# *Table of Contents*

# *Table of Contents (Continued)*

## *List of Figures*



## *List of Tables*

O
P
E
R
A
T
I
O
N
S

# Chapter 1 Concept for GSE Messages

## 1.1 INTRODUCTION

The Group of Scientific Experts (GSE) Experimental International Seismic Monitoring System (EISMS) concept and GSE Technical Test 3 (GSETT-3) is predicated on the reliable communication among the sources, processors and users of the data and data products exchanged within the system. The infrastructure upon which the data exchange relies are the protocols and formats for data exchange. Protocols provide the mechanism for data exchange and formats provide the mechanism for organizing the data exchanged so that its handling can be automated. Because the participants in GSETT-3 are globally distributed, standard protocols and formats that are accessible by the international community have been adopted to provide reliable exchange of data and information.

The GSE message formats adopted for GSETT-3 data exchange are built upon the practical experience gained in the two previous GSE Technical Tests and the experience gained within the Federation of Digital Seismograph Networks (FDSN). The GSE Technical Tests demonstrated the capability of the international community to exchange meaningful information for the mutual benefit of all participating states in a proof of concept for future treaty monitoring activities. FDSN has a wealth of experience in the exchange of seismic information which has been tapped in defining these formats.

The email message structure is based on AutoDRM, an automated system that was developed to provide data, station and event information from local seismic networks in response to request messages.[1] These message formats and the request paradigm have been extended to accommodate the broader requirements of the EISMS and diverse data formats (e.g., GSE, CSS and SEED).

Chapter 1 describes the GSE message concept and provides basic protocol information and message standards. Chapter 2 is devoted to AutoDRM request messages which are used to request data from a station or data center. Data and information sent on a routine basis are requested using subscription messages which are described in Chapter 3. Data returned from request and subscription messages as well as unsolicited data are conveyed via data messages (Chapter 4). Chapter 5 describes the minimum AutoDRM configuration that is needed by a station or NDC participating in GSETT-3. Chapter 6 describes a Problem message system that will be used for reporting problems in the various components of the EISMS. Finally, Chapter 7 provides information on the protocol and formats for the exchange of continuous data.

1) Kradolfer, U., Automating the Exchange of Earthquake Information, *Eos Trans. GU*, **74**, 442, 1993.

## 1.2 PROTOCOLS

There are two protocols designated for the exchange of a GSE Message: electronic mail (email) and File Transfer Protocol (ftp), the primary one being email. While all messages can be exchanged in some form by either of these protocols, each has distinct advantages and disadvantages that make their efficient use dependent on the message length and content. For example, email is better suited to shorter alpha-numeric messages, while ftp is a more appropriate method for longer messages and those containing “binary” data.

## 1.3 MESSAGE CONVENTIONS

### Message Size

There is no limit to the size of a GSE message. Its size may, however, determine which protocol is most appropriate for transmitting the message. Although uncommon, email is sometimes truncated to 100 Kilobytes during transmission. Messages that are longer than 100 Kilobytes should be sent via ftp or should be broken into several smaller emails using the methods for continuing GSE messages as described in section 1.4 of this chapter. The maximum message size for using the ftp protocol is a function of the bandwidth between the two sites and the space available in the anonymous ftp directory. Requests for extremely large amounts of data may be rejected.

GSE messages are not synonymous with computer files or email messages. Several GSE messages may be included in a single email message or ftp file, or a single GSE message may span several emails or files.

### Line Length

ASCII message lines may be up to 1024 characters long excluding the special characters New Line (NL) and Carriage Return (CR). An ASCII message line may be terminated by an NL or by an NL followed by a CR. Although the maximum line length allowed in the GSE message paradigm for ASCII message lines is 1024, the default line length is 132 characters; the formats have been designed not to exceed this limit.

The format for each of the message lines defined in this document determines the “virtual” line length. A “virtual” line may be broken into several “physical” lines at the request of the user, however. To break a line, a backslash (“\”) is inserted at the break point and the virtual line is continued on the next physical line. This improves the readability of lines that may be very long by limiting the physical line length to a reasonable size (say 80 characters for terminals that cannot easily handle longer line lengths). The backslash may occur in any character position of the line, is counted as one of the physical lines characters, and does not hold the place of a blank or any other character. The character preceding the backslash abuts the character in character position 1 of the next line.

A backslash line continuation character is not required for ASCII waveform data, but the limit on physical line length is observed.

Line lengths may also be extended to any number up to 1024. This applies primarily to sampled data where more samples can be put on a single line if the space is available. Fixed format lines with a well defined line length are not extendable.

The user who is requesting data may set the line length with a LINE_LEN command as described in Chapter 2.

## Free Format Lines

Message lines that are not in a fixed format (i.e., free format lines) must be left justified (with the exception of comment lines) and case insensitive. Message lines must also have one or more blank spaces between fields. All of the basic message lines and lines in request and subscription messages are free format lines.

A free format message line consists of a keyword followed (for some) by an argument list. In describing free format message lines in this document the keyword will be followed by a descriptive list of arguments with optional arguments placed in square brackets "[]". Where the arguments may be repeated or continued, three dots "..." are put in their place. Below the descriptive keyword line each of the arguments will be explained. Throughout this document keywords and words reserved for arguments will be capitalized (even though they are case insensitive in ee c
describe free format linet in ths documenthisgievenbBelo:s

## Date-Time Conventions

The standard format for specifying the date and time in GSE messages is in two fields; one for the date, and one for the time with a separating blank or blanks. The date must always be present, but the time field may be dropped in which case the time is assumed to be 00:00:00.000 The date field is formatted as yyyy/mm/dd, where yyyy is the year, mm is the month number, and dd is the day of the month; e.g., 1994/02/28 is February 28, 1994. The time field format is hh:mm:ss.sss where hh is the hour, mm is the minute, and ss.sss is the decimal second (UTC). The range of times over a day is from 00:00:00.000 to 23:59:59.999 (24:00:00.000 is not a valid time).Leading zeros in any of the number fields may be dropped in free format lines, but they must be present in fixed format lines. In addition, some of the accuracy may be dropped from the time field of free format lines. If the seconds, or the minutes and seconds are dropped, then they are assumed to be 0 (e.g., 21:03 = 21:03:00.000 and 9 = 09:00:00.000).

*Example 1.3 - 1* Acceptable date-time formats for free format lines

```
1994/01/01 13:04:12.003
1994/12/23
1995/07/14   01:05
1995/09/10 2:15:33
```

## Station Naming Conventions

Station codes for GSE messages must have been registered with the ISC or NEIC. All station codes must be five or fewer characters in keeping with the rules governing the naming of stations. Note that array stations will have unique station codes for each element of the array as well as a unique array code that refers to the entire array. The code referencing the array should not be the same as the station code of any of the array elements. Throughout this document station codes are capitalized.

## Channel Naming Conventions

The format for channel designators follows that used by the Federation of Digital Seismic Networks. Three characters are used to designate a channel. The first specifies the general sampling rate and the response band of the instrument, as shown in Table 1. The second character specifies the instrument code, as shown in Table 2. The third character specifies the physical configuration of the members of a multiple axis instrument package or other parameters as specified for each instrument, as shown in Table 3.

Table 1. **Channel Band Codes**

| Band Code | Band Type | Sample rate (Hz) | Corner period (seconds) |
|---|---|---|---|
| E | Extremely Short Period | ≥ 80 | < 10 sec |
| S | Short Period | ≥ 10 to < 80 | < 10 sec |
| H | High Broadband | ≥ 80 | ≥ 10 sec |
| B | Broadband | ≥ 10 to < 80 | ≥ 10 sec |
| M | Mid Period | > 1 to < 10 | |
| L | Long Period | = 1 | |
| V | Very Long Period | = 0.1 | |
| U | Ultra Long Period | = 0.01 | |
| R | Extremely Long Period | = 0.001 | |
| W | Weather/Environmental | | |

Table 2. **Channel Instrument Codes**

| Instrument Code | Description |
|---|---|
| H | High Gain Seismometer |
| L | Low Gain Seismometer |
| G | Gravimeter/Accelerometer Seismometer |
| M | Mass Position Seismometer |

Table 3. **Channel Orientation Codes**

| Orientation Code | Description |
|---|---|
| Z, N, or E | Traditional (Vertical, North-South, East-West) |
| A, B, or C | Triaxial (along the edges of a cube turned up on a corner) |
| T or R | For Transverse and Radial rotations |
| 1, 2, or 3 | Orthogonal components but non traditional orientations |
| U, V, or W | Optional components |

### Auxiliary Naming Conventions

The auxiliary designator is used to distinguish between different instruments or data streams that have the same station and channel codes. This is a four letter designator that is used only when a conflict exists. When not needed, this field may be left blank.

### Distance Units Conventions

The units of length or distance in seismology have historically spanned nanometers to degrees. Distance units used for GSE messages are nanometers for ground displacement, degrees for source-receiver distances, and kilometers for all other distance measures (including, e.g., event depth, emplacement depth, and station elevation).

## *1.4* GSE MESSAGE STRUCTURE

The first three lines of a GSE message are BEGIN, MSG_TYPE, and MSG_ID. These lines provide a minimal amount of information that (1) identifies the message system version number, (2) specifies the type of message, and (3) assigns identification codes to the message. If the message refers to a previous message (e.g., a "data" message in response to a "request" message), then the fourth line is a REF_ID line. These lines are followed by information specific to the message type. The final line of a GSE message is the STOP line.

GSE messages may span emails or files. The mechanism for doing so is provided by CONTINUED and CONTINUATION lines.

### BEGIN

Except in the case of a HELP message, the BEGIN line is the first line of a GSE message. The BEGIN line also contains the version identifier. For GSETT-3, the identifier is "GSE2.0".

*Syntax:* **`BEGIN`** `version`

`version`......................version identification (GSE2.0)

*Example 1.4 - 1* Sample BEGIN line for GSETT-3 messages

```
BEGIN GSE2.0
```

### MSG_TYPE

The MSG_TYPE line is the second line of a GSE message. A message type is required to distinguish between the different types of messages. Only one MSG_TYPE is allowed per message; combining different message types in the same GSE message is prohibited.

*Syntax:* **MSG_TYPE** type

`type` ............................ identifies the message as REQUEST, SUBSCRIPTION, DATA, or PROBLEM.

*Example 1.4 - 2* Sample message type line for a request message

```
MSG_TYPE REQUEST
```

### MSG_ID

Message tracking is provided through the use of MSG_ID and REF_ID lines. The MSG_ID is a convenience for the sender in tracking and identifying messages. It is also used for identifying continuations of messages across emails or files. The sender is responsible for providing a MSG_ID that is unique to the sender.

MSG_ID is the third line in a GSE message. The information conveyed by the MSG_ID line includes a unique message identification string containing no blanks or backslash (“\”) characters and a message source code. The message source code should be a recognized International Station Code (ISC) for messages generated from a station. Data centers and NDC’s should adopt a unique code to use as the message source code, as shown in Table 4. Each individual GSETT-3 user should use his IDC login name as his message source code.

Table 4. **GSE Message Source Codes**

| Source Code | Description |
|---|---|
| GSE_IDC | International Data Center for GSETT-3 |
| GSE_WGE | Working Group on Evaluation |
| GSE_WGO | Working Group on Operations |
| GSE_WGP | Working Group on Planning |
| AUS_NDC | Australia NDC |
| BRA_NDC | Brazil |
| BGR_NDC | Bulgaria |
| CAN_NDC | Canada NDC |
| CAF_NDC | Central African Republic NDC |
| COL_NDC | Columbia NDC |
| COK_NDC | Cook Islands |
| CZK_NDC | Czech Republic NDC |
| ETH_NDC | Ethiopia NDC |
| FIN_NDC | Finland NDC |
| FRA_NDC | France NDC |
| DEU_NDC | Germany NDC |
| HUN_NDC | Hungary NDC |
| IND_NDC | India NDC |
| IRA_NDC | Iran NDC |
| ITA_NDC | Italy NDC |
| JPN_NDC | Japan NDC |
| NLD_NDC | Netherlands NDC |
| NZL_NDC | New Zealand NDC |
| NOR_NDC | Norway NDC |
| PAK_NDC | Pakistan NDC |
| PNG_NDC | Papua New Guinea |
| PRY_NDC | Paraguay |
| PER_NDC | Peru NDC |
| ROM_NDC | Romania NDC |
| RUS_NDC | Russian Federation NDC |
| RSA_NDC | South African Republic NDC |
| ESP_NDC | Spain NDC |
| SWE_NDC | Sweden NDC |
| CHE_NDC | Switzerland NDC |
| TUR_NDC | Turkey NDC |
| TMS_NDC* | Turkmenistan |
| GBR_NDC | United Kingdom NDC |
| USA_NDC | United States NDC |
| WSM_NDC | Western Samoa NDC |

* Turkmenistanian code pending approval

The format for the MSG_ID line is given below.

*Syntax:* **MSG_ID** identification source

identification .... Unique identification string (up to 20 characters)
source ....................... Message source

*Example 1.4 - 3* Message from the NDC in country ABC

```
MSG_ID  1994/05/21_0001 ABC_NDC
```

### REF_ID

A REF_ID is used in data messages that are in response to request or subscription messages and in problem messages. The REF_ID is the MSG_ID of the request message to which the response is being given. It follows the MSG_ID line as the fourth line of the message.

*Syntax:* **REF_ID** msg_id_string [msg_id_source]

msg_id_string....... id_string from reference MSG_ID line
msg_id_source....... message source code from reference MSG_ID line

*Example 1.4 - 4* A request message is sent from an NDC to the IDC:

```
BEGIN  GSE2.0
MSG_TYPE REQUEST
MSG_ID  1994/05/21_0001 ABC_NDC
      (request specific information)
STOP
```

*Example 1.4 - 5* The response to the request will have a MSG_ID from the IDC and use the request message MSG_ID as the REF_ID:

```
BEGIN  GSE2.0
MSG_TYPE  data
MSG_ID  00000023 GSE_IDC
REF_ID  1994/05/21_0001 ABC_NDC
      (data specific information)
STOP
```

## CONTINUED

Email provides a widely used method for sending messages of a limited size. Information can be lost, however, if the message is too long. In some cases, the solution to this problem is to send the message via the ftp. It is also possible to break the message into several smaller sections and send them as separate emails.

The mechanism for indicating that a GSE message will be continued in another email or file is a CONTINUED line. A CONTINUED line may appear at any point in a message as a separate line (although not in a section with binary data). All lines that come after the CONTINUED line will be ignored. For example, if a message is split into three smaller sections, the first section must end with CONTINUED, the second section must end with CONTINUED, and the third must end with STOP.

## CONTINUATION

A continued section must begin with a CONTINUATION line. The parameters of the CONTINUATION line are an integral sequence number and the message ID. Sequence numbers must begin with one (1) and increase monotonically. This identifies to which message the continuation belongs and where it fits into the sequence of message sections.

*Syntax:* **`CONTINUATION`** `sequence_number msg_id_string [msg_id_source]`

`sequence_number`...sequence number beginning with 1
`msg_id_string` .......identification string from MSG_ID line
`msg_id_source` .......source from MSG_ID line

*Example 1.4 - 6* A data message in response to a request that has been split into three return emails is given below:

```
BEGIN  GSE2.0                                   (email 1; data message)
MSG_TYPE  data
MSG_ID  00000023 GSE_IDC
REF_ID  1994/05/21_0001 ABC_NDC
      (data-specific information)
CONTINUED

CONTINUATION 1 00000023 GSE_IDC                 (email 2; continuation)
      (data-specific information)
CONTINUED

CONTINUATION 2 00000023 GSE_IDC                 (email 3; continuation)
      (data-specific information)
STOP
```

## STOP

The last line of any GSE message is a STOP line. In the case where two or more GSE messages (with different MSG_ID lines!) are included in one email or file, all lines between the STOP and BEGIN lines are ignored. A GSE message without a STOP line is considered incomplete and is ignored.

# Chapter 2 AutoDRM Request Messages

## 2.1 INTRODUCTION

The request message formats provide a framework in which almost all data can be requested from a station or data center. A request message consists of a single BEGIN / STOP sequence that has a MSG_TYPE REQUEST and that can be uniquely identified by the MSG_ID. Within a request message, several types of data may be requested. For example, requests may be made for a bulletin and associated waveforms, or for specific event information from several different areas; each as a separate request. The order of the requests in the request message is preserved in the response (data) message. The response to a request message must be contained in a single data message that includes the MSG_ID of the request message as the REF_ID.

## 2.2 THE HELP LINE

A HELP line specifies that the requestor would like to have a description of the AutoDRM interface. The HELP line is a special line in that no other message lines are required; the basic message lines BEGIN, MSG_TYPE, and MSG_ID need not be included. HELP may also appear as the message subject in an email that contains no body.

## 2.3 REQUEST FORMAT DESCRIPTION

In addition to the basic message information described in Chapter 1 a request message is a series of free format command lines that provide information about the return message (control lines), set the environment for subsequent request lines (environment lines), or specify the type of data that is to be returned within the limits of the environment (request lines). Some request lines must be preceded by environment lines that, by constraining the request, limit the size of the response.

Implementation of the AutoDRM formats will vary from site-to-site and will depend on the type of data and information that is available from the site. The minimum required configuration for a station or NDC AutoDRM participating in GSETT-3 is outlined in Chapter 5. The defaults in this document are for the IDC. Recommendations for station or NDC AutoDRM defaults are mentioned where they might differ from IDC defaults. The “HELP” mechanism should provide a list of defaults for an AutoDRM installation.

## 2.4 RESPONSE CONTROL LINES

The response control lines determine the protocol of the return data message and the physical line-length to be used. The existing options for specifying the protocol for returning messages in AutoDRM are E-MAIL and FTP. In each GSE message there can only be one method specified (i.e., either one E-MAIL line or one FTP line). If different protocols are desired for return data, separate request messages should be submitted to the AutoDRM.

A request message that contains no email or FTP line will be answered using the return email address of the sender. In some cases, the return address may not be reliable, however, so it is strongly suggested that one of the return mechanisms be specified. email will be used as the default method of transmitting data for small ASCII data messages (under 100 Kilobytes); ftp will be used for larger messages and messages with binary data.

### E-MAIL

The E-MAIL line is followed by the E-MAIL address to which the return message should be sent.

*Syntax:* `E-MAIL return_address`

`return_address`.....email address to send reply

*Default:* Return address from email header

### FTP

The FTP line specifies that the message should be put in a file or files on the AutoDRM computer for transmission using the file transfer protocol (ftp). The argument for the FTP line is the email address to which notification can be sent that the ftp file is ready for transfer.

*Syntax:* `FTP return_address`

`return_address`.....email address to send notification

The notification message in the return email is of the FTP_LOG data type (DATA_TYPE FTP_LOG, Chapter 4). The format of this message contains the name and location of the ftp file(s), allowing automated retrieval of the data.

### LINE_LEN

The line length of the returned data message is specified through the LINE_LEN line. Continuation of one physical line onto the next physical line is designated with the backslash continuation character (\). Waveform data following header information (e.g., the WID2 line) are a special case; the LINE_LEN is used as the maximum virtual, rather than physical, line length, so the continuation character convention does not apply.

*Syntax:* **LINE_LEN** `length`

`length` ...................... maximum length of a physical line in the returned data message

*Default:* 132

## 2.5 REQUEST ENVIRONMENT LINES

The environment in which the response to the request line will be made is specified using environment lines. Environment lines constrain the request based on latitude, longitude, time, depth, station, channel, etc. An environment is defined by its keyword and the arguments that follow. They are set when called for the first time and can be reset with another call to the same keyword. An environment keyword with no arguments resets the constraint on that environmental parameter to the default.

There are two classes of environments; ranges and lists.

A range environment specifies the inclusive limits between which the values satisfy the environment. Ranges are delimited with the word "TO" surrounded by blank spaces.

*Syntax:* **KEYWORD** `[[Low_Limit]  TO  [High Limit]]`

An open ended range is specified by not giving the low or high limit. The following example specifies all magnitudes of 5.0 and above.

*Example 2.5 - 1* All magnitudes of 5.0 and above:

```
MAG  5.0 TO
```

Blank spaces may appear in certain limits; specifically, time limits.

*Example 2.5 - 2* All times between February 23, 1994 at 00:00:00 and March 10, 1994 at 14:37:02:

```
TIME  1994/02/23  TO  1994/03/10 14:37:02
```

List environment lines contain lists of comma delimited parameters specifying discrete constraints such as station names and channels. Some list environments are allowed only one parameter (e.g., BULL_TYPE), but most may have an unlimited number of parameters. Spaces after the commas are neither required nor prohibited. The general format for a list environment is:

*Syntax:* **KEYWORD** `[arg1[, arg2[, arg3[, ...]]]]`

Lists can be quite long, so a wild card character (*) may be used as a substitute for any string of characters in some list environments.

*Example 2.5 - 3* Wild card use

`STA_LIST *` ............. would specify all of the stations
`STA_LIST A*` .......... would specify all stations beginning with "A"
`CHAN_LIST *Z` ........ would specify all channels ending with "Z" (vertical channels)

Descriptions of specific environments follow. For each one, default settings and examples are given. Although there are many environments listed, only certain ones of them may be applicable to a particular AutoDRM installation. Those that have been implemented should be described in the HELP message.

## TIME

The time environment is expressed as a range with both date and time entries. Unlike most range environments, a space is allowed between the date and time entries of the limits.

*Syntax:* **TIME** `[[date1 [time1]] TO [date2 [time2]]]`

`date1 time1`............Low range date and time
`date2 time2`............High range date and time

*Default:* current date and time TO current date and time.

Only the date and time fields necessary to obtain the resolution desired need be specified; all other fields are assumed to be 0 or 1 as appropriate (1 for month and day, 0 for hour, minute, and second).

*Example 2.5 - 4* Sample TIME environments

```
TIME 1994/02 to 1994/03
TIME 1994/02/01 23:14:19.7 TO 1994/03/01 12
TIME 1994/2/1 23:14:19.7 to 1994/3/1 12
```

## LAT

The LAT environment specifies the range of latitude in degrees. Southern latitudes are negative. The low range value must be smaller than the high range value.

*Syntax:* **LAT** `[[low_lat] TO [high_lat]]`

`low_lat`......................Low range latitude
`high_lat`...................High range latitude

*Default:* no constraint

*Example 2.5 - 5* Latitudes constrained between 12.0 degrees South to 17 degrees North

```
LAT -12 TO 17
```

## LONG

The LONG environment specifies the range of longitude in degrees. Western longitudes are negative and the range is interpreted from west to east. It is specific to the LONG environment that either both or neither (to return to the default values) of the longitudes must be given.

*Syntax:* **LONG** `[western_long TO eastern_long]`

`western_long` ......... Western longitude
`eastern_long` ......... Eastern longitude

*Default:* No constraint

*Example 2.5 - 6* A longitude range of 350 degrees

```
LONG -175 TO 175
```

Longitude ranges may span the International Date Line, as shown in the following example.

*Example 2.5 - 7* A longitude range of 10 degrees

```
LONG 175 TO -175
```

## EVENT_STA_DIST

Event - station distance (in degrees) is applied in context to the request. When requesting waveform data associated with specific events, EVENT_STA_DIST helps determine the stations from which the data will be retrieved. When requesting bulletin-type information (bulletins, events, origins, or arrivals), then EVENT_STA_DIST helps determine the events for which the data will be retrieved.

*Syntax:* **EVENT_STA_DIST** `[[low_dist] TO [high_dist]]`

`low_dist` .................. Low distance range
`high_dist` ................ High distance range

*Default:* No constraint

*Example 2.5 - 8* A request for bulletin information from events within 20 degrees of stations ABC or DEF must include these lines:

```
STA_LIST ABC, DEF
EVENT_STA_DIST 0 TO 20
BULLETIN GSE2.0
```

*Example 2.5 - 9* A request for all waveform data from stations within 20 degrees of an event of 1995/01/01 00:12:17 must include these lines:

```
TIME 1995/1/1 00:12:16.9 to 1995/1/1 00:12:17.01
EVENT_STA_DIST 0 to 20
BULLETIN INTERNAL
ASSOCIATE BULLETIN
WAVEFORM GSE2.0
```

## DEPTH

Depth ranges are given in kilometers of depth from the surface. All depths are positive.

*Syntax:* **DEPTH** `[[shallow] TO [deep]]`

`shallow`......................low depth range
`deep`.............................high depth range

*Default:* No constraint

*Example 2.5 - 10* Depth environment

```
DEPTH 0.0 TO 10.0
```

### DEPTH_MINUS_ERROR

To obtain all events that have a 90% probability of being within a certain depth range, the DEPTH_MINUS_ERROR environment is provided. Depth minus error ranges are given in kilometers of depth from the surface.

*Syntax:* **DEPTH_MINUS_ERROR** `[[shallow] TO [deep]]`

`shallow`......................Low depth range
`deep`.............................High depth range

*Default:* No constraint

*Example 2.5 - 11* Anything likely to be within 10 km of the surface

```
DEPTH_MINUS_ERROR 0.0 TO 10.0
```

### MAG

Magnitude ranges specify the range of magnitudes to include in the search. If no magnitude range is specified, all events regardless of magnitude will be selected. The type of magnitude (mb, Ms, etc.) is specified in the MAG_TYPE environment.

*Syntax:* **MAG** `[[low_mag] TO [high_mag]]`

`low_mag`......................Low magnitude range
`high_mag`...................High magnitude range

*Default:* No constraint

*Example 2.5 - 12* Magnitudes above 4.5

```
MAG 4.5 TO
```

### MAG_TYPE

The magnitude type to search with the magnitude environment is given in this list. Valid entries will be determined by the data center, but standard magnitude codes are mb (body wave magnitude, Ms (surface wave magnitude), ML (local magnitude), Mn (Nuttli Lg magnitude), MD (duration), Mw (moment magnitude), and M (unspecified).

*Syntax:* **`MAG_TYPE`** `[mag_type[,...]]`

`mag_type`.................. any of: mb, Ms, ML, Mn, Mw, MD, or M

*Default:* No constraint

*Example 2.5 - 13* mb and Ms magnitudes only

```
MAG_TYPE mb, Ms
```

### MB_MINUS_MS

This difference in magnitude values specifies the range of magnitude differences to include in the search.

*Syntax:* **`MB_MINUS_MS`** `[[low_mag_diff] TO [high_mag_diff]]`

`low_mag_diff`......... Low magnitude difference
`high_mag_diff`....... High magnitude difference

*Default:* No constraint

*Example 2.5 - 14* A difference of magnitudes from 1 to 2

```
MB_MINUS_MS 1.0 TO 2.0
```

### STA_LIST

The station search list is given in the STA_LIST environment. If an array station is specified, then all elements of the array are implied. Specific array elements may be referenced individually. The default for the station list is dependent on the AutoDRM installation. For data centers it is none, preventing inadvertently large requests. For single station AutoDRM's, it is all. The wildcard character (*) is allowed for specifying station codes.

*Syntax:* **`STA_LIST`** `[sta[, sta[, ...]]]`

`sta`.............................. Station Code(s)

*Default:* Installation dependent (Empty list for the IDC)

*Example 2.5 - 15* Four specific stations

```
STA_LIST ARA0, ARA1, ARA2, ARB2
```

*Example 2.5 - 16* All stations beginning with A

```
STA_LIST A*
```

### CHAN_LIST

The channel search list defaults to all vertical channels (*z). The wild card character is allowed.

*Syntax:* **CHAN_LIST** [chan[, chan[, ...]]]

chan .............................Channel Code

*Default:* *z

*Example 2.5 - 17* Three short period channels

```
CHAN_LIST shz, shn, she
```

*Example 2.5 - 18* All short period channels

```
CHAN_LIST s*
```

### AUX_LIST

Station and channel are not always adequate to completely describe a specific data stream for some seismic stations. An auxiliary identification is supplied for completeness in handling these special cases. The instances in which the auxiliary identifications are necessary should be rare. The wild card character is allowed in the list of identification codes.

*Syntax:* **AUX_LIST** [aux[, aux[, ...]]]

aux ...............................Auxiliary identification code

*Default:* *

*Example 2.5 - 19* AUX_LIST chi, med

### BULL_TYPE

Data centers may produce more than one bulletin. The BULL_TYPE environment provides a means to specify which bulletin to retrieve. Only one bulletin may be specified in any BULL_TYPE line. Bulletin naming conventions are not standard, so the valid lists will be data center dependent. The default bulletin will be the best bulletin available that covers the entire period requested.

*Syntax:* **BULL_TYPE** [bulletin_name]

bulletin_name .......any of: IDC_REB, IDC_AEL, IDC_ABEL, or other

*Default:* Installation dependent

The IDC will store the three bulletins regularly produced; the IDC_AEL, the IDC_ABEL, and the IDC_REB. All other bulletins stored at the IDC are considered gamma bulletins and are identified by the three letter country code followed by an underscore and the name of the bulletin.

*Example 2.5 - 20* IDC Reviewed Event Bulletin

```
BULL_TYPE IDC_REB
```

*Example 2.5 - 21* Country ABC's DEF Bulletin (Gamma)

```
BULL_TYPE ABC_DEF
```

## GROUP_BULL_LIST

Events are often common between bulletins; it is sometimes desirable to list the various solutions (origins) together. GROUP_BULL_LIST is a list of the bulletins that should be combined with the bulletin specified in BULL_TYPE. Only the origin information from these other bulletins will be included in the combined bulletin that is eventually returned; the arrival information will be for the BULL_TYPE bulletin. Bulletin naming conventions are not standard, so the valid lists will be data center dependent.

*Syntax:* **GROUP_BULL_LIST** [bulletin[, ...]]

bulletin.................. any of: IDC_AEL, IDC_ABEL, IDC_REB, GAMMA, or Other

*Default:* None

Events in the GROUP_BULL_LIST will be grouped with at most one event in the BULL_TYPE bulletin. Grouping of events at the IDC will be done by including all events with locations within three degrees and origin times within sixty seconds. If the initial criteria are met for more than one event, then the differences in distance and origin time will be weighted and a choice made on that basis.

The IDC will store the three bulletins regularly produced; the IDC_AEL, the IDC_ABEL, and the IDC_REB. Gamma bulletins from NDC's will be designated by the three letter country code followed by an underscore and the Bulletin name. At the IDC a shorthand for all gamma bulletin information will be "GAMMA".

*Example 2.5 - 22* IDC_REB with entries from the IDC_ABEL and all GAMMA origins

```
BULL_TYPE IDC_REB
GROUP_BULL_LIST IDC_ABEL, GAMMA
```

## ARRIVAL_LIST

A unique arrival identification number is assigned to each arrival stored at the IDC. This identification number appears in the bulletin and may be used subsequently to request waveforms or comments associated with the specific arrival.

*Syntax:* **ARRIVAL_LIST** [arid[, arid[, ...]]]

arid............................Arrival identification number or string

*Default:* All arrivals

*Example 2.5 - 23*

```
ARRIVAL_LIST 8971234, 90814
```

### ORIGIN_LIST

A unique origin identification number may be assigned to each origin stored at a data center. This identification number appears in the bulletin and may be used subsequently to request waveforms or comments associated with the specific origin.

*Syntax:* **ORIGIN_LIST** [orid[, orid[, ...]]]

orid............................Origin identification number

*Default:* All origins

*Example 2.5 - 24*

```
ORIGIN_LIST 132456, 190672
```

### EVENT_LIST

A unique event identification number is assigned to each event whose origin(s) are computed at the IDC. This identification number appears in the bulletin and may be used subsequently to request waveforms or comments associated with the specific event.

*Syntax:* **EVENT_LIST** [evid[, evid[, ...]]]

evid............................Event identification number

*Default:* All events

*Example 2.5 - 25*

```
EVENT_LIST 87623495, 87
```

### COMM_LIST

The communications list is the list of communications links to include in status reports. Links are defined by the end of the link furthest from the IDC. Thus, for the link between the USA_NDC and the GSE_IDC the communications link would be designated as USA_NDC. Station codes are used for links from the station to the NDC or from the station to the IDC, etc.

*Syntax:* **COMM_LIST** [comm[, comm[, ...]]]

`comm` ............................ Communications Link Code

*Default:* *NDC

*Example 2.5 - 26*

```
COMM_LIST  ABC,IND_NDC
```

### PROB_TYPE

Problem types provide an environment for requesting problem reports from the IDC. The problem type can be any of a number of categories including hardware, software, communications, facilities, and unknown.

*Syntax:* **PROB_TYPE** [type[, type[, ...]]]

`type` ............................ any of: HARDWARE, SOFTWARE, COMM, FACILITIES, or UNKNOWN

*Default:* Empty list

*Example 2.5 - 27*

```
PROB_TYPE COMM
```

### PROB_LOC

PROB_LOC provides an environment for specifying the location from which to provide problem reports. Problems are generally associated with a station, a communications link, or a data center (IDC or NDC).

*Syntax:* **PROB_LOC** [location[, location[, ...]]]

`location`.................. station, NDC, or IDC code

*Default:* Empty list

*Example 2.5 - 28*

```
PROB_LOC GSE_IDC, ABC_NDC
```

### PROB_STATE

Problem status (PROB_STATE) provides an environment for requesting problem reports from the IDC. A problem may be either open or closed. The PROB_STATE environment enables the user to specify which problems to list in the report.

*Syntax:* **PROB_STATE** status[, status]

`status` ....................... either OPEN or CLOSED

*Default:* OPEN

*Example 2.5 - 29* Constraint for open problems

```
PROB_STATE OPEN
```

*Example 2.5 - 30* All problems, open or closed

```
PROB_STATE OPEN, CLOSED
```

## 2.6 REQUEST LINES

Request lines specify which information to retrieve from the AutoDRM installation. All of the arguments to a request line are optional and include the format for the return message which is specified as a generic term such as GSE2.0, CSS3.0, SEED2.3, or INTERNAL; an optional sub-format specific to the data type being requested; and an optional EMBEDDED designator.

*Syntax:* **REQUEST_KEYWORD** [format [sub-format]] [EMBEDDED]

`format` ........................ either INTERNAL or one of a few generic data types (e.g., GSE2.0);
`sub-format` .............. specifies which internal format to use with this data type. The sub-format is used primarily for WAVEFORM requests;
`EMBEDDED` .................. indicates that the data should be embedded within the return message in a format that allows it to be stripped from the message and placed in independent files.

If no format is specified, then the default (GSE2.0) format will be used. If INTERNAL is specified as the format, then the information is retrieved, but not returned in a response. The INTERNAL format is used in conjunction with an ASSOCIATE environment as described in Section Chapter 2.7. EMBEDDED directs the AutoDRM to place the output in a file embedded in the return message. The file can be stripped from the message using a utility that is available from any AutoDRM installation that is capable of creating EMBEDDED files.

For each request that is made, a subset of the total environment is applied, as shown in Table 5. All applicable environments are enforced for each request. If the environment is not specified explicitly, then the default is used. Because the default values for some environments are an empty list (e.g., STA_LIST) or a zero length range (e.g., TIME), a request made without explicitly defining these environments will result in no data returned. For example, the TIME environment must be satisfied for bulletin requests. Descriptions of the request lines (below) include the applicable environments. The environments that must be explicitly specified to obtain a result from the IDC are in bold type like this: **TIME**.

The order of the request lines is very important, as the environment established prior to the request line is what is used to constrain the request. The environment can be changed between request lines allowing multiple requests for the same type of information, but with different constraints.

*Example 2.6 - 1* To obtain the bulletin information for all events in January, 1994 within the areas defined by 10 to 20 degrees North, 120 to 160 degrees East and 45 to 55 degrees South, 15 to 25 degrees West.

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1994/01/01 TO 1994/01/31
LAT  10.0 TO 20.0
LONG 120.0 TO 160.0
BULLETIN  GSE2.0
LAT  -45.0 TO -55.0
LONG -15.0 TO -25.0
BULLETIN  GSE2.0
STOP
```

Requests attempt to provide information of general interest to seismologists. The requests listed below are only a set of suggestions for standard requests that may be offered by an AutoDRM. A data center such as the IDC will provide most, if not all, of the information listed below. A data source such as an NDC or station might provide a very limited number of the data types through requests. The minimum set of environment and request lines that must be implemented by providers of beta data from either stations or NDC's are given in Chapter 5. Table 5 gives the applicable requirements for requests.

Table 5. **Applicable Environments for Requests**

| envs / requests | time | lat | long | event_sta_dist | depth | depth_minus_error | mag | mag_type | ms_minus_mb | sta_list | chan_list | aux_list | bull_type | group_bull_list | arrival_list | origin_list | event_list | comm_list | prob_type | prob_loc | prob_state |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| waveform | X | | | | | | | | | X | / | / | | | | | | | | | |
| detection | X | | | | | | | | | X | | / | | | | | | | | | |
| arrival | X | | | | | | | | | X | | | / | | / | | | | | | |
| origin | X | / | / | / | / | / | / | / | / | | | | / | | | / | | | | | |
| event | X | / | / | / | / | / | / | / | / | | | | / | / | | | / | | | | |
| bulletin | X | / | / | / | / | / | / | / | / | | | | / | / | | | / | | | | |
| station | | / | / | | | | | | | X | | | | | | | | | | | |
| channel | | / | / | | | | | | | X | / | / | | | | | | | | | |
| response | / | | | | | | | | | X | / | / | | | | | | | | | |
| outage | X | | | | | | | | | X | / | / | | | | | | | | | |
| comment | / | | | | | | | | | X | | | | | X | X | X | | | | |
| sta_status | X | | | | | | | | | X | | / | | | | | | | | | |
| chan_status | X | | | | | | | | | X | / | / | | | | | | | | | |
| comm_status | X | | | | | | | | | | | | | | | | | / | | | |
| problem | X | | | | | | | | | | | | | | | | | | X | X | / |

X Required environment

/ Supplemental environment

## WAVEFORM

Waveforms are digital time series data. Waveform requests will typically accept sub-formats that specify how the digital data are formatted within the general format of the waveform data type. The sub-formats include INT, CM6, CM8, AUT, AU6, and AU8, for GSE2.0 data.

*Environment:* **TIME, STA_LIST,** CHAN_LIST, AUX_LIST

O P E R A T I O N S

*Example 2.6 - 2* Data in 6-bit compressed format from all channels of station ABC for the time between 03:25 and 03:40 on March 1 is requested with the following message:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/03/01 03:25 TO 1994/03/01 03:40
STA_LIST ABC
CHAN_LIST *
WAVEFORM GSE2.0 CM6
STOP
```

## DETECTION

Detections are the result of some detection process run on waveform data. The information in a detection includes time, amplitude, period, and other parameters derived from polarization or array processing.

*Environment:* **TIME**, **STA_LIST**, AUX_LIST

*Example 2.6 - 3* The detections from station ABC on October 10, 1994 may be obtained with the following message:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/10/10 TO 1994/10/11
STA_LIST ABC
DETECTION GSE2.0
STOP
```

## ARRIVAL

An arrival is a detection that has been associated with an origin. Although the arrival time, azimuth and slowness may be determined for an arrival, it is not required that this information be used in determining the location or timing of the origin. An arrival may be assigned a phase name if appropriate.

*Environment:* **TIME**, **STA_LIST**, ARRIVAL_LIST, BULL_TYPE

*Example 2.6 - 4* To obtain the arrivals from stations ABC and DEF for March of 1994:

```
BEGIN GSE2.0
MSG_TYPE Request
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/03/01 TO 1994/04/01
STA_LIST ABC, DEF
ARRIVAL GSE2.0
STOP
```

## ORIGIN

Origins are solutions to the location and time of the source. Several origins may be determined by different organizations (e.g., the GSE IDC, NEIC, and ISC) for any one source (event).

*Environment:* **TIME**, LAT, LONG, DEPTH, MAG, MAG_TYPE, BULL_TYPE, ORIGIN_LIST, EVENT_STA_DIST, MB_MINUS_MS, DEPTH_MINUS_ERROR

*Example 2.6 - 5* Obtain the origin information for the IDC_REB origins for one day in August.

```
BEGIN GSE2.0
MSG_TYPE Request
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/08/08 TO 1994/08/09
BULL_TYPE  IDC_REB
ORIGIN GSE2.0
STOP
```

*Example 2.6 - 6* Limiting Example 2.5.4-1 to a specific geographic region, magnitude, and depth range is done by including more environment lines.

```
BEGIN GSE2.0
MSG_TYPE Request
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/08/08 TO 1994/08/09
LAT -60 TO 10.0
LONG -81 TO -34
MAG 4.5 TO 5.5
DEPTH 0 TO 10
BULL_TYPE  IDC_REB
ORIGIN GSE2.0
STOP
```

## EVENT

An event is representative of the physical occurance that was detected through the network of sensors. There can be many estimates of the time and location of an event, and these estimates are known as origins. Events are the collection of origin estimates. Only those estimates given in the BULL_TYPE and GROUP_BULL_LIST environments are provided. The origin estimates in BULL_TYPE provide the base for associating the origins in the GROUP_BULL_LIST.

*Environment:* **TIME**, LAT, LONG, DEPTH, MAG, MAG_TYPE, BULL_TYPE, GROUP_BULL_LIST, EVENT_LIST, EVENT_STA_DIST, MB_MINUS_MS, DEPTH_MINUS_ERROR

*Example 2.6 - 7* All of the IDC_REB origins within regional distance (20 degrees) of stations ABC and DEF and the associated IDC_ABEL origins are obtained for March of 1994 with the following querry:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/03/01 TO 1994/04/01
BULL_TYPE  IDC_REB
GROUP_BULL_LIST IDC_ABEL
STA_LIST ABC, DEF
EVENT_STA_DIST 0.0 TO 20.0
EVENT GSE2.0
STOP
```

## BULLETIN

Bulletins are composed of arrival, origin and event lines. Only the arrival information associated with the event is given in the bulletin. BULLETIN may be used as an argument on ASSOCIATE lines to constrain waveforms.

*Environment:* **TIME**, LAT, LONG, DEPTH, MAG, MAG_TYPE, BULL_TYPE, EVENT_LIST, GROUP_BULL_LIST, EVENT_STA_DIST, MB_MINUS_MS, DEPTH_MINUS_ERROR

*Example 2.6 - 8* Request for the IDC_REB for May 25, 1994.

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/05/25 TO 1994/05/26
BULL_TYPE  IDC_REB
BULLETIN GSE2.0
STOP
```

*Example 2.6 - 9* Include the IDC_ABEL origins in Example 2.5.6-1:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/05/25 TO 1994/05/26
BULL_TYPE  IDC_REB
GROUP_BULL_TYPE IDC_ABEL
BULLETIN GSE2.0
STOP
```

*Example 2.6 - 10* List only origins whose DEPPTH_MINUS_ERROR is less than 10 kilometers in Example 2.5.6-2:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/05/25 TO 1994/05/26
DEPTH_MINUS_ERROR  TO 10
BULL_TYPE  IDC_REB
GROUP_BULL_TYPE IDC_ABEL
BULLETIN GSE2.0
STOP
```

*Example 2.6 - 11* List only origins whose DEPTH_MINUS_ERROR is less than 10 kilometers and whose MB_MINUS_MS is greater than 0.5:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/05/25 TO 1994/05/26
DEPTH_MINUS_ERROR  TO 10
MB_MINUS_MS 0.5 TO
BULL_TYPE  IDC_REB
GROUP_BULL_TYPE IDC_ABEL
BULLETIN GSE2.0
STOP
```

## STATION

Station information includes station codes, locations, elevations, station type (e.g., array, 3-C), and dates of operation for which data are available through the AutoDRM..

*Environment:* LAT, LONG, **STA_LIST**

*Example 2.6 - 12* Obtain station information for all stations serviced by this AutoDRM:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
STA_LIST *
STATION GSE2.0
STOP
```

*Example 2.6 - 13* For stations in the Southern hemisphere:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
LAT -90 TO 0.0
STA_LIST *
STATION GSE2.0
STOP
```

## CHANNEL

Channel is a complete set of information about the location, emplacement, and type of seismometers at a station.

*Environment:* LAT, LONG, **STA_LIST**, CHAN_LIST, AUX_LIST

*Example 2.6 - 14* To obtain the short period channel information for stations in South America, the LAT and LONG environments are set appropriately:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
LAT -60 TO 10.0
LONG -81 TO -34
STA_LIST *
CHAN_LIST s*
CHANNEL GSE2.0
STOP
```

## RESPONSE

The response is the instrument response of the specified station / channel / auxiliary identification code. Responses are valid at any given time and may change in a time interval. Thus only the first date/time field will be used as the time environment for RESPONSE.

*Environment:* TIME, **STA_LIST**, CHAN_LIST, AUX_LIST

*Example 2.6 - 15* The instrument response information for the broadband vertical channel of station ABC may be obtained with the following request:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1994/01/01 TO 1994/01/02
STA_LIST ABC
CHAN_LIST bhz
RESPONSE GSE2.0
STOP
```

## OUTAGE

OUTAGE reports the data that were not available for the specified time range.

*Environment:* **TIME**, **STA_LIST**, CHAN_LIST, AUX_LIST

*Example 2.6 - 16* To obtain the outage reports from the IDC for all stations and channels for the month of March in 1994, the station and channels must be specified; otherwise the default station list (empty list) and channel list (*z) will be in effect.

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/03/01 TO 1994/04/01
STA_LIST *
CHAN_LIST *
OUTAGE GSE2.0
STOP
```

## COMMENT

Comments may be associated with a station, an event, an origin, or an arrival. To retrieve comments, the station code or the ids of the arrival, origin, or event can be used. These are listed in the bulletins and are obtained with a request (or subscription to) a bulletin or event list. Alternatively, ASSOCIATE may be used to obtain the comments associated with an arrival, origin, or event.

*Environment:* TIME, **STA_LIST**, **ARRIVAL_LIST**, **ORIGIN_LIST**, **EVENT_LIST** (one must be specified)

*Example 2.6 - 17* To obtain the comments for event 510:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
EVENT_LIST 510
COMMENT GSE2.0
STOP
```

## STA_STATUS

Station status is given for the stations in the STA_LIST environment. The TIME environment defines the report period. The minimum report period is one day.

*Environment:* **TIME**, **STA_LIST**, AUX_LIST

*Example 2.6 - 18* To obtain the station status reports for all GSE stations over a one week period:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/11/14 TO 1994/11/21
STA_LIST *
STA_STATUS GSE2.0
STOP
```

## CHAN_STATUS

Channel status is given for the channels in the CHAN_LIST and AUX_LIST environments for the stations in the STA_LIST environment. The TIME environment defines the report period. The minimum report period is one day.

*Environment:* **TIME**, **STA_LIST**, CHAN_LIST, AUX_LIST

*Example 2.6 - 19* To obtain the channel status reports for the short period channels over a four day period at station ARA0:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/11/14 TO 1994/11/17
STA_LIST ARA0
CHAN_LIST s*
CHAN_STATUS GSE2.0
STOP
```

## COMM_STATUS

Communications status is given for the communications links listed in the COMM_LIST environment. The TIME environment defines the report period. The minimum report period is one day. The sub_format field is used to indicate a verbose communications status report.

*Syntax:* **COMM_STATUS** [format [sub_format]]

format........................GSE2.0
sub_format...............VERBOSE

*Environment:* **TIME**, COMM_LIST

*Example 2.6 - 20* To obtain the verbose communications status reports for the links from station ABC to ANY_NDC and from ANY_NDC to the IDC over a one week period:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/11/14 TO 1994/11/21
COMM_LIST ABC, ANY_NDC
COMM_STATUS GSE2.0 VERBOSE
STOP
```

## PROBLEM

Problem reports are produced for the categories listed in the PROB_TYPE environment. The location of the problem is constrained in the PROB_LOC environment, and the status of the problem (open, closed, or both) is specified using the PROB_STATE environment. The TIME environment may also be used to constrain the problems presented. The reports that are generated are sent from the IDC as problem messages (MSG_TYPE PROBLEM).

*Environment:* **TIME**, PROB_TYPE, PROB_LOC, PROB_STATE

*Example 2.6 - 21* To obtain all problem reports that are open for stations ABC and DEF over a one week period:

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/11/14 TO 1994/11/21
PROB_STATE OPEN
PROB_LOC ABC, DEF
PROBLEM GSE2.0
STOP
```

## *2.7* ASSOCIATION

The concept of association is introduced to provide the ability to tie, or associate, one form of data with another. The most common association is the one between waveforms and events allowing a user to request waveforms associated with a particular set of events.

An association is made using the keyword ASSOCIATE. ASSOCIATE has all of the characteristics of a list environment, except that it is active only for the subsequent request line and the arguments are requests (e.g., BULLETIN).

*Syntax:* `ASSOCIATE request_keyword`

`request_keyword`.. request keyword

Usually, a request line is used just prior to the ASSOCIATE line which, in turn, constrains the subsequent request line. Often, the first request line produces unwanted information (such as BULLETIN). The INTERNAL argument to the first request line suppresses the distribution of the request so that only the associated request is delivered.

*Example 2.7 - 1* Request for a REB and associated waveforms

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1995/01/06 19:00:00 to 1995/01/06 20:00:00
BULL_TYPE IDC_REB
BULLETIN GSE2.0
ASSOCIATE BULLETIN
WAVEFORM GSE2.0 CM6
STOP
```

*Example 2.7 - 2* Request for associated waveforms only of Example 2.6.0-1

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1995/01/06 19:00:00 to 1995/01/06 20:00:00
BULL_TYPE IDC_REB
BULLETIN INTERNAL
ASSOCIATE BULLETIN
WAVEFORM GSE2.0 CM6
STOP
```

The common association of waveforms to events requires that the time windows of the associated waveforms be designated. The TIME environment is station dependent for associated waveform data. The standards for time windows depend on the sample rate of the station. For stations with sample rates greater than one sample per second the window extends from 60 seconds before the initial P wave to 120 seconds after the S or after the last phase associated with the event. For stations with sample rates equal to or smaller than one sample per second, the window extends from 60 seconds before the P to 120 seconds after the PKiKP or after the last phase associated with the event. If the waveforms are associated with arrivals or detections, then the window will include the 60 seconds before and after the arrival or detection.

It is very easy to make very large requests using the association capabilities of an AutoDRM and caution should be used when making association requests. Changes are expected in the function of association as experience is gained with its use.

*Example 2.7 - 3* Bulletin and Associate Waveforms

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1994/01/01 TO 1994/01/02
LAT  10.0 TO 20.0
LONG 120.0 TO 160.0
BULLETIN  GSE2.0
ASSOCIATE BULLETIN
WAVEFORM GSE2.0
STOP
```

In example 2.6.0-3 the origin times of the events in the bulletin are constrained by the time and the epicenters are constrained by the latitude and longitude. The waveforms, however, are constrained by their association to the events. The waveforms may be recorded by stations outside of the latitude and longitude ranges and extend beyond the time constraint.

*Example 2.7 - 4* Regional bulletin and associated waveforms

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1994/01/01 TO 1994/01/02
LAT  10.0 TO 20.0
LONG 120.0 TO 160.0
EVENT_STA_DIST 0 TO 20
BULLETIN  GSE2.0
ASSOCIATE BULLETIN
WAVEFORM GSE2.0
STOP
```

For Example 2.6.0-4 the EVENT_STA_DIST environment used by BULLETIN is applied to the WAVEFORM request to obtain only those data from stations within twenty degrees of the events.

*Example 2.7 - 5* REB events and comments associated with them.

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME 1994/01/01 TO 1994/01/02
BULL_TYPE IDC_REB
EVENT  GSE2.0
ASSOCIATE EVENT
COMMENT GSE2.0
STOP
```

*Example 2.7 - 6* Arrivals associated with a specific event.

```
BEGIN GSE2.0
MSG_TYPE REQUEST
MSG_ID example ANY_NDC
E-MAIL name@my.computer
TIME  1994/03/21 19:34:15 TO 1994/03/21 19:35:00
LAT 35.35 TO 35.45
LONG -124.23 TO -124.03
EVENT INTERNAL
ASSOCIATE EVENT
ARRIVAL GSE2.0
STOP
```

## *2.8* AUTODRM IMPLEMENTATION SAFEGUARDS

Responding to requests in an automatic system requires safeguards against repeat requests, excessive numbers of requests, excessively large requests, and failures of the email system (returned mail). Although each installation of the AutoDRM will be different, some general guidelines are suggested to avoid problems.

### Message Size

GSE Messages returned by email will have a maximum size of 100 Kilobytes. The maximum size of an ftp message will be standardized to one Mbyte, but each AutoDRM site may set their own limit and may give priority to trusted users as they see fit.

### Repeat Requests

Repeat requests for the same data by the same requestor within ten minutes of the original request will be ignored by AutoDRM.

## Returned Messages

An error in the address for a data message sent out by an AutoDRM will result in an email returned to the AutoDRM by the email system. The senders name (before the @ character in the mail-address) for such an email will be either mailer-daemon or postmaster (with any combination of upper and lower case letters). The AutoDRM will forward these messages to the local AutoDRM-operator; no other action is taken and no response is sent. The AutoDRM may also recognize returned messages by the DATA_TYPE which will be “data”, or by the presence of a REF_ID line which are not used in request messages.

## Syntax Errors

In case any syntax error is detected while processing a request message, a DATA_TYPE ERROR_LOG message is sent.

## AutoDRM Internal Problem Logging

Any problem other than a syntax error revealed during processing of a GSE request message should be reported to the AutoDRM operator who should take appropriate action. All REQUEST messages must be answered; DATA_TYPE ERROR_LOG is sent as response. Therefore an error message should be returned to the requestor.

## Operation Logs

It is recommended that all local AutoDRM installations keep logs of incoming and outgoing messages, parameters of MSG_ID lines, volume of data transferred, and UTC times of message receipt and dispatch.

# Chapter 3 Subscription Messages

## 3.1 INTRODUCTION

Subscriptions allow authorized users to have IDC data and data products forwarded to them automatically on a regular basis. Included in the products available through subscription are the continuous data from alpha stations in near-real-time, bulletins, waveform segments, arrival information, etc. Subscriptions may be set up for delivery continuously (in the case of continuous data), immediately upon receipt or generation at the IDC (e.g., segmented waveform data), or on a daily basis (e.g., daily bulletins and status reports). The only restriction is that the total amount of data forwarded to an NDC per day may not exceed 100 Megabytes (MB) unless special arrangements are made to provide extra bandwidth on the communications link.

## 3.2 SUBSCRIPTION PROCEDURES

A subscription is made by sending a GSE subscription message to the IDC. The email address is messages@cdidc.org. Upon receipt, the source of a subscription message is first validated for its authenticity. Next the volume of data that will be typically generated by the request is checked. Subscription messages that are not sent by an authorized user from a NDC or Working Group will be rejected. Subscriptions estimated to cause the data volume to exceed the maximum (100 MB) will also be rejected unless special arrangements are made (usually, the special arrangement is provision of a high capacity link between the IDC and NDC.)

After validation, the new subscription is added to the existing subscriptions for that user; and notification of the new subscription, in the form of a DATA_TYPE LOG message, is sent. Each subscription is assigned a unique identification number at the IDC.

## 3.3 SUBSCRIPTION FORMAT DESCRIPTION

Subscription messages follow the same rules as request messages, but because subscription messages provide data on a scheduled basis rather than on demand, they are given a separate message type and have additional capabilities that are not found in request messages. Detailed information relating only to subscriptions is given here along with summary information that is described in detail in Chapter 2.

A subscription request must contain the usual basic GSE message information: BEGIN, MSG_TYPE, MSG_ID, E-MAIL or FTP, and STOP.

*Example 3.3 - 1* Generalized Format for Subscription Message

```
BEGIN  GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
      (Subscription information)
STOP
```

The subscription information contains the information that describes what data (or data products) to send and how often they should be sent.

Subscriptions are a special message type that are very similar to request message types. Like request messages, subscriptions are defined through environment lines that constrain the data to be sent and request lines that specify which data to send.

Separate subscriptions are delimited as separate messages. In other words, each BEGIN / STOP sequence describes a single subscription which will be sent to the user as a single message. To avoid confusion, it is highly recommended that each subscription be kept simple; for example, subscribing to the IDC_ABEL and IDC_REB as separate subscriptions rather than as combined into a single subscription. Several subscription messages may be contained in a single email.

## *3.4* ENVIRONMENT LINES

There are two environment lines that are unique to descriptions: FREQ, which specifies how often the data should be sent to the subscriber; and SUBSCR_LIST, which allows the subscriber to list his subscriptions. Descriptions of the environment lines from request messages applicable to subscriptions are be found in Chapter 2. They are not repeated here since they are identical to those of request messages. Note that not all request environments are applicable to subscriptions. TIME, for example, is not a valid subscription environment and would be ignored or flagged as an unrecognized environment.

### FREQ

FREQ specifies how often the data should be sent to the subscriber. There are only three frequencies allowed: CONTINUOUS, IMMEDIATE, or DAILY. The parameter is CONTINUOUS when requesting continuous data in the alpha protocol; IMMEDIATE when the product is to be delivered as soon as it is available; or DAILY for delivery each day. FREQ may appear only once in each subscription message

*Syntax:* **FREQ** [schedule]

`schedule` .................. any of: CONTINUOUS, IMMEDIATE, or DAILY

*Default:* DAILY

## SUBSCR_LIST

SUBSCR_LIST is a list of subscription IDs. All of the IDs in this list will be un-subscribed when an UNSUBSCRIBE request is given.

*Syntax:* **`SUBSCR_LIST`** `[id[,...]]`

`id` ................................ identification number of the subscription

*Default:* None for UNSUBSCRIBE; all for SUBSCR_PROD

## LAT

See section of same name on page 16.

## LONG

See section of same name on page 16.

## EVENT_STA_DIST

See section of same name on page 17.

## DEPTH

See section of same name on page 17.

## DEPTH_MINUS_ERROR

See section of same name on page 18.

## MAG

See section of same name on page 18.

## MAG_TYPE

See section of same name on page 18.

## MB_MINUS_MS

See section of same name on page 19.

## STA_LIST

See section of same name on page 19.

### CHAN_LIST

See section of same name on page 19.

### AUX_LIST

See section of same name on page 20.

### BULL_TYPE

See section of same name on page 20.

### GROUP_BULL_LIST

See section of same name on page 21.

### COMM_LIST

See section of same name on page 22.

### PROB_TYPE

See section of same name on page 23.

### PROB_LOC

See section of same name on page 23.

### PROB_STATE

See section of same name on page 23.

## 3.5 SUBSCRIPTION REQUEST LINES

Subscription message request lines specify which information to send in the return data message. The arguments to a request line define the format for the return data message which are specified as a generic term such as GSE2.0, CSS3.0, SEED2.3, or INTERNAL; an optional sub-format specific to the data type being requested; and an optional EMBEDDED designator.

*Syntax:* **REQUEST_KEYWORD** [format [sub-format]] [EMBEDDED]

format ...................... either INTERNAL or one of a few generic data types (e.g., GSE2.0, CSS3.0, SEED2.3);
sub-format ............. specifies which internal format to use with this data type. The sub-format is used primarily for WAVEFORM requests;
EMBEDDED .................. indicates that the data should be embedded within the return message in a format that allows it to be stripped from the message and placed in independent files.

If no format or sub-format is specified, then the default (GSE2.0) format will be used. If INTERNAL is specified as the format, then the information is retrieved, but not returned in a response. The INTERNAL format is used in conjunction with an ASSOCIATE environment. EMBEDDED directs the AutoDRM to place the output in a file embedded in the return message. The file can be stripped from the message using a utility that is available from any AutoDRM installation that is capable of creating EMBEDDED files.

The following request data types are available through subscription at the IDC. Other data types may be offered in the future. Detailed descriptions of the formats used for the return data messages are given in Chapter 4. Only the SUBSCR_PROD and UNSUBSCRIBE requests are unique to subscription services, all others are also available through standard request messages at the IDC.

## SUBSCR_PROD

SUBSCR_PROD is a request for a list of the products currently subscribed to by the user. Included in the response to this request is the subscription ID that is a unique reference to the subscription, and a listing of the environment and request lines that define the specific product. The response is sent as a DATA_TYPE LOG message.

*Environment:* SUBSCR_LIST

*Example 3.5 - 1* The current list of subscriptions that are in effect for the user is obtained by using the SUBSCR_PROD request line.

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
SUBSCR_PROD
STOP
```

*Example 3.5 - 2* The response to this message is a LOG data message from the IDC:

```
BEGIN  GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID  example ANY_NDC
DATA_TYPE LOG GSE2.0
 The following data products are subscribed
 to by ndc@swe_ndc.org:
 Subscription ID: 52
   FREQ DAILY
   BULL_TYPE IDC_REB
   BULLETIN GSE2.0
 Subscription ID: 57
   FREQ IMMEDIATE
   LAT  0.0 TO 10.0
   LONG 120.0 TO 140.0
   BULL_TYPE IDC_ABEL
   BULLETIN GSE2.0
STOP
```

## UNSUBSCRIBE

UNSUBSCRIBE informs the IDC that the user wishes to remove the subscriptions referenced by the list in the SUBSCR_LIST environment. A return message is sent confirming that the subscription has been cancelled.

To stop the delivery of a subscription, the *subscription identification numbers* must be known. The previous example of using the SUBSCR_PROD request demonstrates how the *subscription identification numbers* may be determined. The identification numbers of the subscriptions that are to be deleted are listed on the SUBSCR_LIST environment line. This is followed by an UNSUBSCRIBE request line.

*Environment:* SUBSCR_LIST

*Example 3.5 - 3* the following code:

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
SUBSCR_LIST 52, 57
UNSUBSCRIBE
STOP
```

*Example 3.5 - 4* A confirmation log message from the IDC to the subscription user will be sent verifying that the subscription has been terminated.

```
BEGIN  GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID  example ANY_NDC
DATA_TYPE LOG GSE2.0
 The following data products have been removed
 by ndc@swe_ndc.org:
 Subscription ID: 52
   FREQ DAILY
   BULL_TYPE IDC_REB
   BULLETIN GSE2.0
 Subscription ID: 57
   FREQ IMMEDIATE
   BULL_TYPE IDC_ABEL
   BULLETIN GSE2.0
STOP
```

## WAVEFORM

Waveforms are the digital time series data. Waveform requests will typically accept sub-formats that specify how the digital data are formatted within the general format of the waveform data type. The available formats for waveform data from the IDC subscription service are standard continuous data format for continuous data and GSE2.0 format for all other waveform data. The sub-formats supported are INT, CM6, CM8, AUT, AU6, and AU8.

*Environment:* FREQ, **STA_LIST**, CHAN_LIST, AUX_LIST

Continuous data from an Alpha station may be subscribed to very simply using the mechanism provided. The FREQ environment should be set to CONTINUOUS and the stations/channels for forwarding should be specified in STA_LIST and CHAN_LIST environments. Continuous data will be forwarded from the IDC in the alpha protocol (described in Chapter 7). Because the volume of continuous data from more than a few channels could exceed 100 MB, special arrangements could be necessary to receive it.

*Example 3.5 - 5* To subscribe to continuous data from the short-period, high-gain, vertical channels from the ABAR array and from the central site (CDA0) of the CDAR array, the FREQ environment is set to CONTINUOUS, the appropriate station and channel lists are defined (ABAR refers to all sites within the array), and WAVEFORM are requested.

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ CONTINUOUS
STA_LIST ABAR, CDA0
CHAN_LIST shz
WAVEFORM GSE2.0
STOP
```

*Example 3.5 - 6* Waveform segments retrieved from Beta stations by the IDC can be retrieved automatically for all events by constraining only the station and requesting waveforms. The data will be forwarded in the appropriate format using email or ftp. Here, waveforms from station ABC are requested:

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ IMMEDIATE
STA_LIST ABC
WAVEFORM GSE2.0
STOP
```

## DETECTION

Detections are the result of some detection process run on waveform data. The information in a detection includes time, amplitude, period, and other parameters derived from polarization or array processing.

*Environment:* FREQ, **STA_LIST**, AUX_LIST

*Example 3.5 - 7* All of the detections from station ABC reported each day would be given with the following subscription:

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ DAILY
STA_LIST ABC
DETECTION GSE2.0
STOP
```

## ARRIVAL

An arrival is a detection that has been associated with an origin. It is not required that this information be used in determining the location or timing of the origin. At this stage, a phase name may have been associated with the arrival.

*Environment:* FREQ, **STA_LIST**, ARRIVAL_LIST, BULL_TYPE

*Example 3.5 - 8* To obtain the arrivals from stations ABC and DEF each day:

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ DAILY
STA_LIST ABC, DEF
CHAN_LIST *
ARRIVAL GSE2.0
STOP
```

## ORIGIN

Origins are solutions to the location and time of the source. Several origins may be determined by different organizations (e.g., the GSE IDC, NEIC, and ISC) for any one source.

*Environment:* FREQ, LAT, LONG, DEPTH, MAG, MAG_TYPE, BULL_TYPE, EVENT_STA_DIST, MB_MINUS_MS, DEPTH_MINUS_ERROR

*Example 3.5 - 9* The first example shows how to obtain the origin information for the daily IDC_REB delivered when the IDC_REB is finished.

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ  DAILY
BULL_TYPE  IDC_REB
ORIGIN GSE2.0
STOP
```

*Example 3.5 - 10* The request above can be further limited to a specific geographic region, magnitude, and depth range by including more environment lines.

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ  DAILY
LAT -60 TO 10.0
LONG -81 TO -34
MAG 4.5 TO 5.5
DEPTH 0 TO 10
BULL_TYPE  IDC_REB
ORIGIN GSE2.0
STOP
```

## EVENT

An event is defined as the preferred origin according to the provider of the data. The event will thus be different depending on where the information is obtained.

*Environment:* FREQ, LAT, LONG, DEPTH, MAG, MAG_TYPE, BULL_TYPE, EVENT_STA_DIST, MB_MINUS_MS, DEPTH_MINUS_ERROR

*Example 3.5 - 11* In this example, all of the IDC_REB events within regional distance (20 degrees) of stations ABC and DEF are obtained.

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ DAILY
BULL_TYPE  IDC_REB
STA_LIST ABC, DEF
EVENT_STA_DIST 0.0 TO 20.0
EVENT GSE2.0
STOP
```

## BULLETIN

Bulletins are composed of arrival, origin and event lines. Only the arrival information associated with the event is given in the bulletin.

*Environment:* FREQ, LAT, LONG, DEPTH, MAG, MAG_TYPE, BULL_TYPE, GROUP_BULL_LIST, EVENT_STA_DIST, MB_MINUS_MS, DEPTH_MINUS_ERROR

*Example 3.5 - 12* In the first example the daily IDC_REB is requested with no constraints; i.e., all IDC_REB events will be sent regardless of location, magnitude, depth, etc. The frequency of delivery (FREQ) is set to DAILY, which means that the IDC_REB will be delivered once each day, when analysis at the IDC has been completed.

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ DAILY
BULL_TYPE IDC_REB
BULLETIN GSE2.0
STOP
```

To subscribe to the immediate IDC_AEL and IDC_ABEL, two BEGIN / STOP sequences are used. The FREQ environment is set to IMMEDIATE. Soon after an event has been located (about an hour after real time for the IDC_AEL and about four hours after real time for the IDC_ABEL), the subscription software will forward the results to the user. In the example below, messages would be sent to the user quite often (as often as once every twenty minutes) since there are no constraints on the request. This arrangement would be appropriate for an NDC system that accepts the data automatically.

*Example 3.5 - 13* Subscribe to both AEL and ABEL

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ IMMEDIATE
BULL_TYPE IDC_AEL
BULLETIN GSE2.0
STOP
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ IMMEDIATE
BULL_TYPE IDC_ABEL
BULLETIN GSE2.0
STOP
```

To subscribe to the daily IDC_REB for events within some location, depth, and magnitude ranges the proper environments are set prior to the request lines. In the example below two latitude longitude boxes are described and all events shallower than 30 km depth between magnitudes 3.5 and 4.5 within these boxes would be delivered in one message.

*Example 3.5 - 14* Subscribe only to reports in an area of interest

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ DAILY
BULL_TYPE IDC_REB
MAG 3.5 To 4.5
DEPTH  TO 30
LAT -30 TO -20
LONG -180 TO -140
BULLETIN GSE2.0
LAT 75 TO 79
LONG 110 TO 140
BULLETIN GSE2.0
STOP
```

Note that once an environment has been established, it remains in effect until changed. Also note that the depth is given as “DEPTH  TO 30”, which is interpreted as DEPTH £ 30.

## STA_STATUS

Station status is given for the stations in the STA_LIST environment. The FREQ environment defines the time interval between the subscribed reports; it is limited to CONTINUOUS, IMMEDIATE, or DAILY (see “FREQ” on page 40).

*Environment:* FREQ, **STA_LIST**, AUX_LIST

*Example 3.5 - 15* To obtain the daily station status reports for all GSE stations:

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ DAILY
STA_LIST *
STA_STATUS GSE2.0
STOP
```

### CHAN_STATUS

Channel status is given for the channels in the CHAN_LIST and AUX_LIST environments for the stations in the STA_LIST environment. The FREQ environment defines the time interval between the subscribed reports (see “FREQ” on page 40).

*Environment:* FREQ, **STA_LIST**, CHAN_LIST, AUX_LIST

*Example 3.5 - 16* To obtain the channel status reports for the short period channels over a four day period at station ARA0:

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ DAILY
STA_LIST ARA0
CHAN_LIST s*
CHAN_STATUS GSE2.0
STOP
```

### COMM_STATUS

Communications status is given for the communications links listed in the COMM_LIST environment. The FREQ environment defines the time interval between the subscribed reports (see “FREQ” on page 40). a verbose communications status report listing individual circuit dropouts is obtained by using the VERBOSE sub-format.

*Environment:* FREQ, COMM_LIST

*Example 3.5 - 17* To obtain the verbose communications status reports for the links from station ABC to ANY_NDC and from ANY_NDC to the IDC:

```
BEGIN GSE2.0
MSG_TYPE SUBSCRIPTION
MSG_ID example ANY_NDC
E-MAIL name@my.computer
FREQ DAILY
COMM_LIST ABC, ANY_NDC
COMM_STATUS GSE2.0 VERBOSE
STOP
```

## 3.6 ASSOCIATION

Association is allowed in subscriptions, just as they are in request messages. (See section 2.6.)

*Example 3.6 - 1* To subscribe to a daily IDC_REB for events that fall within the given latitude and longitude constraints, and with DEPTH_MINUS_ERROR less than or equal to 10 km and the associated regional waveforms:

```
BEGIN  GSE2.0
MSG_TYPE  SUBSCRIPTION
MSG_ID  example ANY_NDC
E-MAIL  name@my.computer
FREQ DAILY
BULL_TYPE IDC_REB
LAT 30 TO 40
LONG -177 TO -155
DEPTH_MINUS_ERROR TO 10
BULLETIN GSE2.0
EVENT_STA_DIST TO 20
ASSOCIATE BULLETIN
WAVEFORM GSE2.0
STOP
```

# Chapter 4 Data Messages

## 4.1 GSE DATA MESSAGE FORMATS

GSE data formats provide a common format for data and data product exchange. The data formats all contain ASCII options that allow the exchange of information via email (even for waveforms). Waveforms in binary format may also be sent using the GSE message format, but the transmission of data messages with binary information must be via ftp.

Each data message contains the required information described in Chapter 1 for all GSE messages. All messages must contain the BEGIN line and be followed by a MSG_TYPE DATA line and a MSG_ID line using the proper formats for the arguments. Since a data message may be a response to a request, a REF_ID line may also appear. Following the identification line(s) are sections of data specific information.

Many different types of data may be exchanged using the message formats described here. These include seismic waveforms, bulletins, station information, and many others. For some of these data types, multiple data formats may be supported by the AutoDRM (e.g., GSE2.0, CSS3.0, and SEED2.3). Data messages in GSETT-3 must be available in the GSE2.0 formats. Sub-formats may also be available within a specific data type. A classic example of this is for GSE2.0 waveforms in which there are several internal data formats (e.g., INT, CM6, etc.). The type of data that is included in a data section and the format of the data are designated with a DATA_TYPE line.

### DATA_TYPE

Data sections must begin with a DATA_TYPE line. The arguments to DATA_TYPE are the type of data that follows (waveform, bulletin, etc.) and the format (e.g., GSE2.0, CSS3.0, or SEED2.3).

The sub-format, if any, must be included in the data section.

*Syntax:* **`DATA_TYPE`** `type_of_data format`

`type_of_data` ......... the type of data that follows; typical examples are WAVEFORM, BULLETIN, and RESPONSE.
`format` ....................... the general format of the data (e.g., GSE2.0, CSS3.0, or SEED2.3).

An example of the DATA_TYPE line is:

*Example 4.1 - 1* **DATA_TYPE** waveform GSE2.0

There is no line that is used to end a data section. The end of the section is implied by another DATA_TYPE line, or a STOP line. Users should be aware that CONTINUED lines may appear at any line of a data section (with the exception of sections with binary data).

For GSETT-3, all data types used by the IDC will have a GSE2.0 format, a CSS3.0 format, or both a GSE2.0 and a CSS3.0 format. The GSE2.0 format will describe most of the frequently used data message types. The CSS3.0 format is the internal format of the IDC and describes the table format for all of the IDC database tables. NDC and station AutoDRM's will use the GSE2.0 format

## EMBEDDED

Established formats for seismic data (e.g., CSS3.0 and SEED2.3) have a large software base that uses the data in its native format. The GSE message format provides a convenient method of maintaining the file structure of these formats through the EMBEDDED command. An EMBEDDED command line may follow any DATA_TYPE command line. The data to be placed in a file are "embedded" in the GSE message following the EMBEDDED line. EMBEDDED provides the information necessary to unpack the files once the GSE message has reached its destination:

*Syntax:* **`EMBEDDED`** `file bytes msg_form [native_form] [line_len]`

`file` ............................ file into which the data will be placed;
`bytes` .......................... number of bytes (including new lines, etc.) in the file;
`msg_form` .................. either ASCII or BINARY. Specifies the form that the message takes. BINARY messages must be sent by ftp, while ASCII data may be sent by either email or ftp.
`native_form` ........... specifies the form of the destination file (RAW or ASCII);
`line_len` .................. used to define the length of fixed-length lines.

Once the message has been received, a utility that is provided by the center that generated the file unpacks the embedded files leaving a GSE message with the EMBEDDED lines and the embedded file information replaced by an IN_FILE line with the file name into which the data were placed.

The following examples show how embedded files appear before and after they are unpacked. The first message is an example of a response to a request from the ANY_NDC for response and waveform data embedded in CSS3.0 files.

*Example 4.1 - 2* Embedded message before unpacking:

```
BEGIN GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID example ANY_NDC
DATA_TYPE LOG
   Command RESPONSE processed
DATA_TYPE response  CSS3.0
EMBEDDED BLA.response 1234 ASCII ASCII 132
      (Response specific information)
DATA_TYPE LOG
   Command WAVEFORM processed
DATA_TYPE WAVEFORM  CSS3.0
EMBEDDED data.wfdisc 3124 ASCII ASCII 204
      (CSS3.0 wfdisc file)
EMBEDDED data1.w 12048 BINARY BINARY
      (Waveform data)
EMBEDDED data2.w 10345 BINARY BINARY
      (Waveform data)
STOP
```

*Example 4.1 - 3* After unpacking there would be five files placed in the local file system; BLA.response, data.wfdisc, data1.w, data2.w and the GSE message which would look like this:

```
BEGIN GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID example ANY_NDC
DATA_TYPE LOG
   Command RESPONSE processed
DATA_TYPE response  CSS3.0
IN_FILE BLA.response
DATA_TYPE LOG
   Command WAVEFORM processed
DATA_TYPE WAVEFORM  CSS3.0
IN_FILE data.wfdisc
IN_FILE data1.w
IN_FILE data2.w
STOP
```

## 4.2 MESSAGE LOGS

In response to a request, the AutoDRM will log its progress and errors in a LOG data type placed just before the data type section containing the data. Free format ASCII lines are used starting in column 2; i.e., no characters may be placed in column 1. The exact content of the logs is unspecified.

LOG data types contain information about changes that may have been made in the format of the return message (e.g., a default format may be used if the requested format is not available); or the protocol of the return message (e.g., a large return email message may be changed to ftp).

*Example 4.2 - 1* The following example is a data message sent to a requestor of data. Just before the data section, a log section is used to state that the waveform request command was processed.

```
BEGIN GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID example ANY_NDC
DATA_TYPE LOG
   Command waveform processed.
DATA_TYPE WAVEFORM
      (waveform data)
STOP
```

## 4.3 ERROR LOGS

A special data type is reserved for error logs so that they can be identified easily in the case that something goes wrong in a request message. Specific formats have not been defined at this time, although it is recommended that the request message be given with the line or lines causing the error identified.

*Example 4.3 - 1* The following error log example identifies the request message line in which the AutoDRM failed. Note that the log has been indented to avoid placing keywords in column one.

```
BEGIN GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID example ANY_NDC
DATA_TYPE ERROR_LOG
   An error was detected in the following request message:
   BEGIN GSE2.0
   MSG_TYPE request
   MSG_ID example ANY_NDC
   TIME 94/03/01 TO 94/03/02
   *** Unrecognized time format ***
   STA_LIST ARA0
   WAVEFORM
   STOP
STOP
```

## *4.4* FTP LOGS

In response to a large data request, data are sent via ftp and a message sent to the user by email with information on the location of the file to be retrieved by the requestor using ftp. Data type FTP_LOG is provided to convey this information in a consistent manner so that automated data retrieval programs can easily obtain the data.

The first line of the FTP_LOG data type must contain the essential information for retrieving the message file.

*Syntax:* **`FTP_FILE`** `net_address login_mode directory file`

`net_address` ........... address of machine where data reside (although names are preferred, the IP number may be used).
`login_mode` .............. either USER or GUEST. If USER, then the requestor should log in as a user to ftp the data (an account is required). If GUEST, the requestor should log in as anonymous to ftp the data (an account is not required).
`directory` ............... specifies the absolute directory in which the message file will reside. Note that this may depend on the login_mode. Also, the directory name is case sensitive.
`file` ............................ is the name of the file that contains the message.

Free format log information may follow the FTP_FILE line. As with all log and comment information, the first column must be kept blank.

*Example 4.4 - 1* The following example is a data message sent to a requestor of data. It indicates that the data are on machine "cdidc.org" in directory " / temp / data" in file "1994125001.msg". The requestor must log into cdidc.org as a user to obtain the data.

```
BEGIN GSE2.0
MSG_TYPE DATA
MSG_ID example GSE_IDC
REF_ID example ANY_NDC
DATA_TYPE FTP_LOG
FTP_FILE cdidc.org USER /temp/data 1994125001.msg
   The original request could not be satisfied using email due to
   the size of the requested information; ftp was used instead.
   Please log into your user account to retrieve the data.  Data
   will be removed by 1994/10/23.
STOP
```

## 4.5 GSE2.0 Waveform Segments

The GSE2.0 format for waveforms consists of a waveform identification line (see WID2, Table 6) followed by the waveform information itself (DAT2) and a checksum of the data (see CHK2, Table 8). Each data (DAT2) section should end with a checksum so that the validity (or otherwise) of the data can be verified.

The WID2 line gives the date and time of the first data sample; the station, channel, and auxiliary codes; the sub-format of the data; the number of samples and sample rate; the calibration of the instrument represented as the number of nanometers per digital count at the calibration period; the type of instrument, as shown in Table 7; and the horizontal and vertical orientation of the instrument. Note that the auxiliary code will be blank in most cases; it is only used when there is a conflict between two data streams with the same station and channel codes. Instrument response information must be obtained separately using a RESPONSE request.

Data following the DAT2 line may be in any of six different sub-formats recognized in the GSE2.0 waveform format. They are INT, CM6, CM8, AUT, AU6, and AU8. INT is a simple ASCII sub-format, the "CM" sub-formats are for compressed data, and the "AU" sub-formats are for authentication data. All of the GSE formats represent the numbers as integers.

A checksum is computed for the waveform data in the GSE2.0 waveform format. The checksum is computed from integer data values prior to converting them to any of the subformats. To prevent overflow, the checksum is computed modulo 100,000,000 and stored as an eight digit integer without sign. To avoid possible confusion and bypass incompatibilty problems, a C function and a F77 subroutine are provided in Appendix A demonstrating the exact algorithm for checksum computation.

The line length limits for GSE messages are enforced for the GSE2.0 data formats; that is, no line may be longer than 1024 bytes long, the default line length is 132 characters, and the line length can be set using LINE_LEN. The line continuation character (“ \ ”) is not used in waveform data lines.

Table 6. **Waveform Identification Line Format**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-4 | id | a4 | must be “WID2” |
| 6-15 | date | i4,a1,i2,a1,i2 | date of the first example (yyyy/mm/dd) |
| 17-28 | time | i2,a1,i2,a1,f6.3 | time of the first example (hh:mm:ss.sss) |
| 30-34 | station | a5 | ISC station code |
| 36-38 | channel | a3 | FDSN channel designator |
| 40-43 | auxid | a4 | auxiliary identification code |
| 45-47 | datatype | a3 | “INT”, “CMn”, or “AUx”<br>INT is integers as ASCII characters;<br>“CM” denotes compressed data, and n is either 6 (6-bit compression), or 8 (8-bit compression)<br>“AU” signifies authentication and x is T (uncompressed binary integers), 6 (6-bit compression), or 8 (8-bit compression) |
| 49-56 | samps | i8 | number of samples |
| 58-68 | samprat | f11.6 | data sampling rate (Hz) |
| 70-79 | calib | e10.2 | calibration factor; i.e., the ground motion in nanometers per digital count at calibration period (calper) |
| 81-87 | calper | f7.3 | calibration reference period; i.e., the period in seconds at which calib is valid; calper should be near the flat part of the response curve. (in most cases, 1 sec) |
| 89-94 | instype | a6 | instrument type (from Table 7) |
| 96-100 | hang | f5.1 | horizontal orientation of sensor, measured in degrees clockwise from North (-1.0 if vertical) |
| 102-105 | vang | f4.1 | vertical orientation of sensor, measured in degrees from vertical (90.0 if horizontal) |

Table 7. **GSE Instrument Types**

| instype | Description |
|---|---|
| Akashi | Akashi |
| 23900 | Geotech 23900 |
| BB-13V | Geotech BB-13V |
| CMG-3 | Guralp CMG-3 |
| CMG-3N | Guralp CMG-3NSN |
| CMG-3T | Guralp CMG-3T |
| CMG-3E | Guralp CMG3-ESP |
| FBA-23 | Kinemetrics FBA-23 |
| GS-13 | Geotech GS-13 |
| GS-21 | Geotech GS-21 |
| KS3600 | Geotech KS-36000 |
| KS360i | Geotech KS-36000-I |
| KS5400 | Geotech KS-54000 |
| Mk II | Willmore Mk II |
| Oki | Oki |
| Parus2 | Parus-2 |
| S-13 | Geotech S-13 |
| S-500 | Geotech S-500 |
| STS-1 | Streckeisen STS-1 |
| STS-2 | Streckeisen STS-2 |
| TSJ-1e | TSJ-1e |

Table 8. **Data Header Line Format**

| Position | Name | Format | Description |
|---|---|---|---|
| Header Line | | | |
| 1-4 | header id | a4 | must be “DAT2” |
| Data Lines | | | |
| 1-1024 (variable) | data | i, a, or f | data values |
| Checksum Line | | | |
| 1-4 | header id | a4 | must be “CHK2” |
| 6-13 | checksum | i8 | for checksum algorithm see Appendix A |

## Sub-format INT

The INT waveform sub-format represents integer data as blank or NewLine delimited ASCII characters. The number of blank spaces between samples is unspecified and an individual sample value may not be continued on the next line.

*Example 4.5 - 1* Data in INT data type with 105 character line length:

```
WID2 1994/03/10 12:13:14.800 BLA   SHZ      INT    32490   40.000000   1.30e-02     2.0 GS-13   -1.0  0.0
DAT2
1873 1734 1690 1200 873 340 -290 -478 -1300 -209 -1972 -24 13 25 64 81 102 76 53 23 -10 -80 -132 -487
...
12 15 36 75 53 80 27 6 -17 -32 -95 -73 -43 -4 3 29 46 59 100 125 103 76 52 10 -30
CHK2 4968214
```

## Compression Schemes

Two different compression schemes are recognized in the GSE2.0 waveform format; CM6 and CM8; the 6-bit and 8-bit second difference schemes used in GSETT2.

The basis of these compression schemes is that, for seismic data, the difference between data samples is usually very much smaller than their instantaneous magnitudes, and the difference of the differences (the second difference) is even smaller. So, transmitting the second differences requires fewer significant bits. Reductions in the message length can be achieved if the number of bits to convey the information is reduced when the signal level is small and expanded when the signal level rises. Since samples will take a variable number of bits, an Index is required in order to indicate how many bits each sample takes.

Both of the compression schemes use second differences as a first step to reducing the number of significant bits required to convey the information contained in the time series. Second difference algorithms are quite simple. A first difference is computed as the difference between successive samples. Second differences are simply the difference between the differences.

The following paragraphs describe the schemes used in each of the compression schemes to reduce the number of bits and/or to make transmission easy.

## Sub-format CM6

CM6 is a data compression algorithm which was successfully employed in GSETT2 and was referred to as 6-bit compression. The advantage of this method is in its conversion of binary integer data to ASCII characters that can be successfully transmitted using email. The compression algorithm converts waveforms into a set of printable ASCII characters carefully avoiding those which have been found to cause problems to either communications circuits or the computers connected to them. It uses only the 64 characters +, -, 0 - 9, A - Z and a - z.

Initially, all data samples in the packet are represented as 32 bit, 2's complement integer, with a range of $-(2^{31})$ to $+(2^{31}-1)$. Second difference samples are encoded as the difference between the first differences and can be computed using the following formula:

$$D_2(j) = S(j) - 2S(j-1) + S(j-2)$$

where zero and negative indices are ignored. Thus the second difference data for N samples are:

$$S(1), S(2) - 2S(1), S(3) - 2S(2) + S(1), \ldots, S(N) - 2S(N-1) + S(N-2)$$

To compress the numbers, the input data (second difference integers) are first converted from two's complement to sign and magnitude. These numbers are then fit into a variable number of bytes in which only the six most significant bits are utilized. The most significant usable bit of each byte is used as a flag or control bit which, if set, is used to signify that the following byte also contains information relating to the same sample. The second most significant bit is used as a sign bit in the first byte pertaining to a sample and as a data bit in all following bytes of the sample. All other bits are used to represent the value of the second difference of the sample:

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| control | sign/ data | data | data | data | data | unused | unused |

These six-bit bytes are then used to refer to a look-up table (Table 9) from which one of 64 different ASCII characters (+, -, 0-9, A-Z, a-z) is extracted.

Table 9. **ASCII Representation of Bit Patterns for CM6**

| Bit Pattern | Char | Bit Pattern | Char | Bit Pattern | Char | Bit Pattern | Char |
|---|---|---|---|---|---|---|---|
| 000000 | + | 010000 | E | 100000 | U | 110000 | k |
| 000001 | - | 010001 | F | 100001 | V | 110001 | l |
| 000010 | 0 | 010010 | G | 100010 | W | 110010 | m |
| 000011 | 1 | 010011 | H | 100011 | X | 110011 | n |
| 000100 | 2 | 010100 | I | 100100 | Y | 110100 | o |
| 000101 | 3 | 010101 | J | 100101 | Z | 110101 | p |
| 000110 | 4 | 010110 | K | 100110 | a | 110110 | q |
| 000111 | 5 | 010111 | L | 100111 | b | 110111 | r |
| 001000 | 6 | 011000 | M | 101000 | c | 111000 | s |
| 001001 | 7 | 011001 | N | 101001 | d | 111001 | t |
| 001010 | 8 | 011010 | O | 101010 | e | 111010 | u |
| 001011 | 9 | 011011 | P | 101011 | f | 111011 | v |
| 001100 | A | 011100 | Q | 101100 | g | 111100 | w |
| 001101 | B | 011101 | R | 101101 | h | 111101 | x |
| 001110 | C | 011110 | S | 101110 | i | 111110 | y |
| 001111 | D | 011111 | T | 101111 | j | 111111 | z |

*Example 4.5 - 2* Data in CM6 sub-format with 80 character line length:

```
WID2 1994/03/10 12:13:14.800 BLA   SHZ      CM6    32490   40.000000   1.30e-02\
     2.0 GS-13   -1.0  0.0
DAT2
hsYbhas76hJHjhd7sk+bsaybaueJjhgHESKHbs923kjGE+GE6gdas723hs7S7jk2hahsJHAsyd0-hd72
...
kjSKuhlksfkluhAkf874kjklds87kjhZ87iu97Dfiu97iuhDSf796khsdfuhskldf672KSEfkiu++kjh
CHK2 4968214
```

## Sub-format CM8

The CM8 sub-format is similar to the CM6 sub-format. The same algorithm is used, but the compression is more efficient than the 6-bit sub-format since there are no unused bits. The 8-bit scheme is a binary format that cannot be transmitted using email; ftp must be used.

The second difference integers are first converted from 2's complement to sign and magnitude. These numbers are then fit into a variable number of bytes in which all eight significant bits are utilized. The most significant usable bit of each byte is used as a flag or control bit which, if set, is used to signify that the following byte also contains information relating to the same sample. The second most significant bit is used as a sign bit in the first byte pertaining to a sample and as data in all following bytes. All other bits are used to represent the value of the second difference:

| MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|
| control | sign/<br>data | data | data | data | data | data | data |

## Sub-format AUT

Waveform data that have been signed for authentication must contain more than just waveform samples; it must also contain time and status information. The GSE2.0 authentication sub-format includes this information in packets that are individually signed for authentication. The signed data in this sub-format include the time-stamp, the number of samples, the status word, and the data. Waveform segments consist of several of these packets concatenated, as shown in Table 10. The data are binary integers.

Table 10. **Authentication Data Format**

| Name | Format | Description |
|---|---|---|
| length of packet | 4-byte IEEE integer | length of packet in bytes, not counting this word, for channel data that follows (divisible by 4) |
| authentication | 40-byte string | authentication signature |
| timestamp | 8-byte IEEE float | seconds since 1 January 1970 00:00 for first sample. Must be within one sample of nominal time. |
| samples | 4-byte IEEE integer | number of samples in channel packet |
| status word | 4-byte string | Data status byte (most significant byte):<br>bit 31 1=dead channel<br>bit 30 1=zeroed data<br>bit 29 1=clipped<br>bit 28 1=calibration signal<br>bits 24-27 undefined<br>Station status byte:<br>bit 23 1=vault door open<br>bit 22 1=authentication box opened<br>bit 21 1=equipment moved<br>bit 20 1=clock differential too large<br>bits 16-19 undefined<br>Station specific bits:<br>bits 0-15 user defined (e.g., station status counter) |
| data | (length of packet minus 56 bytes - based on original 4-byte IEEE integers) | raw 4-byte integers or compressed data |

O P E R A T I O N S

### Sub-formats AU6 and AU8

Data that have an authentication signature may be compressed for ease in transmission. The same authentication sub-format described above is used for compressed authenticated data, with the difference that the data are compressed. The AU6 and AU8 authentication formats are compressed using the 6-bit and 8-bit compression schemes described above. The compression algorithm is applied only to the data, not to the 60 bytes of header information given in Table 10. Before verifying the authenticity of the data, it must be uncompressed using the appropriate decompression scheme.

## 4.6 STATION INFORMATION

The STATION data type contains information describing the site, location and dates of operation. For arrays, the unique array code should be given along with the information from each element. See Table 11.

Table 11. **Station Header & Data Formats**

| Position | Name | Format | Description |
|---|---|---|---|
| Station Header | | | |
| 1-3 | | a3 | Sta |
| 7-10 | | a4 | Type |
| 13-20 | | a8 | Latitude |
| 23-31 | | a9 | Longitude |
| 36-39 | | a4 | Elev |
| 43-49 | | a7 | On Date |
| 53-60 | | a8 | Off Date |
| Station Data | | | |
| 1-5 | sta | a5 | International Station Code (ISC) |
| 7-10 | statype | a4 | 1C=single component<br>3C=three-component<br>hfa=high frequency array<br>lpa=long period array |
| 12-20 | lat | f9.5 | latitude (degrees, S is negative) |
| 22-31 | long | f10.5 | longitude (degrees, W is negative) |
| 33-39 | elev | f7.3 | elevation (km) |
| 41-50 | ondate | i4,a1,i2,a1,i2 | start of operation (yyyy/mm/dd) |
| 52-61 | offdate | i4,a1,i2,a1,i2 | end of operation (yyyy/mm/dd) |

*Example 4.6 - 1* STATION Data Type

```
DATA_TYPE STATION
Sta   Type  Latitude  Longitude    Elev    On Date   Off Date
ARCES hfa   69.53490   25.50580   0.403 1987/09/30
ARA0  3C    69.53490   25.50580   0.403 1987/09/30
ARA1  1C    69.53630   25.50710   0.411 1987/09/30
ARA2  1C    69.53380   25.50780   0.392 1987/09/30
ARA3  1C    69.53460   25.50190   0.402 1987/09/30
ARB1  1C    69.53790   25.50790   0.414 1987/09/30
ARB2  1C    69.53570   25.51340   0.397 1987/09/30
ARB3  1C    69.53240   25.51060   0.376 1987/09/30
ARB4  1C    69.53280   25.49980   0.378 1987/09/30
ARB5  1C    69.53630   25.49850   0.405 1987/09/30
ARC1  1C    69.54110   25.50790   0.381 1987/09/30
ARC2  3C    69.53830   25.52290   0.395 1987/09/30
ARC3  1C    69.53290   25.52310   0.376 1987/09/30
ARC4  3C    69.52930   25.51170   0.377 1987/09/30
ARC5  1C    69.53000   25.49820   0.374 1987/09/30
ARC6  1C    69.53410   25.48820   0.395 1987/09/30
ARC7  3C    69.53960   25.49360   0.362 1987/09/30
ARD1  1C    69.54830   25.50930   0.395 1987/09/30
ARD2  1C    69.54520   25.53080   0.366 1987/09/30
```

## 4.7 CHANNEL INFORMATION

The channel data type contains information describing the sensors and their emplacement (see Table 12).

*Example 4.7 - 1* CHANNEL Data Type

```
ATA_TYPE CHANNEL
ta  Chan Aux    Latitude  Longitude    Elev  Depth   Hang  Vang Sample_Rate Inst       On Date   Off Date
RA0  SHE        69.53490   25.50580   0.403  0.000   90.0  90.0   40.000000 GS-13   1987/09/30
RA0  SHN        69.53490   25.50580   0.403  0.000    0.0  90.0   40.000000 GS-13   1987/09/30
RA0  SHZ        69.53490   25.50580   0.403  0.000          0.0   40.000000 GS-13   1987/09/30
RA1  SHZ        69.53630   25.50710   0.411  0.000          0.0   40.000000 GS-13   1987/09/30
RA2  SHZ        69.53380   25.50780   0.392  0.000          0.0   40.000000 GS-13   1987/09/30
RA3  SHZ        69.53460   25.50190   0.402  0.000          0.0   40.000000 GS-13   1987/09/30
RB1  SHZ        69.53790   25.50790   0.414  0.000          0.0   40.000000 GS-13   1987/09/30
RB2  SHZ        69.53570   25.51340   0.397  0.000          0.0   40.000000 GS-13   1987/09/30
RB3  SHZ        69.53240   25.51060   0.376  0.000          0.0   40.000000 GS-13   1987/09/30
RB4  SHZ        69.53280   25.49980   0.378  0.000          0.0   40.000000 GS-13   1987/09/30
RB5  SHZ        69.53630   25.49850   0.405  0.000          0.0   40.000000 GS-13   1987/09/30
RC1  SHZ        69.54110   25.50790   0.381  0.000          0.0   40.000000 GS-13   1987/09/30
RC2  SHE        69.53830   25.52290   0.395  0.000   90.0  90.0   40.000000 GS-13   1987/09/30
RC2  SHN        69.53830   25.52290   0.395  0.000    0.0  90.0   40.000000 GS-13   1987/09/30
RC2  SHZ        69.53830   25.52290   0.395  0.000          0.0   40.000000 GS-13   1987/09/30
RC3  SHZ        69.53290   25.52310   0.376  0.000          0.0   40.000000 GS-13   1987/09/30
RC4  SHE        69.52930   25.51170   0.377  0.000   90.0  90.0   40.000000 GS-13   1987/09/30
RC4  SHN        69.52930   25.51170   0.377  0.000    0.0  90.0   40.000000 GS-13   1987/09/30
RC4  SHZ        69.52930   25.51170   0.377  0.000          0.0   40.000000 GS-13   1987/09/30
RC5  SHZ        69.53000   25.49820   0.374  0.000          0.0   40.000000 GS-13   1987/09/30
RC6  SHZ        69.53410   25.48820   0.395  0.000          0.0   40.000000 GS-13   1987/09/30
RC7  SHE        69.53960   25.49360   0.362  0.000   90.0  90.0   40.000000 GS-13   1987/09/30
RC7  SHN        69.53960   25.49360   0.362  0.000    0.0  90.0   40.000000 GS-13   1987/09/30
RC7  SHZ        69.53960   25.49360   0.362  0.000          0.0   40.000000 GS-13   1987/09/30
RD1  SHZ        69.54830   25.50930   0.395  0.000          0.0   40.000000 GS-13   1987/09/30
RD2  SHZ        69.54520   25.53080   0.366  0.000          0.0   40.000000 GS-13   1987/09/30
RD3  SHZ        69.53660   25.54830   0.331  0.000          0.0   40.000000 GS-13   1987/09/30
RD4  SHZ        69.52710   25.53620   0.371  0.000          0.0   40.000000 GS-13   1987/09/30
RD5  SHZ        69.52140   25.51180   0.351  0.000          0.0   40.000000 GS-13   1987/09/30
RD6  SHZ        69.52270   25.49000   0.413  0.000          0.0   40.000000 GS-13   1987/09/30
RD7  SHZ        69.52940   25.47070   0.413  0.000          0.0   40.000000 GS-13   1987/09/30
RD8  SHZ        69.53840   25.46860   0.368  0.000          0.0   40.000000 GS-13   1987/09/30
RD9  SHZ        69.54540   25.48570   0.359  0.000          0.0   40.000000 GS-13   1987/09/30
```

Table 12. **Channel Header & Data Formats**

| Position | Name | Format | Description |
|---|---|---|---|
| | | | Channel Header |
| 1-3 | | a3 | Sta |
| 6-9 | | a4 | Chan |
| 11-13 | | a3 | Aux |
| 17-24 | | a8 | Latitude |
| 27-35 | | a9 | Longitude |
| 40-43 | | a4 | Elev |
| 46-50 | | a5 | Depth |
| 54-57 | | a4 | Hang |
| 60-63 | | a4 | Vang |
| 65-75 | | a11 | Sample_Rate |
| 77-80 | | a4 | Inst |
| 88-94 | | a7 | On Date |
| 98-105 | | a7 | Off Date |
| | | | Channel Data |
| 1-5 | sta | a5 | International Station Code (ISC) |
| 7-9 | chan | a3 | FDSN channel code |
| 11-14 | auxid | a4 | auxiliary code |
| 16-24 | lat | f9.5 | latitude (degrees, S is negative) |
| 26-35 | long | f10.5 | longitude (degrees, W is negative) |
| 37-43 | elev | f7.3 | elevation (km) |
| 45-50 | edepth | f6.3 | emplacement depth (km) |
| 52-57 | hang | f6.1 | horizontal angle of emplacement (degrees clockwise from N, -1.0 if vertical) |
| 59-63 | vang | f5.1 | vertical angle of emplacement (degrees from vertical) |
| 65-75 | samprat | f11.6 | sample rate (samp/sec) |
| 77-83 | inst | a6 | instrument type (e.g., CGM-5, STS-2, GS-13) |
| 85-94 | ondate | i4,a1,i2,a1,i2 | start date of operation (yyyy/mm/dd) |
| 96-105 | offdate | i4,a1,i2,a1,i2 | end date of operation (yyyy/mm/dd) |

## 4.8 INSTRUMENT RESPONSE

The RESPONSE data type allows the complete response to be given as a series of response groups that can be cascaded. Modern instruments are composed of several different components, each with its own response. This format mimics the actual configuration of the instrumentation.

A complete response description is made up of the CAL2 identification line plus one or more of the PAZ2, FAP2, GEN2, DIG2, and FIR2 response sections in any order. The response sections should be given sequential stage numbers (beginning with 1) in the order they occur in the system response.

Each response section is comprised of a header line and sufficient occurrences of the values lines to provide all required coefficients. Note that the DIG2 section may occur only once per response and that it requires no values lines. Comments may be inserted after the CAL2 header and after any response section as desired. Successive channel responses should also be separated by blank lines for readability.

The input of the earth is in nanometers of displacement (i.e., all of the responses are displacement responses). Velocity or acceleration responses can be obtained by multiplying the response curve by $i\omega$ or $i\omega^2$, respectively.

The first line is the "CAL2" line giving general information about the response information that follows in Table 13.

Table 13. **Calibration Identification Line (CAL2) Format**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-4 | id | a4 | must be “CAL2” |
| 6-10 | sta | a5 | International Station Code (ISC) |
| 12-14 | chan | a3 | FDSN channel code |
| 16-19 | auxid | a4 | auxiliary identification code |
| 21-26 | instype | a6 | Instrument type from list below:<br>Akashi Akashi<br>23900 Geotech 23900<br>BB-13V Geotech BB-13V<br>CMG-3 Guralp CMG-3<br>CMG-3N Guralp CMG-3NSN<br>CMG-3T Guralp CMG-3T<br>CMG-3E Guralp CMG3-ESP<br>FBA-23 Kinemetrics FBA-23<br>GS-13 Geotech GS-13<br>GS-21 Geotech GS-21<br>KS3600 Geotech KS-36000<br>KS360i Geotech KS-36000-I<br>KS5400 Geotech KS-54000<br>Mk II Willmore Mk II<br>Oki Oki<br>Parus2 Parus-2<br>S-13 Geotech S-13<br>S-500 Geotech S-500<br>STS-1 Streckeisen STS-1<br>STS-2 Streckeisen STS-2<br>TSJ-1e TSJ-1e |
| 28-37 | calib[*] | e10.2 | system sensitivity (nm/count) at reference period (calper) |
| 39-45 | calper[a] | f7.3 | calibration reference period (seconds) |
| 47-56 | samprat[a] | f10.5 | system output sample rate (Hz) |
| 58-67 | ondate | i4,a1,i2,a1,i2 | effective start date (yyyy/mm/dd) |
| 69-73 | ontime | i2,a1,i2 | effective start time (hh:mm) |
| 75-84 | offdate[**] | i4,a1,i2,a1,i2 | effective end date (yyyy/mm/dd) |
| 86-90 | offtime | i2,a1,i2 | effective end time (hh:mm) |

* Calibration, cal period, and sample rate should be the same as in the WID2 header.

** The start/end date-times specify the time period for which the response is valid. If the response is still valid, the off date-time should be left blank.

## Poles and Zeros

A poles and zeros section (PAZ2) can be used for either an analog filter or an IIR filter. In the data section, poles are always given first followed by zeros, as shown in Table 14.

Table 14. **Format for Poles and Zeros Section**

| Position | Name | Format | Description |
|---|---|---|---|
| Header | | | |
| 1-4 | id | a4 | must be “PAZ2” |
| 6-7 | snum | i2 | stage sequence number |
| 9 | ounits* | a1 | output units code (V=volts, A=amps, C=counts) |
| 11-25 | sfactor** | e15.8 | scale factor |
| 27-30 | deci*** | i4 | decimation (blank if analog) |
| 32-39 | corr | f8.3 | group correction applied (seconds) |
| 41-43 | npole**** | i3 | number of poles |
| 45-47 | nzero | i3 | number of zeros |
| 49-73 | descrip | a25 | description |
| Data | | | |
| 2-16 | rroot | e15.8 | real part of pole or zero |
| 18-32 | iroot | e15.8 | imaginary part of pole or zero |

* Output units are V for volts, A for amps, and C for counts. Seismometers typically output volts or amps while an IIR filter would output counts. However, a simple response might give the seismometer with an output directly in counts implying the digitizer and neglecting the digital filters.

** The scale factor is in output units per input units. If this is the first (seismometer) section the input units are nm. Otherwise, the input units are the output units of the previous section.

*** The decimation factor and group correction must be blank for an analogue filter and must be non-blank (zero for no decimation or group correction) for a digital filter.

****For an analogue filter the poles and zeros specify the Laplace transform. For an IIR filter, they specify the Z-transform.

## Frequency, Amplitude, and Phase

Like PAZ2, the FAP2 section can be used to specify the response of analogue or digital filters, or some combination of them including a complete system response, as shown in Table 15.

Table 15. **Format of Frequency, Amplitude and Phase Section**

| Position | Name | Format | Description |
|---|---|---|---|
| Header | | | |
| 1-4 | id | a4 | must be “FAP2” |
| 6-7 | snum | i2 | stage sequence number |
| 9 | ounits | a1 | output units code (V=volts, A=amps, C=counts) |
| 11-14 | deci | i4 | decimation (blank if analog) |
| 16-23 | corr | f8.3 | group correction applied (seconds) |
| 25-27 | ntrip | i3 | number of frequency, amplitude, phase triplets |
| 29-53 | descrip | a25 | description |
| Data (Triplets) | | | |
| 2-11 | freq | f10.5 | frequency (Hz) |
| 13-27 | amp | e15.8 | amplitude (input units/output units) |
| 29-32 | phase | i4 | phase delay (degrees) |

## Generic Response

Like PAZ2, the GEN2 section can be used to specify the response of analogue or digital filters, or some combination of them including a complete system response, as shown in Table 16.

Table 16. **Format for Generic Response Section**

| Position | Name | Format | Description |
|---|---|---|---|
| Header | | | |
| 1-4 | id | a4 | must be GEN2 |
| 6-7 | snum | i2 | stage sequence number |
| 9 | ounits | a1 | output units code (V=volts, A=amps, C=counts) |
| 11-25 | calib | e15.8 | section sensitivity (input units/output units) |
| 27-32 | calper | f7.3 | calibration reference period (seconds) |
| 35-38 | deci | i4 | decimation (blank if analog) |
| 40-47 | corr | f8.3 | group correction applied (seconds) |
| 49-51 | ncorner | i3 | number of corners |
| 53-77 | descrip | a25 | description |
| Data | | | |
| 2-12 | cfreq | f11.5 | corner frequency (Hz) |
| 14-19 | slope | f6.2 | slope above corner (dB/decade) |

## Digitizer Response

There is no values section for the digitizer as this just specifies the digitizer sample rate and sensitivity, as shown in Table 17. It also gives the user a chance to identify the model of digitizer being used.

Table 17. **Format for Digitizer Response Section**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-4 | id | a4 | must be DIG2 |
| 6-7 | snum | i2 | stage sequence number |
| 9-23 | sensitivity | e15.8 | sensitivity (counts/input unit) |
| 25-35 | samprat | f11.5 | digitizer sample rate (Hz) |
| 37-61 | descrip | a25 | description |

## Finite Impulse Response

The finite impulse response section is used to describe the response of digital filters, as shown in Table 18.

Table 18. **Format for Finite Impulse Response Section**

| Position | Name | Format | Description |
|---|---|---|---|
| Header | | | |
| 1-4 | id | a4 | must be FIR2 |
| 6-7 | snum | i2 | stage sequence number |
| 9-18 | gain | e10.2 | filter gain (relative factor, <u>not</u> in dB) |
| 20-23 | deci | i4 | decimation (blank if analog) |
| 25-32 | corr | f8.3 | group correction applied (seconds) |
| 34 | symflag* | a1 | symmetry flag (A=asymetric, B=symetric (odd), C=symetric (even)) |
| 36-39 | nfactor | i4 | number of factors |
| 41-65 | descrip | a25 | description |
| Data | | | |
| 2-16 | factor(i) | e15.8 | factor(i) |
| 18-32 | factor(i+1) | e15.8 | factor(i+1) |
| 34-48 | factor(i+2) | e15.8 | factor(i+2) |
| 50-64 | factor(i+3) | e15.8 | factor(i+3) |
| 66-80 | factor(i+4) | e15.8 | factor(i+4) |

* The symmetry code may be A (asymmetric filter, all coefficients are given; B (symmetric filter with an odd number of coefficients - the nfactor factors represent 2*nfactor-1 coefficients; and C (symmetric filter with an even number of coefficients - the nfactor factors represent 2*nfactor coefficients).

## Comments

Comments are enclosed in parentheses, as shown in Table 19.

Table 19. **Format of Response Comment Section**

| Position | Name | Format | Description |
|---|---|---|---|
| 2 | | a1 | ( |
| 3-n | comment | a<n-1> | comment |
| n+1 | | a1 | ) |

## Sample Response Section

*Example 4.8 - 1*

```
CAL2 MIAR  BHZ US   CMG-3N  4.11000000E+00  16.000   40.00000 1992/09/23 20:00
    (USNSN station at Mount Ida, Arkansas, USA)
PAZ2  1 V  7.29000000E+04   1.000         6   3       CMG-3 (NSN) Acc-Vel (Std)
 -3.14000000E-02  3.14000000E-04
 -1.97000000E-01  1.97000000E-03
 -2.01000000E+02  2.01000000E+00
 -6.97000000E+02  6.97000000E+00
 -7.54000000E+02  7.54000000E+00
 -1.05000000E+03  1.05000000E+01
  0.00000000E+00  0.00000000E+00
  0.00000000E+00  0.00000000E+00
  0.00000000E+00  0.00000000E+00
    (Theoretical response provided by Guralp Systems, Ltd.)
DIG2  2  4.18000000E+05  5120.00000                   Quanterra QX80
FIR2  3   1.00E+00   16    0.006 C   32          QDP380/900616 stage 1
 -1.11328112e-03 -1.00800209e-03 -1.35286082e-03 -1.73045369e-03 -2.08418001e-03
 -2.38537718e-03 -2.60955630e-03 -2.73352256e-03 -2.73316190e-03 -2.58472445e-03
 -2.26411712e-03 -1.74846814e-03 -1.01403310e-03 -3.51681737e-05  1.23782025e-03
  3.15983174e-03  6.99944980e-03  9.09959897e-03  1.25423642e-02  1.63123012e-02
  2.02632397e-02  2.43172608e-02  2.84051094e-02  3.24604138e-02  3.64142842e-02
  4.01987396e-02  4.37450483e-02  4.69873249e-02  4.98572923e-02  5.22795729e-02
  5.41139580e-02  5.43902851e-02
FIR2  4   1.00E+00    4    0.077 C   36          QDP380/900616 stage 2
  1.50487336e-04  3.05924157e-04  4.42948687e-04  3.87117383e-04 -4.73786931e-05
 -9.70771827e-04 -2.30317097e-03 -3.70637676e-03 -4.62504662e-03 -4.46480140e-03
 -2.86984467e-03  7.00860891e-06  3.38519946e-03  6.00352836e-03  6.55093602e-03
  4.25995188e-03 -5.76023943e-04 -6.43416447e-03 -1.09213749e-02 -1.16364118e-02
 -7.26515194e-03  1.53727445e-03  1.19331051e-02  1.96156967e-02  2.03516278e-02
  1.18680289e-02 -4.64369030e-03 -2.41125356e-02 -3.86382937e-02 -3.98499220e-02
 -2.18683947e-02  1.61612257e-02  6.89623653e-02  1.26003325e-01  1.74229354e-01
  2.01834172e-01
FIR2  5   1.00E+00    2    0.379 C   32          QDP380/900616 stage 3,4,5
  2.88049545e-04  1.55313976e-03  2.98230513e-03  2.51714466e-03 -5.02926821e-04
 -2.81205843e-03 -8.08708369e-04  3.21542984e-03  2.71266000e-03 -2.91550322e-03
 -5.09429071e-03  1.33933034e-03  7.40034366e-03  1.82796526e-03 -8.81958286e-03
 -6.56719319e-03  8.38608573e-03  1.24268681e-02 -5.12978853e-03 -1.84868593e-02
 -1.79236766e-03  2.33604181e-02  1.30477296e-02 -2.51709446e-02 -2.93134767e-02
  2.12669298e-02  5.21898977e-02 -6.61517353e-03 -8.83535221e-02 -3.66062373e-02
  1.86273292e-01  4.03764486e-01
    (Theoretical response provided by Quanterra, Inc.)
```

## *4.9* OUTAGE REPORTS

Outage information is reported using the formats in Table 35.

Table 20. **Report Period**

| Position | Name | Format | Description |
|---|---|---|---|
| | | Report Period | |
| 1-18 | “Report period from” | a18 | Text |
| 20-29 | Start_date | i4,a1,i2,a1,i2 | Date (yyyy/mm/dd) |

*Example 4.9 - 1* Outage data

```
DATA_TYPE outage GSE2.0
Report period from 1994/12/24 00:00:00.0 to 1994/12/25 12:00:00.00
Sta  Chan Aux      Start Date Time          End Date Time        Duration Comment
APL  SHZ  1234 1994/12/24 08:13:05.000 1994/12/24 08:14:10.000     65.000
APL  SHN  1234 1994/12/25 10:00:00.000 1994/12/25 10:00:00.030      0.030
```

## *4.10* DETECTIONS

Detection information describes the features of a detection, including station, detection time, amplitude, period, polarization properties signal-to-noise ratio, and detection quality.

Detection character is based on the estimated uncertainty of the onset time. Table 22 gives the guidelines for assigning a detection character based on the estimated uncertainty in the table.

Table 21. **Format for Detection Information**

| Position | Name | Format | Description |
|---|---|---|---|
| Header | | | |
| 1-3 | "Sta" | a3 | Text |
| 6-9 | "Chan" | a4 | Text |
| 11-13 | "Aux" | a3 | Text |
| 19-22 | "Date" | a4 | Text |
| 30-33 | "Time" | a4 | Text |
| 39-42 | "Azim" | a4 | Text |
| 45-48 | "Slow" | a4 | Text |
| 52-54 | "SNR" | a3 | Text |
| 62-64 | "Amp" | a3 | Text |
| 68-70 | "Per" | a3 | Text |
| 72-75 | "Qual" | a4 | Text |
| 82-83 | "Det_ID" | a2 | Text |
| Data | | | |
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 7-9 | Chan | a3 | FDSN channel code |
| 11-14 | Aux | a4 | auxiliary identification code |
| 16-25 | Date | i4,a1,i2,a1,i2 | date (yyyy/mm/dd) |
| 27-36 | Time | i2,a1,i2,a1,f4.1 | time (hh:mm:ss.s) |
| 38-42 | Azim | f5.1 | azimuth detection (degrees) |
| 44-48 | Slow | f5.1 | slowness detection (seconds/degree) |
| 50-54 | Snr | f5.1 | signal-to-noise ratio |
| 56-64 | Amp | f9.1 | amplitude (nm) |
| 66-70 | Per | f5.2 | period (s) |
| 72 | picktype | a1 | type of pick (a=automatic, m=manual) |
| 73 | direction | a1 | direction of short period motion (c=compression, d=dilatation, or blank) |
| 74 | detchar | a1 | detection character (i=impulsive, e=emergent, q=questionable, or blank) |
| 76-83 | ID | a8 | detection identification string |

Table 22. **Detection Character from Estimated Uncertainty in Onset Time**

| detchar | uncertainty for local phases | uncertainty for regional/teleseismic phases |
|---|---|---|
| i | < 0.05 sec | < 0.2 sec |
| e | < 0.25 sec | < 1.0 sec |
| q | > 0.25 sec | > 1.0 sec |

*Example 4.10 - 1* Detections

```
DATA_TYPE DETECTION GSE2.0
Sta  Chan Aux     Date       Time    Azim  Slow   SNR       Amp   Per Qual   ID
GEC2  shz      1994/02/13 00:40:19.3 123.0  30.0   4.4      14.7  0.90 aci 12345678
GEC2  shz      1994/02/13 00:40:27.9  60.8  17.5  12.2     123.0  0.41 ade 98736409
GEC2  shn      1994/02/13 00:40:32.1  65.1  16.3   7.5      24.0  0.42 a e 67230945
GEC2  shz      1994/02/13 00:40:38.6  61.0  15.5   4.4      19.0  0.33 a e 09834598
GEC2  shz      1994/02/13 00:41:15.8  64.5  26.9   2.2      11.7  0.64 a i 90845343
GEC2  she      1994/02/13 00:41:24.7  65.6  25.3   6.9      17.0  0.65 a e 12346876
NRA0  shz      1994/02/13 00:42:06.2 159.7  11.4   4.1       3.0  0.36 aci 87112345
ARA0  shz      1994/02/13 00:43:47.5 257.8   1.6   9.4       4.8  0.37 a e 34502364
```

## *4.11* ARRIVALS

Arrivals are detections that have been associated with an origin or event. The format for arrival data in GSE2.0 format, as shown in Table 23, is the same as the arrival format for a bulletin.

Table 23. **Arrival Lines**

| Position | Name | Format | Description |
|---|---|---|---|
| | | Header | |
| 1-3 | “Sta” | a3 | Text |
| 9-12 | “Dist” | a4 | Text |
| 15-18 | “EvAz” | a4 | Text |
| 24-28 | “Phase” | a5 | Text |
| 35-38 | “Date” | a4 | Text |
| 46-49 | “Time” | a4 | Text |
| 55-58 | “TRes” | a4 | Text |
| 61-64 | “Azim” | a4 | Text |
| 67-71 | “AzRes” | a5 | Text |
| 74-77 | “Slow” | a4 | Text |
| 80-83 | “SRes” | a4 | Text |
| 85-87 | “Def” | a3 | Text |
| 91-93 | “SNR” | a3 | Text |
| 101-103 | “Amp” | a3 | Text |
| 107-109 | “Per” | a3 | Text |
| 113-116 | “Mag1” | a4 | Text |
| 120-123 | “Mag2” | a4 | Text |
| 131-132 | “ID” | a2 | Text |
| | | Data | |
| 1-5 | Sta | a5 | station code |
| 7-12 | Dist | f6.2 | station to event distance (degrees) |
| 14-18 | EvAz | f5.1 | event to station azimuth (degrees) |
| 20 | picktype | a1 | type of pick (a=automatic, m=manual) |
| 21 | direction | a1 | direction of short period motion (c=compression, d=dilatation, or blank) |
| 22 | detchar | a1 | detection character (i=impulsive, e=emergent, q=questionable, or blank) |
| 24-30 | Phase | a7 | ISC phase code (P, S, pP, etc.) |
| 32-41 | Date | i4,a1,i2,a1,i2 | arrival date (yyyy/mm/dd) |
| 43-52 | Time | i2,a1,i2,a1,f4.1 | arrival time (hh:mm:ss.s) |
| 54-58 | TRes | f5.1 | time residual (seconds) |
| 60-64 | Azim | f5.1 | arrival azimuth (degrees) |
| 66-71 | AzRes | f6.1 | azimuth residual (degrees) |
| 73-77 | Slow | f5.1 | arrival slowness (seconds/degree) |
| 79-83 | SRes | f5.1 | slowness residual (seconds/degree) |
| 85 | tdef | a1 | time defining flag (T or blank) |
| 86 | adef | a1 | azimuth defining flag (A or blank) |
| 87 | sdef | a1 | slowness defining flag (S or blank) |
| 89-93 | Snr | f5.1 | signal-to-noise ratio |
| 95-103 | Amp | f9.1 | amplitude (nanometers) |
| 105-109 | Per | f5.2 | period (seconds) |
| 111-112 | mdef1 | a2 | magnitude type (mb, Ms, ML, MD=duration, Mn=Nuttli, M) |

Table 23. **Arrival Lines**

| | | | |
|---|---|---|---|
| 113-116 | Mag1 | f4.1 | magnitude |
| 118-119 | mdef2 | a2 | magnitude type (mb, Ms, ML, MD=duration, Mn=Nuttli, M) |
| 120-123 | Mag2 | f4.1 | magnitude |
| 125-132 | ID | a8 | Arrival ID |

*Example 4.11 - 1* Arrivals

```
DATA_TYPE ARRIVAL GSE2.0
Sta     Dist  EvAz     Phase      Date       Time     TRes  Azim  AzRes  Slow  SRes Def  SNR       Amp   Per   Mag1   Mag2       ID
GERES  10.56 150.3      P      1995/01/16 07:29:20.7  -0.2 163.7   13.4  13.8   0.1 T     6.8       0.6   0.3 ML 4.0          3586432
GERES  10.56 150.3      S      1995/01/16 07:31:17.5  -0.6 153.4    3.1  23.4  -1.0 T     4.9       2.9   0.6                 3586513
NORES  22.02 161.4      P      1995/01/16 07:31:41.2   0.3 155.0   -6.4  11.5   0.7 T     9.9       3.5   0.3                 3586453
FINES  22.29 191.6      P      1995/01/16 07:31:44.1   0.2 182.0   -9.6   8.6  -2.2 T     7.3       4.5   0.8 mb 3.7          3586555
ARCES  30.27 187.8      P      1995/01/16 07:32:57.8   1.2 191.3    3.5  10.7   1.9 T     7.7       1.2   0.6 mb 3.7          3586456
MBC    61.77  34.6      P      1995/01/16 07:37:03.8   0.5   5.9  -28.6   4.4  -2.3 T     4.6       0.3   0.4 mb 3.3          3586481
FCC    68.12  49.4      P      1995/01/16 07:37:45.3   0.4                          T                                         3604094
YKA    72.17  35.1      P      1995/01/16 07:38:09.5  -0.1                          T                                         3604095
WHY    78.21  19.3      P      1995/01/16 07:38:44.0  -0.5                          T                                         3604093
```

## 4.12 ORIGINS

The origin format is used in GSETT-3 to send gamma informatioun to the IDC. The format for origin data in GSE2.0 format is given in Table 24, Table 25, and Table 26. Blank lines are inserted between each pair of origin lines, as well as between the header lines and the origin lines. One or more comments about a particular origin are included within parentheses after the two origin lines. Only one header line is necessary for origin data.

Table 24. **Origin Header Lines**

| Position | Text | Format | Description |
|---|---|---|---|
| Header Line 1 | | | |
| 4-7 | "Date" | a4 | Text |
| 15-18 | "Time" | a4 | Text |
| 26-33 | "Latitude" | a8 | Text |
| 35-43 | "Longitude" | a9 | Text |
| 48-52 | "Depth" | a5 | Text |
| 57-60 | "Ndef" | a4 | Text |
| 62-65 | "Nsta" | a4 | Text |
| 67-69 | "Gap" | a3 | Text |
| 74-77 | "Mag1" | a4 | Text |
| 80 | "N" | a1 | Text |
| 85-88 | "Mag2" | a4 | Text |
| 91 | "N" | a1 | Text |
| 96-99 | "Mag3" | a4 | Text |
| 102 | "N" | a1 | Text |
| 105-110 | "Author" | a6 | Text |
| 121-122 | "ID" | a2 | Text |
| Header Line 2 | | | |
| 8-10 | "rms" | a3 | Text |
| 14-21 | "OT_Error" | a10 | Text |
| 28-33 | "Smajor" | a6 | Text |
| 35-40 | "Sminor" | a6 | Text |
| 42-43 | "Az" | a2 | Text |
| 52-54 | "Err" | a3 | Text |
| 58-62 | "mdist" | a5 | Text |
| 65-69 | "Mdist" | a5 | Text |
| 75-77 | "Err" | a3 | Text |
| 86-88 | "Err" | a3 | Text |
| 97-99 | "Err" | a3 | Text |
| 105-111 | "Quality" | a7 | Text |

Table 25. **Data Line of an Origin**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| Data Line 1 | | | |
| 1-10 | Date | i4,a1,i2,a1,i2 | date (yyyy/mm/dd) |
| 12-21 | Time | i2,a1,i2,a1,f4.1 | time (hh:mm:ss.s) |
| 23 | fixf | a1 | fixed flag (f=fixed origin time solution, or blank) |
| 26-33 | Latitude | f8.4 | latitude (minus for South) |
| 35-43 | Longitude | f9.4 | longitude (minus for West) |
| 45 | fixf | a1 | fixed flag (f= fixed epicenter solution, or blank) |
| 48-52 | Depth | f5.1 | depth (km) |
| 54 | fixf | a1 | fixed flag (f= fixed depth station, d=depth phases, or blank) |
| 57-60 | Ndef | i4 | number of defining phases |
| 62-65 | Nsta | i4 | number of defining stations |
| 67-69 | Gap | i3 | gap in azimuth coverage (degrees) |
| 72-73 | magtype1 | a2 | magnitude type |
| 74-77 | Mag1 | f4.1 | magnitude value 1 |
| 79-80 | N | i2 | number of stations contributing to magnitude 1 |
| 83-84 | magtype2 | a2 | magnitude type |
| 85-88 | Mag2 | f4.1 | magnitude value 2 |
| 90-91 | N | i2 | number of stations contributing to magnitude 2 |
| 94-95 | magtype3 | a2 | magnitude type |
| 96-99 | Mag3 | f4.1 | magnitude value 3 |
| 101-102 | N | i2 | number of stations contributing to magnitude 3 |
| 105-112 | Author | a8 | e.g., ITA_NDC |
| 115-122 | ID | a8 | unique origin identification string |

Table 25. **Data Line of an Origin**

| Data Line 2 | | | |
|---|---|---|---|
| 6-10 | rms | f5.2 | root mean square of time residuals (seconds) |
| 14-15 | “+-” | a2 | text |
| 16-21 | OT_Error | f6.2 | origin time error (seconds) |
| 26-31 | Smajor | f6.1 | semi-major axis of 90% ellipse or its estimate. (km) |
| 33-38 | Sminor | f6.1 | semi-minor axis of 90% ellipse or its estimate. (km) |
| 41-43 | Az | i3 | strike (0 <= x <= 360) of error ellipse clockwise from North (degrees) |
| 48-49 | “+-” | a2 | Text |
| 50-54 | Err | f5.1 | depth error 90% (km) |
| 57-62 | mdist | f6.2 | distance to closest station (degrees) |
| 64-69 | Mdist | f6.2 | distance to furthest station (degrees) |
| 73-74 | “+-” | a2 | Text |
| 75-77 | Err | f3.1 | standard magnitude 1 error |
| 84-85 | “+-” | a2 | Text |
| 86-88 | Err | f3.1 | standard magnitude 2 error |
| 95-96 | “+-” | a2 | Text |
| 97-99 | Err | f3.1 | standard magnitude 3 error |
| 105 | antype | a1 | analysis type: (a=automatic, m=manual, g=guess) |
| 107 | loctype | a1 | location method: (i=inversion, p=pattern recognition, g=ground truth, o=other) |
| 109-110 | evtype | a2 | event type:<br>uk = unknown,<br>ke = known earthquake<br>se = suspected earthquake<br>kr = known rockburst<br>sr = suspected rockburst<br>ki = known induced event<br>si = suspected induced event<br>km = known mine expl.<br>sm = suspected mine expl.<br>kx = known experimental expl.<br>sx = suspected experimental expl.<br>kn = known nuclear expl.<br>sn = suspected nuclear explosion<br>ls = landslide |

Table 26. **Origin Comment Lines (with Any Origin Group)**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| 5 | “(” | a1 | Text |
| 6-M | comment | a(M-4) | Comment |
| M+1 | “(” | a1 | Text |

*Example 4.12 - 1* GSE2.0 Origins

```
ATA_TYPE ORIGIN GSE2.0
  Date       Time       Latitude Longitude    Depth    Ndef Nsta Gap    Mag1  N    Mag2  N    Mag3  N  Author          ID
       rms   OT_Error      Smajor Sminor Az        Err   mdist  Mdist     Err        Err        Err    Quality

995/01/16 07:26:52.4     39.4500   20.4400    66.8       9    8 322  mb 3.6  3  ML 4.0  1            GSE_IDC     282672
     0.53   +- 12.69      93.6   83.7   27    +- 83.8   10.56  78.21                                  m i

995/01/16 07:27:07.3     50.7700 -129.7600    36.7       7    7 252  mb 4.0  2                        GSE_IDC     281990
     0.79   +-  9.63     129.3   23.5   37    +- 60.1   10.32  25.90                                  m i

995/01/16 08:44:26.4     40.4400  139.7700   188.6      13   12 135  mb 3.6  5                        GSE_IDC     282516
     0.28   +-  3.61      48.0   30.4   30    +- 51.9    4.08  89.03                                  m i

995/01/16 09:31:04.3    -22.0900 -177.4500     0.0 f     3    3 343  mb 3.8  2                        GSE_IDC     282674
     0.30   +-126.09    2296.3  130.0   86              37.59  44.91                                  m i

995/01/16 09:55:05.9    -22.8700  170.0800     0.0 f     7    4 336  mb 4.5  2                        GSE_IDC     282425
     1.23   +-  3.24      87.3   57.9  166              26.78  43.63                                  m i

995/01/16 10:25:06.1     61.6200 -149.7600    56.3      20   14 117  mb 4.1  5                        GSE_IDC     281948
     1.30   +-  5.89      42.6   18.3   56    +- 45.1    7.27 164.93                                  m i

995/01/16 11:08:02.2      9.2200  125.9700     0.0 f    16   16 136  mb 4.4 10                        GSE_IDC     281977
     0.82   +-  0.56      39.4   12.7   84              30.12 145.36                                  m i
```

## *4.13* EVENTS

For any seismic event, there can be several origins derived from many different organizations or procedures. The format for events places these different origins into groups separated by origin headers just as they would appear in a bulletin. The difference between the event format and the bulletin format is that the arrival information is not included in the event data. As in the bulletin format, there is a title line at the beginning of the data section which must include the name of the bulletin that was used as the basis for associating the separate origin estimates, as shown in Table 27. For each event, there is an event identification string and the geographic region name given as shown in Table 28. Following the title, each event group has the following structure:

- 1 event identification
- origin header lines
- 1 blank line
- n origin data lines
- 1 blank line
- geographic region name
- 2 blank lines

The event data format should not be used to submit gamma information for GSETT-3; the origin format is the proper one to use.

Table 27. **Title Line**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-80 | Title | a80 | Bulletin title |

Table 28. **Geographic Region Name**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-80 | geogreg | a80 | geographic region |

*Example 4.13 - 1* Event Information

```
DATA_TYPE EVENT GSE2.0
Reviewed Event Bulletin (REB) of the GSE_IDC for January 16, 1995
EVENT 280435
   Date       Time       Latitude Longitude    Depth    Ndef Nsta Gap    Mag1  N    Mag2  N    Mag3  N  Author          ID
       rms   OT_Error      Smajor Sminor Az        Err   mdist  Mdist     Err        Err        Err     Quality

1995/01/16 07:26:52.4     39.4500   20.4400     66.8       9    8 322  mb 3.6  3  ML 4.0  1            GSE_IDC     282672
      0.53   +- 12.69      93.6   83.7   27    +- 83.8   10.56  78.21                                    m i

GREECE-ALBANIA BORDER REGION


EVENT 280436
   Date       Time       Latitude Longitude    Depth    Ndef Nsta Gap    Mag1  N    Mag2  N    Mag3  N  Author          ID
       rms   OT_Error      Smajor Sminor Az        Err   mdist  Mdist     Err        Err        Err     Quality

1995/01/16 07:27:07.3     50.7700 -129.7600     36.7       7    7 252  mb 4.0  2                       GSE_IDC     281990
      0.79   +-  9.63     129.3   23.5   37    +- 60.1   10.32  25.90                                    m i

VANCOUVER ISLAND REGION


EVENT 280499
   Date       Time       Latitude Longitude    Depth    Ndef Nsta Gap    Mag1  N    Mag2  N    Mag3  N  Author          ID
       rms   OT_Error      Smajor Sminor Az        Err   mdist  Mdist     Err        Err        Err     Quality

1995/01/16 08:44:26.4     40.4400  139.7700    188.6      13   12 135  mb 3.6  5                       GSE_IDC     282516
      0.28   +-  3.61      48.0   30.4   30    +- 51.9    4.08  89.03                                    m i

NEAR WEST COAST OF HONSHU, JAPAN
```

## *4.14* BULLETINS

A bulletin is simply a collection of event and arrival information spanning some time period. The format specifies the information needed for a single event and concatenates them with a title line to form a complete bulletin. Each event group has the following structure:

- 1 event identification line
- origin header lines
- 1 blank line
- n origin lines
- 1 blank line
- geographic region name
- arrival header lines
- n arrival lines
- 2 blank lines

The event and arrival formats are described in the previous sections.

The bulletin data format should not be used to submit gamma information for GSETT-3; the origin format is the proper one to use.

*Example 4.14 - 1* GSE Bulletin

```
DATA_TYPE BULLETIN GSE2.0
Reviewed Event Bulletin (REB) of the GSE_IDC for January 16, 1995
EVENT 280435
   Date       Time       Latitude Longitude    Depth    Ndef Nsta Gap    Mag1  N    Mag2  N    Mag3  N  Author          ID
       rms   OT_Error      Smajor Sminor Az        Err   mdist  Mdist     Err        Err        Err     Quality

1995/01/16 07:26:52.4     39.4500   20.4400     66.8       9    8 322  mb 3.6  3  ML 4.0  1            GSE_IDC     282672
      0.53   +- 12.69      93.6   83.7   27    +- 83.8   10.56  78.21                                    m i

GREECE-ALBANIA BORDER REGION
Sta    Dist  EvAz     Phase       Date      Time     TRes  Azim  AzRes  Slow  SRes Def  SNR        Amp   Per   Mag1   Mag2 Arr ID
GERES  10.56 150.3     P       1995/01/16 07:29:20.7  -0.2 163.7   13.4  13.8   0.1 T    6.8       0.6   0.3 ML 4.0        3586432
GERES  10.56 150.3     S       1995/01/16 07:31:17.5  -0.6 153.4    3.1  23.4  -1.0 T    4.9       2.9   0.6               3586513
NORES  22.02 161.4     P       1995/01/16 07:31:41.2   0.3 155.0   -6.4  11.5   0.7 T    9.9       3.5   0.3               3586453
FINES  22.29 191.6     P       1995/01/16 07:31:44.1   0.2 182.0   -9.6   8.6  -2.2 T    7.3       4.5   0.8 mb 3.7        3586555
ARCES  30.27 187.8     P       1995/01/16 07:32:57.8   1.2 191.3    3.5  10.7   1.9 T    7.7       1.2   0.6 mb 3.7        3586456
MBC    61.77  34.6     P       1995/01/16 07:37:03.8   0.5   5.9  -28.6   4.4  -2.3 T    4.6       0.3   0.4 mb 3.3        3586481
FCC    68.12  49.4     P       1995/01/16 07:37:45.3   0.4                          T                                      3604094
YKA    72.17  35.1     P       1995/01/16 07:38:09.5  -0.1                          T                                      3604095
WHY    78.21  19.3     P       1995/01/16 07:38:44.0  -0.5                          T                                      3604093


EVENT 280436
   Date       Time       Latitude Longitude    Depth    Ndef Nsta Gap    Mag1  N    Mag2  N    Mag3  N  Author          ID
       rms   OT_Error      Smajor Sminor Az        Err   mdist  Mdist     Err        Err        Err     Quality

1995/01/16 07:27:07.3     50.7700 -129.7600     36.7       7    7 252  mb 4.0  2                        GSE_IDC     281990
      0.79   +-  9.63     129.3   23.5   37    +- 60.1   10.32  25.90                                    m i

VANCOUVER ISLAND REGION
Sta    Dist  EvAz     Phase       Date      Time     TRes  Azim  AzRes  Slow  SRes Def  SNR        Amp   Per   Mag1   Mag2 Arr ID
WHY    10.32 161.5     Pn      1995/01/16 07:29:33.7   0.8 161.4   -0.2  13.7  -0.0 T    5.9      52.9   0.4               3586419
WALA   10.37 285.5     Pn      1995/01/16 07:29:34.0   0.4 262.7  -22.8   9.5  -4.2 T    7.6      11.2   0.5               3586401
YKA    14.35 222.0     Pn      1995/01/16 07:30:26.6  -1.3 223.3    1.2  11.3  -2.3 T    6.9       1.4   1.0               3586540
INK    17.69 172.1     Pn      1995/01/16 07:31:10.7   0.0                          T                                      3604812
ULM    21.45 284.5     P       1995/01/16 07:31:51.1  -1.0 287.5    3.0  10.8   0.1 T   15.0      15.7   0.8 mb 4.3        3586452
FCC    21.85 264.2     P       1995/01/16 07:31:56.6   0.3                          T                                      3604813
MBC    25.90 195.2     P       1995/01/16 07:32:34.5  -0.8 217.6   22.4   4.6  -4.5 T    4.4       1.7   1.0 mb 3.6        3586477
...
```

## 4.15 COMMENTS

The first line of data type comment provides a mechanism for associating the comment to a station, arrival, origin, event, etc. If no association is needed, then this line may be left blank. The comment is written in free format and may be up to 1024 characters long. Comments adhere to the line-length conventions (1024 max; continuations marked with backslash; physical lines may not exceed the LINE_LEN environment). See Table 29.

Table 29. **Comment Format**

| Position | Name | Format | Description |
|---|---|---|---|
| Header | | | |
| 1-10 | idtype | a10 | identification type (Station, Arrival, Origin, Event) |
| 12-19 | id | a8 | identification string of the idtype |
| Data | | | |
| 2-1024 | comment | a1023 | Free format comment |

O P E R A T I O N S

*Example 4.15 - 1* Comments

```
DATA_TYPE COMMENT GSE2.0

  Almost anything may be typed into the space between the
  DATA_TYPE line and the STOP line.  No association was
  desired for this comment, so the association line was
  left blank.  Note that this comment is indented so that
  the DATA_TYPE in the second line of this paragraph is
  not interpreted as a command line.
DATA_TYPE COMMENT GSE2.0
Event     7687234
  The referenced event was felt over a wide area (300 square
  kilometers) near the epicenter.
```

## 4.16 COMMUNICATIONS STATUS REPORTS

Communications status is given over the time interval specified in the TIME or FREQ environments for AutoDRM or subscription requests, respectively. The report is comprised of a line giving the report period; a summary section in which each link is described with statistics of link performance for the reporting period; and finally a list of the link outages for each link. The link outages are reported only in the extended format (COMM_E); if a summary report is requested (COMM), these will not be included. The link performance statistics contain a description of the link (nominal link speed in kilobits per second, kbps), the mode of the link (full or half duplex), the percent of time that the link was operational, and the link utilization (1.0 is full utilization) in each direction. See Table 30, Table 31, and Table 32.

Table 30. **Report Period**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| 1-18 | “Report period from” | a18 | Text |
| 20-29 | Start_date | i4,a1,i2,a1,i2 | Start date (yyyy/mm/dd) |
| 31-40 | Start_time | i2,a1,i2,a1,f4.1 | Start time (hh:mm:ss.s) |
| 42-48 | “through” | a7 | Text |
| 50-59 | End_date | i4,a1,i2,a1,i2 | End date (yyyy/mm/dd) |
| 61-70 | End_time | i2,a1,i2,a1,f4.1 | End time (hh:mm:ss.s) |

Table 31. **Communications Status Format**

| Position | Text | Format | Description |
|---|---|---|---|
| | | Header | |
| 1-4 | “Link” | a4 | Text |
| 22-33 | “Nominal_kbps” | a12 | Text |
| 36-39 | “Mode” | a4 | Text |
| 43-46 | “%_up” | a4 | Text |
| 49-57 | “From” | a4 | Text |
| 60-71 | “Util” | a4 | Text |
| 49-57 | “From” | a4 | Text |
| 60-71 | “Util” | a4 | Text |
| | | Data | |
| 1-8 | link | a8 | Link code (farthest from IDC) |
| 10 | “-” | a1 | Text |
| 12-19 | link | a8 | Link code (closest to IDC) |
| 22-27 | speed | f6.1 | Nominal speed of link in kbps |
| 30-33 | mode | a4 | “full” for full-duplex or “half” for half-duplex |
| 35-39 | uptime | f5.1 | Percent uptime |
| 42-49 | link | a8 | Link Code (farthest from IDC) |
| 51-54 | utilization | f4.2 | Utilization of link (dat_rate/speed) |
| 57-64 | link | a8 | Link Code (closest to IDC) |
| 66-69 | utilization | f4.2 | Utilization of link (dat_rate/speed) |

Table 32. **Communications Outage Format**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| | | Link Identification | |
| 1-8 | link | a8 | Link code (farthest from IDC) |
| 10 | “-” | a1 | Text |
| 12-19 | link | a8 | Link code (closest to IDC) |
| 21-32 | “link outages” | a12 | Text |
| | | Header | |
| 10-13 | “From” | a4 | Text |
| 30-36 | “Through” | a7 | Text |
| 50-57 | “Duration” | a8 | Text |
| | | Data | |
| 1-10 | date | i4,a1,i2,a1,i2 | Date of beginning of outage (yyyy/mm/dd) |
| 12-21 | time | i2,a1,i2,a1,f4.1 | Time of beginning of outage (hh:mm:ss.s) |
| 24-33 | date | i4,a1,i2,a1,i2 | Date of end of outage (yyyy/mm/dd) |
| 35-44 | time | i2,a1,i2,a1,f4.1 | Time of end of outage (hh:mm:ss.s) |
| 47-60 | duration | i3,a1,i2,a1,i2,a1,f4.1 | Duration of outage (ddd hh:mm:ss.s) |

*Example 4.16 - 1* Communications Status Report

```
DATA_TYPE COMM_STATUS GSE2.0
Report period from 1994/12/03 00:00:00.0 through 1994/12/04 00:00:00.0
Link               Nom_kbps  Mode  %_Up   From    Util   From    Util
AUS_NDC  - GSE_IDC     56.0  full  88.4  AUS_NDC  0.50  GSE_IDC  0.08
NOR_NDC  - GSE_IDC    128.0  full  99.2  NOR_NDC  0.77  GSE_IDC  0.10
USA_NDC  - GSE_IDC   1000.0  full 100.0  USA_NDC  0.25  GSE_IDC  0.25

AUS_NDC  - GSE_IDC  link outages
         From                Through              Duration
1994/12/02 20:23:14.0  1994/12/03 00:48:28.0  000 00:25:14.0
1994/12/03 02:34:31.0  1994/12/03 02:49:39.0  000 00:15:08.0
1994/12/03 19:02:27.0  1994/12/03 19:12:29.0  000 00:10:02.0

NOR_NDC  - GSE_IDC  link outages
         From                Through              Duration
1994/12/03 04:34:31.0  1994/12/03 06:35:39.0  000 00:45:13.0
```

## *4.17* STATION STATUS REPORTS

Station status is given over the time interval specified in the TIME or FREQ environments for AutoDRM or subscription requests, respectively. The report is comprised of statistics that can be used to evaluate the overall performance of one or more stations. The first line of the report gives the report period. The status lines give the station code and the nominal number of channels for the station. This is followed by the "Station Capability" in which station problems are grouped into four categories depending on the impact each failure has on the capability of that station. Station capability is assessed relative to the maximum performance of that particular station, with given instrument configuration and site characteristics. The station status does not assess the affect of a station problem on the performance of the monitoring network.

In the context of assessing station status, the station consists of the sensors, digitizers, communications within the site and data loggers. Station status is assessed at the IDC based on data that is available at the IDC, and may therefore include the effects of problems with long-haul communications and problems at a National Data Center or Data Relay Center. Moreover, because data may arrive late at the IDC, the station status assessment is a snapshot of station capability at a single time.

- Fully capable

  The system is operating and contributing data to the mission at the level for which it was designed.

- Partially capable

  The system is impaired and contributing significant data to the mission but of degraded quality, reduced quantity, or reduced operational capability.

- Low capability

  The system is severely impaired and is contributing data that do not meet minimum requirements for the designed mission but are still useful for the global monitoring network.

- Not capable

  The system completely inoperative or the data being contributed are not useful for the global monitoring network in any way.

For arrays, capability is estimated based on the theoretical array gain for the available channels relative to maximum array gain with all channels operational. The array gain is estimated from the square root of the number of channels; the geometry of the active channels and the relative values of individual array elements is neglected.

Station mission capability may be estimated based on data available at the IDC. Non-station problems, such as outages of long-haul or tail communication circuits and problems with forwarding the data from NDC's will be folded into the capability estimates. Problems affecting the quality or timing of seismic waveforms will not be included in the automated station capability estimate, at least in the first instance, and thus capability may be overestimated. See Table 33.

Table 33. **Station Capability Criteria**

| Station Type | Fully Capable | Partial Capability | Low Capability | Non-Capable |
|---|---|---|---|---|
| SP or HF array | array gain ≥ 90% max | 70% ≤ array gain < 90% max | array gain < 70% max, at least one channel operational | no channels operational |
| 3-C BB station | all channels operational | one vertical & one horizontal operational | 1 channel operational | no channel operational |
| **Examples:** | **Fully Capable** | **Partial Capability** | **Low Capability** | **Non-Capable** |
| 25 element array | 21-25 | 13-20 | 1-12 | 0 |
| 19 element array | 16-19 | 10-15 | 1-9 | 0 |
| 16 element array | 13-16 | 8-12 | 1-7 | 0 |
| 9 element array | 8-9 | 5-7 | 1-4 | 0 |
| 7 element array | 6-7 | 4-5 | 1-3 | 0 |

Following the station capability entries is the maximum data time which is the cumulative amount of time for which data are expected for this station. For alpha, or primary stations, this will be the entire report period; for beta, or auxiliary stations, this will be the sum of the data segment time intervals requested. Availability indicates the percent of data that are available at the IDC relative to that expected. If an array with ten channels sends nine channels of data to the IDC for the entire period, then the data availability would be 90.0 (even though the data capability may be fully capable 100% of the time!) The median delay measures the time delay between ground motion and receipt of data at the IDC for alpha stations, and the delay between request and receipt for beta stations. Finally, the number of successful retrievals of data from the beta stations and the number of retrieval attempts are given. See Table 34 for the station status format.

*Example 4.17 - 1* Station Status Report

```
DATA_TYPE STA_STATUS GSE2.0
Report period from 1994/12/03 00:00:00.0 through 1994/12/04 00:00:00.0
              Station Capability
Sta   ch  Full   Part   Low   Non  Max_Exp_Time   Avail   Med_Delay Att Suc Pnd
ARCES 33  100     0     0      0   000 24:00:00   98.5  000:00:42.9
ABC    3   90     5     0      5   000 00:23:35  100.0  000:00:55.7   3   3   0
DEF    3   80    20     0      0   000 24:00:00   83.0  000:05:23.6
```

Table 34. **Station Status Format**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| Report Period | | | |
| 1-18 | "Report period from" | a18 | Text |
| 20-29 | Start_date | i4,a1,i2,a1,i2 | Start date (yyyy/mm/dd) |
| 31-40 | Start_time | i2,a1,i2,a1,f4.1 | Start time (hh:mm:ss.s) |
| 42-48 | "through" | a7 | Text |
| 50-59 | End_date | i4,a1,i2,a1,i2 | End date (yyyy/mm/dd) |
| 61-70 | End_time | i2,a1,i2,a1,f4.1 | End time (hh:mm:ss.s) |
| Header LIne 1 | | | |
| 12-25 | "Sta Capability" | a4 | Text |
| Header Line 2 | | | |
| 1-3 | "Sta" | a3 | Text |
| 7-8 | "Ch" | a2 | Text |
| 10-13 | "Full" | a4 | Text |
| 16-18 | "Part" | a4 | Text |
| 21-23 | "Low" | a3 | Text |
| 26-28 | "Non" | a3 | Text |
| 31-42 | "Max_Exp_Time" | a12 | Text |
| 45-49 | "Avail" | a5 | Text |
| 53-61 | "Med_Delay" | a9 | Text |
| 65-67 | "Suc" | a3 | Text |
| 69-71 | "Att" | a3 | Text |
| 73-75 | "Pnd" | a3 | Text |
| Data | | | |
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 7-8 | ch | i2 | Nominal number of channels |
| 11-13 | Full | f5.1 | Full station capability (% of report period) |
| 16-18 | Part | f5.1 | Partial station capability (% of report period) |
| 21-23 | Low | f5.1 | Low station capability (% of report period) |
| 26-28 | Non | f5.1 | Non-capable station (% of report period) |
| 31-42 | Max_Exp_Time | i3,a1,i2,a1,i2,a1,i2 | Maximum data time possible (ddd hh:mm:ss) |
| 45-49 | Avail | f5.1 | Percent of data available at the IDC |
| 52-62 | Med_Delay | i3,a1,i2,a1,f4.1 | Median delay of data from station to IDC |
| 65-67 | Att | i3 | IDC Attempts to retrieve data |
| 69-71 | Suc | i3 | Successful attempts to retrieve data from beta station |
| 73-75 | Pnd | i3 | IDC Pending Data retrievals |

Table 35. **Channel Status Format**

| Position | Name | Format | Description |
|---|---|---|---|
| | | Report Period | |
| 1-18 | “Report period from” | a18 | Text |
| 20-29 | “Start_date” | i4,a1,i2,a1,i2 | Date (yyyy/mm/dd) |
| 31-40 | “Start_time” | i2,a1,i2,a1,f4.1 | Time (hh:mm:ss.s) |
| 42-48 | “through” | a7 | Text |
| 50-59 | “End_date | i4,a1,i2,a1,i2 | Date (yyyy/mm/dd) |
| 61-70 | “End_time” | i2,a1,i2,a1,f4.1 | Time (hh:mm:ss.s) |
| | | Title | |
| 1-28 | “Data Availability Statistics” | a28 | Text |
| | | Header | |
| 1-3 | “Sta” | a3 | Text |
| 7-10 | “Chan” | a4 | Text |
| 12-14 | “Aux” | a3 | Text |
| 19-30 | “Max_Exp_Time” | a12 | Text |
| 34-40 | “%_Avail” | a6 | Text |
| 44-47 | “Gaps” | a4 | Text |
| 51-56 | “Median” | a6 | Text |
| 68-88 | “Min” | a3 | Text |
| 97-99 | “Max” | a3 | Text |
| | | Data | |
| 1-5 | “Sta” | a5 | International Station Code (ISC) |
| 8-10 | “Chan” | a3 | FDSN Channel Code |
| 12-15 | “Aux” | a4 | Auxiliary Identification Code |
| 19-30 | “Max_Exp_Time” | i3,a1,i2,a1,i2,a1,f 4.1 | Maximum data time possible (ddd hh:mm:ss.s) |
| 36-40 | “%_Avail” | f5.1 | Percent of data available at the IDC |
| 43-47 | “Gaps” | i5 | Number of data gaps |
| 50-58 | “Median” | 13,a1,i2,a1,i2 | Median length of data gaps (hhh:mm:ss) |
| 83-91 | “Min” | 13,a1,i2,a1,i2 | Minimum length of data gaps (hhh:mm:ss) |
| 94-102 | “Max” | 13,a1,i2,a1,i2 | Maximum length of data gaps (hhh:mm:ss) |

Table 36. **Data Timeliness Format**

| Position | Text | Format | Description |
|---|---|---|---|
| Title | | | |
| 1-26 | “Data Timeliness Statistics” | a26 | Text |
| Header | | | |
| 1-3 | “Sta” | a3 | Text |
| 7-10 | “Chan” | a4 | Text |
| 12-14 | “Aux” | a3 | Text |
| 18-30 | “Max_Data_Time” | a13 | Text |
| 34-45 | “Delay_Median” | a17 | Text |
| 50-53 | “Mean” | a4 | Text |
| 60-66 | “Std_Dev” | a7 | Text |
| 73-75 | “Min” | a4 | Text |
| 84-86 | “Max” | a7 | Text |
| Data | | | |
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 8-10 | Chan | a3 | Channel Code |
| 12-15 | Aux | a4 | Auxiliary Code |
| 17-30 | Max_Data_Time | i3,a1,i2,a1,i2,a1,f4.1 | Maximum data time possible (ddd hh:mm:ss.s) |
| 33-41 | Delay_Median | 13,a1,i2,a1,i2 | Median delay time (hhh:mm:ss) |
| 44-52 | Mean | 13,a1,i2,a1,i2 | Mean delay time (hhh:mm:ss) |
| 55-63 | Std_Dev | 13,a1,i2,a1,i2 | Standard deviation of delay time (hhh:mm:ss) |
| 66-74 | Min | 13,a1,i2,a1,i2 | Minimum delay time (hhh:mm:ss) |
| 77-85 | Max | 13,a1,i2,a1,i2 | Maximum delay time (hhh:mm:ss) |

*Example 4.18 - 1* Channel Status

```
DATA_TYPE CHAN_STATUS GSE2.0
Report period from 1994/12/03 00:00:00.0 through 1994/12/04 00:00:00.0
Data Availability Statistics
Sta   Chan Aux    Max_Exp_Time   %_Avail   Gaps   Median       Min        Max
OBN    bhz        000 00:05:00     100.0      0  000:00:00  000:00:00  000:00:00
OBN    bhn        000 00:05:00      99.9      6  000:00:10  000:00:00  000:00:24
OBN    bhe        000 00:05:00     100.0      0  000:00:00  000:00:00  000:00:00

ARU    bhz        000 01:23:14      99.9      8  000:00:07  000:00:00  000:00:12
ARU    bhn        000 01:23:14      99.9     12  000:00:10  000:00:00  000:00:12
ARU    bhe        000 01:23:14      99.9     12  000:00:10  000:00:00  000:00:12

KIV0   shz        000 01:53:48      99.7     59  000:00:17  000:00:06  000:16:06
KIV0   shn        000 01:53:48      99.7     79  000:00:00  000:00:06  000:16:06
KIV0   she        000 01:53:48      99.8     55  000:00:00  000:00:06  000:16:06
KIV1   shz        000 01:53:48      99.7     72  000:00:00  000:00:06  000:16:06
KIV2   shz        000 01:53:48      99.7     62  000:00:00  000:00:06  000:16:06
KIV3   shz        000 01:53:48      99.8     64  000:00:00  000:00:06  000:16:06

Data Timeliness Statistics
Sta  Chan Aux   Max_Exp_Time Delay_Med    Mean    Std_Dev     Min       Max
OBN   bhz       000 00:05:00 000:00:00 000:00:00 000:00:00 000:00:00 000:00:00
OBN   bhn       000 00:05:00 000:00:00 000:00:00 000:00:00 000:00:00 000:00:00
OBN   bhe       000 00:05:00 000:00:00 000:00:00 000:00:00 000:00:00 000:00:00

ARU   bhz       000 01:23:14 000:46:22 000:50:17 000:25:50 000:44:14 000:58:01
ARU   bhn       000 01:23:14 000:46:27 000:50:21 000:25:55 000:44:16 000:58:05
ARU   bhe       000 01:23:14 000:45:53 000:49:39 000:26:28 000:43:45 000:57:39

KIV0  shz       000 01:53:48 000:24:19 000:22:21 000:14:00 000:00:01 000:37:57
KIV0  shn       000 01:53:48 000:24:06 000:22:03 000:13:44 000:00:02 000:37:29
KIV0  she       000 01:53:48 000:23:40 000:21:54 000:13:51 000:00:01 000:37:44
KIV1  shz       000 01:53:48 000:23:54 000:21:47 000:13:52 000:00:10 000:37:17
KIV2  shz       000 01:53:48 000:23:16 000:21:52 000:13:59 000:00:10 000:37:37
KIV3  shz       000 01:53:48 000:23:59 000:21:59 000:14:00 000:00:06 000:37:39
```

## *4.19* AUTHENTICATION STATUS REPORTS

Some data channels in GSETT-3 will be contain authentication signatures which will be verified at the IDC. The authentication status report will provide statistics on the authentication process over the time of the report. The first section of the report gives, by station, the number of packets tested, the number that passed, and the number that failed. Below this, the failures are grouped as intervals for each data channel that failed to verify the authentication signature. See Table 37 and Table 38.

Table 37. **Report Period**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| Report Period | | | |
| 1-18 | “Report period from” | a18 | |
| 20-29 | date | i4,a1,i2,a1,i2 | Date (yyyy/mm/dd) |
| 31-40 | time | i2,a1,i2,a1,f4.1 | Time (hh:mm:ss.s) |
| 42-48 | “through” | a7 | |
| 50-59 | date | i4,a1,i2,a1,i2 | Date (yyyy/mm/dd) |
| 61-70 | time | i2,a1,i2,a1,f4.1 | Time (hh:mm:ss.s) |
| Header | | | |
| 1-3 | “Sta” | a3 | Text |
| 6-9 | “Chan” | a4 | Text |
| 11-13 | “Aux” | a3 | Text |
| 17-30 | “Packets_Tested” | a14 | Text |
| 33-46 | “Packets_Failed” | a14 | Text |
| Data | | | |
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 7-9 | Chan | a3 | FSDN Channel Code |
| 11-14 | Aux | a4 | Auxiliary Identification Code |
| 23-30 | Packets_Tested | i8 | Number of packets tested |
| 39-46 | Packets_Failed | i8 | Number of packets failing verification |

Table 38. **Authentication List Format**

| Position | Name | Format | Description |
|---|---|---|---|
| Title | | | |
| 1-23 | “Failed Packet Intervals” | a23 | Text |
| Header | | | |
| 1-3 | “Sta” | a3 | Text |
| 6-9 | “Chan” | a4 | Text |
| 11-13 | “Aux” | a3 | Text |
| 21-30 | “Start_Time” | a10 | Text |
| 45-51 | “End_Time” | a8 | Text |
| 61-67 | “Comment” | a7 | Text |
| Data | | | |
| 1-5 | sta | a5 | International Station Code (ISC) |
| 7-9 | chan | a3 | FDSN Channel Code |
| 11-14 | aux | a4 | Auxiliary Identification Code |
| 16-25 | s_date | i4,a1,i2,a1,i2 | Start date of failure interval (yyyy/mm/dd) |
| 27-36 | s_time | i2,a1,i2,a1,i2,a1,f4.1 | Start time of failure interval (hh:mm:ss.s) |
| 39-48 | e_date | i4,a1,i2,a1,i2,a1,i2,a1,i2,a1,i2,a1,f4.1 | End date of failure interval (yyyy/mm/dd) |
| 50-59 | e_time | i4,a1,i2,a1,i2,a1,i2,a1,i2,a1,i2,a1,f4.1 | End time of failure interval (hh:mm:ss.s) |
| 61-132 | comment | a72 | Comment |

*Example 4.19 - 1* Authentication Status

```
DATA_TYPE AUTH_STATUS GSE2.0
Report period from 1994/12/03 00:00:00.0 through 1994/12/04 00:00:00.0
Sta  Chan Aux   Packets_Tested  Packets_Failed
ABC   shz                 8640               3
DEF   bhz                 8640               0

Failed Packet Intervals
Sta  Chan Aux      Start_Time           End_Time         Comment
ABC   shz      1994/12/03 14:28:40  1994/12/03 14:29:10  Unknown cause
```

## 4.20 NDC EVALUATION REPORTS

Those NDC's which volunteer to provide statistics for evaluation purposes will be able to use an interactive program to produce the evaluation messages for them. Much of the evaluation information may be sent using the data types described above. The Outages data type is an excellent example of useful evaluation information that can be provided using other data types.

For NDC's who prefer to produce the evaluation messages independently, the following section describes the required format for the messages. The format is divided into 13 categories and the NDC's can choose which of these categories they include in the report, according to the data they have available. Categories 1-5 are basic statistics which are important to the evaluation process. The remaining categories present the results of various evaluation tasks which will be essential to the overall evaluation of GSETT-3.

### Message Header

An evaluation data type begins with an evaluation sequence number.

```
EVAL_NO NNNNNN
```

where NNNNNN is a unique message number in a sequence of evaluation messages from one NDC.

*Example 4.20 - 1* as follows:

```
BEGIN GSE2.0
MSG_TYPE DATA
MSG_ID 1994/05/20_005 GBR_NDC
DATA_TYPE EVALUATION GSE2.0
EVAL_NO 000001
```

### Communications Outages

Any failure or breakdown in any communications link used for data transmission should be noted so that the NDC records can be used to verify the IDC records of outages.

**Section start (line 1):**

One line containing the word **COMMSOUT** in columns 1-8

Communications Information (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDCOM** in columns 1-6 (See Table 39.)

Table 39. **Communications Outages**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| 1-7 | Location | a7 | Code for one end of communications link; e.g., NDC, STA, IDC, NODE |
| 10-16 | Location | a7 | Code for other end of communications link; e.g., NDC, IDC, NODE |
| 19-28 | Start date | i4,a1,i2,a1,i2 | Start date of communications breakdown (yyyy/mm/dd) |
| 30-39 | Start time | i2,a1,i2,a1,f4.1 | Start time of communications breakdown (hh:mm:ss.s) |
| 41-42 | "TO" | a2 | Text |
| 44-53 | End date | i4,a1,i2,a1,i2 | End date of communications breakdown (yyyy/mm/dd) |
| 55-64 | End time | i2,a1,i2,a1,f4.1 | End time of communications breakdown (hh:mm:ss.s) |
| 67- | Comment | a | Comment |

## Data Archive Outages

If there is a gap in the archive of data stored at the station or NDC (particularly in the case of beta station data) this should be recorded.

**Section start (line 1):**

One line containing the word **ARCHOUT** in columns 1-7

Data Archive Outage Information (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDARC** in columns 1-6 (See Table 40.)

Table 40. **Data Archive Outages**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| 1-5 | Station code | a5 | International Station Code (ISC) |
| 8-10 | Channel code | a3 | FDSN channel code |
| 13-22 | S_date | i4,a1,i2,a1,i2 | Start date (yyyy/mm/dd) |
| 24-33 | S_time | i2,a1,i2,a1,f4.1 | Start time (hh:mm:ss.s) |
| 35-36 | "TO" | a2 | Text |
| 38-47 | E_date | i4,a1,i2,a1,i2 | End date (yyyy/mm/dd) |
| 47-58 | E_time | i2,a1,i2,a1,f4.1 | End time (hh:mm:ss.s) |
| 61- | Comment | a | Comment - reason for outage; e.g.: computer failure, disc failure, communications failure, seismometer failure |

## NDC Detection System Outages

If NDC's are running signal detectors on their stations, it would be useful to have information regarding any outages in the detection systems.

**Section start (line 1):**

One line containing the word **DETOUT** in columns 1-6

Detection system outages (lines 2 through N) (See Table 41.)

**Section end (line N+1):**

One line containing the word **ENDDET** in columns 1-6

O P E R A T I O N S

Table 41. **NDC Detection System Outages**

| Position | Name/Text | Format | Description |
|---|---|---|---|
| 1-5 | Station code | a5 | International Station Code (ISC) |
| 8-10 | Channel code | a3 | FDSN channel code |
| 13-22 | S_date | i4,a1,i2,a1,i2 | Start date (yyyy/mm/dd) |
| 24-33 | S_time | i2,a1,i2,a1,f4.1 | Start time (hh:mm:ss.s) |
| 35-36 | "TO" | a2 | Text |
| 38-47 | E_date | i4,a1,i2,a1,i2 | End date (yyyy/mm/dd) |
| 49-58 | E_time | i2,a1,i2,a1,f4.1 | End time (hh:mm:ss.s) |
| 61- | Comment | a | Comment (Reason for outage) |

## NDC/IDC Detection Comparisons

It would be very useful to have an independent assessment of the signal detection systems used at the IDC, particularly those used on alpha station data. The format allows the NDC to specify a period of time over which the comparison of detections was made. In this way, the NDC can volunteer to do this evaluation without being committed to running their own detector permanently.

**Section start (line 1):**

One line containing the word **DETCOMP** in columns 1-7 followed by a start date/time and end date/time indicating the period of time when the comparison was made.

DETCOMP yyyy/mm/dd hh:mm:ss.D TO yyyy/mm/dd hh:mm:ss.D

Detection Comparison Results (line 2 through N)

**Section end (line N+1):**

One line containing the word **ENDCOMP** in columns 1-8

Detection Comparison Results

(See Table 42.)

Table 42. **Detection Comparison Results**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 8-10 | Chan | a3 | FDSN Channel code |
| 13-22 | Date | i4,a1,i2,a1,i2 | Date (yyyy/mm/dd) |
| 25-27 | NDC_Detects | i3 | Number of signals detected by NDC |
| 30-32 | IDC_Detects | i3 | Number of signals detected by IDC |
| 35-37 | Matching | i3 | Number of matching detections (arrival times should be within three seconds for a 'match') |
| 40- | Comment | a | Comment |

## Comparison of Extracted Parameters with IDC

If the NDC is running a detector, it may be possible for it to extract the basic parameters from each detected signal and compare the actual measurements to those produced by the IDC. This format allows the NDC to specify a certain period of time over which this comparison was made.

**Section start (line 1):**

One line containing the word **MEASURE** in columns 1-7 followed by a start date and end date.

MEASURE yyyy/mm/dd TO yyyy/mm/dd

Comparison of measurements (lines 2 through N):

**Section end (line N+1):**

One line containing the word **ENDMEAS** in columns 1-7 (See Table 43.)

Table 43. **Automated Parameter Extraction Comparisons**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 8-10 | Chan | a3 | FDSN Channel code |
| 13-22 | ev_date | i4,a1,i2,a1,i2 | Event date (yyyy/mm/dd) |
| 24-33 | ev_time | i2,a1,i2,a1,f4.1 | Event time (hh:mm:ss.s) |
| 36-42 | t_diff | i2,a1,f4.1 | Difference in arrival times picked at NDC compared to IDC (mm:ss.s) |
| 45-53 | amp rat | f9.4 | Ratio of displacement amplitudes picked by NDC compared to IDC (nm) |
| 56- | Comment | a | Comment |

## Data Requests Submitted by NDC

In order to assess whether the IDC is providing the data required by the NDC's for their verification purposes, it would be useful to know how many requests for data the NDC's have submitted and whether they received any response from the IDC. In the case where a large file is prepared by the IDC and stored ready for the requestor to retrieve, the response is taken to be the notification from the IDC that the file is ready for retrieval. If the IDC fails to meet the request for subscription items, the "comment" field should be used to draw attention to this omission.

**Section start (line 1):**

One line containing the word **REQSUB** in columns 1-6

NDC Requests submitted (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDSUB** in columns 1-6 (See Table 44.)

Table 44. **Data Requests Submitted by NDC**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-12 | d_type | a12 | Refer to Table 5 for allowable requests |
| 15-24 | d_date | i4,a1,i2,a1,i2 | Date of data requested (yyyy/mm/dd) |
| 27-36 | r_date | i4,a1,i2,a1,i2 | Date request was sent (yyyy/mm/dd) |
| 38-47 | r_time | i2,a1,i2,a1,f4.1 | Time request was sent (hh:mm:ss.s) |
| 50-69 | msg_id | a20 | Unique identification number of request message |
| 72 | Response | a1 | IDC Response: yes (y) or no (n) |
| 75-84 | res_date | i4,a1,i2,a1,i2 | Date response received (yyyy/mm/dd) |
| 86-95 | res_time | i2,a1,i2,a1,f4.1 | Time response received (hh:mm:ss.s) |
| 98- | Comment | a | Comment |

## Data Requests Received by NDC

In order to assess the performance of NDC's, it is necessary to know what data requests they received (from the IDC) and whether they were able to respond to the request in a timely manner. If possible, the NDC's should distinguish between routine beta data requests and requests forwarded by the IDC on behalf of other NDC's.

**Section start (line 1):**

One line containing the word **REQREC** in columns 1-6

NDC Requests Received (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDREC** in columns 1-6 (See Table 45.)

Table 45. **Data Requests Received by NDCs**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-12 | dat_type | a12 | Refer to Table 5 for allowable requests |
| 15-24 | d_date | i4,a1,i2,a1,i2 | Date of data requested (yyyy/mm/dd) |
| 27-36 | r_date | i4,a1,i2,a1,i2 | Date request was received (yyyy/mm/dd) |
| 38-47 | r_time | i2,a1,i2,a1,f4.1 | Time request was received (hh:mm:ss.s) |
| 50-69 | msg_id | a20 | Unique identification number of request message |
| 72 | Response | a1 | Response sent from NDC: yes (y) or no (n) |
| 75-84 | res_date | i4,a1,i2,a1,i2 | Date response was sent (yyyy/mm/dd) |
| 86-95 | res_time | i2,a1,i2,a1,f4.1 | Time response was sent (hh:mm:ss.s) |
| 98-102 | r_type | a5 | Routine IDC Request (BETA) or NDC Request (NDC) |
| 105- | Comment | a | Comment |

## Gamma Review of REB Quality

Participants in GSETT-3 are asked to review REB events located within their States' territory and use solutions from Gamma networks to assess the accuracy of the REB solutions. The Evaluation Working Group has produced some guidelines on how to perform this comparison (refer to Volume Four Annex One) and the results can be reported using the following format. The format allows the NDC to give a specific time period over which the review of the REB's was made.

**Section start (line 1):**

One line containing the word **GAMREV** followed by the start date and end date of the time period over which the REB's were reviewed.

GAMREV yyyy/mm/dd TO yyyy/mm/dd

Review of REB's using Gamma data (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDGAM** in columns 1-6 (See Table 46, where M = magnitude.)

Table 46. **GAMMA Review of REB Quality**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-10 | Date | i4,a1,i2,a1,i2 | Date of bulletin under review (yyyy/mm/dd) |
| 13-15 | tot_gam | i3 | Total number of gamma events reported for that day |
| 18-20 | tot_REB | i3 | Total number of REB events for that day |
| 23-25 | n_gam1 | i3 | Number of reported gamma events with 2.5 < M < 3.0 |
| 28-30 | n_reb1 | i3 | Number of REB events with 2.5 < M < 3.0 |
| 33-35 | n_match1 | i3 | Number of matching events with 2.5 < M < 3.0 |
| 38-40 | n_gam2 | i3 | Number of reported gamma events with 3.1 < M < 3.5 |
| 43-45 | n_reb2 | i3 | Number of REB events with 3.1 < M < 3.5 |
| 48-50 | n_match2 | i3 | Number of matching events with 3.1 < M < 3.5 |
| 53-55 | n_gam3 | i3 | Number of reported gamma events with 3.6 < M < 4.0 |
| 58-60 | n_reb3 | i3 | Number of REB events with 3.6 < M < 4.0 |
| 63-65 | n_match3 | i3 | Number of matching events with 3.6 < M < 4.0 |
| 68-70 | n_gam3 | i3 | Number of reported gamma events with M > 4.0 |
| 73-75 | n_reb3 | i3 | Number of REB events with m > 4.0 |
| 78-80 | n_match3 | i3 | Number of REB events with m > 4.0 |
| 83- | Comment | a | Comment - general comments on comparison of IDC and gamma solutions |

## Surface Wave Detection

Since the comparison of body and surface wave magnitudes is regarded as a major discrimination technique, it is very important to assess how well the IDC is detecting surface waves from LP data. This section allows NDC's to give a specific time period over which they have compared the IDC surface wave detections with their own automatic or manual surface wave detections.

**Section start (line 1):**

One line containing the word **LPDET** in columns 1-5 followed by the start and end dates of the period of comparison.

LPDET yyyy/mm/dd TO yyyy/mm/dd

LP Detection Comparison (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDLP** in columns 1-5 (See Table 47.)

Table 47. **Comparison of Long Period Detection**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 8-10 | Chan | a3 | FDSN Channel code |
| 13-22 | Date | i4,a1,i2,a1,i2 | Data day for comparison (yyyy/mm/dd) |
| 25-27 | NDC_det | i3 | Number of surface waves detected by NDC |
| 30-32 | IDC_det | i3 | Number of surface waves detected by IDC |
| 35-37 | Matched | i3 | Number of matching detections (arrival times of maxima are within one minute to constitute a 'match' |
| 40- | Comment | a | Comment - comment on comparison |

## Interactive Use of IDC Database

It is hoped that NDC's will access the IDC database to test the interactive services offered there. This section gives NDC's an opportunity to comment on their experiences with the database.

**Section start (line 1):**

One line containing the word **IDCDBASE** in columns 1-8

Summary of Interactive use of IDC database (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDIDC** in columns 1-6 (See Table 48.)

Table 48. **Interactive Use of IDC Database**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-10 | Date | i4,a1,i2,a1,i2 | Date of access to IDC (yyyy/mm/dd) |
| 13-19 | Facility | a7 | Type of service used: IDCVIEW, MOSAIC, other |
| 22-30 | Data type | a9 | Data type accessed: BULLETIN, DETECTION, STATION, GAMMA, OUTAGE, COMMENT, SYSTEM, DOC |
| 33 | connection | a1 | Was the connection was successful? (Y/N) |
| 36 | service | a1 | Was the service you required available? (Y/N) |
| 39 | availability | a1 | Was the data/information you required available? (Y/N) |
| 42 | usefulness | a1 | Was the data you received useful? (Y/N) |
| 36-80 | Comment | a65 | Comment |

## Number of Man-Hours for GSETT-3 Participation

To assess the drain on resources at the various stages of GSETT-3, it would be useful to have information regarding the number of hours spent on the various categories of work. This format allows a time period to be specified over which a detailed record of hours worked by various members of staff would be recorded. It would be extremely useful if participants could submit a table of this kind prior to the start of GSETT-3, i.e., for the period up to 31/12/94. This will enable the evaluation group to assess the situation at the start of the experiment.

**Section start (line 1):**

One line containing the word **HOURS** in columns 1-5 followed by a start and end date for the period of time over which the record was kept.

HOURS yyyy/mm/dd TO yyyy/mm/dd

Man-hours worked at the NDC (line 2)

Man-hours worked at the Station (lines 3 through N)

(One line per station)

**Section end (line N+1):**

One line containing the word **ENDHRS** in columns 1-6

(See Table 49 and Table 50.)

Table 49. **Man-Hours Expended for GSETT-3 at the NDC**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-7 | NDC | a7 | International Station Code (ISC) (e.g., GBR_NDC) |
| 10-13 | Programmers | i4 | Number of hours worked on GSETT-3 tasks by computer programmers |
| 16-19 | Seismologists | i4 | Number of hours worked on GSETT-3 tasks by seismologists |
| 22-25 | Analysts | i4 | Number of hours worked on GSETT-3 tasks by analysts |
| 28-31 | Data processors | i4 | Number of hours worked on GSETT-3 by data processing staff |
| 34-37 | Engineers | i4 | Number of hours worked on GSETT-3 by engineers |
| 40-43 | Support staff | i4 | Number of hours worked on GSETT-3 by support staff (e.g., admin., secretarial, etc.) |
| 46- | Comment | a | Comment - pointing out reasons for any unusually high or low man-hour counts |

Table 50. **Man-hours Expended for GSETT-3 at the Station**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-5 | Station | a5 | International Station Code (ISC) |
| 8-11 | Programmers | i4 | Number of hours worked on GSETT-3 tasks by computer programmers |
| 14-17 | Seismologists | i4 | Number of hours worked on GSETT-3 tasks by seismologists |
| 20-23 | Analysts | i4 | Number of hours worked on GSETT-3 tasks by analysts |
| 26-29 | Data_processors | i4 | Number of hours worked on GSETT-3 by data processing staff |
| 32-35 | Engineers | i4 | Number of hours worked on GSETT-3 by engineers |
| 38-41 | Support_staff | i4 | Number of hours worked on GSETT-3 by support staff (e.g., admin., secretarial, etc.) |
| 46- | Comment | a | Comment - pointing out reasons for any unusually high or low man-hour counts |

Man-hours Expended for GSETT-3 at the NDC

Man-hours Expended for GSETT-3 at the Station

## Use of Source Characterization Parameters

One of the main objectives of the GSETT-3 system is to provide participants with the information they require for their own CTBT verification work. Although participants may not be prepared to disclose their techniques for verification, it would be useful to the Evaluation Group to know whether the IDC provided them with all the information they require. The format allows the NDC to give a specific time period over which the source characterization parameters were used. A list of possible parameters is given in Operations IDC Appendix F.

**Section start (line 1):**

One line containing the word **VERIF** in columns 1-5 followed by the start date and end date of the period of assessment.

VERIF yyyy/mm/dd TO yyyy/mm/dd

Use of Source characterization Parameters (lines 2 through N)

**Section end (line N+1):**

One line containing the word **ENDVER** in columns 1-6

(See Table 51.)

Table 51. **Use of Source Characterization Parameters**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-10 | Date | i4,a1,i2,a1,i2 | Data day of the source characterization parameters (yyyy/mm/dd) |
| 13-20 | parm_type | a8 | Which parameter was used? |
| 23 | usefulness | a1 | Was the parameter useful? (Y/N) |
| 26-28 | n_events | i3 | Number of events for which the above parameter was useful |
| 31- | Comment | a | Comment - NDC point of view on the provision of parameters |

## GSETT-3 Operational Costs

Costs of all aspects of the GSETT-3 experiment are very important with regard to evaluation of the system. This section refers only to those costs involved in the running of the various elements of the GSETT-3 system. It does not include prior investment costs. This format allows NDC's to give their costing details over a certain period of time.It would be extremely useful if participants could submit a table of this kind prior to the start of GSETT-3, i.e., for the period up to 31/12/94. This will enable the evaluation group to assess the situation at the start of the experiment.

### Section start (line 1):

One line containing the word **COST** in columns 1-4 followed by a start and end date for the period over which the costs apply.

COST yyyy/mm/dd TO yyyy/mm/dd

Equipment/communications costs (line 2)

Manpower costs at NDC (line 3)

Manpower costs at Station(s) (lines 4 through N)

### Section end (line 3):

One line containing the word **ENDCOST** in columns 1-7

(See Table 52 for equipment/communications costs, Table 53 for manpower costs at the NDC, and Table 54 for manpower costs at the station.)

Table 52. **Equipment/Communications Costs for Participating in GSETT-3**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 8-14 | NDC | a7 | International Station Code (ISC); e.g., GBR_NDC |
| 17-22 | s_ndc_comm | i6 | Station to NDC communications cost ($US) |
| 25-30 | s_idc_comm | i6 | Station to IDC communications cost ($US) |
| 33-38 | ndc_idc_comm | i6 | NDC to IDC communications cost ($US) |
| 41-46 | sta_new_equip | i6 | Station new equipment cost ($US) |
| 49-54 | sta_repair_equip | i6 | Station repair equipment cost ($US) |
| 57-62 | ndc_new_equip | i6 | NDC new equipment cost ($US) |
| 65-70 | ndc_repair_equip | i6 | NDC repair equipment cost ($US) |
| 73- | Comment | a | Comment |

Table 53. **Manpower Costs for GSETT-3 at the NDC**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-7 | NDC | a7 | International Station Code (ISC); e.g., GBR_NDC |
| 10-15 | Progr | i6 | Cost of programmers ($US) |
| 18-23 | Seis | i6 | Cost of seismologists ($US) |
| 26-31 | Anal | i6 | Cost of analysts ($US) |
| 34-39 | Data_proc | i6 | Cost of data processors ($US) |
| 42-47 | Eng | i6 | Cost of engineers ($US) |
| 50-55 | Support_staff | i6 | Cost of support staff ($US) |
| 58- | Comment | a | Comment - pointing out reasons for any unusually high or low manpower costs |

Table 54. **Manpower Costs for GSETT-3 at the Station**

| Position | Name | Format | Description |
|---|---|---|---|
| 1-5 | Sta | a5 | International Station Code (ISC) |
| 8-13 | Prog | i6 | Cost of programmers ($US) |
| 16-21 | Seis | i6 | Cost of seismologists ($US) |
| 24-29 | Anal | i6 | Cost of analysts ($US) |
| 32-37 | Data_proc | i6 | Cost of data processors ($US) |
| 40-45 | Eng | i6 | Cost of engineers ($US) |
| 48-53 | Support_staff | i6 | Cost of support staff ($US) |
| 56- | Comment | a | Comment - pointing out reasons for any unusually high or low manpower costs |

## *4.21* EXAMPLE OF AN NDC EVALUATION MESSAGE

This NDC Evaluation Message contains all sections.

```
3D:
EGIN GSE2.0
SG_TYPE Data
SG_ID 1995/05/22_005 GBR_NDC
ATA_TYPE Evaluation GSE2.0
VAL_NO 000001
OMMSOUT
BR_NDC  GSE_IDC  1995/05/16 15:50:10.5 TO 1995/05/16 16:01:05.5  ()
NDCOM
RCHOUT
KA    shz  1995/05/18 12:24:13.5 TO 1995/05/18 15:20:26.8 (Disc drive failure)
NDARC
ETOUT
KA    shz  1995/05/18 12:24:13.5 TO 1995/05/18 15:20:26.8 (Disc drive failure)
NDDET
ETCOMP  1995/04/14 00:00:01.0  TO  1995/04/21 23:59:59.9
KA    shz  1995/04/14    5  100    4  (One magnitude 4.0 event was missed by the IDC)
KA    shz  1995/04/15    3  103    3  ()
KA    shz  1995/04/16    5  123    5  ()
KA    shz  1995/04/17    5  110    4  (One magnitude 3.6 event was\
issed by the IDC)
KA    shz  1995/04/18    3  105    3  ()
KA    shz  1995/04/19    3  109    3  ()
KA    shz  1994/04/20    4  130    3  (One magnitude 3.2 event was\
issed by the IDC)
KA    shz  1994/04/21    6  129    6  ()
NDCOMP
EASURE  1995/04/14 TO 1995/04/21
KA    shz 1995/04/14 09:12:20.5  00:25.5  1234.5678  (Only one event with large\
nset time difference due to emergent onset being difficult for automated\
ystem to pick)
KA    shz 1995/04/15 15:50:22.4  00:01.0  1234.5678  (NDC system measured wrong\
mplitude)
KA    shz 1995/04/16 20:24:30.9  00:01.5     4.3     (Good matches)
KA    shz 1995/04/17 03:30:13.4  00:02.0     7.3     (Good matches)
KA    shz 1995/04/18 19:30:31.0  00:10.0    20.0     (IDC system missed\
mergent onset on one event and measured amplitude in wrong place)
KA    shz 1995/04/19 23:30:39.1  00:01.2     3.2     (Good matches)
KA    shz 1995/04/20 15:40:45.7  00:02.2    10.0     (Good matches)
KA    shz 1995/04/21 09:20:30.5  00:03.4     0.8     (Good matches)
NDMEAS
EQSUB
JLLETIN      1995/02/23  1995/02/28 09:30:00.0  xxxx_yyyy_zzzz_00001  Y  1995/02/28 09:50:10.0  ()
NDSUB
EQREC
AVEFORM      1995/05/20  1995/05/21 12:24:00.0  xxxx_yyyy_zzzz_00002  Y  1995/05/21 12:39:40.0    NDC  ()
AVEFORM      1995/05/21  1995/05/21 16:30:35.5  xxxx_yyyy_zzzz_00003  Y  1995/05/21 16:56:45.5   BETA  ()
NDREC
AMREV  1995/04/12 TO 1995/04/13
995/04/12  103  115   70   13    3   21   11    9   20   50   15    2   41    2 (Eight of the events in 2.5 < M < 3.0 range are\
n a suite of small earthquakes in the same region and would be hard to separate out without local knowledge)
NDGAM

PDET  1995/04/22 TO 1995/04/24
OL    lhz  1995/04/22   12    5    4  (IDC surface wave detector is not optimum)
OL    lhz  1995/04/23   15    7    5  ()
OL    lhz  1995/04/24   20   10    9  (Too many surface waves being missed by IDC)
NDLP
OCDBASE
995/05/15  IDCVIEW  STATION    Y  Y  Y  N  (Not all stations were in the database)
995/05/18  MOSAIC   DOC        Y  Y  Y  Y  ()
995/05/20  IDCVIEW  GAMMA      Y  Y  N  N  (Missing section of Gamma data from ING for the period 1995/03/10 to 1995/03/11)
NDIDC
OURS  1995/05/15 TO 1995/05/21
BR_NDC    10    25    50    20     0     3  (Engineers were not needed as there were no hardware problems)
KA      0    30     0     0    20     0  (Engineers hours were spent on routine tasks)
NDHRS
ERIF  1995/05/16 TO 1995/05/17
995/05/16  DEPTH     N    0  (Event 016 - The depth was not well constrained enough to be useful)
995/05/17  mb:Ms     Y    3  (Ms determination did not really have enough surface wave observations)
NDVER
OST  1995/02/01 TO 1995/05/01
KA    GBR_NDC       0       0    1600      20      50       0       0  ()
BR_NDC    4500    7000    3600    3600    7000    3600  ()
KA         0    7000       0       0   20000       0  ()
NDCOST
TOP
```

O
P
E
R
A
T
I
O
N
S

# Chapter 5 Station AutoDRM Basics

## 5.1 INTRODUCTION

Stations must have a minimum capability to provide data to the IDC through the GSE message system. Clearly, all of the functionality of the request and data messages cannot be supported by these stations and a minimal AutoDRM capability is all that is necessary. Stations, at the simplest level, must be able to respond to a request for waveform data from specified stations and channels within a given time period. This chapter describes the minimal AutoDRM configuration necessary to fulfill the duties of a Beta station for GSETT-3.

## 5.2 BASIC MESSAGE SUPPORT

A station providing segmented data must adhere to all of the basic GSE message conventions on size, line length, date-time formats, station and channel naming, and use of units. The basic message formats that must be supported include:

### BEGIN Line and GSE2.0 Format

All messages must contain the BEGIN line and must support the GSE2.0 format.

### MSG_TYPE

The REQUEST message type must be supported for receiving requests; the DATA message type must be supported for sending data messages. Subscription messages need not be supported.

### MSG_ID

The id_string and optional source of the MSG_ID must be recognized in request messages, and an unique id_string must be generated for data messages.

### REF_ID

The message id of the request message must be used as the reference id of the returned data message.

### CONTINUE and CONTINUATION

Breaking data messages into several small messages must be supported by a station AutoDRM.

### HELP

A stations HELP facility must, at a minimum, describe all of the request message lines that are supported.

### Email

Email must be supported as a data return mechanism. FTP is not required.

## 5.3 ENVIRONMENT LINES

Many of the environment lines described in Chapter 2 are not applicable to a limited station capability for AutoDRM's. The ones that are required include TIME, STA_LIST, and CHAN_LIST. AUX_LIST is required only if necessary to distinguish between two different data streams. Using these three environments, simple requests can be made that obtain data from a particular station and channel within a specified time interval.

## 5.4 REQUEST LINES

The request lines specify what data can be obtained from the AutoDRM. A simple station AutoDRM should be able to provide WAVEFORM, STATION, CHANNEL, RESPONSE, and OUTAGE data.

Request lines may have one or more arguments that specify formats, sub-formats, and an EMBEDDED designator. A simple AutoDRM must support the GSE2.0 format as the main format for all requests, as well as one of the ASCII sub-formats (INT, CM6, or AU6) for waveforms. EMBEDDED need not be supported. The sub-format that is supported should be stated in the HELP facility and be the default if none is specified.

## 5.5 DATA TYPES

Data messages are sent in response to requests sent to the AutoDRM. Thus, WAVEFORM, STATION, CHANNEL, RESPONSE, and OUTAGE data types must be supported by a simple AutoDRM in the GSE2.0 format.

# Chapter 6 Problem Reporting Messages

## 6.1 INTRODUCTION

Problems with the EISMS are bound to occur and need to be communicated between the station operators, NDC's, and the IDC so that a solution may be implemented rapidly. The quality of the data and products of the EISMS depend on a rapid and robust exchange of information with the goal of improving overall system performance. Problem messages provide the mechanism for reporting and communicating problems. In some cases, the "problem" may be a shutdown of the data while scheduled maintenance is performed.

## 6.2 PROCEDURES

Problems and their solutions are communicated through GSE problem messages. Any GSETT-3 participant may report a problem and, in general, all problems reported through the problem message mechanism should in some way affect or interfere with the flow of data at a station, NDC, or the IDC. The basic mechanism for problem reporting utilizes a central message handling system at the IDC. All messages reporting problems are sent to the IDC (messages@cdidc.org) which logs the problems and distributes the messages to all who have made entries in the problem log or who are designated as contacts and have a part in the remedy to the problem.

A problem can be any of a number of types; hardware, software, communications, facilities, or unknown. Each problem type covers a major element of the EISMS. Hardware problems include those having to do with seismometers, digitizers, computers, disk drives, etc., at the station, NDC or IDC. Software problems refer to programs such as the AutoDRM and other data forwarding software that are specific to the EISMS mission as well as to problems with operating systems, database management software, etc. Communications problems may be the communications between the station and an NDC, a station and the IDC, an NDC and the IDC, or any other communications link. Facilities are for problems such as power or air conditioning. If the source of a problem is unknown, then it may be specified as such, or if no problem type is given, then it is assumed to be unknown.

Any of the above problems may occur at a station, an NDC, the IDC, or the communication links that connect them. The location of the problem specified as a station, NDC, or IDC code identifies where the problem occurred. If unknown, the location of a problem may be specified as such or not included in the message.

Although a problem may occur at only one place in the EISMS, there may be many stations that are affected. The list of stations affected, if known, should be listed with each problem.

The status of a problem can be "open" for problems that have not been resolved, or "closed" for problems that have been resolved. If a close date is given in the problem message, then the problem is considered closed; if no close date is given, then it is considered open.

All problem messages are sent to messages@cdidc.org regardless of who initiated the problem report. An initial message contains the usual GSE basic message information, and as much information as is known about the problem (type, location, affected stations, potential contacts, effective date, anticipated close date, a short description of the problem, a detailed description of the problem, and a close date if the problem has already been solved). At the IDC a problem id is assigned to the problem and messages are sent to the contact(s) and to the person who initiated the report. Subsequent messages must have the problem id line for reference. Changes or additions to the information provided in the initial problem report may be made by sending a message containing the changes or additions. Unchanged information may also be included in a problem message to the IDC; it will simply be ignored. When a change is made to a problem through adding a new entry or changing any of the other fields in a problem message sent to the IDC, the problem message system at the IDC will send a message to the contact(s) and to all who have made entries to the problem. The message will contain all information associated with the problem id. The author of an entry (from the E-MAIL line) will appear in these messages as well as the date and time that the IDC received the entry.

Scheduled maintenance is reported through the problem message mechanism. In the case of maintenance, the initial report should be sent in prior to the effective date, and the description should specify that scheduled maintenance is the cause of the data delay or loss.

A list of problems and the entries can be can be obtained by making a request to the IDC through the AutoDRM.

## 6.3 PROBLEM FORMAT DESCRIPTION

Problem messages are similar to data messages, but because problem messages must provide a mechanism for opening, closing, and adding notes to the problem, a separate message type has been adopted. The usual GSE message basic information and the accompanying formats must be used as the framework for a problem message: BEGIN, MSG_TYPE, MSG_ID, REF_ID, E-MAIL or FTP, and STOP.

*Syntax:* **BEGIN** GSE2.0

*Example 6.3 - 1* Problem Message

```
MSG_TYPE PROBLEM
MSG_ID  example ANY_NDC_or_IDC
REF_ID  old_msg_id old_source
E-MAIL  name@my.computer
      (Problem information)
STOP
```

Note that problem messages have MSG_TYPE PROBLEM (second line of the example above).

The problem information referred to in the example contains a problem type (similar to data type for data messages), a problem identification number assigned by the IDC, the location of the problem (e.g., a station, NDC, IDC, or communications link code), the email address of a contact or contacts who are responsible for seeing that the problem is resolved, the date and time that the problem began, a brief description of the problem using standard phrases, a series of entries containing a detailed description of the problem and actions taken to remedy the problem with the author and time of each entry, any new entries that may have been added in this message, and the date and time that the problem was resolved (closed).

## PROB_TYPE

The problem type categorizes the problem into a few well defined areas. These areas include hardware, software, communications, facilities, and unknown.

*Syntax:* **PROB_TYPE** `type`

`type` ............................ Type of problem; any of: HARDWARE, SOFTWARE, COMM, FACILITIES, or UNKNOWN

## PROB_LOC

PROB_LOC specifies where the problem occurred. This will be a specific station code, an NDC code, or the IDC code (GSE_IDC). When referring to communications problems, the code used is for the sending side of the link in question. For example, if the IDC is having troubles sending messages, then the PROB_LOC would be GSE_IDC; if there is a problem on the link between station ABC and its NDC, then the PROB_CODE would be ABC (data are being sent from the station to the NDC under normal operating conditions).

*Syntax:* **PROB_LOC** `code`

`code` ............................ Station, NDC, or IDC code of the problem location.

## PROB_ID

The problem id is assigned by the IDC and must be included in all subsequent messages that refer to the associated problem. Failure to include the problem id once assigned, will cause a new problem to be opened and a new id to be assigned. The first message reporting a problem will not contain a problem id.

*Syntax:* **PROB_ID** `id`

`id` ................................. Identification number assigned by the IDC.

### AFFECTED_STA

The stations affected by the problem are given on the AFFECTED_STA line.

*Syntax:* **AFFECTED_STA** `station[, station[, ...]]`

`station`......................Station(s) affected by the problem.

### CONTACT

The contact is the email address(es) of the person or persons who are responsible for resolution of the problem. These persons will be included in all correspondence about this problem. A table of contacts will be maintained at the IDC, and will be automatically added to any message that does not have a contact or contacts designated. The contact table will be associated with the affected stations.

*Syntax:* **CONTACT** `address[, address[, ...]]`

`address`......................email address of a contact.

### EFFECTIVE_DATE

Problems will generally be recognized sometime after they occur. EFFECTIVE_DATE is the date and time that the problem first occurred (not when it was first recognized).

*Syntax:* **EFFECTIVE_DATE** `date time`

`date`.............................Date of problem (yyyy/mm/dd)
`time`.............................Time of problem (hh:mm:ss)

### ANTICIPATED_CLOSE_DATE

After a problem has been diagnosed, it may be some time before repairs can be made. The ANTICIPATED_CLOSE_DATE is the date when repairs are expected to be completed.

*Syntax:* **ANTICIPATED_CLOSE_DATE** `date`

`date`.............................Date of problem (yyyy/mm/dd)

### DESCRIP

A brief description of the problem is included in a few words on the DESCRIP line. Standard descriptions are shown in the table below for each problem type. If none of the descriptions fit, then a new one may be added (see Table 55).

*Syntax:* **DESCRIP** `description`

`description`............Brief description of the problem.

Table 55. **Standard Problem Descriptions**

| Problem Type | Description |
|---|---|
| HARDWARE | Station Down |
| | Dead Channels |
| | Data Dropouts |
| | Timing Error |
| | Data Glitches |
| | DC Offset |
| | CPU Down |
| | Disk Down |
| SOFTWARE | AutoDRM Problem |
| | Continuous Data Send Problem |
| | Version Upgrade |
| | Scheduled Shutdown |
| COMM | Link Down |
| | Link out of Spec |
| FACILITIES | AC Out |
| | Power Out |
| UNKNOWN | |

## ENTRY

An entry is a record of the descriptions and explanations associated with the PROBLEM_ID. It consists of an ENTRY line followed by a free format description or explanation. The description is limited by the physical and virtual line lengths of GSE messages, and must be indented at least one character in accordance with GSE message conventions. All entries are distributed to the contributors and the contact whenever an addition or change is made to a problem. ENTRY sections received at the IDC will be ignored; only new entries will be added to the problem.

*Syntax:* **`ENTRY`** `address date time`

`address` ..................... email address of the person who made the entry
`date` ............................ Date entry received at the IDC (yyyy/mm/dd).
`time` ............................ Time entry received at the IDC (hh:mm:ss).
`description` ........... A free-format description of the problem or explanation of action taken.

*Example 6.3 - 2*

```
ENTRY someone@some_ndc.org 1994/12/10 07:15:32
 A 3 second timing error was discovered at station ABC.
 A new clock has been ordered.
```

### NEW_ENTRY

A new entry is a description or explanation of a problem. If sent in a message to the IDC without a problem identification number, it is considered a new problem and the IDC will assign a problem identification number; if sent in a message with a problem identification number, it will be added as an entry to the problem. A new entry consists of a NEW_ENTRY line followed by a free format description or explanation. The description is limited by the physical and virtual line lengths of GSE messages, and must be indented at least one character in accordance with GSE message conventions.

*Syntax:* **`NEW_ENTRY`**

`description` ............ A free-format description of the problem or explanation of action taken.

*Example 6.3 - 3*

```
NEW_ENTRY
 A 3 second timing error was discovered at station ABC.
 A new clock has been ordered.
```

### CLOSE_DATE

A problem is closed when a CLOSE_DATE line is included in a message sent by the contact. The date and time of the closed date refer to when the problem was remedied, not when a report was sent that it was remedied.

*Syntax:* **`CLOSE_DATE`** `date time`

`date` ............................ Date problem was remedied (yyyy/mm/dd)
`time` ............................ Time that problem was remedied(hh:mm:ss)

## 6.4 PROBLEM MESSAGE SEQUENCE EXAMPLES

The following example shows a sequence of messages referring to a clock problem at station ABC. The hypothetical problem was first recognized by “sam” at the IDC who sent the following message to messages@cdidc.org and guessing that “joe” and “fred” should be the contacts:

*Example 6.4 - 1*

```
BEGIN  GSE2.0
MSG_TYPE PROBLEM
MSG_ID  1 GSE_IDC
E-MAIL  sam@cdidc.org
PROB_TYPE HARDWARE
PROB_LOC ABC
AFFECTED_STA ABC
CONTACT joe@the_ndc.org, fred@the_ndc.org
EFFECTIVE_DATE 1994/08/10 00:00:00
PROBLEM Timing Error
NEW_ENRTY
 ABC has a timing error.  Data times are about 1 min early.
STOP
```

The IDC problem message handler assigned a problem identification number, logged the message, changed the NEW_ENTRY line to an ENTRY line with the author (sam) and the time that the message was received, and sent the following message to both “sam”, “fred”, and “joe”:

*Example 6.4 - 2*

```
BEGIN  GSE2.0
MSG_TYPE PROBLEM
MSG_ID  2 GSE_IDC
REF_ID  1 GSE_IDC
E-MAIL  messages@cdidc.org
PROB_TYPE HARDWARE
PROB_ID 000037
PROB_LOC ABC
AFFECTED_STA ABC
CONTACT joe@the_ndc.org, fred@the_ndc.org
EFFECTIVE_DATE 1994/08/10 00:00:00
PROBLEM Timing Error
ENRTY sam@cdidc.org 1994/08/11 13:32:17
 ABC has a timing error.  Data times are about 1 min early.
STOP
```

Joe recognized that he and fred were not the proper contacts for this problem and sent a message back to the IDC with a new contact "wanda". Joe could have returned the complete message with the ENTRY, but chose to send only the new contact:

*Example 6.4 - 3*

```
BEGIN  GSE2.0
MSG_TYPE PROBLEM
MSG_ID  100 THE_NDC
REF_ID  2 GSE_IDC
E-MAIL  joe@the_ndc.org
PROB_ID 000037
CONTACT wanda@the_ndc.org
STOP
```

The IDC problem message handler changed the contact from "joe" and "fred" to "wanda" and sent the following message to both "sam" and "wanda":

*Example 6.4 - 4*

```
BEGIN  GSE2.0
MSG_TYPE PROBLEM
MSG_ID  3 GSE_IDC
REF_ID  100 THE_NDC
E-MAIL  messages@cdidc.org
PROB_TYPE HARDWARE
PROB_ID 000037
PROB_LOC ABC
AFFECTED_STA ABC
CONTACT wanda@the_ndc.org
EFFECTIVE_DATE 1994/08/10 00:00:00
PROBLEM Timing Error
ENRTY sam@cdidc.org 1994/08/11 13:32:17
 ABC has a timing error.  Data times are about 1 min early.
STOP
```

"Wanda" looked into the problem (which she was able to solve), added an entry to the problem, closed the problem, and sent a message to messages@cdidc.org. Note that no ANTICIPATED_CLOSE_DATE was necessary in this case because the message closing the problem would have been the same in which an anticipated closing date was known.

*Example 6.4 - 5*

```
BEGIN  GSE2.0
MSG_TYPE PROBLEM
MSG_ID  101 THE_NDC
REF_ID  3 GSE_IDC
E-MAIL  wanda@the_ndc.org
PROB_TYPE STATION
PROB_ID 000037
AFFECTED_STA ABC
CONTACT wanda@the_ndc.org
EFFECTIVE_DATE 1994/08/11 12:00:00
PROBLEM Timing Error
ENRTY sam@cdidc.org 1994/08/11 13:32:17
 ABC has a timing error.  Data times are about 1 min early.

NEW_ENTRY
 The ABC station clock was running approximately 1 minute ahead of
 UTC until about 17:00 UTC on Tuesday 16-Aug-94 when it was reset.

CLOSE_DATE 1994/08/16 17:00
STOP
```

The IDC problem message handler added the new entry with “wanda” as the author, closed the problem and sent a final message to both “wanda” and “sam”:

*Example 6.4 - 6*

```
BEGIN  GSE2.0
MSG_TYPE PROBLEM
MSG_ID  4 GSE_IDC
REF_ID  101 THE_NDC
E-MAIL  wanda@the_ndc.org
PROB_TYPE HARDWARE
PROB_ID 000037
PROB_LOC ABC
AFFECTED_STA ABC
CONTACT wanda@the_ndc.org
EFFECTIVE_DATE 1994/08/11 12:00:00
PROBLEM Timing Error
ENRTY sam@cdidc.org 1994/08/11 13:32:17
 ABC has a timing error.  Data times are about 1 min early.

ENTRY wanda@the_ndc.org 1994/08/16 19:09:54
 The ABC station clock was running approximately 1 minute ahead of
 UTC until about 17:00 UTC on Tuesday 16-Aug-94 when it was reset.

CLOSE_DATE 1994/08/16 17:00
STOP
```

Chapter 7

# Protocol for the Exchange of Continuous Data

## 7.1 INTRODUCTION

A special protocol for sending and receiving continuous near-real-time data has been established that provides a reliable method for transmitting and receiving data with a built-in flexibility for handling the data from seismic stations with up to 100 data streams in a variety of formats (including compressed data and data with authentication signatures). The real-time data exchange paradigm includes several special message formats that are used to describe the data being sent, communicate changes, and forward the data.

This Chapter covers the protocols and formats for continuous data passed between different organizations participating in GSE. It is not meant to describe the internal communications protocols or formats within the many organizations participating in GSETT-3.

All other types of data and information is exchanged using the protocols described and GSE Message conventions found in Chapter 1.

A model of communication between the alpha stations and the IDC is shown in Figure 1. Two critical processes will allow the IDC to receive and store incoming alpha data. The Connection Manager will identify requests for data transmission that are made by alpha stations and will return a message to the alpha station assigning a communications port. The Disk Loop Manager will accept incoming data, store the data in disk files, and update the data base. As a minimum, alpha data will be stored in disk loops at the IDC until routine data processing, analysis and long-term archiving are completed, generally 2-3 days. This chapter describes the function of the Connection Manager, describes a protocol for establishing or altering a connection between an alpha station and the IDC, and specifies formats for data and other message packets.

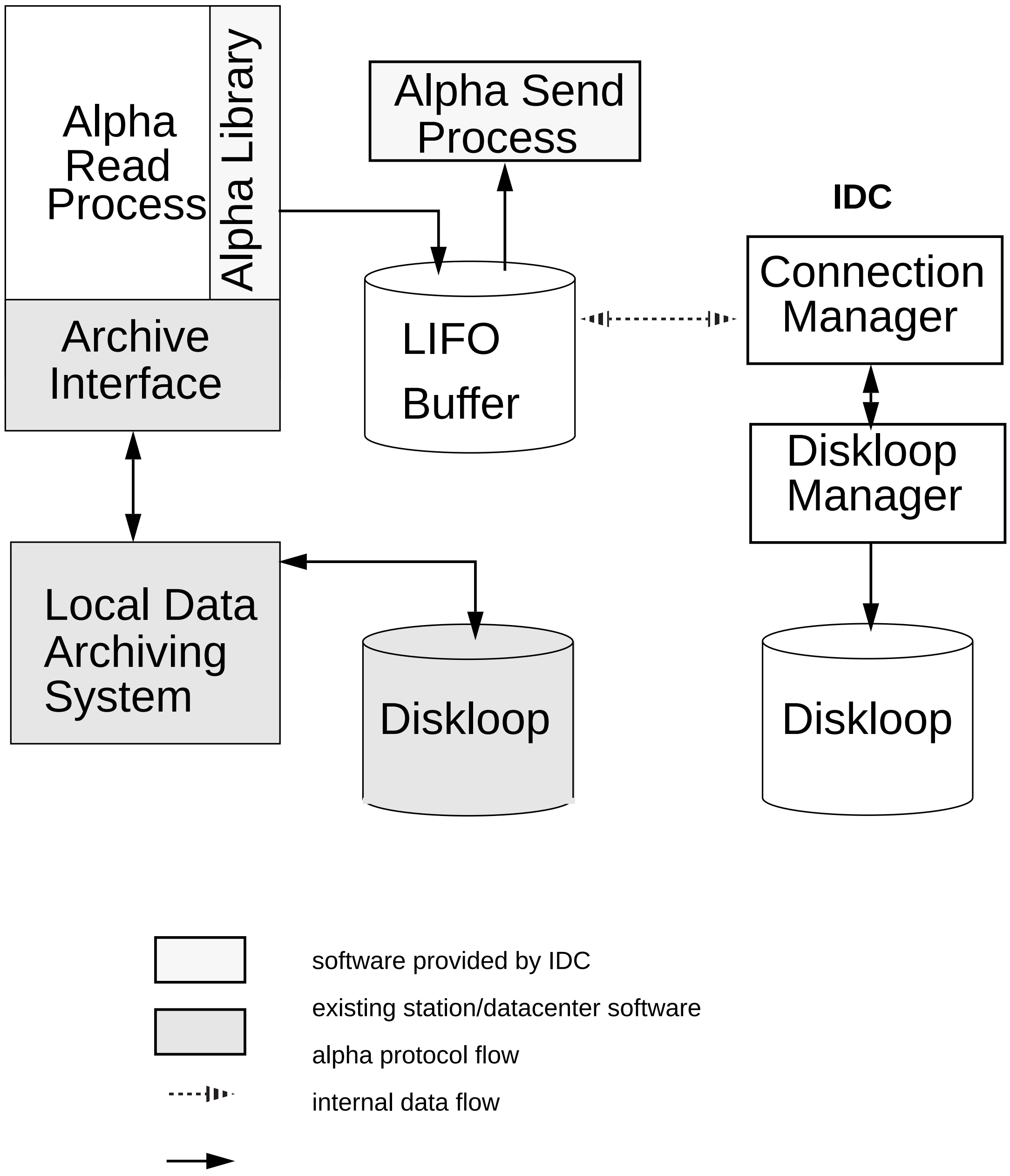


Figure 1. **Data Communications Between Alpha Stations and the IDC**

## 7.2 ESTABLISHING COMMUNICATIONS BETWEEN ALPHA STATIONS AND THE IDC

It is essential to adhere to a fixed protocol for establishing a connection between an alpha station and the IDC. This section proposes a protocol that is short and simple yet meets the requirements for robustness and flexibility. The protocol is shown schematically in Figure 2. In general, a separate connection will be made for each station that is transmitted to the IDC from a data concentration center.

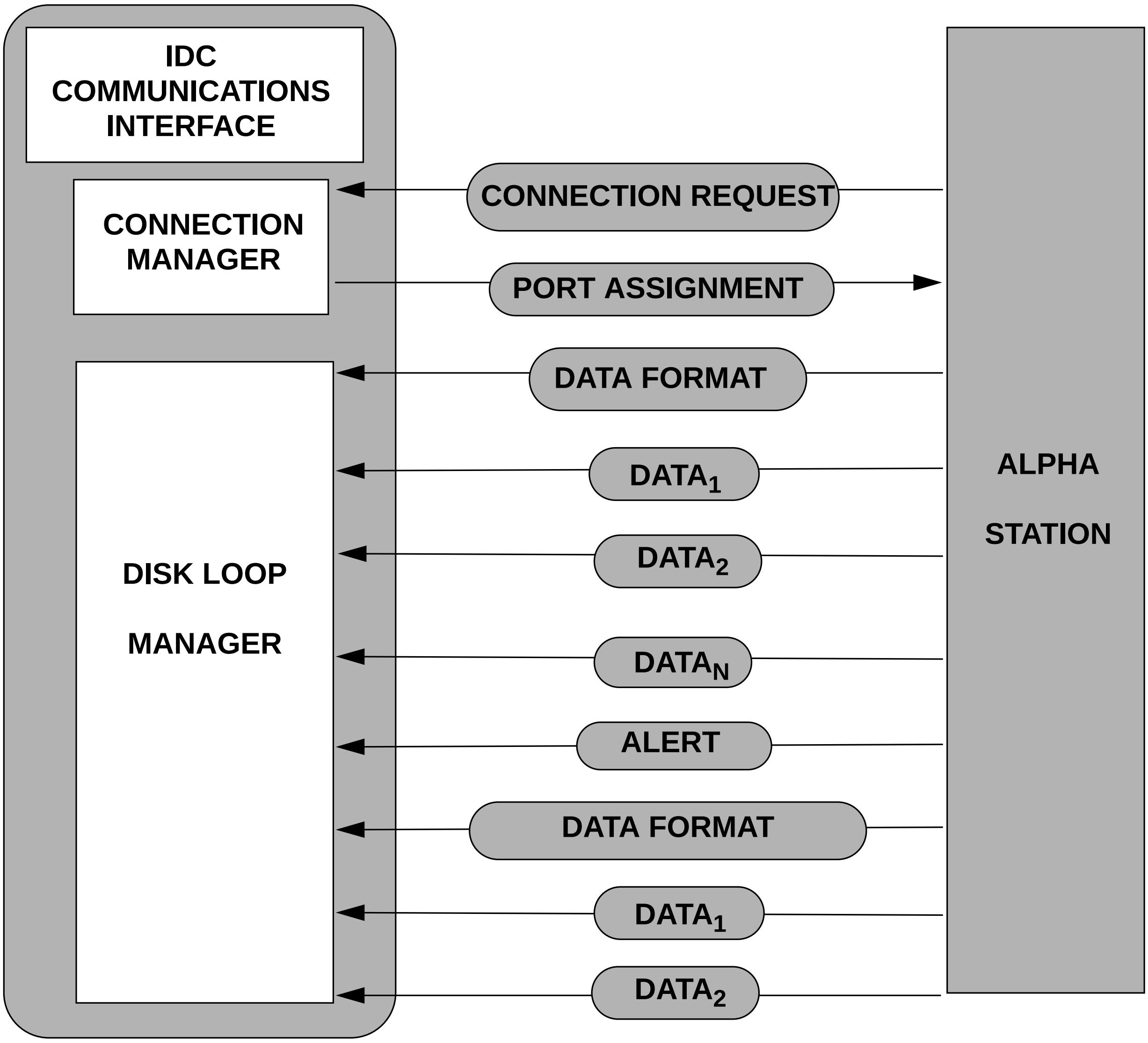


Figure 2. **Schematic Diagram of Communications Protocol Between Alpha Stations and the IDC**
The alpha station requests a connection from the Connection Manager, which assigns a port for data transmission. The alpha station connects to a Disk Loop Manager through an assigned port, transmits a single description of the format of the data and transmits only data frames.

An alpha station will need to establish a connection with the IDC under three circumstances: when the station is first installed, when the alpha station resumes operation after a station failure, and when the station needs to reconnect to the IDC after a failure at the IDC or a failure of communications to the IDC. To initiate communication, the alpha station will attempt to open a connection to the IDC on a pre-designated address (currently idc.css.gov) and port (8000). The station will announce itself to the Connection Manager at the IDC by sending a Connection Request Frame that contains its station code. The format of this frame and others are described in detail in a following section.

A program at the IDC, such as the Unix process "inetd", must be running at the pre-designated address and listening to port 8000 at all times. The Connection Manager will only allow a connection if three conditions are met: 1) the station code is announced, 2) the station code matches a known alpha station code, and 3) the IP address from which the connection originated is valid for that station. The Connection Manager that is executed in response to this connection request will negotiate with a Disk Loop Manager for a port and address for the connection between the alpha station and the Disk Loop Manager. This information will be sent in a Port Assignment Frame to the alpha station in the format specified below. The Connection Manager will then be finished with this connection request and will close the connection. At all times, however, even when the Connection Manager is working on a new request for connection, the designated port will be monitored.

When the alpha station receives the address and port assignments, it should immediately drop the connection to the Connection Manager and connect to the assigned address and port. Once the alpha station and the Disk Loop Manager are connected, the station sends a Data Format Frame, which describes the format of the subsequent Data Frames. The alpha station is then free to send Data Frames as long as it can send data. No polling is done by the Disk Loop Manager, and error correction is handled by TCP/IP.

Safeguards need to be included for lost communication. If the connection is lost between the time the alpha station connects and the Connection Manager replies, after 60 seconds the Connection Manager will assume that the connection is lost and will give up attempting to communicate with the alpha station. Likewise, if the alpha station does not receive a Port Assignment Frame within 60 seconds, the station should assume that the connection is lost and try again by sending another Connection Request Frame.

## *7.3* ALTERING A CONNECTION

To alter a connection, the alpha station may send an Alert Frame to the Disk Loop Manager. In the case where attempts by the alpha station to write to the Disk Loop Manager fail, it should time out after 10 minutes of failures, close the connection, and request a new connection from the Connection Manager. The alpha station must be prepared to receive an alert from the Disk Loop Manager, and alter the connection as ordered.

## 7.4 DATA RECOVERY FOLLOWING LOSS OF COMMUNICATION LINK

It is a goal of the IDC to use as many alpha stations as possible when preparing the Alpha Event List. Therefore, it is essential to resume access to near-real time data as rapidly as possible following failure of a communication link. A last-in-first-out (LIFO) data transmission scheme, illustrated in Figure 3, addresses part of this problem. In a LIFO system, data from a station are stored in a stack and the most recent data available at any given time are transmitted. Circuit capacity exceeding the rate of data production is used to work down through the stack, but as newer data are produced, they arrive at the top of the stack and are sent first. In the pure LIFO scheme, the send pointer, which determines which data are sent next, is always at the top of the stack.

For signal processing at the IDC, recent data are only valuable in sizes large enough to process automatically. Timeliness requirements prevent the length of an individual data packet from being useful for processing so a method has been devised that ensures continuity of data in large blocks. In the modified scheme, a send pointer designates the point within the stack from which to send data, and markers are inserted in the stack at regular intervals. The marker intervals correspond to the minimum length of contiguous data that is useful for processing at the IDC. Under normal circumstances, when data are being transmitted about as fast as they are arriving in the stack, there will only be one marker in the stack at any given time and the stack pointer will be at the top of the stack. However, if data are not being sent faster than they are collected, then the stack accumulates data packets above the send point. Data above the send point are transmitted until a marker is reached. At that time, the stack is checked for other markers. If found, this indicates that at least one useful interval of data has collected above the send point. The send pointer is then moved to the most recent marker, and all markers are deleted. When the send pointer reaches the top of the stack (after the transmission problems have been corrected), data are once again transmitted in pure LIFO order and the data collected below the send point is gradually dispatched. The Disk Loop Manager at the IDC is responsible for properly handling data that arrives out of time order.

Figure 3 illustrates how a pure LIFO and the modified LIFO systems compare in terms of data availability at the IDC. In the example the data link supports only one-half the necessary bandwidth for the first six time units. At that time, the link improves and has a 50% excess capacity for the remainder of the time. In the pure LIFO system, the gaps in data as seen by the IDC (a horizontal slice through the diagram) render the data useless for much of the time. The modified LIFO system, on the other hand, maximizes the useful data arriving at the IDC.

Several circumstances may require the IDC to retrieve data from an alpha station using methods not included in the alpha protocol described here. For example, if data authentication fails, it would be important to confirm that the authentication signature was not corrupted in transmission. Also, if a communications outage was sufficiently long that the data buffer space was exceeded, data would fall off the stack. In these cases the IDC would retrieve data using the Beta waveform request interface. In some cases of very long outages, it may not be practical to recover using either the alpha or beta protocols, and it may be necessary to send the data to the IDC on magnetic media.

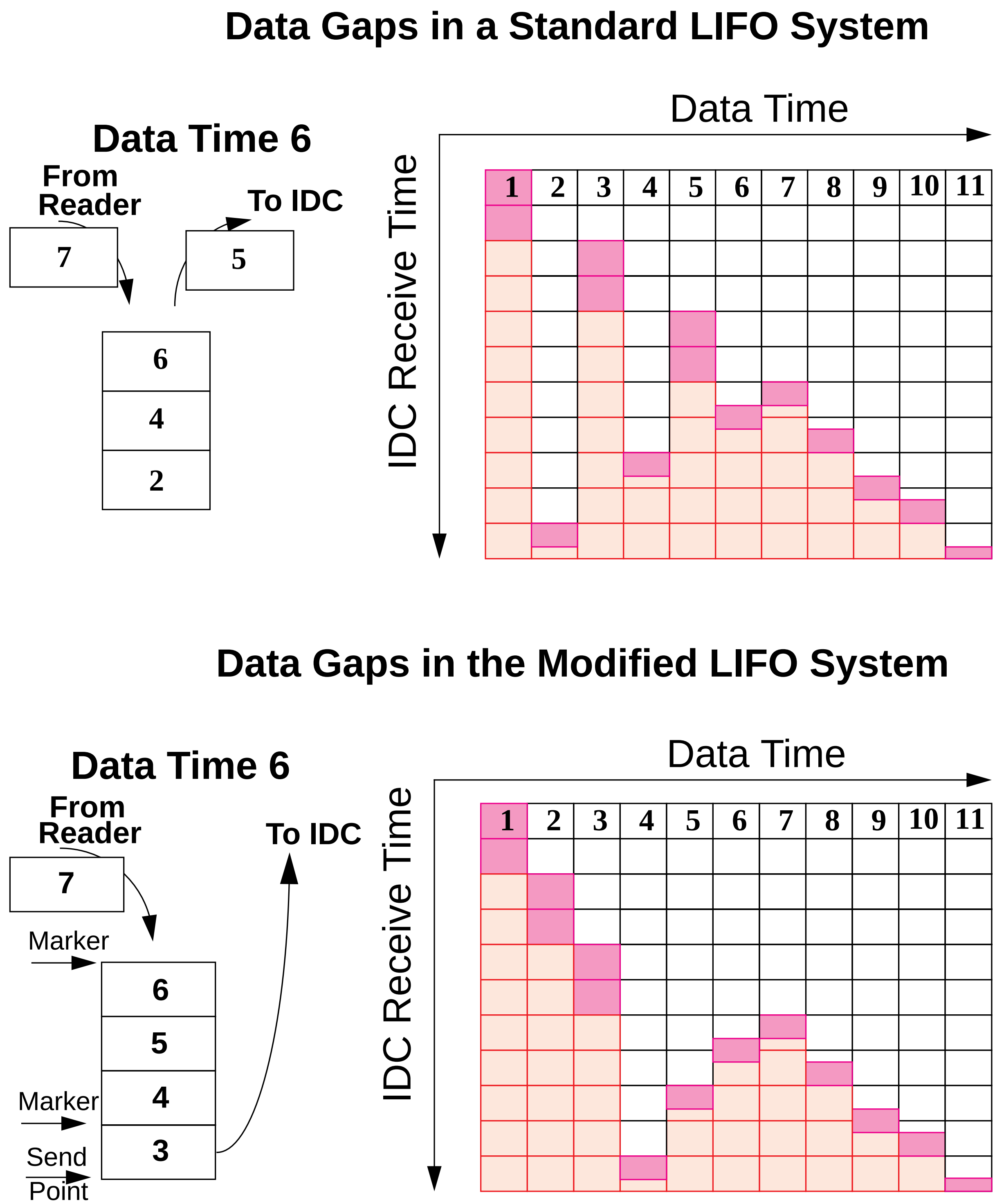


Figure 3. **Data Gaps in Standard and Modified LIFO Systems**
Diagram comparing the sequence of data transmission and the resultant data gaps in a pure last-in first-out (LIFO) system and a modified LIFO system that incorporates the importance of data continuity. The number in each packet shows the original sequence of the data.

## 7.5 TRANSMISSION OF CONTINUOUS WAVEFORMS

Continuous waveform data are pushed to the IDC in near-real-time. In some cases the data may be sent directly to the IDC from the station, and in others, the data are passed through an NDC prior to forwarding them to the IDC. Data and messages related to the transmission of continuous alpha data to the IDC or from the IDC must adhere to the formats specified below.

The connection and transmission requirements can be satisfied by five distinct types of messages, which adhere to prescribed frame formats. These messages are Transmission Request Frame, Port Assignment Frame, Data Format Frame, Data Frame and Alert Frame. The formats are described below.

Each Data Frame must be short enough to allow near-real-time transmission and to prevent any large losses due to a small error somewhere in the frame. On the other hand, the Data Frame must be large enough to be processed efficiently. Data Frames of duration between one and ten seconds are most appropriate. All channels for a given station should be incorporated into a single Data Frame. Data Frames should include authentication signature when required.

The format specifies IEEE numerical representation. Use of IEEE representation implies forward byte ordering of all multi-byte values. Values generated on machines that do not use IEEE as their native numerics may need to swap bytes before transmitting data to the IDC. All portions of this format that are of arbitrary length are required to be divisible by 4, which saves unnecessary byte manipulation for machines with IEEE native numerics. Note that all unused bytes in any frame should be set to 0 (nil), including unused bytes at the end of strings.

### Transmission Request Frame

To initiate transmission of data, the alpha station sends a Transmission Request Frame to the IDC.

Table 56. **Transmission Request Frame**

| Field | Format | Description |
|---|---|---|
| station name | 8-byte string | Registered station code |

### Port Assignment Frame

To assign a port for data transmission, the IDC sends a Port Assignment Frame to the alpha station.

Table 57. **Port Assignment Frame**

| Field | Format | Description |
|---|---|---|
| IP address and port | 32-byte string | IP address number and port number separated by "/" |

## Data Format Frame

The Data Format Frame describes the format of the alpha data that is to be transmitted. The Data Format Frame will be transmitted to the IDC whenever communications are first established or re-established following a disconnection. The Data Format Frame consists of 2020 bytes as shown below. The maximum allowable frame size is 10,000,000. Note that while the format provides for data authentication, that no authentication signature will be sent if the authentication flag is off.

Table 58. **Data Format Frame**

| Field | Format | Description |
|---|---|---|
| frame id | 4-byte IEEE integer | id for Data Format Frame = 2 |
| size | 4-byte IEEE integer | length of data format frame in bytes (including size and id) |
| maximum frame size | 4-byte IEEE integer | maximum size of any Data Frame in bytes ( 10,000,000; divisible by 4) |
| number of channels | 4-byte IEEE integer | number of channels in Data Frame |
| frame time length | 4-byte IEEE integer | time in milliseconds that each Data Frame transmitted will represent |
| site and channel flags, calibrations, and names | 100 28-byte sections:<br>Flags - 4 bytes<br>Calib - 4 byte IEEE int<br>Calper - 4 byte IEEE flt<br>Sta/Chan - 16 bytes | Flags<br>Byte 1: authentication (0=off, 1=on).<br>Byte 2: compression (0= no compression, 1=Canadian compression applied after authentication, 2=8-bit compression (CM8) applied after authentication, other values as required)<br>Bytes 3 and 4: unused.<br>Bytes 5-8<br>Bytes 9-12<br>Bytes 13-28: site/channel names, with slash character required as separator. String begins at first byte and contains no spaces. Remaining bytes zeroed. Unused strings zeroed. Site/channel designations are required to appear in the order the data will appear in the Data Frames. If a single site has more than 100 channels, multiple connections to the IDC will be required. |

## Data Frame

In normal operation, once communications are established between an alpha station and the IDC, only Data Frames will be transmitted to the IDC. The Data Frame consists of a header and Channel Sub-Frames. The number of Channel Sub-Frames is the same as the number of channels specified in the Data Format Frame. The header contains size and time information and possibly weather or other state of health data, whereas the Channel Sub-Frame contains the authentication as well as the actual data.

The header for the Data Frame is shown below. All portions of this format that are of arbitrary length are required to be divisible by 4, which saves unnecessary byte manipulation for machines with IEEE native numerics.

The description field will initially be stored as a station log at the IDC. Systematic reporting of other types of data (e.g., weather) could, in the future, be stored in a more useful format. Descriptions shall not be longer than 1024 bytes.

Table 59. **Data Frame**

| Field | Format | Description |
|---|---|---|
| frame id | 4-byte IEEE integer | id for Data Frame = 1 |
| size | 4-byte IEEE integer | length of data frame with Channel Sub-Frames in bytes (including size and id, and channel data) |
| nominal time | 8-byte IEEE float | nominal start time of all channels in frame, in seconds since 1 Jan 1970, 00:00 and not counting leap seconds |
| description length | 4-byte IEEE integer | number of bytes in the following description ( 1024; divisible by 4). |
| description | description-length bytes | description. Contains any data the station operators want to log at the IDC (e.g., weather data). |

## Channel Sub-Frame

There is one *Channel Sub-Frame* for each channel listed in the *Format Frame* when communications were first established, and the Channel Sub-Frames are arranged in the order that was specified.

In order to avoid unnecessary communication overhead, no authentication string will be sent for channels that are not authenticated as flagged in the Data Format Frame, as shown in Table 60.

Table 60. **Channel Sub-Frame**

| Name | Format | Description |
|---|---|---|
| packet_len | 4-byte IEEE integer | length of packet in bytes, not counting this word, for channel that follows (divisible by 4) |
| authentication | 40-byte string | authentication signature for this channel packet |
| timestamp | 8-byte IEEE float | seconds since 1 January 1970 00:00 (excluding leap seconds) for first sample. Must be within one sample of nominal time. |
| samples | 4-byte IEEE integer | number of samples in channel packet |
| status | 4-byte string | Data status byte (most significant byte)<br>bit 31 1=dead channel<br>bit 30 1=zeroed data<br>bit 29 1=clipped<br>bit 28 1=calibration signal<br>bit 24-27 undefined<br>Station status byte<br>bit 23 1=vault door open<br>bit 22 1=authentication box opened<br>bit 21 1=equipment moved<br>bit 20 1=clock differential too large<br>bit 16-19 undefined<br>Station specific bits<br>bit 0-15 user defined (e.g., station status counter) |
| data | (length of packet minus 16 or 56 bytes - based on original 4-byte IEEE integers) | raw, compressed data for channel |

The 32-bit status word contains automatic indicators of the status and problems with the station equipment or data. As part of the general station information provided to the IDC, station operators should notify the IDC of status conditions that are not defined in the above table. In order to allow the IDC to monitor the completeness of each channel data stream, data should be sent to the IDC even when zeroed, clipped or consisting of calibration pulses. The IDC is responsible for handling these data conditions properly.

## Canadian Compression Scheme

The Canadian compression scheme is preferred for continuous data transmission. It is based on the same principles as the CM6 and CM8 schemes: second difference data compressed using an indexing formula.

The “second difference” algorithm used for Canadian compression is slightly different than that used for CM6 and CM8. The second value in Canadian compression is S(2) - S(1) rather than S(2) - 2S(1), and there is an additional sample at the end that incorporates the first sample of the next block and is used as a check of the data.

Any data sample whose value is unknown is given an integer value of -231 before compression takes place.

In the “second difference” algorithm, the first data sample in the packet, S(1), is represented as 32 bit, 2's complement integer, with a range of - (231) to +(231- 1).

The second sample is encoded as the difference between the second and first time series values:

$$D(2) = S(2) - S(1)$$

The remaining samples in the packet are encoded as second differences; i.e., the difference between the current difference and the preceding difference:

$$D_2(j) = D(j) - D(j-1) = S(j) - 2S(j-1) + S(j-2)$$

Thus the second difference data for N samples are:

$$S(1), D(2), D_2(3), D_2(4), \ldots, D_2(N+1)$$

or

$$S(1), S(2) - S(1), S(3) - 2\,S(2) + S(1), \ldots, S(N+1) - 2\,S(N) + S(N-1)$$

Note that there are N + 1 samples in the packet. The final (N + 1th) sample is used as an independent error check of the compression and decompression processes. It has the same value as the first data sample in the next Data Packet.

The 'N' differences are represented as 4, 6, 8, 10, 12, 14, 16, 18, 20, 24, 28 or 32 bit, 2's complement integers, with ranges from -8 to +7 for four bits, -32 to +31 for six bits, and so on, up to -(231) to +(231 - 1) for 32 bits.

An Index is used to specify how many bits are used to encode each difference. The Index contains a number of 16 bit Index Entries, one for each block of 20 differences. Thus a 6-second data packet, with 60 samples / second, will have an Index containing 18 Entries totaling 36 bytes.

Each Index Entry is encoded as follows:

| MSB | | | | | | | | | | | | | | | LSB |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| h | a | a | a | b | b | b | c | c | c | d | d | d | e | e | e |

| | |
|---|---|
| h = 0 | all 20 samples can be encoded in < = 18 bits/sample |
| h = 1 | at least one sample needs > = 20 bits for encoding |
| aaa | encodes the number of bits/sample required to specify the first four data sample differences (i.e., for samples #2 - #5). Its interpretation depends on whether h = 0 or 1. |
| bbb | bits/sample encoding for samples #6 - #9 |
| ccc | bits/sample encoding for samples #10 - #13 |
| ddd | bits/sample encoding for samples #14 - #17 |
| eee | bits/sample encoding for samples #18 - #21 |

Table 61 shows some sample codes.

Table 61. **Bits / Sample Codes**

| aaa | h = 0 | h = 1 |
|---|---|---|
| 000 | 4 | 4 |
| 001 | 6 | 8 |
| 010 | 8 | 12 |
| 011 | 10 | 16 |
| 100 | 12 | 20 |
| 101 | 14 | 24 |
| 110 | 16 | 28 |
| 111 | 18 | 32 |

The next Index Entry corresponds to samples #22 - #41. And so on for each of the remaining Index Entries.

## Alert Frame

Alert Frames are sent by an alpha station to catch the attention of the Communications Manager and Disk Loop Manager at the IDC.

The IDC will ignore any message having a “frame id” other than 1 for a Data Frame, 2 for a Data Format Frame, or 4 for an Alert Frame. Furthermore, it will ignore any packet with a “frame id” other than 1 if not preceded by an Alert Frame. However, on initial connect or after alert key 2, the IDC will ignore any frame without the id “2” for Data Format Frame. If at all possible, the station should send an Alert Frame announcing termination to the IDC before disconnecting. More alert types may be added as this standard evolves. The IDC will be required to handle all alert types, but stations do not need to be able to send all types. Alerts should always be used to announce that a frame other than data is about to arrive. Other frames, such as calibration reports, will be added to this specification. The station must also be capable of receiving a termination alert from the Disk Loop Manager. This can be either (key 1000) an immediate termination, or (key 1001). It could be followed by new connection information (like the original connection frame sent by the IDC to the station).

Table 62 defines the Alert Frame.

Table 62. **Alert Frame**

| Name | Format | Description |
|---|---|---|
| frame_id | 4-byte IEEE integer | id for Alert Frame = 4 |
| size | 4-byte IEEE integer | length of alert frame in bytes (including size and id) |
| alert_type | 4-byte IEEE integer | key for alert type.<br>0 no action<br>1 immediate termination by station<br>2 flush format, wait for new (from station)<br>1000 immediate termination by IDC<br>1001 new connection information (from IDC) |

## Connecting an Alpha Station to the IDC and Transmitting Data

In this example, we assume the station computer has been down and has recently returned to operation and is seeking a connection to the IDC. Our station for this discussion will be "ZZA0", a nine-element array having nine short-period verticals at 40 Hz and a three-component broadband at 80 Hz. The software managing the sending of the station data immediately attempts to open a connection to idc.css.gov on port 8000. The time-out is set to 60 seconds. The time-out expires because communications are not yet fully established after restarting the station systems. The program immediately retries the connection. This time it is successful, and a TCP connection is established to the Connection Manager at the IDC. The station immediately sends the Connection Request Frame as shown in Table 63, where "0" is an ASCII character 0, and nil is a zero byte.

Table 63. **Example Connection Request Frame**

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|
| Z | Z | A | 0 | nil | nil | nil | nil |

While the IDC finds a disk loop connection for this session, the station program has again set a 60 second time-out. There are several reasons why this time-out may be exceeded:

- communications may have failed
- the IDC may reject the station name or the originating address
- the IDC may be slowed by excessive requests from other stations or NDC's and fail to meet the deadline

Nevertheless, the station should close the connection and attempt it again from the start. Under normal circumstances, within 60 seconds the IDC will send a Port Assignment Frame to the alpha station as shown in Table 64.

Table 64. **Example: Port Assignment Frame**

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4 | 0 | . | 1 | 6 | 2 | . | 1 | . | 5 | / | 7 | 1 | 4 | 4 |
| nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil |

In this case, the IDC is telling the station to connect to the Internet address “140.162.1.5” on port “7144”. At this point, the station immediately drops the original connection to the IDC and tries connecting to the new address and port. When successful, the station now has a connection to a Disk Loop Manager at the IDC. Again, the station should set a time-out of 60 seconds for this connection to be established. If it fails, the station must start the process over again by connecting to idc.css.gov.

Once connected to the Disk Loop Manager, the station must first send the Data Format Frame as illustrated in Table 65. Refer to Table 58 for a description of the fields used in Table 65.

Table 65. **Example of a Data Format Frame**

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Frame ID (2) | | | | Frame Length (2820) | | | | Max. Data Bytes (12200) | | | | Number of Channels (9) | | | |
| nil | nil | nil | 002 | nil | nil | 017 | d | nil | nil | / | 250 | nil | nil | nil | 014 |
| Millisec of Data (6000) | | | | Flags | | | | Calibration Factor | | | | Calibration Period | | | |
| nil | nil | 027 | p | 1 | 1 | nil | nil | 013 | e | G | 017 | p | j | 004 | q |
| Station and Channel Designation | | | | | | | | | | | | | | | |
| Z | Z | A | 0 | / | s | z | nil | nil | nil | nil | nil | nil | nil | nil | nil |
| Flags | | | | Calibration Factor | | | | Calibration Period | | | | etc. | | | |
| 0 | 1 | nil | nil | c | e | G | 017 | p | j | 004 | q | Z | Z | A | 1 |
| / | s | z | nil | nil | nil | nil | nil | nil | nil | nil | nil | 0 | 1 | nil | nil |
| 013 | e | G | 017 | p | j | 004 | q | Z | Z | A | 2 | / | s | z | nil |
| nil | nil | nil | nil | nil | nil | nil | nil | 0 | 1 | nil | nil | 013 | e | G | 017 |
| p | j | 004 | q | Z | Z | B | 1 | / | s | z | nil | nil | nil | nil | nil |
| nil | nil | nil | nil | 0 | 1 | nil | nil | 013 | e | G | 017 | p | j | 004 | q |
| Z | Z | B | 2 | / | s | z | nil | nil | nil | nil | nil | nil | nil | nil | nil |
| 0 | 1 | nil | nil | c | e | G | 017 | p | j | 004 | q | Z | Z | B | 3 |
| / | s | z | nil | nil | nil | nil | nil | nil | nil | nil | nil | 0 | 1 | nil | nil |
| 013 | e | G | 017 | p | j | 004 | q | Z | Z | B | 4 | / | s | z | nil |
| nil | nil | nil | nil | nil | nil | nil | nil | 0 | 1 | nil | nil | 013 | e | G | 017 |
| p | j | 004 | q | Z | Z | B | 5 | / | s | z | nil | nil | nil | nil | nil |
| nil | nil | nil | nil | 0 | 1 | nil | nil | 013 | e | G | 017 | p | j | 004 | q |
| Z | Z | A | 3 | / | s | z | nil | nil | nil | nil | nil | nil | nil | nil | nil |
| nil | nil | nil | nil | nil | nil | (2548 nil bytes) | | | nil | nil | nil | nil | nil | nil | nil |
| nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil | nil |

Note that channel order reflects the order in which the data will be transmitted (ZZA3/sz is at the end in this case, not after ZZA2). The first 20 bytes (shown in the example as an ASCII representation of a binary number with non-printing characters written in octal) give the “frame id” 2 (for Data Format Frame), size 2820 bytes, maximum Data Frame size 12200 bytes, nine channels, and 6000 milliseconds of data per Data Frame. The maximum Data Frame size is computed by assuming at most three bytes per sample, three 80 Hz and nine 40 Hz channels (with 100 bytes additional information per channel), 20 bytes required header, plus 180 bytes of descriptive data (this station will limit itself to 180 bytes in that field). The ZZA0/sz data have authentication, and all data are compressed using the Canadian compression scheme.

After the Data Format Frame is sent, the station is free to write Data Frames to the IDC Disk Loop Manager as long as the write is successful. The start of a typical Data Frame printed in octal is shown in Table 66.

Table 66. **Example of a Data Frame**

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| nil | nil | nil | 001 | nil | nil | 027 | p | A | 306 | @ | 356 | 371 | ) | 231 | 232 |
| nil | nil | nil | nil | | | | | | | | | | | | |

This means the Data Frame has a "frame id" of 1, will be 6000 bytes long, and all channels have a nominal start time of 746708466.325 seconds (30 August, 1993 at 11:01:06.325 GMT). There is no descriptive information in this frame. Immediately after the last byte above, the channel data for 6000 milliseconds starting at the sample at approximately 11:01:06.325 for ZZA0/sz will start, including the authentication, the actual time of the first sample, the number of samples (40), and status. Then the rest of the channels follow in the order specified in the Data Format Frame.

After some time of operation like this, the station operators decide that the broadband must be removed for maintenance. The communications may be left running during this, by instructing the sending software to interrupt communications and send a new format. First the station would send an alert to the Disk Loop Manager, as shown in Table 67.

Table 67. **Example of an Alert Frame**

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| nil | nil | nil | 004 | nil | nil | nil | 014 | nil | nil | nil | 002 | | | | |

This notifies the Disk Loop Manager to pause and expect a new format. Then the station will send a new Data Format Frame, specifying nine channels and the order of those. It may then resume sending data (this time without the broadband channels).

Other circumstances all are prompted by alerts or loss of communications. The station may send an alert for immediate shutdown, if station operators need to work on communications, sending software or computer, etc. The station may be required to respond to an new address/port for Data Frames or disconnect without a new address. Regardless, when the station first connects to an address/port it must send the Data Format Frame. If the station ever gets an unrecoverable error on write, it should assume communications are lost, and start again at the Connection Manager. In the case of an active data connection the time-out on writes should be 10 minutes. This number is chosen to try to limit the number of broken connections due merely to slow software or communications. The IDC will similarly time-out on waiting to read from the station.

# Appendix A Checksum Algorithm

## *A.1* C FUNCTION FOR COMPUTING THE CHK2 CHECKSUM

```
#include <stdlib.h>
#include <math.h>

/*
    This function computes the GSE2.0 checksum used in the CHK2 line
*/
void
compute_checksum(signal_int, number_of_samples, _checksum)
        int     *signal_int;
        int     number_of_samples;
        int     *_checksum;
{
        int     i_sample;
        int     sample_value;
        int     modulo;
        int     checksum;

        int     MODULO_VALUE = 100000000;

        checksum = 0;

        modulo = MODULO_VALUE;

        for (i_sample=0; i_sample < number_of_samples; i_sample++)
        {
                /* check on sample value overflow */

                sample_value = signal_int[i_sample];

                if (abs(sample_value) >= modulo)
                {
                        sample_value = sample_value -
                                (sample_value/modulo)*modulo;
                }

                /* add the sample value to the checksum */

                checksum += sample_value;

                /* apply modulo division to the checksum */

                if (abs(checksum) >= modulo)
                {
                        checksum = checksum -
                                (checksum/modulo)*modulo;
                }
        }

        /* compute absolute value of the checksum */

        *_checksum = abs(checksum);
}
```

## *A.2* FORTRAN SUBROUTINE FOR COMPUTING CHK2 CHECKSUM

```
      subroutine
   compute_checksum(signal_int,number_of_samples,checksum)
c*****************************************************************
c  This subroutine computes GSE2.0 checksum used in the CHK2 line
c*****************************************************************
c  declarations
c
      implicit none
c
      integer*4 signal_int(*)        ! (input)  seismic signal
                                 !        (counts, integer values)
      integer*4 number_of_samples    ! (input)  number of used
   samples
      integer*4 checksum             ! (output) computed checksum
c
      integer*4 i_sample                ! index
      integer*4 sample_value            ! value of one sample after
                                        ! sample overflow check
      integer*4 modulo                  ! overflow protection value
c
      integer*4 MODULO_VALUE            ! overflow protection value
      parameter (MODULO_VALUE = 100 000 000)
c
c  initialize the checksum
c
      checksum = 0
c  use modulo variable besides MODULO_VALUE parameter to suppress
c  optimizing compilers to bypass local modulo division computation
c
      modulo = MODULO_VALUE
c
c  loop over all samples (counts, integer values)
c
      do i_sample = 1, number_of_samples
c
c  check on sample value overflow
c
          sample_value = signal_int(i_sample)
          if(abs(sample_value) .ge. modulo)then
              sample_value = sample_value-
     *        (sample_value/modulo)*modulo
          endif
c
c  add the sample value to the checksum
c
          checksum = checksum+sample_value
c
c  apply modulo division to the checksum
c
          if(abs(checksum) .ge. modulo)then
              checksum = checksum-
     *        (checksum/modulo)*modulo
          endif
c
c  end of loop over samples
      enddo
c
c  compute absolute value of the checksum
c
      checksum = abs(checksum)
c
      return
      end
```

O P E R A T I O N S