“Translation API” to the WorldLingo System

Technical Summary

WorldLingo

Table of Contents

Table of Figures

List of Tables

1. Purpose

This document describes the *Translation API*, which allows connection to the WORLDLINGO.COM SYSTEM and submitting requests, which are then processed. When finished the result can be retrieved by the client and used as required.

2. Intended Audience and Reading Suggestions

The content of this document is aimed at technically skilled persons, such as webmasters, network administrators or software developers. It requires certain knowledge of Internet techniques and protocols such as HTTP. It is recommended that you follow the references in section 3 and read the material provided in the listed locations. Since the way translated text is returned to the calling party is HTTP based, the calling system needs to implement scripts or programs that parse these results and effectively convert them into the desired end result, such as a web page. Hence, the user of the API needs detailed knowledge in these techniques.

3. References

The following Internet Specifications provide relevant information to the development and implementation of the Translation API:

- ISO 639 Language Codes
- ISO 3166 Country Codes
- ISO 4217 Currency Codes
- RFC 1738 Uniform Resource Locators (URL)
- RFC 1808 Relative Uniform Resource Locators
- RFC 1945 Hypertext Transfer Protocol (HTTP/1.0)
- RFC 2046 MIME Part Two: Media Types

On-line versions of any of these RFC's (ie. "Request For Comments") can be located at http://www.rfc-editor.org. The World Wide Web Consortium (http://www.w3.org) is the definitive source of HTTP related information that affects this specification and its implementations.

4. API Description

The Translation API is based on HTTP (Hypertext Transfer Protocol). This allows different programming environments to be used to access the API. One might use a PERL script, Java classes or C based CGI applications. Even simple Telnet connections to the server are possible, although it is the purpose of this API to integrate the services provided by WorldLingo into other applications.

Please refer to Figure 1 for a graphical representation of the communication involved in this API. The process works as follows:

1. The client sends a request to the WorldLingo server. This request must be a valid HTTP request in the form of

   ```
   http://www.worldlingo.com/S000.1/api?parameterlist...
   ```

   that might result in the following data being sent to the server

```
GET /S000.1/api?parameterlist... HTTP/1.1
Accept: */*
Accept-Language: en,de;q=0.7,ja;q=0.3
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 5.0; Windows NT)
Host: www.worldlingo.com
Connection: Keep-Alive
```

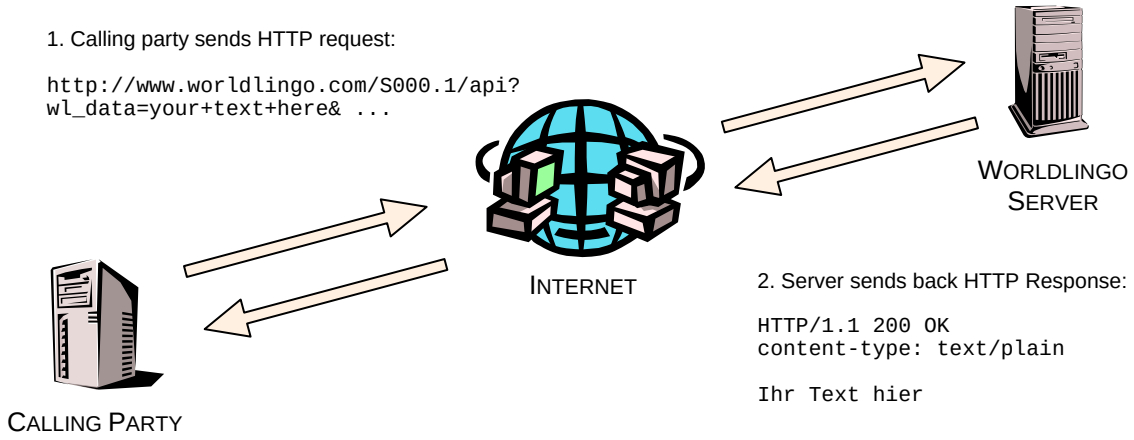
2. The server returns the translated text to the client. In this example it might look like the following

```
HTTP/1.1 200 OK
Date: Fri, 31 Mar 2000 03:21:18 GMT
Server: Apache
Content-Type: text/plain

translated text...
```

The result can then be processed on the calling side. It can be embedded in a resulting web page or sent off by e-mail.

Figure 1: Schematic Calling Sequence



The following table gives a comprehensive list of all mandatory request parameters, please be advised that you must url-encode the values of the parameters:

Parameter	Description
wl_data	The data to be translated. The format of the data is specified by the wl_mimetype parameter and defaults to plain text.
wl_password	The password of the calling party, issued by WorldLingo upon sevice creation.
wl_srclang	The selected source language for translation.
wl_trglang	The selected target language for translation.

Table 1: List of Mandatory Parameters

The following table gives a comprehensive list of all optional request parameters:

Parameter	Description
wl_opt	0 – Translation Only (default) 1 – Quote Only (Quoting is not currently available) 2 – Quote and Translation (Quoting is not currently available)
wl_mimetype	text/plain – The data is in plain text format (default) text/html – The data is in HTML format
wl_dictno	Specifies the custom dictionary to use. Refer to section 8 for further information.
wl_gloss	Use a context specific glossary. Refer to section 8 for further information.
wl_srcenc	Specify the input encoding of the data (see "wl_data" above). If the parameter is not given then the server assumes the data is encoded in the default encoding of “UTF-8”
wl_trgenc	Specify the output encoding for the translation. If the parameter is not given then the server returns the result encoded in the default encoding of “UTF-8”
wl_errorstyle	0 – The API will return a simple number, as the first line of the result, referencing the particular error. Refer to section 11 for error information. 1 – The error code will be placed in the http response header field “X-WL-ERRORCODE”. Refer to section 11 for error information.

Table 2: List of Optional Parameters

The parameters that specify the encodings (wl_srcenc, wl_trgenc) have to be provided in the same way as done inside a HTML page within the "Content-Type" and its "charset" part. An example might be the following webpage encoded in a Japanese Shift JIS:

```
<html>
<head>
<title>Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=x-sjis">
</head>
<body bgcolor="#FFFFFF">
Japanese Text here...
</body>
</html>
```

The text of this web page that is sent to the WorldLingo Server must provide the following parameter as part of the request to get the correct results

```
...?wl_data=Japanese+text+here&wl_srclang=ja&wl_trglang=en&wl_trgenc=shift_
jis
```

5. Mime Types

The WorldLingo Translation API currently handles two types of input data. The Translation API is primarily designed to handle the translation of text. However, functionality for html translation has also been built into the system. The following example illustrates how the API can be used to translate HTML.

Request:

```
http://www.worldlingo.com/S000.1/api?wl_srclang=en&wl_trglang=de&wl_passwor
d=secret&wl_mimetype=text%2Fhtml&wl_data=%3Chtml%3E%0D%0A%3Chead%3E%0D%0A%3
Ctitle%3Ethis+is+a+test%3C%2Ftitle%3E%0D%0A%3C%2Fhead%3E%0D%0A%3Cbody%3E%0D
%0AThis+is+a+test+of+the+HTML+translation+via+the+API%0D%0A%3C%2Fbody%3E%0D
%0A%3C%2Fhtml%3E%0D%0A
```

Response:

```
HTTP/1.1 200 OK
Server: Apache
Content-Type: text/html;charset=UTF-8
Content-Length: 137

0
<html>
<head><title>dieses ist ein Test</title></head>
<body>Dieses ist ein Test der HTML-Übersetzung über die API</body>
</html>
```

6. Supported Languages

Language Code	Language
de	German
en	English
es	Spanish
fr	French
it	Italian
ja	Japanese
pt	Portuguese
ru	Russian
zh_CN	Chinese (Simplified)
zh_TW	Chinese (Traditional)
ko	Korean
el	Greek
nl	Dutch
ar	Arabic
sv	Swedish
lwa_bg	Bulgarian
lwa_cs	Czech

Language Code	Language
lwa_da	Danish
lwa_fi	Finnish
lwa_ha	Hausa
lwa_he	Hebrew
lwa_hi	Hindi
lwa_hu	Hungarian
lwa_no	Norwegian
lwa_ps	Pashto
lwa_fa	Farsi
lwa_pl	Polish
lwa_ro	Romanian
lwa_sr	Serbian
lwa_so	Somali
lwa_th	Thai
lwa_tr	Turkish
lwa_ur	Urdu

Table 3: Used Language Codes

We support multilingual language pairs between all of the above languages.

7. Supported Encodings

The parameter "wl_srcenc" or “wl_trgenc” determines what the encoding (input or output) of the given text is. If the parameter is not given as part of the request it is assumed by using a default value of UTF-8.

Language Code	Default	Additional
de	UTF-8	ISO-8859-1
en	UTF-8	ISO-8859-1
es	UTF-8	ISO-8859-1
fr	UTF-8	ISO-8859-1
it	UTF-8	ISO-8859-1
pt	UTF-8	ISO-8859-1
ru	UTF-8	KOI8-R, Cp1251
ja	UTF-8	Shift_JIS
zh_CN	UTF-8	GB2312
zh_TW	UTF-8	Big5
ko	UTF-8	EUC-KR
el	UTF-8	ISO-8859-7
nl	UTF-8	ISO-8859-1
ar	UTF-8	ISO-8859-6
sv	UTF-8	ISO-8859-1
lwa_bg	UTF-8	
lwa_cs	UTF-8	
lwa_da	UTF-8	
lwa_fi	UTF-8	
lwa_ha	UTF-8	
lwa_he	UTF-8	
lwa_hi	UTF-8	
lwa_hu	UTF-8	
lwa_no	UTF-8	
lwa_ps	UTF-8	
lwa_fa	UTF-8	
lwa_pl	UTF-8	
lwa_ro	UTF-8	
lwa_sr	UTF-8	
lwa_so	UTF-8	
lwa_th	UTF-8	
lwa_tr	UTF-8	
lwa_ur	UTF-8	

Table 4: Supported Encodings

8. Improving Translation Quality

To improve the quality of your translations you can use a custom dictionary of words and phrases either not to be translated or to be translated to a specific word or phrase. To create a custom dictionary, go to **http://www.worldlingo.com/login/dictionaries.html.** The dictionary number assigned is the value you need to pass to the API under the parameter 'wl_dictno'.

Another way to improve the translation quality is to specify the subject matter of the text by using one of our pre-defined glossaries.

Glossary Subject	Code
General	gl1
Automotive	gl2
Aviation/Space	gl3
Chemistry	gl4
Colloquial	gl5
Computers/IT	gl6
Earth Sciences	gl7
Economics/Business	gl8
Electronics	gl9
Food Science	gl10
Legal	gl11
Life Sciences	gl12
Mathematics	gl13
Mechanical Engineering	gl14
Medicine	gl15
Metallurgy	gl16
Military Science	gl17
Naval/Maritime	gl18
Photography/Optics	gl19
Physics/Atomic Energy	gl20
Political Science	gl21

Table 5: Glossary Codes

9. Access via SOAP

The Translation API is also accessible via a SOAP interface; the WSDL for the SOAP interface is available at http://www.worldlingo.com/soap/ServiceApi.wsdl. SOAP can be used from within a number of languages including Visual Basic, C# & .NET

Example - Access from within a C# application
To access the SOAP interface from within a C# application you simply need to open your Visual Studio (.NET) project, go to the Project menu, and select Add Web Reference. In the URL field enter http://www.worldlingo.com/soap/ServiceApi.wsdl. It will immediately check the WSDL and display that it found 1 Service (Translation API) and list the method(s) the service has available. In the Web reference name field you need to enter in the name the object will be called in the actual code eg. WorldLingo. You can now click the Add Reference button.

To use the object in the application go to the class source code where you wish to access it from and create a new object of the type 'WorldLingo' (or the name you entered in as the 'Web reference name').

```
WorldLingo.ServiceAPIService service = new WorldLingo.ServiceAPIService();
```

Now to send a translation request you simply call the 'translate' method on this object. The translate method expects the following parameters:
serviceNo - your service number
password - your service password.
data - the data you wish to be translated.
mimeType - the mimetype of the data to be translated (text/plain or text/html).
srcLang - the source language.
trgLang - the target language.
srcEnc - the encoding the source data is encoded in. (Can be null, defaults to UTF8).
trgEnc - the encoding translation should return. (Can be null, defaults to UTF8).
dictionary - the number of the dictionary to be used. (Can be null).
glossary - the glossary to be used (Can be null).

```
WorldLingo.TranslateResult result = service.translate("S000.1", "secret",
"text to be translated.", "text/plain", "en", "fr", "utf8", "utf8", null,
null);
```

The result is returned as a 'TranslateResult' object that allows you easy access to the translation result. It also provides you with an error code, the result encoding and the mimetype.
If you find the result data you are getting is not readable you might need to change the encoding:

```
Encoding encoding = System.Text.Encoding.UTF8;
new String(encoding.GetChars(encoding.GetBytes(result.data)));
```

10. Testing and Security

To allow a user of the API to test the services and evaluate its functionality there is a test mode built into it. The Service Number S000.1 (as used in the examples) in combination with the password “secret” will translate the given text from the given source language into a random target language. The target languages are always Latin based (ISO-8859-1 encoded), therefore excluding the Asian encodings. This ensures that most users of the API can handle the random result. Additionally the test account is limited 25 words per translation request.

This test mode delivers the same result as in a normal use of the API with an established service. But because the result is not predictable it is not useful in a normal production environment.

When you have been given access to the API, you will need to use your own service number (built into the url) and password. You are able to change the password for the service at any time. As an extra security feature, our API is accessible via https as well as http.

11. Examples

A number of examples on how the API can be used have been prepared; they include the following languages Java, PHP, Perl and HTML and are available at
http://www.worldlingo.com/services/service1?serviceaction=download

The following examples give an overview of how to use the service. Please note, for ease of reading, this document only illustrates GET requests. You are also able to use POST requests.

1. Example

This example displays a valid request to the server, providing the text, language pair and authorization information.

Request:

```
http://www.worldlingo.com/S000.1/api?wl_data=This+is+a+test+translation&wl_
srclang=en&wl_trglang=de&wl_password=secret
```

Response:

```
HTTP/1.1 200
Server: Apache
Content-Type: text/plain;charset=UTF-8
Content-Length: 34

0
 Dieses ist eine TestÃ¼bersetzung
```

2. Example

Here the user provided the wrong password for authorization.

Request:

```
http://www.worldlingo.com/S000.1/api?wl_data=This+is+a+test+translation&wl_
srclang=en&wl_trglang=de&wl_password=wrong
```

Response:

```
HTTP/1.1 200
Server: Apache
Content-Type: text/plain;charset=UTF-8
Content-Length: 29

26
  This is a test translation
```

3. Example

Here is a valid request, this time using the request header error style (wl_errorstyle=1).

Request:

```
http://www.worldlingo.com/S000.1/api?wl_data=This+is+a+test+translation&wl_
srclang=en&wl_trglang=de&wl_password=secret&wl_errorstyle=1
```

Response:

```
HTTP/1.1 200
Server: Apache
X-WL-ERRORCODE: 0
Content-Type: text/plain;charset=UTF-8
Content-Length: 32

Dieses ist eine TestÃ¼bersetzung
```

12. Error codes

The following outline describes the possible error codes given by the API:

Error	Error code
Successful	0
Incorrect Password	26
Source language not subscribed to.	28
Target language not subscribed to.	29
Invalid Language Pair	176
No input data	177
Invalid Mime-type	502
Translation timed out	1176
TEEngineErrorException. Occurs when single words or short phrases in isolation cause the translation engine to fail.	1181

Table 6: Error codes

If you find you are receiving an error code that is not listed above please contact WorldLingo Technical Support **http://www.worldlingo.com/contact/**

To subscribe to WorldLingo's Translation API solution, please visit:
http://www.worldlingo.com/products/worldlingo_api.html
or contact us http://www.worldlingo.com/contact/