GCF Reference

Güralp Compressed Format

Document Number: SWA-RFC-GCFR

Issue F - December 2021

Designed and manufactured by
Güralp Systems Limited
3 Midas House, Calleva Park
Aldermaston RG7 8EA
England

Table of Contents

1 Proprietary Notice

The information in this document is proprietary to Güralp Systems Limited and may be copied or distributed for educational and academic purposes but may not be used commercially without permission.

Whilst every effort is made to ensure the accuracy, completeness and usefulness of the information in the document, neither Güralp Systems Limited nor any employee assumes responsibility or is liable for any incidental or consequential damages resulting from the use of this document.

1.1 Warnings, Cautions and Notes

Warnings, cautions and notes are displayed and defined as follows:

Warning: A black cross indicates a chance of injury or death if the warning is not heeded.

Caution: A yellow triangle indicates a chance of damage to or failure of the equipment if the caution is not heeded.

Note: A blue circle indicates indicates a procedural or advisory note.

1.2 Manuals and Software

All manuals and software referred to in this document are available from the Güralp Systems website: https://www.guralp.com/ unless otherwise stated.

In general, a manual can be accessed by appending it's part number to https://www.guralp.com/documents/ so, for example, the URL for the document SWA-RFC-BRPR will be https://www.guralp.com/documents/SWA-RFC-BRPR.

2 Introduction

GCF is a block-based format for storing seismic data. Each block is 1024 bytes long, including a 16-byte header, and is completely self contained. There are currently six distinct types of block: data (containing sample differences), textual status, CD status (a deprecated binary status format), unified status, strong motion results and byte pipe.

The 16-byte header contains a system ID and a stream ID, which, taken together, generally identify the digitiser producing the data and the channel with which it is associated. It also contains a start time in seconds (possibly with a fractional component), a sample rate (which is set to zero if the packet does not contain time-series data), a number of samples (or bytes of non-sample data), the format of the samples and some further ancillary information.

Related to the GCF format (although not directly part of it) are various transport layers:

- BRP (the Block Recovery Protocol, for use over serial links; see document SWA-RFC-BRPR); and
- Scream (the Scream protocol, for use with TCP or UDP over IP networks; see document SWA-RFC-SCRM).

2.1 Byte order

Unless otherwise stated, multi-byte integers are stored in big-endian (most significant byte first) format.

2.2 GCF Time

The time-stamp for the packet is always in UTC. It is given as two fields: the number of days elapsed since 1989-11-17 (the first day of operation of the GCF format) and the number of seconds elapsed since midnight (00:00:00Z). This allows it to represent correctly negative and positive leap seconds (a value of 86400 in the seconds field means 23:59:60Z, the clock time of a positive leap second). The time-stamp refers to the start of the data in the packet (i.e. it is the time-stamp of the first sample); it is left up to applications to compute the end time.

This format for time is used throughout the GCF standard and in many programs dealing with it; it is referred to as ‘GCF time’.

Some higher sample rates require data packets with time-stamps with fractional seconds. The fractional seconds are represented by a numerator, encoded into the header, and a denominator which can be determined from the sample rate. See section 4.1.2 on page 12 for more details.

2.3 Base 36 IDs

Two codes which are used to identify the source of GCF data, System IDs and Stream IDs, are encoded in the packet header using base 36.

To decode, take the integer read from the ID field in the header and consider it as a base 36 number (i.e. each digit spans the decimal range 0_{10} to 35_{10}). The base 36 digits are mapped to ASCII characters as follows: 0_{10} →0, 9_{10} →9, 10_{10} →A, 35_{10} →Z. Alternatively stated, each base 36 digit may be used as a (zero-referenced) character position offset into the string “`0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ`”.

The least significant base 36 digit is the remainder after dividing the integer by 36. The associated character is the last character of the encoded ID.

The integer can then be divided by 36, discarding the the fractional remainder, and the next least significant digit used to determine the next-to-last character of the ID.

This process is repeated until the integer itself hits zero. At this point, there are no more characters to decode (i.e. you can never have an ID that starts with the character 0).

To encode, take the first character of the string to be encoded and place the base 36 equivalent in an accumulator. The, as long as there are additional characters to encode, multiply the contents of the accumulator by 36 and add the base 36 equivalent of the next character from the string. When there are no more characters, the contents of the accumulator is the encoded string.

3 Common Header Fields

Each GCF packet has a four word (16-byte) header. The meaning of fields in the header depends on the type of block, but there is a lot of commonality. The fields whose interpretation does not change with packet type are described in this section.

For any type of packet, the System ID and Stream ID are determined in the same manner. The Stream ID can also be used to further differentiate between certain types of block. The sample rate, date, compression and number of samples are determined in a similar manner for different types of block, although the details vary slightly.

The first word encodes the System ID (in base 36 - see section 2.3 on page 5) and the digitiser type. The gain is also encoded here for certain digitiser types.

There are three different variants of the first word, determined by the contents of bit 31 (the Extended SysID flag) and bit 30 (the Double-extended SysID flag). The three variants are shown below:

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td colspan="31">System ID (up to 6 characters)</td></tr>
</table>

This variant (non-extended SysID) is described in section 3.1.1 on page 7.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td colspan="3">Gain</td><td>t</td><td colspan="26">System ID (up to 5 characters)</td></tr>
</table>

This variant (Extended SysID) is described in section 3.1.2 on page 8.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
</table>

This variant (Double-extended SysID) is described in section 3.1.3 on page 9.

In all cases, the *next* three words are laid out as follows:

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">№ of records</td></tr>
</table>

Note: The "`Timestamp (days)`" field comprises fifteen (15) bits and the "`Timestamps (seconds)`" field comprises seventeen (17) bits.

Note: The "`Numerator`" field comprises five (5) bits and the "`Comp`" field (the compression code) comprises three (3) bits.

Bit 31 of the second word is reserved and must be set to 0. The rest of the second word encodes the Stream ID in base 36 - see section 2.3 on page 5.

The third word contains the packet's timestamps, as described in section 2.2 on page 4 (and, for data packets with fractional-second offsets, in section 4.1.2 on page 12).

The fourth word contains five fields

- The 8-bit TTL (Tap-table look-up) field is described in section 4.1.3 on page 13.
- The 8-bit "Sample rate" is described in section 4.1.1 on page 11.
- The 5-bit "Numerator" field is described in section 4.1.2 on page 12.
- The 3-bit "Comp" field (compression code) is described in section 4.1.4 on page 13.
- The 8-bit "№ of records" field specifies the number of 4-byte data records. Each record holds one, two or four sample differences, depending on the compression code. The number of samples encoded in the packet is equal to the compression code multiplied by the contents of the "№ of records" field.

3.1 First words variants

The three different variants of the first word - the System ID word - are described in the following subsections. Each subsection begins with a test specification. The interpretation which follows only applies if the test specification is satisfied.

3.1.1 Non-extended SysID

This early format is not typically used by current Güralp digitiser models but must be accepted by any compliant software implementation.

Test:

- Bit 31 of the first word (System ID) is clear (0)

The interpretation of the first four words is as shown below:

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td colspan="31">System ID (up to 6 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">№ of records</td></tr>
</table>

There is no information in the first word other than the System ID itself. Decoding the base 36 IDs is explained in section 2.3 on page 5. The type of digitiser that produces this sort of block cannot be determined from the GCF block but it is typically a DM24 Mk2.

3.1.2 Extended SysID

This format is typically used by Güralp CD24 and DM24 digitisers.

Test:

- Bit 31 of the System ID word is set (1) and
- Bit 30 of the System ID word is clear (0)

The interpretation of the first four words is as shown below:

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>1</td><td>0</td><td colspan="3">Gain</td><td>t</td><td colspan="26">System ID (up to 5 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">№ of records</td></tr>
</table>

Bit 26 of the first word, *t*, identifies the digitiser type. It is set (1) if the digitiser is a CD24 and clear (0) if the digitiser is a DM24.

Note: Early versions of the CD24 firmware set the extended SysID bit but left the 'Type' bit as 0, so the digitiser would be misidentified as a DM24 Mk3. This was fixed in release 273 of the CD24 firmware.

Bits 29, 28 and 27 are used to specify the digitiser gain (where appropriate) as follows:

Bits 29-27	Gain setting
000	No variable-gain stage fitted
001	x1
010	x2
011	x4
100	x8
101	x16
110	x32
111	x64

3.1.3 Double-extended SysID

This format is typically used by Güralp Minimus and Affinity digitisers.

Test:

- Bit 31 of the System ID word is set (1) and
- Bit 30 of the System ID word is set (1)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td><u>t</u></td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">№ of records</td></tr>
</table>

Bit 26 of the first word, *t*, identifies the digitiser type. It is set (1) if the digitiser is a Minimus and clear (0) if the digitiser is an Affinity.

The digitiser gain is encoded in bits 29, 28 and 27 but the interpretation depends on the model of digitiser.

Note: Early versions of the Minimus firmware did ***not*** set the gain bits in the first word of the packet header.

Bits 29, 28 and 27 of the first word are used to specify the digitiser gain as follows:

Bits 29-27	Gain setting - Affinity	Gain setting - Minimus
000	*not specified - check config*	*not specified - check config*
001	x1	x1
010	x2	x2
011	x4	x4
100	x8	x8
101	x16	x12
110	x32	*not used*
111	x64	*not used*

3.2 Digitiser type summary

The digitiser type can be determined from bits 31, 30 and 26 of the first word of the packet header:

Bit 31	Bit 30	Bit 26	Digitiser type
0	*x*	*x*	unknown - probably DM24 Mk2
1	0	0	DM24
1	0	1	CD24
1	1	0	Affinity
1	1	1	Minimus

Note: Early versions of the CD24 firmware set bit 31 but left bit 26 as 0, so the digitiser would be misidentified as a DM24 Mk3. This was fixed in release 273 of the CD24 firmware.

3.2.1 Range of valid SysIDs

Because the number of bits used to encode the SysID is different for different packet formats, the range of valid SysIDs also differs. The ranges are given in the table below. Note that

- Valid characters are the digits '`0`' to '`9`' and the uppercase characters '`A`' to '`Z`'
- No SysID may start with the character '`0`';
- In the ranges given below, digits sort before letters.

Digitiser type	Range of valid SysIDs
DM24 MkII	All 1-, 2-, 3-, 4- and 5-character IDs plus `100000` to `ZIK0ZJ`
DM24 MkIII and CD24	All 1-, 2-, 3-, 4- and 5-character IDs plus `100000` to `13YDJ3`
Affinity and Minimus	All 1-, 2-, 3-, 4-character IDs plus `10000` to `18Y67`

4 Decoding Blocks

With the common header details determined, the next step is to categorise the block. The various block types are described in the following sub-sections.

4.1 Data block

Test:

- Sample rate (the second byte of the fourth header word) is non-zero

Data blocks contain raw time-series data, encoded as a series of integer differences. The data samples follow immediately after the header.

4.1.1 Sample Rate

The value of the sample rate field (the second byte of the fourth header word) generally indicates the number of samples per second. However, because it is an unsigned 8-bit integer, it cannot directly represent sample rates higher than 255 sps or lower than 1 sps. Therefore, some "unlikely" values have been chosen to represent such sample rates. These are shown in the table below.

Code value (from header)	**Actual sample rate (samples per second)**	**Time fractional offset denominator** (see section 4.1.2 on page 12)
157	0.1	not applicable
161	0.125	not applicable
162	0.2	not applicable
164	0.25	not applicable
167	0.5	not applicable
171	400	8
174	500	2
175	800	16
176	1000	4
179	2000	8
181	4000	16
182	625	5
191	1250	5
193	2500	10
194	5000	20

4.1.2 Fractional-second time-stamps

When a sample rate higher than 250 sps is in use, it is possible that a block will need to be started at a time that does not lie on an integral second boundary. To represent such times, a fractional-second offset is added to the time-stamp. The numerator of the fraction is encoded in the header; the denominator is given in the table in section 4.1.1 on page 11.

> **Note:** The numerator originally used *four* bits - 12 to 15 - of the fourth header word). When a digitiser capable of 5000 sps was released, this proved inadequate and an extra bit was required. Bit 11 of the fourth header word was co-opted for this purpose. Note that it encodes the high-order bit - the MSB - of the numerator, despite its position.

If the fourth header word is `W`, the numerator is equal to

```
( ( W & 0x08 ) << 1 ) + ( ( W & 0xF0 ) >> 4 )
```

where `&` is the bit-wise logical-AND operator, `<<` and `>>` are the left and right bit-shift operators and `0x08` and `0xF0` are numeric constants expressed in hexadecimal notation.

Dividing the numerator by the denominator gives a fraction between zero and one, which is the offset (in seconds) from the start time specified in the third header word.

For example, when we decode this packet header

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>1</td><td colspan="5">Reserved</td><td colspan="21">System ID</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">0 0000 1110 0001 0000 (=3600₁₀)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">10111111 (=191₁₀)</td><td colspan="5">0001 0</td><td colspan="3">001</td><td colspan="8">11111010 (=250₁₀)</td></tr>
</table>

- The sample rate code is 191, corresponding to 1250 sps
- From the table, the "time fractional offset denominator" for 1250 sps is 5
- The numerator (computed as described above) is 1
- The time fractional offset is, therefore, 1 ÷ 5 = 0.2 seconds
- The time-stamp of this packet (ignoring the date) is, therefore, 3600 seconds + 0.2 seconds past midnight - i.e. 01:00:00.2

4.1.3 TTL and decimation

The "Tap-table look-up" (TTL) field is the first byte of the fourth word of the header. It is an index into one of several a digitiser-specific tables of decimation chain configurations. Knowing the digitiser type, the sample rate of the data and the configuration of the decimation chain allows you to determine the precise series of decimation filters through which the data have passed.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">Nº of records</td></tr>
</table>

The tap tables themselves are available from our web site at the following addresses:

- **DM24 MkII** - the TTL value will be between 17–59 inclusive. The relevant tap-table is www.guralp.com/documents/SWA-D24-2D00.
- **DM24 MkIII** - The TTL value will be between 1–95 inclusive. The relevant tap-table is www.guralp.com/documents/SWA-D24-3D00.
- **CD24** (and **EDU**) - The TTL value will be between 1–226 inclusive. The relevant tap-table is www.guralp.com/documents/SWA-C24-D000.

4.1.4 GCF Compression

The packet header is followed immediately by the sample data. In order to efficiently represent the time-series, the data are not presented as raw samples but as one initial (absolute) value followed by a sequence of signed differences.

The GCF format supports variable compression, based on the magnitude of the differences. The number of bits used to encode each difference in a packet is specified by the three-bit Compression code, contained in the fourth word of the header.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">Nº of records</td></tr>
</table>

- Code 4 is used to store up to 1000 difference values in 8-bit fields, four per record. Each value must be between -128 and +127 inclusive.

- Code 2 is used to store up to 500 difference values in 16-bit fields, two per record. Each value must be between -32768 and +32767, inclusive.
- Code 1 is used to store up to 250 difference values in 32-bit fields, one per record.

The number of samples encoded in a packet can be computed by multiplying the Compression code and the contents of the "№ of records" field (the fourth byte of the fourth word of the header), which specifies how many four-byte data records are present in the packet.

The "№ of records" has a maximum value of 250, allowing a maximum of 1000 samples per packet.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">Sample rate</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">№ of records</td></tr>
</table>

The meanings of the compression codes and the calculation used to determine the number of samples are summarised in the following table:

Compression	Description	Number of samples
1	Signed 32-bit differences	1 × № of records
2	Signed 16-bit differences	2 × № of records
4	Signed 8-bit differences	4 × № of records

4.1.5 Time-series data

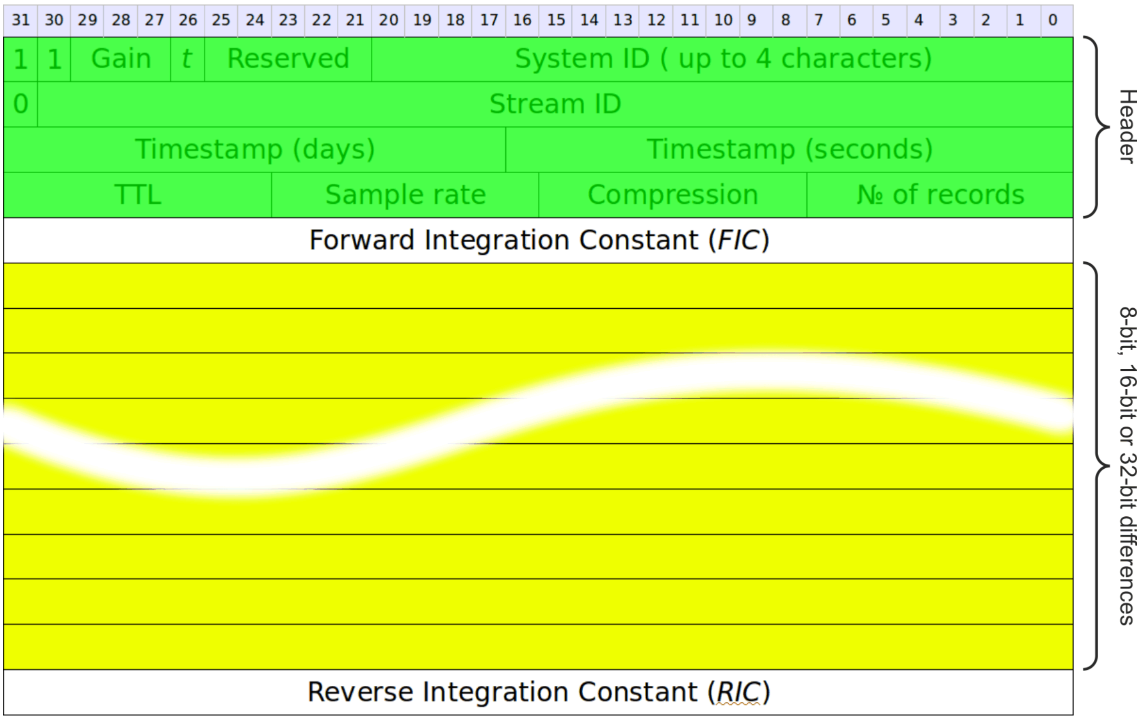
The absolute value of the first sample is given in 32-bit signed integer format in the first word of the data. This value is known as the "Forward Integration Constant", or *FIC*. The time-stamp of this first sample is the time-stamp that is encoded in the packet header, as described in section 2.2 on page 4 and,for time-stamps with fractional seconds, in section 4.1.2 on page 12.

Following the *FIC* are *n* signed differences (where *n* is the number of samples in the block, computed as described above). The first difference is always zero because the first encoded sample *is* the *FIC*. Each difference is represented by 1, 2 or 4 bytes, depending on the Compression code: code 1 signifies 4 bytes per difference; code 2 signifies 2 bytes per difference and code 4 signifies 1 byte per difference.

A decoder should initialise an accumulator with the *FIC* and then add each consecutive difference in order to obtain the absolute value at each time step.

After the differences and acting as a final check, the block contains the absolute value of the last sample (the “Reverse Integration Constant”, or *RIC*). The *RIC* is a signed 32-bit integer stored immediately after the last difference. The value in the accumulator should match the *RIC*. If it does not, a transmission error has occurred.

The layout of a data packet is, therefore, as follows:



The largest permitted GCF packet contains 250 records. This can hold up to 250, 500 or 1000 sample differences, depending on the Compression code. Together with the four-word header and a word each for the FIC and RIC, the total packet is 256 words, or 1 kB long.

4.2 Status block

Test:

- Sample rate (the second byte of the fourth header word) is equal to zero
- Compression code (bits 8, 9 and 10 of the fourth header word) is 4
- The rightmost two characters of the stream ID are “00” (i.e. the value of the second header word, modulo 1296, is zero)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID decodes to "XXXX00"</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">00000000 (=0_{10})</td><td colspan="5">Numerator</td><td colspan="3">100</td><td colspan="8">№ of records</td></tr>
</table>

A status block contains textual information such as state of health in ASCII (although there are various control codes and other unprintable characters that may occur, so the data must be sanitised before displaying on a terminal etc.).

The status information is written to the packet immediately after the header. The length of the status text, in bytes, is determined by multiplying the unsigned 8-bit value of the № of records field by 4.

4.3 Unified status

Test:

- Sample rate (the second byte of the fourth header word) is equal to zero
- Compression code (bits 8, 9 and 10 of the fourth header word) is 4
- The rightmost two characters of the stream ID are “01” (i.e. the value of the second header word, modulo 1296, is 1)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID decodes to "XXXX01"</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">00000000 (=0_{10})</td><td colspan="5">Numerator</td><td colspan="3">100</td><td colspan="8">№ of records</td></tr>
</table>

A Unified Status block is a binary, machine-readable block of status information that is emitted every second by a digitiser (when this mode is enabled). The length of the data, in bytes, is determined by multiplying the unsigned 8-bit value of the № of records field by 4.

See the document www.guralp.com/documents/SWA-RFC-UNIS for the full specification of Unified Status blocks.

4.4 Strong motion

Test:

- Sample rate (the second byte of the fourth word) is equal to zero
- Compression code (bits 8, 9 and 10 of the fourth header word) is 4
- The rightmost two characters of the stream ID are “SM” (i.e. the value of the second header word, modulo 1296, is 1030)

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID decodes to "XXXXSM"</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">00000000 (=0₁₀)</td><td colspan="5">Numerator</td><td colspan="3">100</td><td colspan="8">№ of records</td></tr>
</table>

A Strong Motion packet contains the results of strong motion calculations, computed every second by a digitiser (if this mode is enabled). The length of the data, in bytes, is determined by multiplying the unsigned 8-bit value of the № of records field by 4.

See the document www.guralp.com/documents/SWA-RFC-STMN for the full specification of Strong Motion packets.

4.5 Byte pipe

Test:

- Sample rate (the second byte of the fourth word) is equal to zero
- Compression code (bits 8, 9 and 10 of the fourth header word) is 4
- The rightmost two characters of the stream ID are “BP” (i.e. the value of the second header word, modulo 1296, is 421)

<table>
<tr><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>t</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID decodes to "XXXXBP"</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">00000000 (=0₁₀)</td><td colspan="5">Numerator</td><td colspan="3">100</td><td colspan="8">№ of records</td></tr>
</table>

Some digitisers have a 'byte pipe' mode, used to package arbitrary binary data read from an auxiliary serial port into a GCF packet for transmission to a host. The length of the packaged data, in bytes, is given by multiplying the unsigned 8-bit value of the № of records field by 4. The data are transmitted raw, with no encoding.

4.6 CD status (deprecated)

Test:

- Sample rate (the second byte of the fourth word) is equal to zero
- The rightmost two characters of the stream ID are "CD" (i.e. the value of the second header word, modulo 1296, is 445)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td colspan="3">Gain</td><td>*t*</td><td colspan="5">Reserved</td><td colspan="21">System ID (up to 4 characters)</td></tr>
<tr><td>0</td><td colspan="31">Stream ID decodes to "*XXXX***CD**"</td></tr>
<tr><td colspan="15">Timestamp (days)</td><td colspan="17">Timestamp (seconds)</td></tr>
<tr><td colspan="8">TTL</td><td colspan="8">00000000 (=0_{10})</td><td colspan="5">Numerator</td><td colspan="3">Comp</td><td colspan="8">№ of records</td></tr>
</table>

This format is deprecated. Use unified status instead. When CD status packets are enabled, the digitiser will emit one such packet each second.

5 Revision history

2007-09-28	A	New document
2009-09-22	B	Update formatting to Güralp standard. Document variable gain header fields. Remove detailed CD status documentation.
2011-01-03	C	Clarification of “№ of records”
2018-03-14	D	Re-formatted to reflect new branding Re-write to include new features to support Minimus and Affinity digitisers. Improved explanations and graphics.
2020-03-03	E	Corrected contradictory bit-width specifications in the time-stamp header field explanation. Corrected description and clarified algorithm for the numerator of fractional time-stamps.
2021-12-03	F	Added section on ranges of valid SysIDs