**The** **GLIDEnumber READ API interface.**

This chapter describes how to programmatically retrieve the information stored in the GLIDEnumber database in either Extensible Markup Language (XML) or JavaScript Object Notation (JSON ) formats.

Using this interface, as software program can access the data providing all the indications required to mimic the interactive capabilities of the GLIDEnumber website by exchanging XML or JSON documents over a secure HTTPS connection.

With this API a computer program can transmit the specification of a subset of data desired, and then receive and interpret the resulting data.

This interface is particularly useful for those software tools that require searching the database for the existence of an event within the database, a process that is required before attempting to create a new disaster event in the system. It is also useful for software used for result monitoring and analysis of disaster data where the GLIDE number standard is used, or simply for retrieval and display purposes.

This API complements the separate GLIDEnumber CREATE XML/JSON API interface, described in the second part of this document.

Before using it, please be aware that the owner of the software MUST confirm in written to ADRC (email to [gliderep@adrc.asia](mailto:gliderep@adrc.asia)) that the owner agrees and will comply with the following:

**The following fields are RESERVED and should NOT be published under any circumstance:**   
  killed   
  injured   
  affected   
  homeless

destroyedhouses

**They are internal measures, not guaranteed to be precise and may not appear at all, or be removed or changed in the future.**

**GLIDE XML/JSON Application Program Interface API**   
    
A process can retrieve one single GLIDE record, retrieve a specific subset of the database or obtain the entire data set and receive the response in either XML or JSON format for a specific subset that matches the following criteria invoking the following URLs, for XML or JSON formats respectively:   
    
<https://www.glidenumber.net/glide/xmlglideset.jsp>?.... [parameters]   
<https://www.glidenumber.net/glide/jsonglideset.jsp>?.... [parameters]

Where parameters can be (ALL parameters are optional):   
    
**level0**=N  (n=1 Africa, 2 Americas, 3 Asia, 4 Europe, 5 Oceania)   
**level1**=ISO (ISO= ISO3 standard country code)

*(Please note level0 and level1 are mutually exclusive. If both are specified level1 will take precedence)*

**ymin**= minimum Latitude of bounding box**xmin**= minimum Longitude of bounding box**ymax**= maximum Latitude of bounding box**xmax**= maximum Longitude of bounding box

**events**= event type (hazard) code list, separated by commas. See appendix A for the list of approved event type codes

**fromyear**=YYYY, **frommonth=MM, fromday=DD** startrange of dates to consider. If month or day are missing they will be assumed as 1 in both cases

**toyear=**YYYY**, tomonth=MM, today=DD** end ofrange of dates to consider. If month or day are missing they will be assumed as 12 and 28/29/30/31 (depending on the last day of the month).

**keywords**=   search for words (this parameter must be URL-encoded).

**glide**= get one specific GLIDE record. GLIDE is of the form ISO-SSSSSSSS-EV

**Sortby**= X

Where X can be

0 GLIDEnumber   
1 Country, Event, Date   
2 Country, Date, Event   
3 Event, Country, Date   
4 Event, Date, Country   
5 Date, Country, Event   
6 Date, Event, Country   
    
For pagination:   
**nStart**= retrieve results starting at hit # nStart  
**maxhits** = max records to retrieve  
    
Example:

XML format example:    
<https://www.glidenumber.net/glide/xmlglideset.jsp?level0=3&events=FL&fromyear=2003>

Equivalent for JSON format:

<https://www.glidenumber.net/glide/xmlglideset.jsp?level0=3&events=FL&fromyear=2003>   
   
Will return all Flood events in Asia after Jan 1, 2003   
    
The response XML (or JSON) stream will be of the form

|  |  |
| --- | --- |
| **XML format** | **JSON format** |
| <glideset>    <glide>    .... One glide event    </glide>    <glide>    .... One glide event    </glide>   ....   <glide>    .... One glide event    </glide>  </glideset> | {  " glideset":  [  { ...one glide event},  { ...one glide event},  ....  { ...one glide event}  ]  } |

    
As can be seen, the underlying search engine (same as in the search page) processes each request with as many parameters as needed. The reader may want to check the search page and experiment with the potential results selecting different parameter.   
    
Each glide event looks like:

|  |  |
| --- | --- |
| **XML format** | **JSON format** |
| <glide>    <event>**FL**</event>    <number>**2004-000098**</number>    <geocode>**DOM**</geocode>    <location>**Ramon Santana, east of the**  **Dominican Republic**</location>    <year>**2004**</year>    <month>**9**</month>    <day>**18**</day>    <time />   < duration>**0**</duration>    <magnitude />    <source>**Reuters AlertNet**</source>    <killed>**7**</killed>    <injured />    <affected />    <homeless>**2200**</homeless>  <destroyedhouses>333</destroyedhouses>  <latitude>41.3275459</latitude>  <longitude>19.8186982</longitude>    <docid>**16958**</docid>    <status>**A**</status>    <comments>***Jeanne killed seven people, forcing 22,000 people to flee their homes and flooded parts of the Dominican Republic***</comments>  <id>6</id>  <idsource>IFRC</idsource>   </glide> | {  "event":"**FL**",  "number":"**2004-000098**",  "geocode":"**DOM**",  "location":"**Ramon Santana, east of the**  **Dominican Republic**",  "year":"**2004**",  "month":"**9**",  "day":"**18**",  "time":""  "duration":"**0**",  "magnitude":"",  "source":"**Reuters AlertNet**",  "killed":"**7**",  "injured":"",  "affected":"",  "homeless":"**2200**",  "destroyedhouses":"333",  "latitude":"41.3275459",  "longitude":"19.8186982",  "docid":"**16958**",  "status":"**A**",  "comments":"**Jeanne killed seven people, forcing 22,000 people to flee their homes and flooded parts of the Dominican Republic**",  id=6,  idsource=”IFRC”  }  ***IMPORTANT NOTE****: the actual order of the fields in the JSON string may be different and vary according to certain conditions in the server. No line feeds or carriage returns are produced either.* |

 The field **docid** can be used to generate a fast link to the disaster page in Glide:   
    
<https://www.glidenumber.net/glide/public/search/details.jsp?glide=16958>   
    
    
Where the parameter value (16958 in the example) is the docid.

REMINDER NOTE:   
    
The following fields are RESERVED and should NOT be published under any circumstance.   
  killed   
  injured   
  affected   
  homeless

destroyedhouses

**They are internal measures, not guaranteed to be precise and may not appear at all.**   
 

**The GLIDEnumber CREATE API interface.**

This chapter describes how to programmatically create, extend and modify disaster records to be stored in the GLIDEnumber database, using either Extensible Markup Language (XML) or JavaScript Object Notation (JSON ) formats.

Using this interface, as software program can create new GLIDE numbers, extend an existing disaster record to another country or to modify the contents of the fields of an existing GLIDE number record.

Access to data requires providing all the indications required to mimic the interactive capabilities of the GLIDEnumber website by exchanging XML/JSON documents over a secure HTTPS connection.

With this API a computer program can transmit the specification of the changes required, and then receive and interpret the results of the operation.

This interface is intended for software systems that can detect or receive information in near real time about disasters that should be recorded in the GLIDE database.

This API complements the separate GLIDEnumber READ XML/JSON API interface, described in the first part of this document.

The READ API is particularly useful for searching of the database for the existence or not of an event within the database, a process that is MUST BE CARRIED OUT before attempting to create a new disaster event in the system using the CREATE API.

The CREATE API uses a simple messaging exchange schema, either passing an XML or JSON formatted message specifying the data to be used, and would return the success code of the operation and the generated GLIDEnumber.

It is very important to note that, as opposed to the READ API, which is open to the public, the CREATE API requires the specification of User Credentials with enough privileges to execute the requested operations.

These User Credentials must correspond to an existing user that has been granted OPERATOR or higher privileges in the GLIDEnumber website. Only ADRC and the GLIDE Steering Committee can grant this access level to a specific user.

The authentication mechanisms in this proposal is very simple, but other authentication mechanisms could be adopted in the future if the community considers it essential or highly convenient, and not too complicated for both the server and client sides.

Given the communication protocol with the site will be SSL (https:// protocol), ALL the messages sent to the server must contain authentication information (username and password) of an authorized OPERATOR user, and other data required to create, extend, or modify a GLIDE record, the disaster data itself.

The CREATE API is foreseen to change over the years as new versions of the system are developed and possibly new indicators, and other fields will be required. Therefore the API general design is such that includes a “handshake” protocol by which the remote software wishing to create, extend or modify a GLIDE record specifies which version of the API is using.

**This document describes Version 1 of the API**. Each version will require a specific set of fields and/or XML or JSON attributes to be exchanged with the server.

The general flow of information is as follows:

1. The remote software sends a request to the server specifying the API version number, the User Credentials, the operation to be performed, and the data itself to be used.
2. The server will verify that all fields are valid and, if feasible will execute the operation requested.
3. If the request is not feasible (because of a version not supported, or because the user specified does not have the correct level of privileges, or because record data is not properly specified, among other reasons) the server will return an XML or JSON formatted error message, indicating that the operation was not successful, a code and description of the error.
4. If the request is feasible, the server will return an XML response indicating the operation was successful, and the GLIDE number and docid (primary key) of the record that was created, extended or modified.

For example, a CREATE request posted to the server should contain the following message:

|  |  |
| --- | --- |
| **XML format** | **JSON format** |
| **POST to:**  [**https://glidenumber.net/glide/admin/API/glide\_api.jsp**](https://glidenumber.net/glide/admin/API/glide_api.jsp)  **message is of the form:** | |
| Parameter:  **api\_xml\_message** | Parameter: **api\_json\_message** |
| <GLIDEAPI>  <version>1</version>  <username>user</username>  <password>xxxxx</password>  <api\_action>CREATE</action>  <!--legal values are CREATE|EXTEND|UPDATE-->   <glide>    <event>**FL**</event>    <geocode>**DOM**</geocode>    <location>Lac-a-Foin</location>    <year>**2020**</year>    <month>**9**</month>    <day>**18**</day>    <time />    <duration>**3**</duration>    <magnitude />    <source>**GDACS Alert system**</source>    <killed>**1**</killed>    <injured>**22**</injured>    <affected>**385**</affected>    <homeless>**2200**</homeless>  <destroyedhouses>333</destroyedhouses>  <latitude>41.3275459</latitude>  <longitude>19.8186982</longitude>    <comments>Hurricane Teddy is be the second hurricane to strike in Republica Dominicana in a week, leaving a trail of destruction on its way</comments>    </glide>  </GLIDEAPI> | {  "version":"1",  "username":"user",  "password":"xxxx",  "api\_action":"CREATE",  "glide":  { "event":"FL",  "geocode":"DOM",  "location":"Ramon Santana, east of the Dominican Republic",  "year":"2020",  "month":"9",  "day":"18",  "time":"",  "duration":"0",  "magnitude":"",  "source":"GDACS Alert system",  "killed":"1",  "injured":"22",  "affected":"385",  " homeless":"2200",  "destroyedhouses":"333",  "latitude":"41.3275459",  "longitude":"19.8186982",  "comments":" Hurricane Teddy is be the second hurricane to strike in República Dominicana in a week, leaving a trail of destruction on its way"  }  } |
|  |  |

Upon successful completion, the server would response with a message like:

|  |  |
| --- | --- |
| **XML format** | **JSON format** |
| <GLIDEAPI\_RESPONSE>  <version>1</version>  <result>**success**</result>  <glidenumber>**FL-2020-000632-DOM**</glidenumber>  <docid>**26345**</docid>  </GLIDEAPI\_RESPONSE> | {  "version":1,  "result":"**success",**  "Glidenumber":"**FL-2020-000632-DOM”,**  "docid":**26345**  } |

If there is an error in the operation the response will be like:

|  |  |
| --- | --- |
| **XML format** | **JSON format** |
| <GLIDEAPI\_RESPONSE>  <version>1</version>  <result>**ERROR**</result>  <code>**ERROR\_CODE**</code>  <description>**ERROR DESCRIPTION**</description>  </GLIDEAPI\_RESPONSE> | {  "version":1,  "result":"**error",**  "code":"error\_code**”,**  "description":"error description**"**  } |

As it can be seen the API supports three operations, CREATE, EXTEND, UPDATE. Each operation requires a slightly different set of attributes in the GLIDE record section as follows (JSON format attributes would be similar on each case):

**CREATE operation:**

<GLIDEAPI>

<version>**1**</version>

<username>**USERNAME**</username>

<password>**PASSWORD**</password>

<api\_action>**CREATE**</api\_action>

<glide>   
  <event>**EV**</event>   
  <geocode>**ISO**</geocode>   
  <location>**Location description**</location>   
  <year>**YYYY**</year>   
  <month>**MM**</month>   
  <day>**DD**</day>

*[rest of glide record fields as in above example]*

  </glide>   
</GLIDEAPI>

**EXTEND operation:**

This request requires two additional attributes, the GLIDEnumber to extend and (for consistency verification) its DOCID, primary key. **Please note that an event will be extended to another country with the same event code as the source event record. If present, event code attribute will be ignored.**

<GLIDEAPI>

<version>**1**</version>

<username>**USERNAME**</username>

<password>**PASSWORD**</password>

<api\_action>**EXTEND**</api\_action>

<glide>   
 <glidenumber>**EV-YYYY-NNNNNN-ISO**</glidenumber>

<docid>26345</docid>

 <geocode>**ISO**</geocode>   
  <location>**Location description**</location>   
  <year>**YYYY**</year>   
  <month>**MM**</month>   
  <day>**DD**</day>

*[rest of glide record fields as in above example]*

  </glide>   
</GLIDEAPI>

**UPDATE operation:**

This request requires two additional attributes, the GLIDEnumber to update and (for consistency verification) its DOCID, primary key, **and does not allow neither the Event code, the country code, or year to be changed.** Those attributes must be the same as the original record, otherwise an error will be produced. Please also note that ALL fields will be updated. If some of the fields are missing the will be cleared in the update (to zero, or an empty string)

<GLIDEAPI>

<version>**1**</version>

<username>**USERNAME**</username>

<password>**PASSWORD**</password>

<api\_action>**UPDATE**</api\_action>

<glide>   
 <glidenumber>**EV-YYYY-NNNNNN-ISO**</glidenumber>

<docid>26345</docid>

  <location>**Location description**</location>   
  <year>**YYYY**</year>   
  <month>**MM**</month>   
  <day>**DD**</day>

*[rest of glide record fields as in above example]*

  </glide>   
</GLIDEAPI>

**IMPORTANT NOTE:**

The GLIDE API does not include DELETE nor INTEGRATE operations, which were intentionally not included for security reasons. Please note these operations must be done manually exclusively

**ERROR CODES AND DESCRIPTION**

|  |  |  |  |
| --- | --- | --- | --- |
| Code | Description | Explanation | Possible Solutions |
| 999 | Internal Server Error | Sever server error | Please communicate with ADRC. Ensure that messages are correctly being posted. |
| 001 | No message was received | The API requires either an XML or a JSON format message. | Ensure a proper message is POSTed, either as XML (parameter "api\_xml\_message") or JSON (parameter "api\_json\_message") |
| 002 | Invalid XML message | The XML message is either empty (NULL) or is not a syntactically valid XML message | Ensure the validity and completeness of the XML message |
| 003 | Invalid JSON message | The JSON message is either empty (NULL) or is not a syntactically valid JSON message | Ensure the validity and completeness of the JSON message |
| 090 | Invalid or non supported API Version | Currently API version must be 1. | Ensure message has Version 1. |
| 091 | Invalid user name or password | Either the user name or password are wrong | Provide correct credentials |
| 092 | User has not enough privileges | User provided must have Operator or higher privileges in order to access the CREATE API functions | Communicate with ADRC in order to elevate the privileges of the user provided |
| 099 | Invalid operation | Operation must be CREATE, EXTEND or MODIFY | Provide a correct operation |
| 101 | Invalid Event Type Code | Not a valid hazard code. It must be one of the codes present in the corresponding drop down list in the GLIDE website | Provide a correct Event type (Hazard) code. |
| 102 | Invalid Country code | Not a valid ISO-3 Country code. | Correct the country code supplied |
| 103 | Invalid Year | Year must be between 1900 and current year | Provide a year that is within the range 1900-current year |
| 104 | Invalid Month | Month must be between 1 and 12, or left blank | Provide a correct month |
| 105 | Invalid Day | Day must be between 1 and the last day of the month provided, or left blank | Provide a correct day |
| 106 | Invalid date – in the future | Date of event must be today or in the past | Provide a correct date |
| 107 | Invalid Latitude | Latitude must be between 90.0 and -90. | Provide a latitude within the range |
| 108 | Invalid Longitude | Longitude must be between 180.0 and -180. | Provide a Longitude within the range |
| 109 | Comment too long - truncated | Comment must be less than 8000 characters long | Shorten the comment |
| 201 | Disaster already exists | A glide number has been issued for that country, same event, same date and same location. | Provide data that differs in one of the main fields (country, event, date and location) |
| 301 | GLIDE number does not exist | An UPDATE or EXTEND request provided an erroneous GLIDE number | Provide a correct GLIDE number. |
| 302 | DOCID does not exist | An UPDATE or EXTEND request provided an erroneous DOCID number | Provide an existing and correct DOCID. Use the XML interface to obtain it. |
| 303 | DOCID does not correspond to GLIDE number | An UPDATE or EXTEND request provided an inconsistent specification of a GLIDE number which does not correspond to the DOCID provided | Provide a correct DOCID. Use the XML interface to obtain it. |
| 304 | Invalid GLIDEnumber | An UPDATE or EXTEND request provided a GLIDEnumber in Invalid format | Provide a correct GLIDE number in the format “HH-YYYY-nnnnnn-ISO”. |