INTERMAGNET Technical Note

Title: INTERMAGNET Web Data Delivery

Document number: TN9	**Version number:** 1.0	**Creation date:** 09/07/2018

Related documents: *(Links to other documents with additional information, previous documents that this document supersedes, and later documents that supersede this one.)*

Keywords:

Lead author: S. Flower	**Contributors:**

Purpose of document: Acceptance of Discussion Document 2

Terms of Reference

1. Describe an INTERMAGNET protocol for the exchange of data using http

Date due:	**Date submitted:** **Date revised:**

Outcomes: *(Actions assigned and decisions made)*

Note for information

INTERMAGNET Technical Notes are designed to record information on how to accomplish technical tasks. They will often describe alternative approaches, and make recommendations on the practice to be adopted by INTERMAGNET. In many cases a Technical Note will be used as the basis for an entry in the Technical Manual.

The Secretary of the Operations Committee assigns the Document Number, maintains the metadata on the cover sheet, and keeps an index of all documents. Documents to be archived and indexed will be given integer version numbers. Working documents will have fractional version numbers. Only the lead author may edit the main text.

Exchange of INTERMAGNET data via the web

This note describes a mechanism for upload of INTERMAGNET data from INTERMAGNET Observatories (IMOs) to Geomagnetic Information Nodes (GINs) using the http protocol.

1. Summary

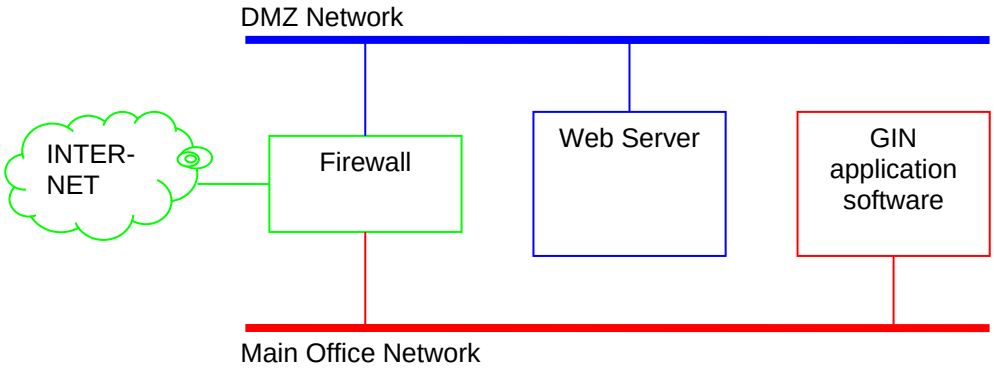
To be successful a protocol for data exchange must be designed with security in mind. Most organisations already allow web traffic into an area of their network. This note describes using the web server as a data cache, allowing IMOs to upload data into the cache and GINs to download data from the cache.

2. Choice of underlying transport service

Modern organisations must protect themselves from the dangers of connecting to the global INTERNET. The data exchange service described by INTERMAGNET must exist in a world of firewalls and complex security policies. In implementing the new data exchange service, it is unlikely that significant changes to an institute's security policy can be made. In designing the new service, it makes sense to try to use protocols that are already widely in use. Most organisations already allow e-mail traffic in and out of their sites. In addition most organisations operate web servers, allowing web traffic into their site (often into a special area of their network, separate from other services). The network protocols that support these services are SMTP (for e-mail) and HTTP/HTTPS (for web services).

The original INTERMAGNET data delivery service relies on e-mail for its transport mechanism. E-mail is not suitable for exchange of real-time data and larger data files (such as 1-second data), first because it is cumbersome to use, secondly because it is not always reliable, thirdly because it may not be fast enough to support the demand for real-time data exchange.

This note describes a system that can be built around an organisation's web servers. The exact details of the implementation are not described. The system could be implemented using a 3rd generation language and the Common Gateway Interface (CGI) provided by nearly all web servers. Alternatively it could be implemented in Java Server Pages (JSP) or similar technologies, which are again widely used. **BGS have implemented the protocol in JSP - this implementation is available to other GINs on request.** This system assumes (but does not require) that an organisation's web server will be segregated (conventionally in an area called a Demilitarized Zone, DMZ) from the part of the network where data processing applications will run. A typical network layout is shown below.



In order to implement this system an organisation's security policy must allow the following:

1. The organisation must have access to a web server.

2. Staff responsible for INTERMAGNET data must be able to deploy an application on the web server.
3. The web server must allow file upload from the INTERNET.
4. It must be possible for the computer running the 'GIN application software' (the computer where INTERMAGNET data processing takes place) to download files from the web server.

Of these four requirements the 2nd and 3rd are perhaps the most likely to cause a problem.

3. INTERMAGNET Data Exchange Protocol

The exchange of data described here relies on the file upload capabilities of most modern web servers (the file upload mechanism is described in RFC1867 "Form-based File Upload in HTML" - see www.ietf.org/rfc/rfc1867.txt). As a very simple description:

1. IMOs upload their data into an application on the web server. The application is called **IDEA** (INTERMAGNET Data Exchange Application).
2. IDEA indexes and caches the data on the web server, making it available for download.
3. IDEA provides a search mechanism by which a GIN can find out what data is in the cache.
4. IDEA provides a download form which allows GIN application software (at any GIN, not just the GIN hosting the web server) to download data from the cache.
5. The responses which IDEA makes to these actions are standardised, making it easy to automate the processes of uploading and downloading data.

These four parts are now described in greater detail.

3.1 Protocol Section 1 - IMO Data Upload

IMOs connect to the IDEA web server in one of three ways:

1. Manually using a standard web browser. Once connected the file upload is very simple, similar to the method used by online mail services to upload files as attachments when sending mail.
2. Automatically using a command line based tool that supports file upload. One such tool is CURL (**http://curl.haxx.se/**) which is available for SUN Solaris, Linux and Windows.
3. Automatically using their own software. File upload to a web server is supported by Java and other languages.

The protocol will accept data in the following formats:

1. One-second data in IAGA 2002 format.
2. One-minute data in IAGA 2002 format for entry into the GIN's existing minute-mean data set (optional – a GIN may decide not to accept data in this format).
3. One-minute data in IMF V1.22 format for entry into the GIN's existing minute-mean data set (optional – a GIN may decide not to accept data in this format).

Data files may contain a maximum of 24 hours data. There is no minimum quantity of data, but an IMO may not make more than 1440 uploads per day. The same file may not be uploaded more than once. These limits are imposed to help prevent Denial Of Service (DOS) attacks.

On successful upload, a message will be returned, either in plain text or HTML format (the format is under control of the user – see the 'Protocol Section 3' and the section on URLs for details of setting the format). The HTML response is designed to be easy for an interactive user to understand. The plain text response is designed to be easy to process from software and will consist of either a two or three line message. If the data was loaded and there were no problems removing old elements from the cache then this message will be sent:

```
Success
```

```
Data loaded OK, data file id = <ID>
```

If the data was loaded, but there were problems removing old cache elements, then this message will be sent:

```
Warning
Data loaded OK, data file id = <ID>
<Cache warning message>
```

If there was an error, then the response will conform to the description in section 5 of the protocol.

3.2 Protocol Section 2 – Internals of the Web Server Cache and Index

The first operation that IDEA performs when a file is uploaded is to parse the file to check:

1. The file name and contents are correct
2. The file conforms to the maximum data size and maximum number of file upload rules described in section 1.

These checks are designed to protect this system from DOS attacks and other abuse.

Once a file has been verified it is assigned a unique ID number. The ID number increments for each new file uploaded. The ID number never gets smaller. ID numbers can be used to sort files by their time of arrival on the server. This is an important feature for GIN applications that will download the data – they may maintain a copy of the ID number of the most recently processed file. Subsequently they may query IDEA, asking it to list all files with an ID greater than the number they supply. This mechanism allows a GIN to request copies of all data received since its previous connection to IDEA.

To ensure that the ID number cannot 'run out' it is implemented as a (signed) 64-bit integer, starting at zero. This allows for values that range from 0 to 2305843009213693952. If IDEA were to receive the maximum number of daily uploads from 100 IMOs, the ID would last for 175482725206 years before being reset.

In addition to the ID number, IDEA gives a number of other attributes to each uploaded file:

- The date/time that the file was uploaded (information taken from the computer's clock)
- The station code of the data (information parsed from the data file)
- A code for the format of data (IMF, IAGA 1-second, IAGA 1-minute).
- A code for the type of data (reported, adjusted, provisional, definitive, test)
- The start time of the data in the file (information parsed from the data file)
- The duration of the data in the file (information parsed from the data file)

All these attributes are available to the user when searching the cache.

The maximum amount of disk space occupied by the cache is a configurable parameter of IDEA, ensuring that the application uses a predictable amount of disk space on the web server.

3.3 Protocol Section 3 – Searching the Cache

IDEA supports a web form that allows searches to be made in the data file cache. Data may be filtered using the following fields:

Field	**Description**
Station code	Leave blank to include all stations in the results, otherwise a comma separated list of IAGA station codes to include in the results.
Format of data	A special code allows all data formats to be retrieved.
Type of data	i.e. Reported, adjusted, etc. A special code allows all data types to be retrieved.

Earliest date of upload	Data uploaded before the specified date is not included in the results. Leave blank for no limit on earliest date.
Latest date of upload	Data uploaded after the specified date is not included in the results. Leave blank for no limit on latest date.
Earliest date of data	Data stating before the specified date is not included in the results. Leave blank for no limit on earliest date.
Latest date of data	Data ending after the specified date is not included in the results. Leave blank for no limit on latest date.
Smallest ID number	Data with an ID number smaller than the given number will be removed from the results. Leave blank for no limit.
Largest ID number	Data with an ID number larger than the given number will be removed from the results. Leave blank for no limit.
Maximum number of results	This allows you to limit the number of results returned. For example, set this field to 10 and only the first 10 results (provided there are at least 10 results) will be returned. Leave blank for no limit.

As described in section 2, a GIN may search with all fields blank, except the 'smallest ID number', which is set to the number following the most recently processed data file – this search will return all newly uploaded files.

Similarly the 'Maximum number of results' field provides a means to find the earliest ID number in the cache – simply set this field to 1 and leave all other fields empty.

Search results are presented in a predictable form to make it simple to automatically process the results. For each data file returned in the search the following information is given:

- The data file's unique ID number.
- The station code for the data.
- The format of the data.
- The type of data in the file.
- The start time of the data in the file.
- The duration of the data in the file.

Two different formats are available (selectable by the user) for the search results.

Format	**Suggested use**
Plain text	Automatic data processing
HTML	Interactive searching of the cache

Both formats share the following features:

- Dates and times are represented in the form YYYY-MM-DD^hh:mm:ss where YYYY is the four digit year, MM is the month in the year (01-12), DD is the day in the month (01-31), ^ is a single space character, hh is the hour of the day (00-23), mm is the minute of the hour (00-59), ss is the second of the minute (00-59). All fields in the date/time are fixed width (years are 4 characters wide, all other fields 2 characters wide).
- Dates and times should not include time zone or daylight savings offsets, but should be given in UTC.

3.3.1 Plain Text Search Results

Each data file is represented by a row in the returned text file. Each row has the following fields:

<ID> <Station code> <Type-code> <Format-code> <Data date/time>
<Data duration> <Upload date/time>

All fields have a fixed width and are separated by a single space. Numbers less than the maximum width are padded to the left with spaces or zeroes. Alphanumeric strings less than the maximum width are padded to the right with spaces.

The <ID> has a width of 19 characters. The <Station code> has a width of 3 characters. The <Type-Code> and <Format-code> (described in the tables below) have a field width of 1. The <Data duration>, which is specified in seconds, has a width of 5 (allowing for a maximum value of 86400 – or 24 hours). Date / time formats have already been described.

Data format code	Meaning
A	IMF (1-minute) data
B	IAGA 2002 1-minute data
C	IAGA 2002 1-second data
Z	All data formats (used during searching)

Data type code	Meaning
R	Reported
A	Adjusted
P	Provisional
D	Definitive
T	Test
Z	All data types (used during searching)

There is no header on the search results.

3.3.2 HTML Search Results

The data is returned in an HTML table, with a header, making it easy for an interactive user to read. Each data file is represented by a row in the table. The fields in the row are hyperlinked, allowing the data file to be downloaded using the hyperlink.

Precise formatting definitions are not given, since this format is not designed for processing by computer programs.

3.4 Protocol Section 4 – Downloading Files from the Cache

Data files may be downloaded from the IDEA cache for processing. IDEA itself does not perform any data processing and must not make any changes to uploaded data files. Any GIN (not just the one hosting IDEA) may download data files, thus allowing GINs to share data.

A well-defined URL pattern allows applications to construct the URL needed to download any particular data file. The query string portion of the URL contains only a single parameter – the unique ID for the data file. For further information see the section on URL details.

3.5 Protocol Section 5 – Response to errors

Using any of the services that IDEA provides may result in an error. As with all other parts of the system, responses may be formatted as HTML or plain text. HTML responses are designed to be clear to read for an interactive user. Plain text responses are ‘well formatted’ – that is to say that they correspond to a set or rules that make them easy to parse with a simple computer program.

The first line of the response indicates the level of fault. It may contain one of the following words:

Word	Description
Success	The operation succeeded.
Information	The operation succeeded and there is some additional information.
Warning	The operations succeeded, but there was a problem with a peripheral part of the system.

Error	The operation failed.
Exception	The operation failed with an unforeseen problem. A full description of the software fault is included.
Fatal	The entire IDEA application is not working.

The second line of the response is a message describing the problem. The response may include subsequent lines with further detail.

4. URL Details

This section maps the protocol's services to specific URLs. The base of the URL will be dependent on the web server where IDEA is running. For example, at BGS, the base URL will be similar to http://app.bgs.ac.uk/GINFileUpload/. Below this base URL, IDEA will use the following relative URLs:

URL	**Description**
Index.html	Help information for the GIN file upload system. Links to the system's interactive forms.
HelpAutomate.html	A page containing information (duplicated from this document) describing how to automate the process of uploading and downloading data.
UploadForm.html	An interactive form allowing the user to upload a data file from their computer.
Cache?Request=Upload& Format=<Format>& File=<GIN-data-file>	The page where data is loaded into the cache. <Format> is one of 'plain' or 'html' (default is plain) – it applies to the response message indicating whether the upload was successful or not.
SearchForm.html	An interactive form allowing the user to search the cache contents.
Cache?Request=Search& Format=<Format>& FormatCode=<FormatCode>& TypeCode=<Type>& IAGACode=<code>& EarliestUpload=<Date>& LatestUpload=<Date>& EarliestData=<Date>& LatestData=<Date>& SmallestID=<ID>& LargestID=<ID>& MaxNResults=<Number>& TypeCode=<Code>	This page returns a list of files (may be zero length) that meet the given criteria. None of the elements in the query string are compulsory. Codes for <typeCode> and <formatCode> are defined in the 'plain text search results' description. <Format> is one of 'plain' or 'html' (default is plain). Dates and times are represented in the form YYYY-MM-DD^hh:mm:ss where YYYY is the four digit year, MM is the month in the year (01-12), DD is the day in the month (01-31), ^ is a single space character, hh is the hour of the day (00-23), mm is the minute of the hour (00-59), ss is the second of the minute (00-59). Dates and times should not include time zone or daylight savings offsets, but should be given in UTC.
DownloadForm.html	An interactive form allowing the user to download a data file from the cache.
Cache?Request=Download& Format=<Format>& ID=<ID>	Download the given data file. <Format> is one of 'plain' or 'html' (default is plain).

5. Further Security Considerations

A number of security features are built into the protocol (such as limiting and parsing data files to prevent DOS attacks).

The IDEA will require configuration. To prevent any danger of remote attack, the configuration must be implemented using a file that is not accessible through the web server. IDEA must not allow any ‘remote management’ of the configuration.

As an option the upload, search and download pages can be password protected, to prevent access by un-authorised users.

Another possible option would be to allow access to these pages only from a limited set of IP addresses – this option is unwise because IMOs may not always have static IP addresses.