBJ-SOURCE, -OBJ-TARGET, and -OBJ-TEXTOUT are all considered to be subordinate to ZCOM-OBJ-DIST. However, no hierarchical naming is allowed or supported.

The distributor commands that the extended programmatic interface supports are ZCOM-CMD-ADD, ZCOM-CMD-ALTER, ZCOM-CMD-DELETE, and ZCOM-CMD-STATUS (note that all have a "ZCOM-" prefix in the command name). The extended programmatic interface requires specification of a ZCOM object type and verb when the command is initialized by SSINIT, as well as a ZCOM-TKN-OBJNAME token.

Distributor commands that are still supported by the original basic SPI interface are ZEMS-CMD-CONTROL, ZEMS-CMD-GETEVENT, ZEMS-CMD-GETVERSION, and ZEMS-CMD-REPLACE. Note that these basic SPI interface-compliant commands all have a "ZEMS-" prefix in the command name. While the ZEMS-CMD-REPLACE command requires an object name, the other ZEMS- commands do not.

The EMS distributor distinguishes between the two interfaces by checking for the presence of the ZSPI-TKN-OBJECT-TYPE token, which is only present in the ZCOM-commands.

The following SPI tokens and values can be used to add or delete filters, event sources, event destinations (textout, target), and to set distributor attributes. The following table identifies the ZCOM- commands that can be used in place of their corresponding CONTROL commands.

SPI Commands to Use Instead of CONTROL Commands

ZSPI-TKN-COMMAND	ZSPI-TKN-OBJECT-TYPE	CONTROL Command
ZCOM-CMD-ADD	zcom-obj-source	ZEMS-TKN-CONNECT-SRC-COLL
ZCOM-CMD-ADD	zcom-obj-source	ZEMS-TKN-CONNECT-LOG
ZCOM-CMD-ADD	zcom-obj-filter	ZEMS-TKN-FILTERFILE
ZCOM-CMD-ADD	zcom-obj-textout	ZEMS-TKN-ADD-TEXTOUT
ZCOM-CMD-ALTER	zcom-obj-filter	ZEMS-TKN-REPLACE-PARAM
ZCOM-CMD-ALTER	zcom-obj-dist	ZEMS-TKN-GMTTIME
ZCOM-CMD-ALTER	zcom-obj-dist	ZEMS-TKN-LOGTIME
ZCOM-CMD-ALTER	zcom-obj-target	ZEMS-TKN-REPLACE-TGT-COLL
ZCOM-CMD-ALTER	zcom-obj-dist	ZEMS-TKN-EOFDELAY
ZCOM-CMD-ALTER	zcom-obj-dist	ZEMS-TKN-SEQ-BLOCKING
ZCOM-CMD-DELETE	zcom-obj-filter	ZEMS-TKN-RESET-FILTER
ZCOM-CMD-DELETE	zcom-obj-source	ZEMS-TKN-DISCONNECT-SRC-COLL
ZCOM-CMD-DELETE	zcom-obj-source	ZEMS-TKN-DISCONNECT-LOG
ZCOM-CMD-DELETE	zcom-obj-textout	ZEMS-TKN-DELETE-TEXTOUT

The ZCOM-CMD-ALTER command can also be used to allow positioning by generation time or logtime and altering the settings of the EOF delay and sequential blocking ON/OFF parameters. All of these attributes are submitted in a control structure; that is, more than one attribute can be changed at a time:

```
DEFINITION ZEMS-DDL-CONTROL-DIST.
  02 ZDIST-GMTTIME           TYPE ZSPI-DDL-TIMESTAMP.
  02 ZDIST-LOGTIME          TYPE ZSPI-DDL-TIMESTAMP.
  02 ZDIST-DELAY-PRESENT    TYPE ZSPI-DDL-BOOLEAN.
  02 ZDIST-EOFDELAY         TYPE ZSPI-DDL-INT.
  02 ZDIST-BLOCKING-PRESENT TYPE ZSPI-DDL-BOOLEAN.
  02 ZDIST-SEQ-BLOCKING     TYPE ZSPI-DDL-BOOLEAN.
END
```

Object Support Summary

Tables 4-1, 4-2, and 4-3 show which distributor commands can be issued to which object, whether or not multiple objects can be specified, and whether or not object wild cards (*) are supported. Table 4-1 shows distributor command support for the distributor object (ZCOM-OBJ-DIST) and the filter object (ZCOM-OBJ-FILTER). Table 4-2 shows distributor command support for the event source object (ZCOM-OBJ-SOURCE) and the event destination object (ZCOM-OBJ-TEXTOUT). Table 4-3 shows distributor command support for the forwarding target object (ZCOM-OBJ-TARGET).

In each table's object columns, the values presented have the following meaning:

- “Yes” signifies that the distributor command supports the object type; “No” signifies that the command does not support the object type.
- “Only 1” signifies that only one instance of that object type is supported; “>=1” signifies that more than one instance of that object type is supported; “*” signifies that wild-card object types are supported.

Table 4-1. Distributor and Filter Object Support

Distributor Command	ZCOM-OBJ-DIST	ZCOM-OBJ-FILTER
ZCOM-CMD-ADD	No	Yes, >=1
ZCOM-CMD-ALTER	Yes, Only 1	Yes, >=1, *
ZCOM-CMD-DELETE	No	Yes, >=1, *
ZEMS-CMD-REPLACE	No	Yes, Only 1
ZCOM-CMD-STATUS	Yes, Only 1, *	Yes, Only 1, *
ZEMS-CMD-GETVERSION	Yes, Only 1	No
ZEMS-CMD-GETEVENT	Yes, Only 1	No

Table 4-2. Event Source and Event Destination Object Support

Distributor Command	ZCOM-OBJ-SOURCE	ZCOM-OBJ-TEXTOUT
ZCOM-CMD-ADD	Yes, >=1	Yes, >=1
ZCOM-CMD-ALTER	No	No
ZCOM-CMD-DELETE	Yes, >=1, *	Yes, >=1, *
ZEMS-CMD-REPLACE	No	No
ZCOM-CMD-STATUS	Yes, Only 1, *	Yes, Only 1, *
ZEMS-CMD-GETVERSION	No	No
ZEMS-CMD-GETEVENT	No	No

Table 4-3. Forwarding Target Object Support

Distributor Command	ZCOM-OBJ-TARGET
ZCOM-CMD-ADD	No
ZCOM-CMD-ALTER	Yes, Only 1
ZCOM-CMD-DELETE	No
ZEMS-CMD-REPLACE	No
ZCOM-CMD-STATUS	Yes, Only 1, *
ZEMS-CMD-GETVERSION	No
ZEMS-CMD-GETEVENT	No

Common SPI Tokens

Describes SPI tokens that appear in more than one distributor command.

Header Tokens

Describes SPI header tokens that are assigned values for distributor commands.

Token (ZCOM-TKN- or ZSPI-TKN-)	Type (ZSPI-TYP-)	Description
COMMAND	ENUM	Standard SPI command-number token; values are: ZEMS-CMD-GETVERSION 0 ZEMS-CMD-STATUS 1 ZEMS-CMD-CONTROL 2 ZEMS-CMD-GETEVENT 3 ZCOM-CMD-STATUS 8 ZCOM-CMD-ALTER 5003 ZCOM-CMD-ADD 5005 ZCOM-CMD-DELETE 5007 ZEMS-CMD-REPLACE 30000
OBJECT-TYPE	ENUM	Contains the object, if any, to which the command is applied. The supported object types and their respective values are: ZCOM-OBJ-DIST 70 ZCOM-OBJ-SOURCE 72 ZCOM-OBJ-TEXTOUT 73 ZCOM-OBJ-TARGET 74 ZCOM-OBJ-FILTER 75
MAXREP	INT	If not zero, the distributor encapsulates the response in a data list. If multiple objects occur, they are all enclosed in the same data list.

Token (ZCOM-TKN- or ZSPI-TKN-)	Type (ZSPI-TYP-)	Description
OBJNAME	STRING	Specifies the fully-qualified name of the object to be selected for processing. Used only by ZCOM- type commands.
SERVER-VERSION	UINT	Version number of distributor's programmatic interface
SSID	SSID	Contains the subsystem ID for EMS (ZSPI-VAL-TANDEM.ZSPI-SSN-ZEMS.ZSPI-VAL-VERSION).

Data Tokens

Describes SPI tokens that are used in distributor commands.

Token (ZCOM-TKN- or ZSPI-TKN-)	Type (ZSPI-TYP-)	Description
ENDLIST	SSCTL	SPI token used to end lists, including error
ERRLIST	LIST	SPI token that begins an error list
RETCODE	ENUM	Standard SPI return token. If zero, the command was successful. If other than zero, the command was unsuccessful. The response message will contain an error list if an error or warning event has occurred.
SERVER-BANNER	CHAR50	Standard banner information.

Distributor Command Errors

Describes the errors returned in ZSPI-TKN-RETCODE if RETCODE is not zero.

Command Interpretation Errors

Describes the errors that can occur while interpreting a distributor command.

Error Name (ZEMS-ERR-)	Error Number	Parm Err?	Description
VERSION	1001	No	Version not supported
INV-CMD	1002	No	Invalid command
INV-SSID	1003	No	Invalid subsystem ID (SPI buffer not owned by EMS)
INV-TKN	1004	Yes	Extra or unrecognized token
INV-VALUE	1005	Yes	Invalid token value
DUP-TKN	1006	Yes	Duplicate token code
MODE-CONFLICT	1007	Yes	Command not applicable to distributor type
INV-OBJECT	1008	Yes	Invalid object type—must be null
INV-OP	1014	Yes	Token not allowed in this context
REQ-TKN	1015	Yes	Missing required token
INV-HEADERTYPE	1016	No	Invalid SPI header type

Distributor Operation Errors

Describes the errors that can occur while executing a distributor command.

Error Name (ZEMS-ERR-)	Error Number	Description
COLL-ACCESS	1018	Distributor cannot access collector.
LOG-ACCESS	1031	Distributor cannot access log file.
DEST-ACCESS	1036	Printing distributor cannot access TEXTOUT, or forwarding distributor cannot access remote collector.
DEVTYPE	1045	Distributor cannot access collector's log file.
COLL-PROTOCOL	1046	Distributor cannot interpret collector status reply.
COLL-DISCONNECT	1050	Distributor disconnected specified collector.

ADD Command (ZCOM-CMD-ADD)

Adds objects to the distributor. Supported objects are filters (ZCOM-OBJ-FILTER), event source (ZCOM-OBJ-SOURCE), and event destination (ZCOM-OBJ-TEXOUT).

Command

ZCOM-CMD-ADD

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
<filter param tokens>	token-type	ZSPI-TYP-ENUM.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

ALTER Command (ZCOM-CMD-ALTER)

Alters objects in the EMS distributor. Supported objects are distributors (ZCOM-OBJ-DIST), filters (ZCOM-OBJ-FILTER), and target collectors (ZCOM-OBJ-TARGET).

Command

ZCOM-CMD-ALTER

Tokens in Command Buffer (-OBJ-DIST)

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZEMS-MAP-CONTROL-DIST	token-type	ZEMS-DDL-CONTROL-DIST.

Tokens in Command Buffer (-OBJ-FILTER)

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
<filter parameter token(s)>		

Tokens in Command Buffer (-OBJ-TARGET)

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

CONTROL Command (ZEMS-CMD-CONTROL)

Controls the distributor operations environment.

Note. IUse the distributor's ADD, ALTER, and DELETE commands instead of the CONTROL command. The CONTROL command does not support multiple filters.

Command

ZEMS-CMD-CONTROL

Tokens in Command Buffer (excluding header tokens)

ZEMS-TKN-FILTERFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-REPLACE-PARAM	token-type	ZSPI-TYP-BOOLEAN
ZEMS-TKN-RESET-FILTER	token-type	ZSPI-TYP-BOOLEAN
ZEMS-TKN-GMTTIME	token-type	ZSPI-TYP-TIMESTAMP
ZEMS-TKN-LOGTIME	token-type	ZSPI-TYP-TIMESTAMP
ZEMS-TKN-REPLACE-TGT-COLL	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-CONNECT-SRC-COLL	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-DISCONNECT-SRC-COLL	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-CONNECT-LOG	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-DISCONNECT-LOG	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-ADD-TEXTOUT	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-DELETE-TEXTOUT	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-DISCONNECT-LOG	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-EOFDELAY	token-type	ZSPI-TYP-INT
ZEMS-TKN-SEQ-BLOCKING	token-type	ZSPI-TYP-BOOLEAN

Tokens in Response Buffer

ZSPI-TKN-RETCODE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
. . .		
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

CONTROL Command RETCODE Values

Gives the name, number, and description of each CONTROL command error (or warning).

Error Name	Error Number	Description
ZEMS-ERR-FLT-FORM	1019	Bad filter format
ZEMS-ERR-FLT-LOAD	1020	Failed to load filter
ZEMS-ERR-REQ-PARAM	1022	Required parameter missing
ZEMS-ERR-HIST-MODE	1024	Bad operation on log file source
ZEMS-ERR-MAX-COLLECTOR	1025	Five collectors already
ZEMS-ERR-COLLECTOR-EXISTS	1026	Already connected
ZEMS-ERR-COLL-NOT-FOUND	1027	Not connected; cannot disconnect
ZEMS-ERR-EOF	1032	Last record read from log file
ZEMS-ERR-NO-POOL	1043	No space for another collector
ZEMS-WRN-TOO-EARLY	502	Position too far in the past
ZEMS-WRN-TOO-LATE	503	Position too far in the future
ZEMS-ERR-FORWARD-SEARCH	1033	Missing next file to search
ZEMS-ERR-NO-EVENT-SOURCE	1044	Specify source collector
ZEMS-ERR-MAX-DEST	1035	Exceeds five TEXTOUT destinations.
ZEMS-ERR-DEST-NOT-FOUND	1038	Destination not found
ZEMS-ERR-DEST-EXISTS	1037	Destination already added
ZEMS-ERR-STAT-ONLY	1052	Access to specified distributor not permitted

Entries marked with an asterisk (*) are warnings (which use the prefix ZEMS-WRN-) rather than errors (which use the prefix ZEMS-ERR-).

DELETE Command (ZCOM-CMD-DELETE)

Deletes objects from the EMS distributor, including filters (ZCOM-OBJ-FILTER), events source (ZCOM-OBJ-SOURCE), and event destination (ZCOM-OBJ-TEXTOUT). At least one object name token is required.

Command

ZCOM-CMD-DELETE

Tokens in Command Buffer

ZPSI-TKN-MAXRESP

token-type ZSPI-TYP-INT.

ZCOM-TKN-OBJNAME

token-type ZSPI-TYP-STRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

GETEVENT Command (ZEMS-CMD-GETEVENT)

Provides the next event message to pass to the current filter.

Command

ZEMS-CMD-GETEVENT

Tokens in Command Buffer (excluding header tokens)

ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.
ZEMS-TKN-EOFSTOP	token-type	ZSPI-TYP-BOOLEAN.

Tokens in Response Buffer

ZEMS-TKN-EVENT	token-type	ZSPI-TYP-BYTESTRING.
ZEMS-TKN-PASSVAL	token-type	ZSPI-TYP-INT.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.
ZSPI-TKN-RETCODE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
. . .		
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

GETEVENT Command RETCODE Values

Describes the primary error numbers for the GETEVENT command.

Error Name	Error Number	Description
ZEMS-WRN-EOF	501	End-of-file requested by ZEMS-TKN-EOFSTOP
ZEMS-ERR-LOG-ACCESS	1031	Distributor cannot access event-log file
ZEMS-ERR-EOF	1032	GETEVENT called past end-of-file
ZEMS-ERR-CONTEXT	1039	Submitted and saved context do not match
ZEMS-ERR-BAD-FILTER	1042	Filter failed on indicated event message

Error Name	Error Number	Description
ZEMS-ERR-NO-EVENT-SOURCE	1044	Distributor has no source collector or log file
ZEMS-ERR-BAD-EVENT	1047	Indicated event message not recognized
ZEMS-ERR-STAT-ONLY	1052	Access to specified distributor not permitted

GETVERSION Command (ZEMS-CMD-GETVERSION)

Provides the version number of the EMS programmatic interface.

Command		
ZEMS-CMD-GETVERSION		
Tokens in Command Buffer		
None		
Tokens in Response Buffer		
ZSPI-TKN-SERVER-BANNER	token-type	ZSPI-TYP-CHAR50
ZSPI-TKN-SERVER-VERSION	token-type	ZSPI-TYP-UINT
ZSPI-TKN-RETCODE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
. . .		
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

The banner displayed by this command is of the form:

```
EMSDIST - T9632D30 - 01MAR95 - EMS Distributor
```

REPLACE Command (ZEMS-CMD-REPLACE)

Replaces a configured object in the distributor with another object. Currently, only the filter object (ZCOM-OBJ-FILTER) is supported. Only one filter at a time can be replaced.

Command		
ZEMS-CMD-REPLACE		
Tokens in Command Buffer		
ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZEMS-TKN-NEW-OBJNAME	token-type	ZSPI-TYP-STRING.
<filter param tokens>		

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

STATUS Command (ZCOM-CMD-STATUS)

Requests both configuration and status information from the EMS distributor. Supported object types are the distributor (ZCOM-OBJ-DIST), filters (ZCOM-OBJ-FILTER), event source (ZCOM-OBJ-SOURCE), event destination (ZCOM-OBJ-TEXTOUT), and forwarding target (ZCOM-OBJ-TARGET).

Command

ZEMS-CMD-REPLACE

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.

Tokens in Response Buffer (ZCOM-OBJ-DIST)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.
ZEMS-MAP-STATUS-SOURCE	token-type	ZEMS-DDL-STATUS-SOURCE.
ZEMS-MAP-STATUS-FILTER	token-type	ZEMS-DDL-STATUS-FILTER.
ZEMS-MAP-STATUS-TEXTOUT	token-type	ZEMS-DDL-STATUS-
TEXTOUT.		
ZEMS-MAP-STATUS-TARGET	token-type	ZEMS-DDL-STATUS-TARGET.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

Tokens in Response Buffer (ZCOM-OBJ-FILTER)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZEMS-MAP-STATUS-FILTER	token-type	ZEMS-DDL-STATUS-FILTER.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

Tokens in Response Buffer (ZCOM-OBJ-SOURCE)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZEMS-MAP-STATUS-SOURCE	token-type	ZEMS-DDL-STATUS-SOURCE.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

Tokens in Response Buffer (ZCOM-OBJ-TEXTOUT)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZEMS-MAP-STATUS-TEXTOUT TEXTOUT.	token-type	ZEMS-DDL-STATUS-
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

Tokens in Response Buffer (ZCOM-OJB-TARGET)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZEMS-MAP-STATUS-TARGET	token-type	ZEMS-DDL-STATUS-TARGET.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.

If the object type is ZCOM-OBJ-SOURCE, ZCOM-OBJ-TEXTOUT, ZCOM-OBJ-FILTER, or ZCOM-OBJ-TARGET, and the object name is either omitted or set to (*), an asterisk, information about all object instances is returned (equivalent to the existing STATUS command, which is described later in this section). The following ZEMS-MAP-STATUS-TEXTOUT structure includes the record length:

```
DEFINITION ZEMS-MAP-STATUS-TEXTOUT.
  02 ZDIST-TEXTOUT-NAME          TYPE ZSPI-DDL-NAME.
  02 ZDIST-TEXTOUT-TYPE          TYPE ZSPI-DDL-UINT.
  02 ZDIST-TEXTOUT-STATE        TYPE ZSPI-DDL-UINT.
  02 ZDIST-TEXTOUT-RECLLEN      TYPE ZSPI-DDL-UINT.
END
```

For ZCOM-OBJ-FILTER, the distributor returns the filter object name, the filter name, size, and type:

```
DEFINITION ZEMS-MAP-STATUS-FILTER.
  02 ZDIST-FILTERNAME-NAME      TYPE ZSPI-DDL-FNAME.
  02 ZDIST-FILTER-NAME          TYPE ZEMS-DDL-CHAR30.
  02 ZDIST-FILTER-SIZE          TYPE ZSPI-DDL-UINT.
  02 ZDIST-FILTER-FLAGS        TYPE ZSPI-DDL-UINT.
END
```

See “Distributor Commands and Responses,” in the *EMS Manual* for detailed definitions of these and the other token map fields returned by the ZCOM-CMD-STATUS command (ZEMS-DDL-STATUS-DIST, ZEMS-MAP-STATUS-SOURCE, ZEMS-MAP-STATUS-TARGET).

STATUS Command (ZEMS-CMD-STATUS)

Provides status information for the distributor and associated collector, log files, or TEXTOUT.

Command

ZEMS-CMD-STATUS

Tokens in Command Buffer

None

Tokens in Response Buffer

```

ZEMS-MAP-STATUS-DIST
  def ZEMS-DDL-STATUS-DIST.
    02 ZDIST-NAME                type ZSPI-DDL-PROCNAME.
    02 ZDIST-TYPE                type ZSPI-DDL-ENUM.
    02 ZDIST-PRIMARY-CPU        type ZSPI-DDL-INT.
    02 ZDIST-BACKUP-CPU        type ZSPI-DDL-INT.
    02 ZDIST-PRIORITY          type ZSPI-DDL-INT.
    02 ZDIST-FILTERFILE        type ZSPI-DDL-FNAME.
    02 ZDIST-FILTERNAME        type ZEMS-DDL-CHAR30.
    02 ZDIST-LOGFILE           type ZSPI-DDL-DISCNAME.
    02 ZDIST-LAST-POSITION     type ZSPI-DDL-
TIMESTAMP.
    02 ZDIST-CURRENT-POSITION   type ZSPI-DDL-
TIMESTAMP.
    02 ZDIST-LAST-ERROR        type ZSPI-DDL-INT.
    02 ZDIST-EVENT-PASS        type ZSPI-DDL-INT2.
    02 ZDIST-EVENT-TOTAL       type ZSPI-DDL-INT2.
    02 ZDIST-FILTER-CHANGE     type ZSPI-DDL-INT.
    02 ZDIST-SUSPENDED         type ZSPI-DDL-BOOLEAN.
    02 ZDIST-EOFDELAY          type ZSPI-DDL-INT.
    02 ZDIST-SEQ-BLOCKING      type ZSPI-DDL-BOOLEAN.
  end

ZEMS-MAP-STATUS-SOURCE
  def ZEMS-DDL-STATUS-SOURCE.
    02 ZDIST-COLL-NAME         type ZSPI-DDL-PROCNAME.
    02 ZDIST-COLL-LOGNAME      type ZSPI-DDL-DISCNAME.
    02 ZDIST-CUR-STATE         type ZSPI-DDL-ENUM.
    02 ZDIST-CUR-IO-PENDING    type ZSPI-DDL-ENUM.
    02 ZDIST-CUR-LOGNAME      type ZSPI-DDL-DISCNAME.
    02 ZDIST-CUR-RECORD-ADDRESS type ZSPI-DDL-INT2.
    02 ZDIST-EVENT-PASS        type ZSPI-DDL-INT2.
    02 ZDIST-EVENT-TOTAL       type ZSPI-DDL-INT2.
    02 ZDIST-LOGSTOPPED       type ZSPI-DDL-BOOLEAN.
  end

```

```

ZEMS-MAP-STATUS-TARGET
  def ZEMS-DDL-STATUS-TARGET .
    02 ZDIST-TARGET-NAME                type ZSPI-DDL-FNAME .
    02 ZDIST-TARGET-EVENTS-PER-BLOCK    type ZSPI-DDL-UINT .
    02 ZDIST-TARGET-STATE                type ZSPI-DDL-UINT .
  end

ZEMS-MAP-STATUS-TEXTOUT
  def ZEMS-DDL-STATUS-TEXTOUT .
    02 ZDIST-TEXTOUT-NAME                type ZSPI-DDL-FNAME .
    02 ZDIST-TEXTOUT-TYPE                type ZSPI-DDL-UINT .
    02 ZDIST-TEXTOUT-STATE                type ZSPI-DDL-UINT .
    02 ZDIST-TEXTOUT-RECLLEN            type ZSPI-DDL-UINT .
  end

ZSPI-TKN-RETCODE                        token-type ZSPI-TYP-ENUM
ZSPI-TKN-ERRLIST                        token-type ZSPI-TYP-LIST
. . .
ZSPI-TKN-ENDLIST                        token-type ZSPI-TYP-SSCTL

```

Distributor Event Messages

Describes event messages that are generated by printing or forwarding distributors and are related to their operation.

Distributor Event Message Summary

Summarizes the distributor event messages and shows which of them are critical.

Number	Name (ZEMS-EVT-)	Description	Critical?
538	BURST-START	Event burst detected	Yes
539	BURST-END	Event burst ended	No
1000	LOG-ACCESS	Cannot access log file	Yes
1001	COLL-ACCESS	Cannot access collector	Yes
1002	DEST-ACCESS	Cannot access destination	No
1003	LOGFILE-EOF	Reached end of log file	No
1005	BAD-FILTER	Encountered filter error	No
1006	COLL-PROTOCOL	Got bad collector response	Yes
1007	BAD-EVENT	Read bad event message	No
1008	DEVTYPE	Detected bad device	No
1009	INTERNAL-ERROR	Encountered internal error	No
1010	CHECKOPEN-FAILED	Notified of bad backup	No
1011	TAKEOVER	Switched control to backup	No
1012	CREATEBACKUP-FAILED	Could not create backup	No

Number	Name (ZEMS-EVT-)	Description	Critical?
1013	BACKUP-CREATED	Created backup process	No
1014	BACKUP-ABENDED	Terminated backup process	No
1015	BACKUP-DELETED	Missing backup process	No
1016	CHECKPOINT-FAILED	Encountered I/O error	No
1017	BAD-LOG	Detected bad log file	No
1018	FILES-LOST	Detected missing log file	Yes
1019	COLL-DISCONNECT	Disconnected from this collector	Yes
1020	STARTUP-FAILED	Cannot create destination process	No
1021	STARTUP-OK	Destination successfully started	No
1022	WRITE-FAILED	Write to destination failed	Yes

Header Tokens

Describes the header tokens that appear in every event message; they are not described with the event syntax.

SPI Header Tokens

Describes the SPI header token that appears in the header of every event message.

ZSPI-TKN-	ZSPI-TYP-	Description
MAX-FIELD-VERSION	INT	The highest field version among the non-null fields of structures added to the buffer with EMSADDTOKENMAPS.
SSID	SSID	Contains subsystem ID for EMS.
USEDLEN	INT	The number of bytes actually used in the buffer. A subsystem that reports events can use this token to determine how many bytes to send to \$0 through the WRITEREAD procedure.

EMS Header Tokens

Describes the EMS header tokens that appear in the header of every distributor event message.

ZSPI-TKN-	ZSPI-TYP-	Description
EVENTNUMBER	ENUM	Shared; identifying number assigned to event by subsystem; unique to subsystem.
GENTIME	TIMESTAMP	Shared; Greenwich mean time when subsystem created event message.
LOGTIME	TIMESTAMP	Shared; Greenwich mean time when collector wrote event-message to log files.
NODENUM	INT2	Shared; Expand system number of the system where event was reported.
CPU	UINT	Shared; CPU number of reporting subsystem process.
PIN	UINT	Shared; PIN of reporting subsystem process.
PROC-DESC	STRING	Shared; process descriptor of the event creator.
USERID	BYTE-PAIR	Shared; user ID of reporting subsystem process.
CONSOLE-PRINT	-	Is no longer used by the compatibility distributor.
EMPHASIS	BOOLEAN	Shared; set to TRUE to emphasize event message as critical.
SUPPRESS-DISPLAY	BOOLEAN	Shared; set to TRUE to suppress display on ViewPoint display.
CONTENT-STANDARD	ENUM	Shared; indicates the type of standard event.
CONTENT-USER	ENUM	Shared; identifies the type of a subsystem-defined event.

Common Data-Portion Tokens

Describes the nonheader tokens that appear in many distributor event messages.

SPI Data Tokens

Describes the SPI token that appears in distributor event messages.

ZSPI-TKN-	ZSPI-TYP-	Description
PROC-ERR	ENUM	Procedure associated with event; ZEMS-VAL- can be:
		FILTER-EVAL 20
		ZFILEAWAITIO 21
		ZFILOPEN 22
		ZFILPOSITION 23
		ZFILREAD 24
		ZFILWRITE 25
		ZFILWRITEREAD 26
		FILTER-READ 27
		FILTER-VERIFY 28
		EMSADDBUFFER 29
		EMSADDSUBJECT 30
		EMSADDTOKENS 31
		EMSGET 32
		EMSINIT 33
		EMSSEND 34
		CHECKOPEN 35
		NEWPROCESS 36
		CHECKPOINT 37
		CHECKMONITOR 38

EMS Data Tokens

Describes the EMS tokens that appear in distributor event messages.

ZSPI-TKN-	ZSPI-TYP-	Description
BLOCKLENGTH	INT	Nonshared; block length of log file.
COLNAME	FNAME	Nonshared; collector for distributor reporting event; otherwise, blanks.
COLNAME-ENUM	ENUM	Nonshared; presence or absence of COLNAME value:
		COLNAME-PRESENT 0
		COLNAME-NOTPRESENT 1
DEVICE-TYPE	INT	Nonshared; device-type of device for distributor.
DEVTYPE-ENUM	ENUM	Nonshared; problem with log file (0-3) or collector (4-5).
		FILECODE-BAD 0
		BLOCKLENGTH-BAD 1
		DEVICE-TYPE-BAD 2
		LOGNAME-BAD 3
		VERSION-INCOMPATIBLE 4
		DUP-SOURCE-TARGET 5

ZSPI-TKN-	ZSPI-TYP-	Description
DIST-NAME	FNAME	Nonshared; name of distributor reporting event.
FAILFILENAME	FNAME	Nonshared; file name of bad log file.
FAIL-REASON	ENUM	Nonshared; reason distributor failed to access log file. EVENT-GENERATED 0 NO-LINK 1 BAD-LINK 2
FILECODE	INT	Nonshared; file code of log file.
FILTER-ERROR	ENUM	Nonshared; private to Tandem.
FILTERNAME	CHAR30	Nonshared; name of the filter at compilation.
LASTLOGFILE	FNAME	Nonshared; collector log that distributor tried to access.
LOGNAME	FNAME	Nonshared; log file in use when event occurred.
NEWLOGFILE	FNAME	Nonshared; next collector log file after inaccessible file.
NEWPROCESS-CPU	INT	Nonshared; CPU number in NEWPROCESS command.
NEWPROCESS-ERROR	ENUM	Nonshared; NEWPROCESS error: NO-ERROR 0 UNDEFINED-EXTERNALS 1 NO-PCB-AVAILABLE 2 PROGRAMFILE-ERROR 3 NO-MAP 4 BAD-SWAPFILE 5 BAD-FILE-FORMAT 6 UNLICENSED 7 BAD-PROCESS-NAME 8 LIBRARY-CONFLICT 9 MONITOR-COMM 10 LIBRARY-FILE 11 PROGRAM-FILE 12 NO-EXT-SEGMENT 13 EXT-SEGMENT-SWAP 14 BAD-HOMETERM 15
NEWPROCESS-PRIORITY	INT	Nonshared; priority in NEWPROCESS.
PROGRAMFILE	FNAME	Nonshared; file name of program file.
RECORD-ADDRESS	INT2	Nonshared; entry-sequenced file address computed as: $(blknum * blklen) + recnum$

ZSPI-TKN-	ZSPI-TYP-	Description
RECOVERY-ENUM	ENUM	Nonshared; recovery after failing to access log file. RECOVERY-OK 0 NO-RECOVERY 1
TAKEOVER-REASON	ENUM	Nonshared; reason for backup takeover: PRIMARY-STOPPED 0 PRIMARY-ABEND 1 PRIMARY-CPU-DOWN 2 CHECKSWITCH 3
ZFILERR	UINT	Nonshared; Error code associated with event.

EMS Extensible Structured Token

Describes the extensible structured token that appears in many distributor event messages.

ZSPI-TKN-	ZSPI-TYP-	Description
STATUS-DIST	STATUS-DIST	Status information describing current state of distributor

538: ZEMS-EVT-BURST-START

CRITICAL: An event burst has been detected.

Unconditional Token (excluding header tokens)	
ZSPI-TKN-COLLECTOR	token-type ZSPI-TYP-ENUM.
ZEMS-TKN-BURST-EVT-NUMBER	token-type ZSPI-TYP-ENUM.
ZEMS-TKN-SSID	token-type ZSPI-TYP-SSID.
ZEMS-TKN-SUBJ-CODE	token-type ZSPI-TYP-TOKENCODE.
ZEMS-TKN-SUBJ-VALUE	token-type ZSPI-TYP-STRING.
ZEMS-TKN-SUBJ-SSID	token-type ZSPI-TYP-SSID.
ZEMS-TKN-BURST-TIME-START	token-type ZSPI-TYP-TIMESTAMP.
ZEMS-MAP-BDS-INFO	token-type ZEMS-DDL-BDS-INFO.
Event-Message Text	
EMS: EVENT BURST DETECTED FOR EVENT NO. <i>eventno</i> OF SUBSYSTEM <i>ssid</i> , BY DISTRIBUTOR <i>proc-desc</i>	

Subject

ZEMS-TKN-BURST-EVT-
NUM The event number of the bursting event

Header Token Value

ZSPI-TKN-
EMPHASIS TRUE

Text Values

eventno ZEMS-TKN-BURST-EVT-NUM
ssid ZEMS-TKN-BURST-SSID
proc-name ZSPI-TKN-BURST-PROC-DESC

539: ZEMS-EVT-BURST-END

An event burst has ended.

Unconditional Token (excluding header tokens)

ZSPI-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-BURST-EVT-NUMBER	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SSID	token-type	ZSPI-TYP-SSID.
ZEMS-TKN-SUBJ-CODE	token-type	ZSPI-TYP-TOKENCODE.
ZEMS-TKN-SUBJ-VALUE	token-type	ZSPI-TYP-STRING.
ZEMS-TKN-SUBJ-SSID	token-type	ZSPI-TYP-SSID.
ZEMS-TKN-BURST-TIME-START	token-type	ZSPI-TYP-TIMESTAMP.
ZEMS-TKN-BURST-END	token-type	ZSPI-TYP-TIMESTAMP
ZEMS-TKN-EVTS-DELETED	token-type	ZSPI-TYP-INT2.
ZEMS-TOKEN-BURST-END-REASON	token-type	ZSPI-TYP-ENUM.

Event-Message Text

If ZEMS-TKN-BURST-END-REASON = ZEMS-VAL-BDS-ENABLED, then the event message text is:

EMS: BURST SUPPRESSION TERMINATED: count OCCURRENCES OF EVENT NO. *eventno*. OF SUBSYSTEM *ssid* WERE NOT PROCESSED BY DISTRIBUTOR *proc-desc*

If ZEMS-TKN-BURST-END-REASON = ZEMS-VAL-NO-EVENTS, then the event message text is:

EMS: BURST END DETECTED: count OCCURRENCES OF EVENT NO. *eventno*. OF SUBSYSTEM *ssid* WERE NOT PROCESSED BY DISTRIBUTOR *proc-desc*

Subject

ZEMS-TKN-BURST-EVT-
NUM The event number of the bursting event

Header Token Value

ZSPI-TKN-
EMPHASIS FALSE

Text Values

eventno ZEMS-TKN-BURST-EVT-NUM
ssid ZEMS-TKN-BURST-SSID
proc-name ZSPI-TKN-BURST-PROC-DESC
count ZSPI-TKN-BURST-EVTS-DELETED

1000: ZEMS-EVT-LOG-ACCESS

CRITICAL: Distributor cannot access event log file.

Unconditional Token (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.

Event-Message Text

```
EMS: EVENT LOG logname CANNOT BE ACCESSED -
GUARDIAN ERROR zfilerr,
  ( COLLECTOR colname      }
  ( (ACCESSED AS LOGFILE) }
USING PROCEDURE proc-name
```

Subject

ZEMS-TKN-
LOGNAME Name of log file that cannot be accessed

Header Token Value

ZSPI-TKN-
EMPHASIS TRUE

Proc-Err Values

ZSPI-TKN-PROC-ERR	ZEMS-VAL-ZFILOPEN	22
	ZEMS-VAL-ZFILPOSITION	23
	ZEMS-VAL-ZFILREAD	24

Text Values

<i>logname</i>	ZEMS-TKN-LOGNAME
<i>zfilerr</i>	ZEMS-TKN-ZFILERR
<i>colname</i>	ZEMS-TKN-COLNAME
<i>proc-name</i>	ZSPI-TKN-PROC-ERR

1001: ZEMS-EVT-COLL-ACCESS

CRITICAL: Distributor cannot access collector.

Unconditional Tokens (excluding header tokens)

SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-DIST	Def:	ZEMS-DDL-STATUS-DIST.
ZEMS-TKN-ZFILLER	token-type	ZSPI-TYP-UINT.

Event-Message Text

EMS: COLLECTOR *colname* CANNOT BE ACCESSED
 GUARDIAN ERROR *zfilerr*, USING PROCEDURE *proc-name*

Subject

ZEMS-TKN-COLNAME Collector that distributor cannot access

Header Token Value

ZEMS-TKN-EMPHASIS TRUE

Proc-Err Values

ZSPI-TKN-PROC-ERR	ZEMS-VAL-ZFILOPEN	22
	ZEMS-VAL-ZFILWRITEREAD	26

Text Values

<i>colname</i>	ZEMS-TKN-COLNAME
<i>zfilerr</i>	ZEMS-TKN-ZFILERR
<i>proc-name</i>	ZSPI-TKN-PROC-ERR

1002: ZEMS-EVT-DEST-ACCESS

Printing distributor cannot access text or forwarding distributor cannot access remote collector.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.

Event-Message Text

```
EMS: failfilename CANNOT BE ACCESSED FOR
    { PRINTING }
    { FORWARDING }
- GUARDIAN ERROR zfilerr
```

Subject

ZEMS-TKN-
FAILFILENAME Destination device or process

Header Tokens Value

ZEMS-TKN-
EMPHASIS FALSE

Proc-Err Values

ZSPI-TKN-PROC-	ZEMS-VAL-ZFILOPEN	22
ERR:	ZEMS-VAL-ZFILWRITE	25

Text Values

<i>failfilename</i>	ZEMS-TKN-FAILFILENAME
<i>zfilerr</i>	ZEMS-TKN-ZFILERR

1003: ZEMS-EVT-LOGFILE-EOF

Distributor reached end-of-file on log file it accesses directly.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.

Event-Message Text

```
EMS: EOF ENCOUNTERED FOR EVENT LOG logname
```

Subject

```
ZEMS-TKN-          Log file in use when end-of-file was reached
LOGNAME
```

Header Token Value

```
ZEMS-TKN-          FALSE
EMPHASIS
```

Text Value

```
logname  ZEMS-TKN-LOGNAME
```

1005: ZEMS-EVT-BAD-FILTER

Distributor detects filter error while filtering event messages.

Unconditional Tokens (excluding header tokens)

```
ZEMS-TKN-SUBJECT-MARK      token-type ZSPI-TYP-SSCTL.
ZEMS-TKN-FILTERNAME        token-type ZEMS-TYP-CHAR30.
ZSPI-TKN-PROC-ERR          token-type ZSPI-TYP-ENUM
ZEMS-MAP-STATUS-DIST       token-type ZEMS-DDL-STATUS-DIST.
ZEMS-TKN-RECORD-ADDRESS    token-type ZSPI-TYP-INT2.
ZEMS-TKN-FILTER-ERROR      token-type ZSPI-TYP-ENUM.
ZEMS-TKN-LOGNAME           token-type ZSPI-TYP-FNAME.
ZEMS-TKN-COLNAME-ENUM      token-type ZSPI-TYP-ENUM.
ZEMS-TKN-COLNAME           token-type ZSPI-TYP-FNAME.
```

Event-Message Text

```
EMS: FILTER ERROR filter-error ON FILTER filtername, EVENT
FROM logname,
    { COLLECTOR colname }
    { ACCESSED AS LOGFILE )}
```

Subject

```
ZEMS-TKN-          Filter with error
FILTERNAME
```

Header Token Value

```
ZEMS-TKN-          FALSE
EMPHASIS
```

Proc-Err Values

ZSPI-TKN-PROC- ERR:	ZEMS-VAL-ZFILOPEN	22
	ZEMS-VAL-ZPOSITION	23
	ZEMS-VAL-ZFILREAD	24

Text Values

<i>filter-error</i>	ZEMS-TKN-FILTER-ERROR
<i>filtername</i>	ZEMS-TKN-FILTERNAME
<i>logname</i>	ZEMS-TKN-LOGNAME
<i>colname</i>	ZEMS-TKN-COLNAME

1006: ZEMS-EVT-COLL-PROTOCOL

CRITICAL: Distributor detects invalid response to command message issued to collector.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.

Event-Message Text

EMS: INVALID RESPONSE FROM COLLECTOR *colname* USING PROCEDURE *proc-name*, COLLECTOR WAS CLOSED

Subject

ZEMS-TKN-COLNAME Collector that responded to distributor with invalid SPI response.

Header Token Value

ZEMS-TKN-EMPHASIS TRUE

Proc-Err Value

ZSPI-TKN-PROC- ERR:	ZSPI-VAL-SSGET	2
------------------------	----------------	---

Text Values

<i>colname</i>	ZEMS-TKN-COLNAME
<i>proc-name</i>	ZSPI-TKN-PROC-ERR

1007: ZEMS-EVT-BAD-EVENT

Distributor detects invalid event message from a log file.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.
ZEMS-TKN-RECORD-ADDRESS	token-type	ZSPI-TYP-INT2.
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.

Event-Message Text

```
EMS: BAD EVENT AT RECORD ADDRESS record address IN LOG FILE
logname { , COLLECTOR colname },
          {(ACCESSED AS LOGFILE)}, USING PROCEDURE proc-name
```


Subject

ZEMS-TKN-FAILFILENAME Name of log file or collector that distributor cannot access.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Text Values

failfilename ZEMS-TKN-FAILFILENAME
dev-type- ZEMS-TKN-DEVICE-TYPE
text ZSPI-TKN-PROC-ERR
proc-name

1009: ZEMS-EVT-INTERNAL-ERROR

Distributor cannot find event definition for internal error message.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.

Event-Message Text

EMS: INTERNAL ERROR IN DISTRIBUTOR, SSID *ssid*, ERROR *error*
PROGRAM FILENAME *failfilename*

Subjects

ZEMS-TKN-PROGRAMFILE File name of distributor object program.
 ZSPI-TKN-ERROR Standard SPI error token.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Text Values

ssid ZSPI-TKN-SSID
error ZSPI-TKN-ERROR
failfilename ZEMS-TKN-PROGRAMFILE

1010: ZEMS-EVT-CHECKOPEN-FAILED

Distributor backup unable to “checkopen” a file.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-DIST-NAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.

Event-Message Text

EMS: CHECKOPEN FAILED, ERROR *zfilerr* ON *failfilename*

Subjects

ZEMS-TKN-FAILFILENAME Name of distributor program.
 ZEMS-TKN-DIST-NAME Name of primary process that starts backup process.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Proc-Err Value

ZSPI-TKN-PROC-ERR: ZEMS-VAL-CHECKOPEN 35

Text Values

zfilerr ZEMS-TKN-ZFILERR
failfilename ZEMS-TKN-FAILFILENAME

1011: ZEMS-EVT-TAKEOVER

Takeover by distributor backup process.

Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-DIST-NAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-TAKEOVER-REASON	token-type	ZSPI-TYP-ENUM.

Event-Message Text

EMS: TAKEOVER BY BACKUP (*takeover-text*)

Unconditional Subject

ZEMS-TKN-DIST-NAME Name of new distributor process.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Proc-Err Value

ZSPI-TKN-PROC-ERR: ZEMS-VAL-CHECKPOINT 37

Text Value

takeover-text ZEMS-TKN-TAKEOVER-REASON

1012: ZEMS-EVT-CREATEBACKUP-FAILED

Distributor cannot create backup due to NEWPROCESS error.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-DIST-NAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TYP-INT.
ZEMS-TKN-PROCCREATE-ERROR	token-type	ZSPI-TYP-ENUM.

Conditional Tokens

ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
------------------	------------	----------------

Event-Message Text

EMS: FAILED TO CREATE BACKUP PROCESS IN CPU *cpu*, PRIORITY *pri*, PROGRAM FILE *name* - ERROR *newprocerr* : *ertxt*

Subjects

ZEMS-TKN-PROGRAMFILE Name of distributor program file.
 ZEMS-TKN-DIST-NAME Name of distributor process.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Proc-Err Value

ZSPI-TKN-PROC-ERR: ZEMS-VAL-NEWPROCESS 36

Text Values

cpu ZEMS-TKN-NEWPROCESS-CPU
pri ZEMS-TKN-NEWPROCESS-PRIORITY
name ZEMS-TKN-PROGRAMFILE
newprocerr : ertxt ZEMS-TKN-PROCCREATE-ERROR

1013: ZEMS-EVT-BACKUP-CREATED

New distributor process in backup CPU successfully created.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-DIST-NAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TYP-INT.
ZEMS-TKN-PROCCREATE-ERROR	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

EMS: CREATED BACKUP IN CPU *newprocess-cpu*, PRIORITY
newprocess-pri, PROGRAM FILE *programfile*

Subjects

ZEMS-TKN-DIST-NAME Name of new distributor process.
 ZEMS-TKN-PROGRAMFILE Name of distributor program file.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Proc-Err Value

ZSPI-TKN-PROC-ERR: ZEMS-VAL-NEWPROCESS 36

Text Values

newprocess-cpu ZEMS-TKN-NEWPROCESS-CPU
newprocess-pri ZEMS-TKN-NEWPROCESS-PRIORITY
programfile ZEMS-TKN-PROGRAMFILE

1014: ZEMS-EVT-BACKUP-ABENDED

Backup process stopped abruptly.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK token-type ZSPI-TYP-SSCTL.
 ZEMS-TKN-DIST-NAME token-type ZSPI-TYP-FNAME.

Event-Message Text

EMS: BACKUP PROCESS ABENDED

Subject

ZEMS-TKN-DIST-NAME Name of distributor process.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

1015: ZEMS-EVT-BACKUP-DELETED

Distributor process in backup CPU failed.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK token-type ZSPI-TYP-SSCTL.
 ZEMS-TKN-DIST-NAME token-type ZSPI-TYP-FNAME.

Event-Message Text

EMS: BACKUP PROCESS DELETED (CPU DOWN)

Subject

ZEMS-TKN-DIST-NAME Name of distributor process.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

1016: ZEMS-EVT-CHECKPOINT-FAILED

I/O error during checkpoint operation.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK token-type ZSPI-TYP-SSCTL.
 ZEMS-TKN-DIST-NAME token-type ZSPI-TYP-FNAME.
 ZEMS-TKN-ZFILERR token-type ZSPI-TYP-UINT.
 ZSPI-TKN-PROC-ERR token-type ZSPI-TYP-ENUM.

Event-Message Text

EMS: CHECKPOINT FAILED, ERROR *zfilerr*

Subject

ZEMS-TKN-DIST-NAME Name of new distributor process.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Proc-Err Value

ZSPI-TKN-PROC-ERR: ZEMS-VAL-CHECKPOINT 37

Text Value

zfiler ZEMS-TKN-ZFILERR
r

1017: ZEMS-EVT-BAD-LOG

Distributor detected five invalid event messages in log file.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.

Event-Message Text

```
EMS: LOG CLOSED; TOO MANY BAD EVENTS ENCOUNTERED -
LOG FILE file-name { , COLLECTOR colname },
                  {(ACCESSED AS LOGFILE)},
LAST RECORD ADDRESS record-address
```

Subject

ZEMS-TKN-LOGNAME Name of log file declared invalid by distributor.

Header Token Value

ZEMS-TKN-EMPHASIS FALSE

Text Values

file-name ZEMS-TKN-LOGNAME
colname ZEMS-TKN-COLNAME
record-address ZEMS-MAP-STATUS-DIST

1018: ZEMS-EVT-FILES-LOST

CRITICAL: Distributor at end of collector log file; cannot access next file.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-LASTLOGFILE	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-FAIL-REASON	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-RECOVERY-ENUM	token-type	ZSPI-TYP-ENUM.

Conditional Tokens

ZEMS-TKN-NEWLOGFILE	token-type	ZSPI-TYP-FNAME.
---------------------	------------	-----------------

Event-Message Text

EMS: DISTRIBUTOR COULD NOT COMPLETE BACK CHAIN TO
 FILE *lastlogfile* BECAUSE OF *failreason recoveryenum*
 COLLECTOR: *colname*

Subject

ZEMS-TKN-COLNAME Collector whose log files cannot be accessed by distributor.

Header Token Value

ZEMS-TKN-EMPHASIS TRUE

Text Values

lastlogfile ZEMS-TKN-LASTLOGFILE
failreaso ZEMS-TKN-FAIL-REASON

Possible values:

ZEMS-VAL-EVENT-GENERATED 0
 ZEMS-VAL-NO-LINK 1
 ZEMS-VAL-BAD-LINK 2

recoveryenu ZEMS-TKN-RECOVERY-ENUM
m

Possible values:

ZEMS-VAL-RECOVERY-OK 0
 ZEMS-VAL-NO-RECOVERY 1

colname ZEMS-TKN-COLNAME

1019: ZEMS-EVT-COL-DISCONNECT

CRITICAL: Distributor cannot retrieve event messages from any log file associated with collector.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.

Event-Message Text

EMS: COLLECTOR *colname* HAS BEEN DISCONNECTED DUE TO INACCESSIBLE EVENT LOG(S).

Subject

ZEMS-TKN-COLNAME Collector that distributor is disconnecting.

Header Token Value

ZEMS-TKN-EMPHASIS TRUE

Text Value

colname ZEMS-TKN-COLNAME

1020: ZEMS-EVT-STARTUP-FAILED

Distributor cannot create a destination process due to a NEWPROCESS error after one retry has failed.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TYP-INT.
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TYP-INT.
ZEMS-TKN-NEWPROCESS-ERROR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-STARTUP-LOGTIME	token-type	ZSPI-TYP-TIMESTAMP
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.

Conditional Token

ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
------------------	------------	----------------

Event-Message Text

EMS: FAILED TO CREATE DESTINATION PROCESS IN CPU *cpu*,
 PRIORITY *pri*, PROGRAM FILE *name* - ERROR *error* :
 {NO ERROR | UNDEFINED EXTERNALS | NO PROCESS CONTROL BLOCK
 AVAILABLE | ERROR ON PROGRAMFILE: *zfilerr* | UNABLE TO
 ALLOCATE MAP | ERROR ON SWAP FILE : *zfilerr* | ILLEGAL FILE
 FORMAT | UNLICENSED PRIVILEGED PROGRAM | PROCESS NAME ERROR:
zfilerr | LIBRARY CONFLICT | UNABLE TO COMMUNICATE WITH
 SYSTEM MONITOR | ERROR ON LIBRARY FILE: *zfilerr* | LIBRARY
 AND PROGRAM FILE ARE THE SAME | EXTENDED SEGMENT ERROR:
zfilerr | SWAP FILE ERROR: *zfilerr* | ILLEGAL HOME
 TERMINAL, ERROR *zfilerr* }. TRIGGER LOGTIME *time*

Subjects

ZEMS-TKN-PROGRAMFILE	Name of distributor program file.
ZEMS-TKN-FAILFILENAME	Name of destination process.

Header Token Value

ZEMS-TKN-EMPHASIS	FALSE
-------------------	-------

Proc-Err Value

ZSPI-TKN-PROC-ERR:	An ordinal identifying the failing procedure, the value of ZEMS-VAL-NEWPROCESS.
--------------------	---

Text Values

cpu ZEMS-TKN-NEWPROCESS-CPU
pri ZEMS-TKN-NEWPROCESS-PRIORITY
name ZEMS-TKN-PROGRAMFILE
error ZEMS-TKN-NEWPROCESS-ERROR
zfilerr ZEMS-TKN-ZFILERR
time ZEMS-TKN-STARTUP-LOGTIME

1021: ZEMS-EVT-STARTUP-FAILED

Destination successfully started after previous failure and retries.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TYP-INT.
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TYP-INT.
ZEMS-TKN-NEWPROCESS-ERROR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-STARTUP-LOGTIME	token-type	ZSPI-TYP-TIMESTAMP
ZEMS-MAP-STATUS-DIST	token-type	ZEMS-DDL-STATUS-DIST.

Event-Message Text

EMS: DESTINATION PROCESS SUCCESSFULLY CREATED IN CPU *cpu*,
 PRIORITY *pri*, PROGRAM FILE *name*, TRIGGER LOGTIME *time*.

Subjects

ZEMS-TKN-PROGRAMFILE Name of distributor program file.
 ZEMS-TKN-
 FAILFILENAME Name of destination process.

Header Token Value

ZEMS-TKN- FALSE
 EMPHASIS

Proc-Err Value

ZSPI-TKN-PROC-
 ERR: An ordinal identifying the affected procedure, the value of
 ZEMS-VAL-NEWPROCESS.

Text Values

cpu ZEMS-TKN-NEWPROCESS-CPU
pri ZEMS-TKN-NEWPROCESS-PRIORITY
name ZEMS-TKN-PROGRAMFILE
time ZEMS-TKN-STARTUP-LOGTIME

1022: ZEMS-EVT-STARTUP-FAILED

CRITICAL: A write operation to a destination has repeatedly timed out over an interval of 24 hours. The event contains a count of the number of timeouts that have occurred during the 24 hour period. If the error is recovered, the message will not be generated.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-WRITE-FAIL-COUNT	token-type	ZSPI-TYP-UINT.
ZEMS-MAP-STATUS-DIST DIST.	token-type	ZEMS-DDL-STATUS-

Event-Message Text

EMS: WRITE OPERATION TO DESTINATION *name* HAS TIMED OUT
n TIMES DURING A 24-HOUR PERIOD.

Subject

ZEMS-TKN-
FAILFILENAME Name of destination process.

Header Token Value

ZEMS-TKN-
EMPHASIS TRUE

Proc-Err Value

ZSPI-TKN-PROC-
ERR: An ordinal identifying the failing procedure, the value of
ZEMS-VAL-ZFILWRITEREAD.

Text Values

name ZEMS-TKN-PROGRAMFILE
n Number of times the write operation timed out

Distributor Errors

Describes the error codes and warning codes returned by the distributor.

Common SPI Tokens

Describes the SPI tokens that occur in distributor error lists.

ZSPI-TKN-	ZSPI-TYP-	Description																																						
COMMAND	ENUM	Specifies the distributor command.																																						
CONTEXT	BYTESTRING	The token saved by the distributor to record its position in the log file.																																						
ENDLIST	SSCTL	The SPI token that ends an error list.																																						
ERRLIST	LIST	Begins an error list.																																						
ERROR	ERROR	Standard SPI error token for ZSPI-TKN-ERROR																																						
PARAM-ERR		<p>Standard SPI error token. DDL for ZSPI-TKN-PARAM-ERR:</p> <pre>def ZSPI-DDL-PARAM-ERR . 02 Z-TOKENCODE type ZSPI-DDL- TOKENCODE . 02 Z-INDEX type ZSPI-DDL-UINT . 02 Z-OFFSET type ZSPI-DDL-UINT . end</pre> <p>The token code and index are given (but not the value) of a parameter token that is in error. If the value was in error, error list also included token and erroneous value.</p>																																						
PROC-ERR	ENUM	<p>Specifies a procedure associated with the error. Possible values:</p> <table border="0"> <tr><td>ZEMS-VAL-FILTER-EVAL</td><td>20</td></tr> <tr><td>ZEMS-VAL-ZFILAWAITIO</td><td>21</td></tr> <tr><td>ZEMS-VAL-ZFILOPEN</td><td>22</td></tr> <tr><td>ZEMS-VAL-ZFILPOSITION</td><td>23</td></tr> <tr><td>ZEMS-VAL-ZFILREAD</td><td>24</td></tr> <tr><td>ZEMS-VAL-ZFILWRITE</td><td>25</td></tr> <tr><td>ZEMS-VAL-ZFILWRITEREAD</td><td>26</td></tr> <tr><td>ZEMS-VAL-FILTER-READ</td><td>27</td></tr> <tr><td>ZEMS-VAL-FILTER-VERIFY</td><td>28</td></tr> <tr><td>ZEMS-VAL-EMSADDBUFFER</td><td>29</td></tr> <tr><td>ZEMS-VAL-EMSADDSUBJECT</td><td>30</td></tr> <tr><td>ZEMS-VAL-EMSADDTOKENS</td><td>31</td></tr> <tr><td>ZEMS-VAL-EMSGET</td><td>32</td></tr> <tr><td>ZEMS-VAL-EMSINIT</td><td>33</td></tr> <tr><td>ZEMS-VAL-EMSSEND</td><td>34</td></tr> <tr><td>ZEMS-VAL-CHECKOPEN</td><td>35</td></tr> <tr><td>ZEMS-VAL-NEWPROCESS</td><td>36</td></tr> <tr><td>ZEMS-VAL-CHECKPOINT</td><td>37</td></tr> <tr><td>ZEMS-VAL-CHECKMONITOR</td><td>38</td></tr> </table>	ZEMS-VAL-FILTER-EVAL	20	ZEMS-VAL-ZFILAWAITIO	21	ZEMS-VAL-ZFILOPEN	22	ZEMS-VAL-ZFILPOSITION	23	ZEMS-VAL-ZFILREAD	24	ZEMS-VAL-ZFILWRITE	25	ZEMS-VAL-ZFILWRITEREAD	26	ZEMS-VAL-FILTER-READ	27	ZEMS-VAL-FILTER-VERIFY	28	ZEMS-VAL-EMSADDBUFFER	29	ZEMS-VAL-EMSADDSUBJECT	30	ZEMS-VAL-EMSADDTOKENS	31	ZEMS-VAL-EMSGET	32	ZEMS-VAL-EMSINIT	33	ZEMS-VAL-EMSSEND	34	ZEMS-VAL-CHECKOPEN	35	ZEMS-VAL-NEWPROCESS	36	ZEMS-VAL-CHECKPOINT	37	ZEMS-VAL-CHECKMONITOR	38
ZEMS-VAL-FILTER-EVAL	20																																							
ZEMS-VAL-ZFILAWAITIO	21																																							
ZEMS-VAL-ZFILOPEN	22																																							
ZEMS-VAL-ZFILPOSITION	23																																							
ZEMS-VAL-ZFILREAD	24																																							
ZEMS-VAL-ZFILWRITE	25																																							
ZEMS-VAL-ZFILWRITEREAD	26																																							
ZEMS-VAL-FILTER-READ	27																																							
ZEMS-VAL-FILTER-VERIFY	28																																							
ZEMS-VAL-EMSADDBUFFER	29																																							
ZEMS-VAL-EMSADDSUBJECT	30																																							
ZEMS-VAL-EMSADDTOKENS	31																																							
ZEMS-VAL-EMSGET	32																																							
ZEMS-VAL-EMSINIT	33																																							
ZEMS-VAL-EMSSEND	34																																							
ZEMS-VAL-CHECKOPEN	35																																							
ZEMS-VAL-NEWPROCESS	36																																							
ZEMS-VAL-CHECKPOINT	37																																							
ZEMS-VAL-CHECKMONITOR	38																																							

EMS Token Codes

Describes the EMS tokens that occur in distributor error lists.

ZSPI-TKN-	ZSPI-TYP-	Description
BLOCKLENGTH	INT	Nonshared. The block length of a log file.
COLNAME	FNAME	Nonshared. If specified, the name of collector associated with distributor; otherwise set to blanks.
COLNAME-ENUM	ENUM	Nonshared. Indicates presence or absence of a collector name in ZEMS-TKN-COLNAME. ZEMS-VAL-COLNAME-PRESENT 0 ZEMS-VAL-COLNAME-NOTPRESENT 1
SRC-COLL	FNAME	Nonshared. Names a collector whose log files are source of event messages for distributor.
DEVICE-TYPE	INT	Nonshared. Device type of distributor reporting the error.
DEVTYPE-ENUM	ENUM	Uses one of the following values to indicate a device-type problem: ZEMS-VAL-FILECODE-BAD 0 ZEMS-VAL-BLOCKLENGTH-BAD 1 ZEMS-VAL-DEVICE-TYPE-BAD 2 ZEMS-VAL-LOGNAME-BAD 3 ZEMS-VAL-VERSION-INCOMPATIBLE 4 ZEMS-VAL-DUP-SOURCE-TARGET 5
DISCONNECT-SRC-COLL	FNAME	Nonshared. Name of a collector to be disconnected as a source of event messages.
FAILFILENAME	FNAME	Nonshared. File name of a bad log file.
FILECODE	INT	Nonshared. File code of a log file.
FILTER-ERROR		Nonshared. Private to Tandem.
LOGNAME	FNAME	Nonshared. File name of log file in use when error occurred.
RECORD-ADDRESS	INT2	Nonshared. Entry-sequenced file address. Suppose that <i>blknum</i> is the block number (numbering from zero), <i>blklen</i> is the block length (in bytes), and <i>recnum</i> is the record number (within the block). Then <i>record-address</i> is computed as follows: $(\textit{blknum} * \textit{blklen}) + \textit{recnum}$
ZFILERR	UINT	Error code associated with some value of ZSPI-TKN-ERROR for the distributor.

Distributor Warning Codes

The following distributor codes represent warnings.

501: ZEMS-WRN-EOF

ZEMS-TKN-EOFSTOP token was sent, and the distributor reached an end-of-file indication before another event message passed the filter.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

502: ZEMS-WRN-TOO-EARLY

The requested log file position appears to far behind—no appropriate event log is available.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

503: ZEMS-WRN-TOO-LATE

The requested log file position is too far ahead—beyond the end-of-file position on the last log file.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

504: ZEMS-WRN-STARTUP-OK

A routing distributor has attempted to launch an application and has failed. After some retries, the process creation succeeds.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-STARTUP-LOGTIME TIMESTAMP	token-type	ZSPI-TYP-
ZEMS-TKN-PROCCREATE-ERROR	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

Distributor Error Codes

The following distributor errors vary in seriousness, depending on the context in which they occur.

1001: ZEMS-ERR-VERSION

The distributor does not support the version supplied in the command message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1002: ZEMS-ERR-INV-CMD

The distributor has received an invalid command message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-COMMAND	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1003: ZEMS-ERR-INV-SSID

The distributor has received a command message with an invalid subsystem ID (from a command-message buffer not owned by EMS).

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-SSID-ERR	token-type	ZSPI-TYP-SSID
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1004: ZEMS-ERR-INV-TKN

The distributor has received a command message with an unnecessary or unrecognized token.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1005: ZEMS-ERR-INV-VALUE

The distributor has received a command message with a bad token; the token value is invalid.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1006: ZEMS-ERR-DUP-TKN

The distributor has received a command message with duplicate tokens.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1007: ZEMS-ERR-MODE-CONFLICT

The distributor has received a command message that is inappropriate for this type of distributor.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1008: ZEMS-ERR-INV-OBJECT

The distributor has received a command message with an invalid object type. The object type must be null for distributor commands.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1014: ZEMS-ERR-INV-OP

The distributor has received a command message that is not allowed in this context.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1015: ZEMS-ERR-REQ-TKN

The distributor has received a command message that lacks a required token.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1016: ZEMS-ERR-INV-HEADERTYPE

The distributor has received a command message with an invalid header type.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-COMMAND	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1018: ZEMS-ERR-COLL-ACCESS

The distributor cannot access a collector process.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZEMS-VAL-ZFILOPEN	22
ZEMS-VAL-ZFILWRITEREAD	26

1019: ZEMS-ERR-FLT-FORM

The distributor has received a command message in which the filter is not in the expected format.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-FILTER-ERROR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZEMS-TKN-FILTER-ERROR can have the following values:

Value	Description
-1	Filter too big to fit into supplied buffer.
-2	Filter format invalid. If the cause of the problem cannot be determined, please contact your Tandem representative.
-3	Version incompatibility. Filter has been compiled with a version newer than the distributor.

1020: ZEMS-ERR-FLT-LOAD

The distributor could not load the filter.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZEMS-VAL-ZFILOPEN	22
ZEMS-VAL-ZFILREAD	24

1022: ZEMS-ERR-REQ-PARAM

One or more filter parameter tokens are missing.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1024: ZEMS-ERR-HIST-MODE

The distributor cannot perform this operation in log file mode.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1025: ZEMS-ERR-MAX-COLLECTOR

The distributor cannot connect another collector because the number of collectors is already at its maximum (five).

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1026: ZEMS-ERR-COLLECTOR-EXISTS

The collector is already associated with the distributor and cannot be connected.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZEMS-TKN-CONNECT-SRC-COLL	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1027: ZEMS-ERR-COLL-NOT-FOUND

The collector is not associated with the distributor and cannot be disconnected.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZEMS-DISCONNECT-SRC-COLL	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1031: ZEMS-ERR-LOG-ACCESS

The distributor cannot access an event log.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZEMS-VAL-ZFILOPEN	22
ZEMS-VAL-ZFILPOSITION	23
ZEMS-VAL-ZFILREAD	24

1032: ZEMS-ERR-EOF

The distributor is in log file mode, and you have requested a log file position that is too far in ahead—beyond the end-of-file position on the log file.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1033: ZEMS-ERR-FORWARD-SEARCH

The distributor cannot find the next file to be searched. Either a legitimate file is missing, or the distributor's first event message after the cold-load event message contains an erroneous file name as its previous log name.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1035: ZEMS-ERR-MAX-DEST

A printing distributor has received more than the maximum (five) number of requests for TEXTOUT destinations.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1036: ZEMS-ERR-DEST-ACCESS

Either a printing distributor cannot access a TEXTOUT destination or a forwarding distributor cannot access a target collector.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZEMS-VAL-ZFILOPEN	22
ZEMS-VAL-ZFILWRITE	25

1037: ZEMS-ERR-DEST-EXISTS

The TEXT destination is already in the list of print destinations for this printing distributor and cannot be added.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZEMS-TKN-ADD-TEXTOUT	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1038: ZEMS-ERR-DEST-NOT-FOUND

The TEXT destination is not in the list of print destinations for this printing distributor and cannot be deleted.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZEMS-TKN-DELETE-TEXTOUT	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1039: ZEMS-ERR-CONTEXT

The context submitted did not match the context saved in the distributor.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1041: ZEMS-ERR-ZSPI

An SPI error has been returned by the PROC-ERR procedure.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARAM-ERR	token-type	ZSPI-DDL-PARAM-ERR
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-PROC-ENUM
ZSPI-TKN-SSID-ERR	token-type	ZSPI-TYP-SSID
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZSPI-VAL-SSGETTKN	3
ZSPI-VAL-SSGET	2
ZSPI-VAL-SPUTTKN	8
ZSPI-VAL-SSMOVE	4

1042: ZEMS-ERR-BAD-FILTER

The event filter failed while the distributor was processing the indicated event message. Report the value of ZEMS-TKN-FILTER-ERROR to your Tandem representative; this is an internal error.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-FILTER-ERROR	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-RECORD-ADDRESS	token-type	ZSPI-TYP-INT2
ZEMS-TKN-FILTERNAME	token-type	ZEMS-TYP-CHAR30
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1043: ZEMS-ERR-NO-POOL

The pool space for an additional collector is unavailable.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1044: ZEMS-ERR-NO-EVENT-SOURCE

The distributor has no event-message source collector or log file source.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1045: ZEMS-ERR-DEVTYPE

The collector's log file has an incorrect file-name format, file type, or block length.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-DEVTYPE-ENUM	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-FILECODE	token-type	ZSPI-TYP-INT
ZEMS-TKN-BLOCKLENGTH	token-type	ZSPI-TYP-INT
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZEMS-TKN-DEVTYPE-ENUM (nonshared) indicates what failed and can have the following values:

ZEMS-VAL-FILECODE-BAD	0
ZEMS-VAL-BLOCKLENGTH-BAD	1
ZEMS-VAL-DEVICE-TYPE-BAD	2
ZEMS-VAL-LOGNAME-BAD	3
ZEMS-VAL-VERSION-INCOMPATIBLE	4
ZEMS-VAL-DUP-SOURCE-TARGET	5

1046: ZEMS-ERR-COLL-PROTOCOL

The distributor cannot interpret the status reply from a collector.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following value:

ZEMS-VAL-EMSGET	32
-----------------	----

1047: ZEMS-ERR-BAD-EVENT

The distributor does not recognize the indicated event message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-RECORD-ADDRESS	token-type	ZSPI-TYP-INT2
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZSPI-VAL-SSGETTKN	3
ZEMS-VAL-FILTER-EVAL	20

1049: ZEMS-ERR-BAD-LOG

The distributor declares the current log inaccessible after five consecutive bad event messages have been encountered.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-LOGNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-RECORD-ADDRESS	token-type	ZSPI-TYP-INT2
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1050: ZEMS-ERR-COLL-DISCONNECT

The specified collector has been disconnected. This error message is accompanied by another error message that explains the original problem.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME-ENUM	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1051: ZEMS-ERR-FILES-LOST

The log file that should be next is unavailable or inaccessible. The distributor is using the file whose name is next in the queue of log file names.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-COLNAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LASTLOGFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-NEWLOGFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-RECOVERY-ENUM	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-FAIL-REASON	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1052: ZEMS-ERR-STAT-ONLY

An application issued a GETEVENT or CONTROL command but did not have permission to use these commands to access the specified distributor.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1059: ZEMS-ERR-MAXFLT

An ADD FILTER command was used and the maximum number of filters allowed (10) was exceeded.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1060: ZEMS-ERR-FLT-ALLOC

A resource for a filter could not be allocated, such as filter buffer or filter parameter buffer.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1061: ZEMS-ERR-DIST-ALLOC

A resource for a routing distributor destination could not be allocated: for instance, a print buffer.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1062: ZEMS-ERR-INV-PROFILE

A destination profile has incompatible attributes; for instance, FORMAT ON is specified and the destination is a collector or a process that is set up for launching.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1063: ZEMS-ERR-DEST-CONFLICT

A destination profile was specified in a filter, and TEXTOUT was also given as a run time parameter.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1064: ZEMS-ERR-STARTUP-FAILED

An attempt to launch a destination process was unsuccessful. This could be a process create, an open error, or a write setup message error.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following values:

ZEMS-VAL-NEWPROCESS
 ZEMS-VAL-ZFILOPEN
 ZEMS-VAL-ZFILWRITE

1065: ZEMS-ERR-WRITE-FAILED

An timeout occurred; also, an attempt was made to write an event to a destination process. The error message is repeated every 24 hours if fewer timeouts occur for this destination.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-WRITE-FAIL-COUNT	token-type	ZSPI-TYP-UINT
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

5

Collector Programmatic Interface

This is a reference summary. For detailed information, see the *EMS Manual*.

Collector Commands

Describes the commands used to manage and monitor the operational environment of the primary and alternate collectors and the compatibility distributor. For more detailed information, see “Collector Commands and Responses” in the *EMS Manual*.

Command Summary

Summarizes each collector command.

Command Name	Description
ZCOM-CMD-ADD	Allows you to add objects to the collector. The only supported object is an EMS filter (ZCOM-OBJ-FILTER), which can be a compiled filter, a filter table, or a single burst filter. A maximum of 10 filters can be added to a collector.
ZCOM-CMD-ALTER	Alters objects in the collector. The supported objects are EMS collectors (ZCOM-OBJ-COLL) and EMS filters (ZCOM-OBJ-FILTER).
ZEMS-CMD-CONTROL	Allows you to control the operational environment of the primary and alternate collectors and the compatibility distributor. You can select attributes of new event-message log files, switch the CPU used by a collector, switch the event-message log subvolume, turn event-message buffering on or off, and make other operational decisions.
ZCOM-CMD-DELETE	Allows you to delete objects from the collector. The only supported object is an EMS filter (ZCOM-OBJ-FILTER).
ZCOM-CMD-GETVERSION	Returns the version number of the programmatic interface. This extended-SPI version of the ZEMS-CMD-GETVERSION command returns the conditional response token ZCOM-TKN-GETVSN-LVL to indicate that the collector can process enlightened tokens and data structures.
ZEMS-CMD-GETVERSION	Returns the version number of the programmatic interface.
ZCOM-CMD-INFO	Requests the configuration values for the specified collector object. These configuration values are initially set at collector startup and might have since been modified by the ALTER command or CONTROL commands. The supported objects are the collector (ZCOM-OBJ-COLL) and EMS filter (ZCOM-OBJ-FILTER).
ZCOM-CMD-LISTOBJECTS	Returns the name and type of all or a subset of the object known by the collector.
ZEMS-CMD-REPLACE	Replaces a configured object in the collector with another object. The only supported object is the EMS filter (ZCOM-OBJ-FILTER). Only one filter at a time can be replaced.

Command Name	Description
ZCOM-CMD-STATUS	Request both configuration and status information from the collector. The supported objects are EMS collectors (ZCOM-OBJ-COLL) and EMS filters (ZCOM-OBJ-FILTER). At least one object name must be specified with the ZCOM-CMD-STATUS command.
ZEMS-CMD-STATUS	Returns primary and alternate collector operational information and event-message statistics.
ZCOM-CMD-STOP	Performs an orderly shutdown of the alternate collector and supports the collector object (ZCOM-OBJ-COLL).
ZEMS-CMD-STOP	Performs an orderly shutdown of the alternate collector.

Common SPI Header Tokens

Describes the SPI header tokens that are assigned values for collector commands.

Token (ZCOM-TKN- or ZSPI-TKN-)	Type (ZSPI-TYP-)	Description
COMMAND	ENUM	Standard SPI command-number token; values can be: ZEMS-CMD-GETVERSION 0 ZEMS-CMD-STATUS 1 ZEMS-CMD-CONTROL 2 ZEMS-CMD-STOP 9 (Alternate collector only)
MAXRESP	INT	If not zero, the collector encapsulates the response record in a data list.
OBJECT-TYPE	ENUM	Contains the type of object, if any, to which the command is applied.
OBJNAME	STRING	Specifies the fully-qualified name of the object to be selected for processing.
RETCODE	ENUM	Standard SPI return token.
SERVER-VERSION	UINT	Version number of primary or alternate collector's programmatic interface.
SSID	SSID	Subsystem ID for EMS.

Common Collector Error Numbers

Describes the errors returned in ZSPI-TKN-RETCODE if a command message cannot be interpreted correctly.

Error Name (ZEMS-ERR-)	Error Number	Description
VERSION	1001	Version not supported
INV-CMD	1002	Invalid command
INV-SSID	1003	Invalid subsystem ID (SPI buffer not owned by EMS)
INV-TKN	1004	Extra or unrecognized token
INV-VALUE	1005	Invalid token value
DUP-TKN	1006	Duplicate token code
INV-OBJECT	1008	Invalid object type
INV-OCCURS	1017	Application buffer too small for response
ZSPI	1041	SPI error while decoding command buffer or building response buffer

ADD Command (ZCOM-CMD-ADD)

Adds one or more filter objects (ZCOM-OBJ-FILTER) to a collector. The filter can be a compiled filter, a filter, table, or a burst filter. A maximum of 10 filters can be added to a collector, including any combination of compiled filters, filter tables, and one burst filter.

Command

ZCOM-CMD-ADD

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
<filter param tokens>	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

ALTER Command (ZCOM-CMD-ALTER)

Alters objects in the EMS collector. The supported objects are collectors (ZCOM-OBJ-COLL) and EMS filters (ZCOM-OBJ-FILTER). Note that exactly one object name token is required with this command.

When the object is the EMS collector, this command functions like its basic SPI-compliant counterpart (ZEMS-CMD-CONTROL). The only additional tokens required are the collector control maps.

When the object is an EMS filter, the additional tokens are filter parameter tokens when filter parameters are specified in the filter. Each ZCOM-CMD-ALTER command resets the filter parameters for the specified filters, so that all required filter parameters must have tokens present in the command. The presence of tokens for the optional filter parameters is optional.

Command

ZCOM-CMD-ALTER

Tokens in Command Buffer (-OBJ-COLL)

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZEMS-MAP-COL-CONTROL	token-type	ZEMS-DDL-ACOL-CONTROL.
ZEMS-MAP-ACOL-CONTROL	token-type	ZEMS-DDL-ACOL-CONTROL.
ZEMS-KTN-CDSITPRICPU	token-type	ZSPI-TYPE-UINT
ZEMS-MAP-COL-CONTROL-CDIST	token-type	ZEMS-DDL-COL-CONTROL-CDIST.
ZEMS-MAP-BDS-CONFIG	token-type	ZEMS-DDL-BDS-CONFIG.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Command Buffer (-OBJ-FILTER)

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
<filter param tokens>		
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

CONTROL Command (ZEMS-CMD-CONTROL)

Manages operation of the primary and alternate collectors and the compatibility distributor.

Attributes Changed by CONTROL Command

Lists attributes that can be set with the CONTROL command and associated EMS processes.

Option	Primary Collector	Alternate Collector	Compatibility Distributor
Primary CPU	x	x	x
Log subvolume	x	x	
Next log file	x	x	
Primary extent	x	x	
Secondary extent	x	x	
Rotate files	x	x	
Maxfile	x	x	
EOF refresh	x	x	
Write through cache	x	x	
Protection	x	x	
Blocking	x	x	

Option	Primary Collector	Alternate Collector	Compatibility Distributor
BDS	x	x	
PLF	x	x	
Discaccessid	x		
Allocate		x	
Console out			x
Textout set			x
Mode set			x
User set			x

CONTROL Command Syntax

Lists the tokens and token maps in a CONTROL command buffer.

Command
ZEMS-CMD-CONTROL

Required Tokens or Token Maps

Primary Collector	ZEMS-MAP-COL-CONTROL
Alternate Collector	ZEMS-MAP-COL-CONTROL
	or
	ZEMS-MAP-ACOL-CONTROL
Compatibility Distributor	ZEMS-TKN-CDISTPRICPU
	or
	ZEMS-MAP-COL-CONTROL-CDIST

Tokens in Command Buffer

```
ZEMS-MAP-COL-CONTROL
  def ZEMS-DDL-COL-CONTROL.
    02 ZCOL-PRIMARYCPU           type ZSPI-DDL-UINT.
    02 ZCOL-LOGSUBVOL            type ZSPI-DDL-SUBVOL.
    02 ZCOL-NEXTLOGFILE          type ZSPI-DDL-BOOLEAN.
    02 ZCOL-ROTATEFILES          type ZSPI-DDL-BOOLEAN.
    02 ZCOL-MAXFILENNNN         type ZSPI-DDL-UINT.
    02 ZCOL-PRIMARYEXTENT        type ZSPI-DDL-UINT.
    02 ZCOL-SECONDARYEXTENT      type ZSPI-DDL-UINT.
    02 ZCOL-WRITETHRUCACHE       type ZSPI-DDL-BOOLEAN.
    02 ZCOL-EOFREFRESH           type ZSPI-DDL-BOOLEAN.
    02 ZCOL-DISACCESSID          type ZSPI-DDL-UINT.
    02 ZCOL-PROTECTION           type ZSPI-DDL-UINT.
    02 ZCOL-EVENT-BLOCKING       type ZSPI-DDL-BOOLEAN.
    02 ZCOL-BURSTSUPDETECT       type ZSPI-DDL-BOOLEAN.
    02 ZCOL-PRELOGFILTER         type ZSPI-DDL-BOOLEAN.
  end

ZEMS-MAP-ACOL-CONTROL
  def ZEMS-DDL-ACOL-CONTROL.
    02 ZCOL-ALLOCATE             type ZSPI-DDL-BOOLEAN.
  end

ZEMS-TKN-CDISTPRICPU           type ZSPI-TYP-UINT.

ZEMS-MAP-COL-CONTROL-CDIST
  def ZEMS-DDL-COL-CONTROL-CDIST.
    02 ZCOL-CDIST-PRICPU         type ZSPI-DDL-UINT.
    02 ZCOL-CDIST-CONSOLE-OUT    type ZSPI-DDL-BOOLEAN.
    02 ZCOL-CDIST-TEXTOUT-SET    type ZSPI-DDL-FNAME.
    02 ZCOL-CDIST-MODE-SET       type ZSPI-DDL-ENUM.
    02 ZCOL-CDIST-USER-SET       type ZSPI-DDL-USERID.
  end
```

Tokens in Response Buffer

ZSPI-TKN-RETCODE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-SERVER-VERSION	token-type	ZSPI-TYP-UINT.
ZSPI-TKN-SSID	token-type	ZSPI-TYP-SSID.
ZSPI-TKN-COMMAND	token-type	ZSPI-TYP-ENUM.

CONTROL Command Errors

Lists the names and numbers of the ZSPI-TKN-RETCODE values for the CONTROL command.

Name (ZEMS-ERR-)	Number	Description
INV-CPU	1009	CPU number in ZCOL-PRIMARYCPU was not backup CPU for \$0 or for alternate collector.
CPU-RANGE	1010	CPU number in ZCOL-PRIMARYCPU invalid; must be 0 to 15.
CDIST-CPU	1013	CPU number in ZEMS-TKN-CDISTPRICPU was not the primary or backup CPU for \$Z0.
REQ-TKN	1015	Collector CONTROL command needs ZEMS-TKN-CDISTPRICPU, ZEMS-MAP-COL-CONTROL-CDIST, or ZEMS-MAP-ACOL-CONTROL.
LOG-ACCESS	1031	Command could not be completed because an I/O error occurred while collector was accessing log file or a ZZEVCNF. Response error list contains ZEMS-TKN-ERROR, ZEMS-TKN-ZFILERR, and ZEMS-TKN-FAILFILENAME.
OPEN-LOG	1034	Could not access subvolume specified in ZCOL-LOGSUBVOL.
ACC-VIOL	1048	Security violation.
INV-MODE	1053	Invalid mode for \$Z0. Must be CRITICAL-ONLY, ALL, or STOP.
CDIST-DOWN	1054	\$Z0 is down.
NO-BACKUP	1055	Contents of ZCOL-PRIMARYCPU field is alternate collector's backup CPU, but alternate collector does not have backup.
ALLOC-LOG	1056	Log file could not be allocated. Response error list contains ZEMS-TKN-ERROR, ZEMS-TKN-ZFILERR, and ZEMS-TKN-FAILFILENAME.

DELETE Command (ZCOM-CMD-DELETE)

Deletes objects from the collector. The only object type supported is the EMS filter (ZCOM-OBJ-FILTER).

Command

ZCOM-CMD-DELETE

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

GETVERSION Command (ZCOM-CMD-GETVERSION)

Returns collector version information and optional supplemental information. Unlike ZEMS-CMD-GETVERSION, this command returns the ZCOM-TKN-GETVSN-LVL token.

Command

ZCOM-CMD-GETVERSION

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZEMS-TKN-GETVSN-SUPP-PARAMS	token-type	ZSPI-TYP-BOOLEAN.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-SERVER-BANNER	token-type	ZSPI-TYP-CHAR50.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

GETVERSION Command (ZEMS-CMD-GETVERSION)

Returns the version number of the programmatic interface to the primary or alternate collector.

Command

ZEMS-CMD-GETVERSION

Tokens in Command Buffer

None

Tokens in Response Buffer

ZSPI-TKN-SERVER-BANNER	token-type	ZSPI-TYP-CHAR50.
ZSPI-TKN-SERVER-VERSION	token-type	ZSPI-TYP-UINT.
ZSPI-TKN-RETCODE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-SSID	token-type	ZSPI-TYP-SSID.
ZSPI-TKN-COMMAND	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.

INFO Command (ZCOM-CMD-INFO)

Requests the configuration values of the specified object. These values are initially set at collector startup and might have subsequently been modified by ZCOM-CMD-ALTER or ZEMS-CMD-CONTROL. Supported objects are collectors (ZCOM-OBJ-COLL) and filters (ZCOM-OBJ-FILTER).

Command

ZCOM-CMD-INFO

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-SUB	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-INFO-ENQ	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-INFO-DEFLT	token-type	ZSPI-TYP-BOOLEAN.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

LISTOBJECTS Command (ZCOM-CMD-LISTOBJECTS)

Returns the name and type of all or a subset of objects known by the EMS collector. Two forms of the LISTOBJECTS command are supported: the first requests all object types and names; the second requests only the object names of a specified object type. Supported object types are ZCOM-OBJ-NULL, ZCOM-OBJ-COLL, and ZCOM-OBJ-FILTER.

Command

ZCOM-CMD-LISTOBJECTS

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

REPLACE Command (ZEMS-CMD-REPLACE)

Replaces one configured object in the EMS collector with another object. The only supported object is an EMS filter (ZCOM-OBJ-FILTER). The filter can be a compiled filter, a filter table, or a burst filter.

Command

ZEMS-CMD-REPLACE

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-NEW-OBJNAME	token-type	ZSPI-TYP-STRING.
<filter param token(s)>		
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

STATUS Command (ZCOM-CMD-STATUS)

Requests both configuration and status information from the EMS collector. The supported object types are collectors (ZCOM-OBJ-COLL) and filters (ZCOM-OBJ-FILTER). At least one object name token is required. Note that the ZEMS-CMD-STATUS command does not support these objects.

Command

ZEMS-CMD-REPLACE

Tokens in Command Buffer

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-SUB	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer (OBJ-COLL)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-OBJSTATE	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-COL-STATUS	token-type	ZEMS-DDL-COL-STATUS.
ZEMS-MAP-ACOL-STATUS	token-type	ZEMS-DDL-ACOL-STATUS.
ZEMS-MAP-COL-CDIST-STATUS	token-type	ZEMS-DDL-COL-CDIST-STATUS.
ZEMS-MAP-BDS-INFO	token-type	ZEMS-DDL-BDS-INFO.
ZEMS-MAP-BDS-STATS	token-type	ZEMS-DDL-BDS-STATS.
ZEMS-MAP-PLF-STATUS	token-type	ZEMS-DDL-PLF-STATS.
ZEMS-MAP-BURST-STATUS	token-type	ZEMS-DDL-BURST-STATUS.
ZEMS-MAP-BURST-SUBJ-VALUE	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer (OBJ-FILTER)

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-OBJSTATE	token-type	ZSPI-TYP-ENUM.
ZEMS-MAP-STATUS-FILTER	token-type	ZEMS-DDL-STATUS-FILTER.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

STATUS Command (ZEMS-CMD-STATUS)

Returns event message statistics and current collector attribute information to the distributor or other application programs.

Command

ZEMS-CMD-STATUS

Tokens in Command Buffer

None

Tokens in Response Buffer

```

ZEMS-MAP-COL-STATUS
  def ZEMS-DDL-COL-STATUS.
    02 ZCOL-PRIMARYCPU           type ZSPI-DDL-UINT.
    02 ZCOL-BACKUPCPU           type ZSPI-DDL-UINT.
    02 ZCOL-PRIORITY            type ZSPI-DDL-UINT.
    02 ZCOL-LOGDISCERROR        type ZSPI-DDL-UINT.
    02 ZCOL-CURRENTFILENAME     type ZSPI-DDL-FNAME.
    02 ZCOL-CURRENTRECORD       type ZSPI-DDL-INT2.
    02 ZCOL-DEFAULTFILENAME     type ZSPI-DDL-FNAME.
    02 ZCOL-PRIMARYEXTENT       type ZSPI-DDL-UINT.
    02 ZCOL-SECONDARYEXTENT     type ZSPI-DDL-UINT.
    02 ZCOL-ROTATEFILES         type ZSPI-DDL-BOOLEAN.
    02 ZCOL-MAXFILENNNN        type ZSPI-DDL-UINT.
    02 ZCOL-EOFREFRESH          type ZSPI-DDL-BOOLEAN.
    02 ZCOL-WRITETHRUCACHE      type ZSPI-DDL-BOOLEAN.
    02 ZCOL-PROTECTION          type ZSPI-DDL-UINT.
    02 ZCOL-EVENTSRECEIVED      type ZSPI-DDL-INT2.
    02 ZCOL-EVENTSLOGGED        type ZSPI-DDL-INT2.
    02 ZCOL-EVENTSDISCARDED     type ZSPI-DDL-INT2.
    02 ZCOL-OPENSRECEIVED       type ZSPI-DDL-UINT.
    02 ZCOL-CLOSESRECEIVED      type ZSPI-DDL-UINT.
    02 ZCOL-FILESWITCHES        type ZSPI-DDL-UINT.
    02 ZCOL-DISCERRORS          type ZSPI-DDL-UINT.
    02 ZCOL-INVALIDEVENTS      type ZSPI-DDL-UINT.
    02 ZCOL-BUFFERFAILURES      type ZSPI-DDL-UINT.
    02 ZCOL-EVENTBLOCKING       type ZSPI-DDL-BOOLEAN.
    02 ZCOL-BURSTSUPDETECT      type ZSPI-DDL-BOOLEAN.
    02 ZCOL-BURST-PRELOGFILTER  type ZSPI-DDL-BOOLEAN.
    02 ZCOL-CURRENTBURSTS       type ZSPI-DDL-UINT.
  end
ZEMS-MAP-ACOL-STATUS
  def ZEMS-DDL-ACOL-STATUS
    02 ZCOL-POOLPAGES           type ZSPI-DDL-UINT.
    02 ZCOL-REPLYAFTERWRITE     type ZSPI-DDL-UINT.
  end
ZEMS-MAP-COL-CDIST-STATUS
  def ZEMS-DDL-COL-CDIST-STATUS.
    02 ZCOL-CDISTPRICPU         type ZSPI-DDL-UINT.
    02 ZCOL-CDISTBKUPCPU        type ZSPI-DDL-UINT.
    02 ZCOL-STOPCOMPATDIST      type ZSPI-DDL-BOOLEAN.
    02 ZCOL-DISTR-ERROR         type ZSPI-DDL-UINT.
    02 ZCOL-OPRLOG-ERROR        type ZSPI-DDL-UINT.
    02 ZCOL-CDIST-MODE          type ZSPI-DDL-ENUM.
    02 ZCOL-CDIST-TEXTOUT       type ZSPI-DDL-FNAME.
    02 ZCOL-CDIST-USER          type ZSPI-DDL-USERID.
    02 ZCOL-CDIST-DEF-TEXTOUT   type ZSPI-DDL-FNAME.
  end
ZSPI-TKN-RETCODE               token-type ZSPI-TYP-ENUM.
ZSPI-TKN-SERVER-VERSION       token-type ZSPI-TYP-UINT.
ZSPI-TKN-SSID                 token-type ZSPI-TYP-SSID.
ZSPI-TKN-COMMAND              token-type ZSPI-TYP-ENUM.

```

STOP Command (ZCOM-CMD-STOP)

Stops the alternate collector. Unlike its basic SPI-compliant counterpart, ZEMS-CMD-STOP, the ZCOM-CMD-STOP command supports the collector object (ZCOM-OBJ-COLL).

Command:

ZCOM-CMD-STOP

Tokens in Command Buffer:

ZPSI-TKN-MAXRESP	token-type	ZSPI-TYP-INT.
ZCOM-TKN-XMGR	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-REQID-xxxx		
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZCOM-TKN-CMD-POWER	token-type	ZSPI-TYP-ENUM-CMDPOWER.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

Tokens in Response Buffer:

ZSPI-TKN-DATALIST	token-type	ZSPI-TYP-LIST.
ZCOM-TKN-XMGR	token-type	ZPSI-TYP-STRING.
ZCOM-TKN-RETCODE	token-type	ZSPI-TYP-INIT.
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST.
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR.
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL.
ZSPI-TKN-CONTEXT	token-type	ZSPI-TYP-BYTESTRING.

STOP Command (ZEMS-CMD-STOP)

Performs an orderly shutdown of the alternate collector.

Command:

ZEMS-CMD-STOP

Tokens in Command Buffer:

None

Tokens in Response Buffer:

ZSPI-TKN-SERVER-VERSION	token-type	ZSPI-TYP-UINT.
ZSPI-TKN-COMMAND	token-type	ZSPI-TYP-ENUM.
ZSPI-TKN-SSID	token-type	ZSPI-TYP-SSID.
ZSPI-TKN-RETCODE	token-type	ZSPI-TYP-ENUM.

STOP Command Errors

Describes the two values for ZSPI-TKN-RETCODE.

Name (ZEMS-ERR)	Number	Description
ACC-VIOL	1048	Requester's user ID not allowed to use STOP command. ZSPI-TKN-ERROR is returned in response error list.
LOGGING-STOPPED	1057	Alternate collector is currently holding events in event buffer pool and cannot write them to disk because logging has stopped.

Collector Event Messages

Describes the event messages generated by the primary collector or an alternate collector.

Primary Collector Event Message Summary

Summarizes the event messages generated by the primary collector.

Event Number	Symbolic Name (ZEMS-EVT-)	Description	Critical
1-511	(Operator Messages)	Numbered operator-console messages	BEL*
512	WRITE-TO-0	Events reported by WRITES of text to \$0	BEL*
513	COLD-LOAD	System cold loaded	Y
514	FILESWITCH	Next log file used	Y
515	COLL-DISC-FAILED	Log file inaccessible	Y
516	COMPAT-DISTR-STARTED	\$Z0 was started	N
517	COMPAT-DISTR-STOPPED	\$Z0 inaccessible	Y
518	COL-EVENT-DISCARDS	Flood recovery	Y
519	MSGR-EVENTS-DISCARDED	Special flood recovery	Y
520	FILE-ROTATE-PURGE	Automatic file purge	Y
521	LOGGING-STOPPED	Logging has stopped	Y
524	LOGTIME-DECREASE	Log time decreased	Y
525	INVALIDEVENT	Invalid event message	N
538	BURST-START	An event burst, as defined by the BDS configuration parameters, has been detected.	Y

Event Number	Symbolic Name (ZEMS-EVT-)	Description	Critical
539	BURST-END	An event burst, as defined by the BDS configuration parameters, has ended.	N
540	PLF-ERROR	A filter evaluation error occurred when a filter was applied to an event.	Y

* This message is critical if the event message text contains the ASCII equivalent of a BEL character.

Alternate Collector Event Message Summary

Summarizes the event messages generated by the alternate collector.

Event Number	Symbolic Name (ZEMS-EVT-)	Description	Critical
512	WRITE-TO-0	Events reported by WRITES of text to alternate collector	BEL*
514	FILESWITCH	Next log file used	Y
520	FILE-ROTATE-PURGE	Automatic file purge	Y
521	LOGGING-STOPPED	Logging has stopped	Y
522	COLLECTOR-RUN	First event after RUN	N
523	ACOL-EVENT-DISCARDS	Number of events discarded after flooding	Y
524	LOGTIME-DECREASE	Log time decreased	Y
525	INVALIDEVENT	Invalid event message	N
526	ACOL-INTERNAL-ERR	Inconsistency in alternate collector operation	Y
527	ACOL-SHUTDOWN	Collector shutdown	N
528	ACOL-ALLOCATESEG-ERR	Cannot allocate extended data segment	Y
529	ACOL-CHECKOPEN-FAILED	Unable to open file	Y
530	ACOL-TAKEOVER	Takeover by collector backup process	Y
531	ACOL-CREATEBACKUP-ERR	Cannot create backup	Y
532	ACOL-BACKUP-CREATED	Backup process created	N
533	ACOL-BACKUP-ABENDED	Backup process abended	Y
534	ACOL-BACKUP-DELETED	CPU failure of backup process	Y
535	ACOL-CHECKPOINT-ERROR	I/O error during checkpoint	Y
536	COL-PURGETABLE-OVRFLO	Log file could not be purged automatically	N

Event Number	Symbolic Name (ZEMS-EVT-)	Description	Critical
537	CONFIG-WARNING	Configuration file is specifying the unused configuration bits	N
538	BURST-START	An event burst, as defined by the BDS configuration parameters, has been detected.	Y
539	BURST-END	An event burst, as defined by the BDS configuration parameters, has ended.	N
540	PLF-ERROR	A filter evaluation error occurred when a filter was applied to an event.	Y

* This message is critical if the event message text contains the ASCII equivalent of a BEL character.

Header Tokens

Header tokens appear in every event message; they are not described with the event syntax.

SPI Header Token

Describes the SPI header token that appears in the header of every event message.

ZSPI-TKN-	ZSPI-TYP-	Description
MAX-FIELD-VERSION	INT	The highest field version among the non-null fields of structures added to the buffer with EMSADDTOKENMAPS.
SSID	SSID	Contains subsystem ID for EMS.
USEDLEN	INT	The number of bytes actually used in the buffer. A subsystem that reports events can use this token to determine how many bytes to send to \$0 through the WRITEREAD procedure.

EMS Header Tokens

Describes the EMS header tokens that appear in the header of every collector event message.

ZEMS-TKN-	ZSPI-TYP-	Description
EVENTNUMBER	INT	Shared. Number of an event message.
GENTIME	TIMESTAMP	Shared. Greenwich mean time when subsystem created event message.
LOGTIME	TIMESTAMP	Shared. Greenwich mean time when collector wrote event-message to log files.

ZEMS-TKN-	ZSPI-TYP-	Description
NODENUM	INT2	Shared. Expand system number of the system where event was reported; must be 16 bits or less.
CPU	INT	Shared. CPU number of reporting subsystem process.
PIN	INT	Shared. PIN of reporting subsystem process.
PROC-DESC	STRING	Shared. <i>proc-desc</i> of reporting subsystem process; PID in token must be less than 8 bits.
USERID	UINT	Shared. User ID of reporting subsystem process.
CONSOLE-PRINT	BOOLEAN	No longer used by the compatibility distributor.
EMPHASIS	BOOLEAN	Shared. Set to TRUE to emphasize event message as critical.
SUPPRESS-DISPLAY	BOOLEAN	Shared. If TRUE, the ViewPoint application will not display the event message. ViewPoint will display it if the token is either FALSE or missing.
CONTENT-STANDARD	ENUM	Shared; indicates the type of standard event.
CONTENT-USER	ENUM	Shared; identifies the type of a subsystem-defined event.

Common Data Portion Tokens

Describes the data tokens used in collector commands.

ZEMS-TKN-	ZSPI-TYP-	Description
SUBJECT-MARK	SSCTL	Shared. Marks following token as subject of event message.

1-511: Tokenized Operator Console Messages

These represent the numbered operator console messages. Each event message corresponds to a numbered console message; the event message number is the same as the number of the console message.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-OPMSG	token-type	ZEMS-DDL-OPMSG.
ZEMS-TKN-XSYSPID	token-type	ZEMS-DDL-XSYSPID.

Conditional Tokens

ZEMS-MAP-EXIOADDR	token-type	ZSPI-TYP-EXIOADDR.
-------------------	------------	--------------------

Event-Message Text

Depends on `opmsg`

Subject

ZEMS-TKN- Private to Tandem.
XSYSPID

Header Token Values

ZEMS-TKN- TRUE, depending on message.
EMPHASIS

512: Tokenized Text Message

ZEMS-EVT-WRITE-TO-0 event message.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-XSYSPID	token-type	ZEMS-DDL-XSYSPID.
ZEMS-TKN-TEXT	token-type	ZSPI-TYP-STRING.

Event-Message Text

Depends on ZEMS-TKN-TEXT.

Subject

ZEMS-TKN- Private to Tandem.
XSYSPID

Header Token Values

ZEMS-TKN- TRUE, depending on BEL character.
EMPHASIS

Unconditional Token

Describes the unconditional token specific to this text message.

ZEMS-TKN-	ZSPI-TYP-	Description
TEXT	STRING	Text sent to the collector by WRITE procedure. Text more than 102 bytes is truncated.

513: ZEMS-EVT-COLD-LOAD

CRITICAL. System cold load and beginning of disk logging.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

```
EMS: PRIMARY COLLECTOR $0,
DISK EVENT LOGGING STARTED
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

514: ZEMS-EVT-FILESWITCH

CRITICAL. Primary and alternate collectors successfully switched log files. Can be action-completion message for ZEMS-EVT-LOGGING-STOPPED (521).

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-LASTLOGFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-NEWLOGFILE	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-LOGSWITCHREASON	token-type	ZSPI-TYP-ENUM
ZEMS-MAP-COL-STATUS	token-type	ZEMS-DDL-COL-STATUS
ZEMS-TKN-LCT-LOGTIME	token-type	ZSPI-TYP-TIMESTAMP

Conditional Tokens

ZEMS-TKN-ACTION-NEEDED	token-type	ZSPI-TYP-BOOLEAN
ZEMS-TKN-ACTION-ID	token-type	ZSPI-TYP-INT

Event-Message Text

```
EMS: COLLECTOR proc-desc SWITCHED LOG FILES
FROM FILE lastlogfile TO newlogfile
  { BY OPERATOR }
  { BECAUSE DISK INACCESSIBLE }
  { BECAUSE OLD FILE FULL }
(depending on logswitchreason)
```


Subject

ZEMS-TKN-COLLECTOR Collector type. Primary collector value is 1; alternate collector value is 2.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Text Values

proc-desc ZEMS-TKN-PROC-DESC
lastlogfile ZEMS-TKN-LASTLOGFILE
newlogfile ZEMS-TKN-NEWLOGFILE
logswitchreason ZEMS-TKN-LOGSWITCHREASON
n

515: ZEMS-EVT-COLL-DISC-FAILED

CRITICAL. Inaccessible log file.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.

Event-Message Text

EMS: COLLECTOR *proc-desc* DISK LOG FILE INACCESSIBLE,
Guardian FILE ERROR *zfilerr* REPORTED ON FILE *failfilename*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Text Values

proc-desc ZEMS-TKN-PROC-DESC
zfilerr ZEMS-TKN-ZFILERR
failfilename ZEMS-TKN-FAILFILENAME

517: ZEMS-EVT-COMPAT-DISTR-STOPPED

Primary collector cannot successfully access the compatibility distributor.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-COMPATDISTCRTPID	token-type	ZSPI-TYP-CRTPID.
ZEMS-TKN-CDIST-CPU-PIN	token-type	ZSPI-TYP-UINT-PAIR.

Event-Message Text

EMS: COMPATIBILITY DISTRIBUTOR FAILED -
 Guardian ERROR - *zfilerr*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Text Values

zfilerr ZEMS-TKN-ZFILERR

518: ZEMS-EVT-COL-EVENT-DISCARDS

CRITICAL. Primary collector has recovered from an event-flooding condition.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-COL-EVENT-DISCARDS	token-type	ZEMS-TYP-COL-EVENT-
ZEMS-TKN-OPMSG	token-type	ZEMS-TYP-OPMSG.

Event-Message Text

```
EMS:  count-0 SENDOPMSGs, count-1 ESPSENDS, count-2 WRITES,
AND count-3 WRITEREADS DISCARDED BY COLLECTOR proc-desc, PID
cpu, pin
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

ZEMS-TKN-COL-EVENT-DISCARDS Four integer array with the DDL definition:

```
def ZEMS-TKN-COL-EVENT-DISCARDS.
  02 ZSENDOPMSG type ZSPI-DDL-UINT.
  02 ZMESSENGER type ZSPI-DDL-UINT.
  02 ZUSERTEXT type ZSPI-DDL-UINT.
  02 ZUSEREVENT type ZSPI-DDL-UINT.
end
```

Text Values

<i>count-0</i>	ZSENDOPMSG
<i>count-1</i>	ZMESSENGER
<i>count-2</i>	ZUSERTEXT
<i>count-3</i>	ZUSEREVENT
<i>proc-desc</i>	ZEMS-TKN-PROC-DESC
<i>cpu</i>	ZEMS-TKN-CPU
<i>pin</i>	ZEMS-TKN-PIN

519: ZEMS-EVT-MSGR-EVENTS-DISCARDED

CRITICAL. The SENDVIAMESSSENGER procedure (details internal to Tandem) had to discard event information.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-MSGR-EVENTS-DISCARDED	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-MESSENGERCPU	token-type	ZSPI-TYP-UINT.

Event-Message Text

```
EMS:  count EVENTS DISCARDED BY SENDVIAMESSSENGER IN CPU
      messenger-cpu
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
MSGR-EVENTS-DISCARDED	UINT	Number of messages discarded by SENDVIAMESSSENGER procedure.
MESSENGERCPU	UINT	CPU of SENDVIAMESSSENGER procedure that discarded the event messages.

Text Values

count ZEMS-TKN-MSGR-EVENTS-DISCARDED
messenger-cpu ZEMS-TKN-MESSENGERCPU

520: ZEMS-EVT-FILE-ROTATE-PURGE

Primary or alternate collector has purged data of oldest log file or performed a rename/purge data operation on a log file.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-PURGEDLOGFILE	token-type	ZSPI-TYP-FNAME.

Event-Message Text

```
EMS: LOG FILE purgedlogfile PURGED BY COLLECTOR proc-desc,
      ROTATEFILES OPTION
```

Subject

ZEMS-TKN-COLLECTOR	Collector type. Value is 1 for primary collector. Value is 2 for alternate collector.
--------------------	---

Header Token Values

ZEMS-TKN-EMPHASIS	TRUE.
-------------------	-------

Unconditional Token Description

Describes the unconditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
PURGEDLOGFILE	FNAME	Name of file the collector renamed or purged.

Text Values

purgedlogfile ZEMS-TKN-PURGEDLOGFILE

proc-desc ZEMS-TKN-PROC-DESC

521: ZEMS-EVT-LOGGING-STOPPED

CRITICAL and ACTION-NEEDED. Primary or alternate collector cannot log event messages. Action-completion message is ZEMS-EVT-FILESWITCH (514).

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-LOGSTOPREASON	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-ACTION-NEEDED	token-type	ZSPI-TYP-BOOLEAN.
ZEMS-TKN-ACTION-ID	token-type	ZSPI-TYP-INT.

Event-Message Text

```
EMS: COLLECTOR proc-desc EVENT LOGGING STOPPED BECAUSE
      { DISK FAILED }
      { NOROTATE OPTION }
      (depending on logstopreason)
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
LOGSTOPREASON	ENUM	Reason collector stopped logging.
ACTION-NEEDED	BOOLEAN	TRUE. Indicates whether some action is required.
ACTION-ID	INT	Value is 0; corresponding message (514) has same value.

Text Values

proc-desc ZEMS-TKN-PROC-DESC

logstopreason ZEMS-TKN-LOGSTOPREASON

522: ZEMS-EVT-COLLECTOR-RUN

First message placed in alternate collector log file after collector RUN command is issued. Informational only.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK token-type ZSPI-TYP-SSCTL.
 ZEMS-TKN-COLLECTOR token-type ZSPI-TYP-ENUM.

Event-Message Text

```
EMS: ALTERNATE COLLECTOR processname, DISK EVENT LOGGING
      STARTED
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS FALSE.

Text Values

processname ZEMS-TKN-PROC-DESC

523: ZEMS-EVT-ACOL-EVENT-DISCARDS

CRITICAL. Alternate collector has recovered from event-flooding or discarded invalid events.

Unconditional tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-ACOL-EVENTS-DISCARDS	token-type	ZSPI-TYP-UINT.

Event-Message Text

EMS: *count* EVENTS DISCARDED BY COLLECTOR *name*, PID *cpu*, *pin*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Description

Describes the unconditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
ACOL-EVENTS-DISCARDS	UINT	Number of events discarded by alternate collector.

Text Values

count ZEMS-ACOL-EVENTS-DISCARDS
name ZEMS-TKN-PROC-DESC
cpu ZEMS-TKN-CPU
pin ZEMS-TKN-PIN

524: ZEMS-EVT-LOGTIME-DECREASE

CRITICAL. Log timestamp of message is out of sequence.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-OLD-LOGTIME	token-type	ZSPI-TYP-TIMESTAMP.
ZEMS-TKN-NEW-LOGTIME	token-type	ZSPI-TYP-TIMESTAMP.

Event-Message Text

EMS: LOGTIME ERROR ENCOUNTERED - TIME DECREASED FROM *time-1*
date-1 TO *time-2* *date-2*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
OLD-LOGTIME	TIMESTAMP	Log time of event message before system-time change.
NEW-LOGTIME	TIMESTAMP	Log time of event message after system-time change.

Text Values

time-1 date-1 ZEMS-TKN-OLD-LOGTIME

time-2 date-2 ZEMS-TKN-NEW-LOGTIME

525: ZEMS-EVT-INVALIDEVENT

Primary or alternate collector has received an invalid event message.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-XSYSPID	token-type	ZEMS-DDL-XSYSPID.

Event-Message Text

EMS: COLLECTOR RECEIVED BAD EVENT BUFFER FROM SYSTEM *sysnum*,
PID *cpu,pin*

Subject

ZEMS-TKN-COLLECTOR	Collector type. Value is 1 for primary collector. Value is 2 for alternate collector.
--------------------	---

Header Token Values

ZEMS-TKN-EMPHASIS	FALSE.
-------------------	--------

Unconditional Token Description

Describes the unconditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
SYSPID	SYSPID	Private to Tandem.

Text Values

sysnum ZEMS-TKN-SYSPID
cpu ZEMS-TKN-SYSPID
pin ZEMS-TKN-SYSPID

526: ZEMS-EVT-ACOL-INTERNAL-ERR

CRITICAL. Alternate collector detected an inconsistency in its own operation. This event message should never occur.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-CODESEG	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-PREG	token-type	ZSPI-TYP-UINT.

Event-Message Text

EMS: INTERNAL ERROR AT %codeseg.preg

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
CODESEG	UINT	Code segment number where error was detected.
PREG	UINT	P register contents where error was detected.

Text Values

%codeseq ZEMS-TKN-CODESEG

preg ZEMS-TKN-PREG

527: ZEMS-EVT-ACOL-SHUTDOWN

Alternate collector shut down in response to a STOP SPI command. Informational message only.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

EMS: COLLECTOR *processname* SHUTDOWN

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS FALSE.

Text Values

processname ZEMS-TKN-PROC-DESC

528: ZEMS-EVT-ACOL-ALLOCATESEG-ERR

CRITICAL. Alternate collector process cannot allocate extended data segment.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-SEGALLOC-ERROR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

```

EMS:  SEGMENT ALLOCATION ERROR zfilerr ON VOLUME failfilename
Where segalloc-error can be one of the following:
{
CREATE OR OPEN ERROR
PARAMETER ERROR
BONDS ERROR
ILLEGAL SEGMENT ID
ILLEGAL SEGMENT SIZE
UNABLE TO ALLOCATE SEGMENT SPACE
UNABLE TO ALLOCATE SEGMENT PAGE TABLE SPACE
}

```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

ZEMS-TKN-
FAILFILENAME Name of the swap volume.

Header Token Values

ZEMS-TKN-
EMPHASIS TRUE.

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
FAILFILENAME	FNAME	Swap volume name.
ZFILERR	UINT	Error code associated with an event.

Text Values

zfilerr ZEMS-TKN-ZFILERR

failfilename ZEMS-TKN-FAILFILENAME

529: ZEMS-EVT-ACOL-CHECKOPEN-FAILED

CRITICAL. Backup alternate collector process is unable to open a file previously opened by the primary process.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.

Event-Message Text

```
EMS: CHECKOPEN FAILED, ERROR zfilerr ON failfilename
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

ZEMS-TKN-FAILFILENAME File that could not be opened by backup alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

Unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
FAILFILENAME	FNAME	File that could not be opened by backup alternate collector.
ZFILERR	UINT	Error code associated with an event.

Text Values

zfilerr ZEMS-TKN-ZFILERR

failfilename ZEMS-TKN-FAILFILENAME

530: ZEMS-EVT-ACOL-TAKEOVER

CRITICAL. The alternate collector generates this event message after a takeover by the alternate collector backup process.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-TAKEOVER-REASON	token-type	ZSPI-TYP-ENUM.

Event-Message Text

```
EMS: TAKEOVER BY BACKUP ( takeover text )
Where takeover text depends on the value of takeover-reason.
The values of takeover text are:
```

```
{ PRIMARY STOPPED }
{ PRIMARY ABENDED }
{ PRIMARY CPU IS DOWN }
{ OPERATOR REQUESTED SWITCH }
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Description

Describes the unconditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
TAKEOVER-REASON	ENUM	Reason for the takeover.

531: ZEMS-EVT-ACOL-CREATEBACKUP-ERR

CRITICAL. Alternate collector cannot create a backup due to a NEWPROCESS error.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TKN-FNAME.
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TKN-ENUM.
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TKN-INT.
ZEMS-TKN-PROCREATE-ERROR	token-type	ZSPI-TYP-ENUM.

Conditional Tokens

ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
------------------	------------	----------------

Event-Message Text

EMS: UNABLE TO CREATE BACKUP PROCESS IN CPU *cpu*, ERROR
 Errnum: *errtext*

Where *errtext* depends on the value of *newprocess-error*. The values of *errtext* are:

```

  { NO PROCESS CONTROL BLOCK AVAILABLE
  { ERROR ON PROGRAMFILE: zfilerr
  { UNABLE TO ALLOCATE MAP
  { ERROR ON SWAP FILE : zfilerr
  { ILLEGAL FILE FORMAT
  { PROCESS NAME ERROR : zfilerr
  { LIBRARY CONFLICT
  { UNABLE TO COMMUNICATE WITH SYSTEM MONITOR
  { ERROR ON LIBRARY FILE: zfilerr
  { LIBRARY AND PROGRAM FILE ARE THE SAME
  { EXTENDED SEGMENT ERROR: zfilerr
  { SWAP FILE ERROR: zfilerr
  { ILLEGAL HOME TERMINAL, ERROR: zfilerr
  }
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

ZEMS-TKN-PROGRAMFILE Alternate collector object file name.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
NEWPROCESS-CPU	ENUM	CPU specified in the NEWPROCESS command.
NEWPROCESS-PRIORITY	INT	Alternate collector priority.
NEWPROCESS-ERROR	ENUM	New process error.

Conditional Token Description

Describes the conditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
ZFILERR	UINT	Error code associated with an event.

Text Values

<i>cpu</i>	ZEMS-TKN-NEWPROCESS-CPU
<i>newprocess-error</i>	ZEMS-TKN-NEWPROCESS-ERROR
<i>errtext</i>	ZEMS-TKN-NEWPROCESS-ERROR
<i>zfilerr</i>	ZEMS-TKN-ZFILERR

532: ZEMS-EVT-ACOL-BACKUP-CREATED

New collector process in backup CPU was successfully created.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-PROGRAMFILE	token-type	ZSPI-TYP-FNAME.
ZEMS-TKN-NEWPROCESS-CPU	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-NEWPROCESS-PRIORITY	token-type	ZSPI-TYP-INT.

Event-Message Text

EMS: BACKUP CREATED IN CPU *cpu*

Subject

ZEMS-TKN-COLLECTOR	Collector type. Value is 2 for alternate collector.
ZEMS-TKN-PROGRAMFILE	Program file for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS	FALSE.
-------------------	--------

Unconditional Token Descriptions

Describes the unconditional tokens specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
NEWPROCESS-CPU	ENUM	CPU specified in the NEWPROCESS command.
NEWPROCESS-PRIORITY	INT	Alternate collector priority.

Text Values

cpu ZEMS-TKN-NEWPROCESS-CPU

533: ZEMS-EVT-ACOL-BACKUP-ABENDED

CRITICAL. Alternate collector backup process has terminated abnormally.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

EMS: BACKUP PROCESS ABENDED

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

534: ZEMS-EVT-ACOL-BACKUP-DELETED

CRITICAL. CPU in which alternate collector backup process is running has failed.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

EMS: BACKUP PROCESS DELETED (CPU DOWN)

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

535: ZEMS-EVT-ACOL-CHECKPOINT-ERR

CRITICAL. An I/O error has occurred during a checkpoint operation.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-SUBJECT-MARK	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.

Event-Message Text

EMS: CHECKPOINT FAILED, ERROR *zfilerr*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 2 for alternate collector.

Header Token Values

ZEMS-TKN-EMPHASIS TRUE.

Unconditional Token Description

Describes the unconditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
ZFILERR	UINT	Error code associated with an event.

Text Values

zfilerr ZEMS-TKN-ZFILERR

536: ZEMS-EVT-COL-PURGETABLE-OVRFLO

Internal table containing log file names overflow, indicating the file must be purged manually.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-NODENUM	token-type	ZSPI-TYP-INT2.
ZEMS-TKN-PURGEDLOGFILE	token-type	ZSPI-TYP-FNAME.

Event-Message Text

```
EMS: COLLECTOR \NODE.00,003 COULD NOT PURGE LOG FILE
$SYSTEM.ZLOG77.ZZEV0003 AUTOMATICALLY.
FILE MUST BE PURGED MANUALLY.
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1.

Header Token Values

ZEMS-TKN-EMPHASIS FALSE.

537: ZEMS-EVT-COL-CONFIG-WARNING (*D-series only*)

The event collector, \$0, generates this event during the initialization process after a cold load if bit 15 of the EMSFLAGS word is set to 1 by SYSGEN.

Unconditional Tokens (excluding header tokens)

ZEMS-TKN-CONFIG-ITEM	token-type	ZSPI-TYP-UINT.
ZEMS-TKN-SPEC-CONFIG-VALUE	token-type	ZSPI-TYP-SSCTL.
ZEMS-TKN-USED-CONFIG-VALUE	token-type	ZSPI-TYP-ENUM.

Event-Message Text

```
EMS: SYSTEM_PROCESS_MODIFIERS SECTION OF THE SYSGEN
CONFIGURATION FILE CONTAINS AN INCORRECT VALUE.
ITEM: config-item, SPECIFIED VALUE: spec-config-value, USED
VALUE: used-config-value.
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1.

Header Token Values

ZEMS-TKN-EMPHASIS FALSE.

Unconditional Token Description

Describes the unconditional token specific to this event message.

ZEMS-TKN-	ZSPI-TYP-	Description
CONFIG-ITEM	ENUM	Value specified by SYSGEN
SPEC-CONFIG-VALUE	UINT	Value specified by SYSGEN
USED-CONFIG-VALUE	UINT	Actual value used by \$0

Text Values

<i>config-item</i>	ZEMS-TKN-CONFIG-ITEM
<i>spec-config-value</i>	ZEMS-TKN-SPEC-CONFIG-VALUE
<i>used-config-value</i>	ZEMS-TKN-USED-CONFIG-VALUE

538: ZEMS-EVT-BURST-START

CRITICAL: An event burst has been detected.

Unconditional Token (excluding header tokens)

ZSPI-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-BURST-EVT-NUMBER	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SSID	token-type	ZSPI-TYP-SSID.
ZEMS-TKN-SUBJ-CODE	token-type	ZSPI-TYP-TOKENCODE.
ZEMS-TKN-SUBJ-VALUE	token-type	ZSPI-TYP-STRING.
ZEMS-TKN-SUBJ-SSID	token-type	ZSPI-TYP-SSID.
ZEMS-TKN-BURST-TIME-START	token-type	ZSPI-TYP-TIMESTAMP.
ZEMS-MAP-BDS-INFO	token-type	ZEMS-DDL-BDS-INFO.

Event-Message Text

EMS: EVENT BURST DETECTED FOR EVENT NO. *eventno* OF SUBSYSTEM *ssid*, BY COLLECTOR *proc-desc*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector; 2 for alternate collector.

Header Token Value

ZSPI-TKN-EMPHASIS TRUE

Text Values

eventno ZEMS-TKN-BURST-EVT-NUM
ssid ZEMS-TKN-BURST-SSID
proc-name ZSPI-TKN-BURST-PROC-DESC

539: ZEMS-EVT-BURST-END

An event burst has ended.

Unconditional Token (excluding header tokens)

ZSPI-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-BURST-EVT-NUMBER	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-SSID	token-type	ZSPI-TYP-SSID.
ZEMS-TKN-SUBJ-CODE	token-type	ZSPI-TYP-TOKENCODE.
ZEMS-TKN-SUBJ-VALUE	token-type	ZSPI-TYP-STRING.
ZEMS-TKN-SUBJ-SSID	token-type	ZSPI-TYP-SSID.
ZEMS-TKN-BURST-TIME-START	token-type	ZSPI-TYP-TIMESTAMP.
ZEMS-TKN-BURST-END	token-type	ZSPI-TYP-TIMESTAMP.
ZEMS-TKN-EVTS-DELETED	token-type	ZSPI-TYP-INT2.
ZEMS-TOKEN-BURST-END-REASON	token-type	ZSPI-TYP-ENUM.

Event-Message Text

If ZEMS-TKN-BURST-END-REASON = ZEMS-VAL-BDS-ENABLED, then the event message text is:

```
EMS: BURST SUPPRESSION TERMINATED: count OCCURRENCES OF EVENT
NO. eventno. OF SUBSYSTEM ssid WERE NOT LOGGED BY COLLECTOR
proc-desc
```

If ZEMS-TKN-BURST-END-REASON = ZEMS-VAL-NO-EVENTS, then the event message text is:

```
EMS: BURST END DETECTED: count OCCURRENCES OF EVENT NO.
eventno. OF SUBSYSTEM ssid WERE NOT LOGGED BY COLLECTOR
proc-desc
```

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector; 2 for alternate collector.

Header Token Value

ZSPI-TKN-EMPHASIS FALSE

Text Values

count ZEMS-TKN-BURST-EVTS-DELETED
eventno ZEMS-TKN-BURST-EVT-NUM
ssid ZEMS-TKN-BURST-SSID
proc-name ZSPI-TKN-BURST-PROC-DESC

540: ZEMS-EVT-COL-PLF-ERROR

CRITICAL: A filter evaluation error occurred when a filter was applied to an event.

Unconditional Token (excluding header tokens)

ZSPI-TKN-COLLECTOR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-FILTER-ERROR	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-FILTERNAME	token-type	ZSPI-TYP-CHAR30.
ZEMS-TKN-EVTNUM	token-type	ZSPI-TYP-ENUM.
ZEMS-TKN-GENTIME	token-type	ZSPI-TYP-TIMESTAMP.

Event-Message Text

EMS: FILTER ERROR *filter-error* ON FILTER *filtername* ON EVENT *eventno*,TIMESTAMP *gentime*, BY COLLECTOR *proc-desc*

Subject

ZEMS-TKN-COLLECTOR Collector type. Value is 1 for primary collector; 2 for alternate collector.

Header Token Value

ZSPI-TKN-EMPHASIS TRUE

Text Values

filter-error ZEMS-TKN-FILTER-ERROR
filtername ZEMS-TKN-FILTERNAME
eventno ZEMS-TKN-BURST-EVT-NUM
gentime ZEMS-TKN-EVT-GENTIME
proc-name ZSPI-TKN-BURST-PROC-DESC

Collector Errors

Describes the error codes returned by the primary and alternate collector.

Common SPI Tokens

Describes the SPI tokens that occur in collector error lists.

ZSPI-TKN-	ZSPI-TYP-	Description
ENDLIST	SSCTL	Ends an error list.
ERRLIST	LIST	Begins an error list.
ERROR	ERROR	Standard SPI error token. DDL for ZSPI-TKN-ERROR: <pre>def ZSPI-DDL-ERROR. 02 Z-SSID type ZSPI-DDL- SSID. 02 Z-ERROR type ZSPI-DDL- ENUM. end</pre>
PARAM-ERR		Standard SPI error token. DDL for ZSPI-TKN-PARAM-ERR: <pre>def ZSPI-DDL-PARAM-ERR. 02 Z-TOKENCODE type ZSPI-DDL- TOKENCODE. 02 Z-INDEX type ZSPI-DDL- UINT. 02 Z-OFFSET type ZSPI-DDL- UINT. end</pre> <p>Token code and index are given (but not value) of a command parameter token that is in error.</p>
PROC-ERR	ENUM	Specifies a procedure associated with the error. Values that can occur depend on error message.
SSID-ERR	SSID	Subsystem ID used in error.

Common EMS Tokens

Describes the EMS tokens that occur in collector error lists.

ZEMS-TKN-	ZSPI-TYP-	Description
FAILFILENAME	FNAME	Nonshared. File name of the bad log file.
ZFILERR	UINT	Nonshared. Error code associated with a distributor ZSPI-TKN-EROR value.

Collector Error Codes

The following collector errors are ordered numerically by error number.

-3: ZCOM-ERR-CMD-INV-IN-SUMSTATE

The command cannot be applied to the object in its current summary state for one of the following reasons:

- Attempted to turn SUPPRESS ON when there was a burst filter installed.
- Attempted to change a SUPPRESS parameter when there was a burst filter installed.
- Attempted to change the S or L SUPPRESS parameters while BDS was enabled.
- Attempted to add a burst filter while BDS was enabled through the SUPPRESS keyword.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-OBJSTATE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-4: ZCOM-ERR-CMD-MISMATCH

A mismatch occurred between the command and object type. The command does not support the specified object type.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-5: ZCOM-ERR-CMD-NOT-SUPP

The request specifies a command that is not supported by the EMS collector.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-15: ZCOM-ERR-OBJ-ALRDY-DEF

The specified object (a filter) is already configured. This error occurs when the ZCOM-CMD-ADD is used to add a filter that is already configured.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-17: ZCOM-ERR-OBJ-NOT-FOUND

The specified object was not found.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-22: ZCOM-ERR-SECUR-VIOL

A sensitive command was issued by an unauthorized user. The command requires that the requester have a Super ID.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-23: ZCOM-ERR-SPI-ERROR

An SPI error occurred while processing this command. Contact your Tandem representative when this occurs.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ERROR.Z-SSID	token-type	ZSPI-DDL-ENUM
ZSPI-TKN-ERROR.Z-ERROR	token-type	ZSPI-DDL-ENUM
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-INT
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-24: ZCOM-ERR-SSID-INV

The subsystem ID (SSID) in the command message header is invalid.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-SSID-ERR	token-type	ZSPI-TYP-SSID
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-25: ZCOM-ERR-SUB-NOT-FOUND

The command contained the ZCOM-TKN-SUB token, but no subordinate objects of the specified type were not found.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-26: ZCOM-ERR-TKN-CODE-INV

The specified token code is invalid (that is., it should not be used or appear in the command).

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYPE-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-PARM-ERR	token-type	ZSPI-TYP-PARM-ERR
ZSPI-TKN-SSID-ERR	token-type	ZSPI-TYP-SSID
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-27: ZCOM-ERR-TKN-DUP

The command contains duplicate occurrences of a token that can appear only once.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-PARM-ERR	token-type	ZSPI-TYP-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-28: ZCOM-ERR-TKN-LEN-INV

The specified variable length token is too large.

Object name tokens are fully resolved. If the resolved name is larger than 50 bytes, then this error is reported. This is most likely an internal error and should be reported to your Tandem representative.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-PARM-ERR	token-type	ZSPI-TYP-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-29: ZCOM-ERR-TKN-REQ

The command is missing a required token.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-30: ZCOM-ERR-TKN-VAL-INVAL

The command contains a token with an illegal value.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-PARM-ERR	token-type	ZSPI-TYP-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-32: ZCOM-ERR-VSN-INCOMP

The command contains token maps with a higher version number than can be supported by this collector.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-SERVER-VERSION	token-type	ZSPI-TYP-VERSION
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

-33: ZCOM-ERR-EMPT-RSP

The EMS collector has no response records to return.

The only token in this response is ZSPI-TKN-RETCODE, which contains the error number. Unlike other ZCOM-ERR tokens, ZCOM-ERR-EMPT-RSP does not contain an error list. It does not indicate an error condition, and no error message should be displayed when this response is received.

The EMS collector returns this response when it receives a command with a context token indicating that more objects await processing, but no qualifying objects are found (for example, if there had been another object to process but that object has since been deleted). The EMS collector also returns this response when a command includes the value ZSPI-VAL-ERR-WARN, but no errors or warnings apply to any object referred to by the command.

-39: ZCOM-ERR-CMD-NOT-SUPP-BY-OBJ

A token, required within the context established by other tokens in the request, is missing from the command buffer. The context in question is defined by the tokens other than ZSPI-TKN-COMMAND, ZCOM-TKN-OBJNAME, and ZSPI-TKN-OBTYP. If the context is based on the command and object tokens, then the error ZCOM-ERR-TKN-REQ is issued instead of this error.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZSPI-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-PARM-ERR	token-type	ZSPI-TYP-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1001: ZEMS-ERR-VERSION

The collector does not support the version specified in the command message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1002: ZEMS-ERR-INV-CMD

The collector has received an invalid command message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1003: ZEMS-ERR-INV-SSID

The collector has received a command message with an invalid subsystem ID (from a message buffer not owned by EMS).

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-SSID-ERR	token-type	ZSPI-TYP-SSID
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1004: ZEMS-ERR-INV-TKN

The collector has received a command message with an unnecessary or unrecognized token.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
[ZSPI-TKN-SSID-ERR]	token-type	ZSPI-TYP-SSID
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

Error Notes

The bracketed token, SSID-ERR, appears in the error message only if the unrecognized token is associated with a subsystem ID other than EMS.

1005: ZEMS-ERR-INV-VALUE

The collector has received a command message with a bad token; the token value is invalid.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1006: ZEMS-ERR-DUP-TKN

The collector has received a command message with duplicate tokens.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1008: ZEMS-ERR-INV-OBJECT

A pre-D31 release collector has received a command message with an invalid object type. The object type must be zero for collector commands.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1009: ZEMS-ERR-INV-CPU

ZCOL-PRIMARYCPU contains an invalid CPU number. CPU number of the current backup CPU is required.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1010: ZEMS-ERR-CPU-RANGE

The collector has received a command message with an invalid CPU number. The CPU number must be in the range 0 to 15.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1013: ZEMS-ERR-CDIST-CPU

The collector has received a command message with an invalid CPU number.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1015: ZEMS-ERR-REQ-TKN

The collector has received a collector CONTROL command message missing a required token.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1017: ZEMS-ERR-INV-OCCURS

The collector has received a command message request from an application whose message buffer is not large enough to contain the entire response message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1019: ZEMS-ERR-FLT-FORM

The filter specified in the command is not in the correct format. If the specified filter was a burst filter or a filter table (EDIT file), it could not be converted into an object form.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZEMS-TKN-FAIL-REASON	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-FILTER-ERROR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1020: ZEMS-ERR-FLT-LOAD

This error can have the following causes:

- The filter specified in the command could not be loaded or read into memory.
- The specified filter contains destination profile data, which is not supported by the EMS collectors.

The parameter tokens in the command are in error.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERROR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1031: ZEMS-ERR-LOG-ACCESS

The command could not be completed because an I/O error occurred while the collector was accessing a log or ZZEVCNF file.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1034: ZEMS-ERR-OPEN-LOG

The collector has received a collector CONTROL command message that contains a ZEMS-TKN-LOGSUBVOL token, but cannot access the specified volume or subvolume.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1036: ZEMS-ERR-DEST-ACCESS

The collector cannot access a TEXTOUT destination.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

ZSPI-TKN-PROC-ERR can have the following value:

ZEMS-VAL-FINDDEV 39

1040: ZEMS-ERR-ZFIL

A file system error occurred while attempting to process a filter. Contact your Tandem representative.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-PROC-ERROR	token-type	ZSPI-TYP-????
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1041: ZEMS-ERR-ZSPI

A message-format error occurred when the collector tried to decode a command message.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-PROC-ERR	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

The token ZSPI-TKN-PROC-ERR, in the second (nested) ZEMS-ERR-ZSPI error list, can have the following values:

ZSPI-VAL-SSGET	2
ZSPI-VAL-SSGETTKN	3

1048: ZEMS-ERR-ACC-VIOL

The collector has received a collector CONTROL command message from a management application that has insufficient access privileges; this is a security violation.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1053: ZEMS-ERR-INV-MODE

The collector received a command message with an invalid mode for the compatibility distributor.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1054: ZEMS-ERR-CDIST-DOWN

The collector received a command message with instructions for the compatibility distributor. However, the distributor is down.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1055: ZEMS-ERR-NO-BACKUP

The content of the ZCOL-PRIMARYCPU was the alternate collector's backup CPU, but the alternate collector does not currently have a backup process.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-PARM-ERR	token-type	ZSPI-DDL-PARM-ERR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1056: ZEMS-ERR-ALLOC-LOG

An alternate collector log file could not be allocated.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZEMS-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZEMS-TKN-ZFILERR	token-type	ZSPI-TYP-UINT
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1057: ZEMS-ERR-LOGGING-STOPPED

The alternate collector is currently holding events in its event buffer pool and cannot write them to disk because logging has stopped for some reason.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1059: ZEMS-ERR-MAXFLT

A collector command attempted to add an eleventh filter to a collector. A maximum of 10 filters per collector are allowed.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1060: ZEMS-ERR-FLT-ALLOC

Resources for the filter specified in the command could not be allocated.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-FAILFILENAME	token-type	ZSPI-TYP-FNAME
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1067: ZEMS-ERR-ZOPR-SEND

A communication error occurred between \$0 and \$ZOPR. Retry the command. If this error persists, contact your Tandem representative.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

1068: ZEMS-ERR-ZOPR-SYNC

An internal error where \$0 and \$ZOPR are not synchronized with respect to filter configuration data has occurred. Contact your Tandem representative if this error occurs. Corrective action is to delete all filters until the error goes away.

ZSPI-TKN-ERRLIST	token-type	ZSPI-TYP-LIST
ZSPI-TKN-ERROR	token-type	ZSPI-TYP-ERROR
ZCOM-TKN-OBJNAME	token-type	ZSPI-TYP-STRING
ZCOM-TKN-OBJTYPE	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-ZOPR-CMD	token-type	ZSPI-TYP-ENUM
ZEMS-TKN-ZOPR-NUM-FILTERS	token-type	ZSPI-TYP-UINT
ZEMS-TKN-PCOLL-NUM-FILTERS	token-type	ZSPI-TYP-UINT
ZSPI-TKN-ENDLIST	token-type	ZSPI-TYP-SSCTL

