

maXbox6

all about code blocks

About

EKON 30

maXbox

RockChain

TEE

X-Mas 2025

recent posts

[Free Google Translator](#)

[API II](#)

[Free Google Translator API](#)

[TEE Kraftwerk Concert](#)

[Starter165 Addons](#)

[Machinelearning Steps](#)

about



Free Google Translator API II

February 11, 2026

As a second edition from maXbox starter168 , we show 4 more solutions to use the translator in yourscripsts. You can combine maXbox with the (official or unofficial) Google Translate API for many more things than just "translate one text and show the result", [scribd+4](#)

Another 4 more solutions:

1. the class THttpRequestC from library HttpClasses.pas
2. OLE Automation and late binding WinHttp.WinHttpRequest.5.1
3. with a Rest-Client TRestClient
4. Wininet library direct procedure Wininet_HttpGet

Ideas inside maXbox scripts for translating

- Batch translation of your own data
Read lines from files, databases, or logs and translate them in loops (e.g. all help texts or UI strings for localization). Save the results as new resource files or JSON dictionaries. [apyflux+2](#)
- Language detection plus routing
Use `s1=auto` to detect the source language, then route texts into different processing pipelines (e.g. German texts to one classifier, English texts to another). [deepwiki+2](#)
- Interactive console / GUI translator
Build a small maXbox form where the user types text, selects `s1/t1` combo boxes, and you call `https://translate.googleapis.com/translate_a/single?client=gtx&dt=t&s1=...&t1=...&q=...` in the background. [scribd+1](#)
- Background localization tool
Make a "translation builder": read all caption strings from a Delphi/Lazarus form (or from a JSON of labels) and produce a translated version for each configured language in one run. [apyflux+2](#)

Example use cases for your projects

- Developer tools

- In-IDE helper translating comments or error messages fetched from logs into English for easier debugging.
- Script that scans code for **TODO** comments and translates them into a common team language.[blogs.doctranslate+1](#)
- Documentation pipeline
 - maXbox script that converts markdown or text manuals into several languages and writes translated files with predictable naming (`readme.de.txt`, `readme.ja.txt`, etc.).[apyflux+2](#)
- Data analysis helpers
 - Translate scraped web data, support tickets, or product reviews before running your own classification / sentiment analysis tools in maXbox or external Python/R.[blogs.doctranslate+1](#)

Real-time and event-driven ideas

- Live chat / support bridge

Let a support client written in maXbox send each incoming message through Translate (auto-detect → your language, and back for replies). Similar patterns are used in chat systems.[developer.chrome+2](#)
- On-the-fly translation in tools

Translate any text the user selects in a memo or SynEdit control as soon as the selection changes, without needing a button (like the JS example that translates on input events).[deepwiki](#)
- File-watcher translation

Watch a directory for new `.txt` or `.csv` files, then automatically produce translated copies in one or more languages.[apyflux+1](#)

Working with the unofficial endpoint

From your Starter 109 text, the key pattern is:

- Endpoint: `https://translate.googleapis.com/translate_a/single`
- Minimal params:
 - `client=gtx` (or `dict-chrome-ex`),
 - `s1` = source (`auto` allowed),
 - `t1` = target,
 - `dt=t`,
 - `q` = URL-encoded text.[scribd+1](#)

That means within maXbox you can generalize your existing function into a mini “translation service” that:

- Wraps retry logic and error handling (4xx/5xx, JSON parse failures).
- Accepts arrays/lists of strings and returns arrays of translations.
- Exposes both “detect language only” and “translate text” as separate functions.[deepwiki+1](#)

THttpRequestC

a more simple way with lesser footprint (no external stream, just one interface) is the following get from the class THttpRequestC:

```

1  function Text_to_translate_API54(AURL, aclient, lngorig, lngtarg
2  var httpq: THttpRequestC;
3      jo: TJSON; jarr:TJSONArray2; urlback: string;
4      begin
5      httpq:= THttpRequestC.create(self);
6      httpq.headers.add('Accept: application/json; charset=utf-
7      //httpq.SecurityOptions:= [soSsl3,soPct,soIgnoreCertCNInv
```

```

8      try
9          httpq.Get(wideFormat(AURL,[lngorig,lngtarget,atext]));
10         urlback:= httpq.response.ContentAsUTF8String;
11         jo:= TJSON.Create();
12         jo.parse(urlback);
13         jarr:= jo.JsonArray;
14         if httpq.response.statusCode2=200 Then
15             result:=jarr[0].decode(urlback)
16         else result:='REST API Failed:'+
17             itoa(Httpq.response.statusCode2)+httpq.respo
18     except
19         writ('EWI_HTTP: '+ExceptionToString(exceptiontype,exce
20     finally
21         httpq.free;
22         httpq:= Nil;
23         jo.free;
24     end;
25 end;

```

OLE Automation

Another solution seems to be the easiest one with OLE Automation and late binding:

```

1  function Text_to_translate_API55(AURL,aclient,lngorig,lngtarg
2  var httprq,hr: olevariant;
3  begin
4      Result:= '';
5      httprq:= CreateOleObject('WinHttp.WinHttpRequest.5.1');
6      try
7          hr:= httpRq.Open('GET',wideFormat(AURL,[lngorig,lngtarget
8          httprq.setRequestHeader('content-type','application/json'
9          if hr= S_OK then HttpRq.Send('');
10         If HttpRq.Status = 200 Then
11             result:= HttpRq.responseText
12         Else result:= 'Failed response:'+itoa(HttpRq.Status)+Http
13             writeln('debug response: '+HttpRq.GetAllResponseHeaders
14     finally
15         httprq:= unassigned;
16     end;
17 end;

```

I have a pas code that pulls in a parses JSON data. The issues I'm having is that usually when I open the workbook and run the macro, I get an error "the data necessary to complete this operation is not available" on the line "Set json = ParseJson(http.responseText)". Then you have to set the async parameter to false:

```
hr:= httpRq.Open('GET',wideFormat(AURL,
[lngorig,lngtarget,atext]),false);
```

Rest-Client TRestClient

The RestClient interface provides a functional style imperative API. RestTemplate is a synchronous HTTP client provided by Stream Framework for making RESTful requests. It offers a convenient way to interact with REST services by abstracting away much of the boilerplate code involved in handling HTTP requests and responses; Or really at runtime dynamic typing, casting, no local variables with a Rest-Client:

```

1  function Text_to_translate_API57(AURL,aclient,lngorig,lngtarg
2  begin
3      with TRestClient.create(self) do begin
4          ConnectionType:= hctWinInet;
5          //OnResponse:= @TRestOnResponseEvent2;
6          try
7              result:=Resource(wideFormat(AURL,[lngorig,lngtarg,atext
8          finally
9              Free;
10     except

```

```

11     writeln('GoT_Error: '+ExceptionToString(exceptiontype,ex
12     end;
13     end; //with
14 end;

```

Wininet library Wininet_HttpGet

A last solution is to call direct the wininet library as a procedure (functional programming) without explicit class constructors, The WinINet API uses the same stuff InternetExplorer / Edge is using so you also get any connection and proxy settings set by InternetExplorer for free:

```

1  function Text_to_translate_API59(AURL, aclient, lngorig, lngtarg
2  var tstrm: TMemoryStream;
3  begin
4      tstrm:= TMemoryStream.Create;
5      try
6          Wininet_HttpGet(wideFormat(AURL, [lngorig, lngtarg, atext]
7          tstrm.Position:= 0;
8          result:= utf8toString(StreamToString2(tstrm))
9      finally
10         tstrm.Free;
11     except
12         writ('GoT_Error: '+ExceptionToString(exceptiontype, excep
13     end;
14 end;

```

In alle 4 cases the url defined as a const is the same:

```

1  const
2  TBASE_URLS='https://clients5.google.com/translate_a/t?client=d
3
4  // call of function:
5  atext1:= 'Programmieren mit Design';
6  writeln((Text_to_translate_API57(TBASE_URLS, 'dict-chrome-ex', '
7  >>[["デザインを取り入れたプログラミング","de"]]

```

Higher-level solution patterns

- Localization toolchain for Delphi/Lazarus
 - maXbox script that reads .po/.resx/custom INI or JSON, fills in missing translations via the API, and writes back files for your main projects.[apyflux+2](#)
- Educational helpers
 - Vocabulary/phrase trainer that shows original + translation, maybe with a simple spaced-repetition schedule managed in maXbox. [\[apyflux\]](#)
- Model-railway / hobby utilities
 - Generate multilingual labels, descriptions, and documentation for rolling stock, layouts, or control software UIs so you can share projects internationally.[\[apyflux\]](#)
- Linguistic training and testing
 - Teststring: Bonjour Bronze Edition:
 - ボンジュール ブロンズエディション
 - Bonjūru buronzuedishon
 - プログラミング Puroguramingu programming

If you tell me what kind of maXbox project you want to extend (CLI tool, GUI helper, data pipeline, data science), I can sketch a concrete structure for the units and translation functions that fits your style idea.

You can also build a chat system to build it by wiring three pieces together in maXbox: a simple chat UI, a send/receive loop, and your existing Google Translate endpoint wrapper.

Basic architecture

For each incoming message, call the same function with `s1=auto` and `t1` = your language, then append both original and translated text to the chat window.

Two text areas: one for your original messages, one showing translated chat (yours and partner's messages). For each outgoing message, call `Text_to_translate_API5` (or similar) with `s1=auto` and `t1` = your partner's language, then send the translated text over your chat channel (socket/HTTP/MQTT/etc.).

Appendix: What are HTTP Response Headers?

HTTP response headers are part of the HTTP protocol, which facilitates communication between clients (like web browsers) and servers. When a client makes a request to a server, the server responds with data and includes headers that convey additional information about the response, in our case that the endpoint sends a file to download called `filename="json.txt"` as reporting-endpoints:

```
debug response: Cache-Control: no-cache, no-store, max-age=0,
must-revalidate
Date: Tue, 10 Feb 2026 16:01:48 GMT
Pragma: no-cache
Transfer-Encoding: chunked
Content-Type: application/json; charset=utf-8
Expires: Mon, 01 Jan 1990 00:00:00 GMT
Accept-Ranges: none
Server: ESF
Vary: Accept-Encoding
X-Content-Type-Options: nosniff
Content-Disposition: attachment; filename="json.txt";
filename=UTF-8'json.txt Cross-Origin-Resource-Policy: cross-
origin Cross-Origin-Opener-Policy: same-origin Content-Security-
Policy: script-src 'report-sample' 'nonce-
qP8Usb9kt4QOKv89MfTgSw' 'unsafe-inline';object-src 'none';base-
uri 'self';report-uri //TranslateApiHttp/cspreport;worker-src
'self' Content-Security-Policy: require-trusted-types-for
'script';report-uri //TranslateApiHttp/cspreport
Accept-CH: Sec-CH-UA-Arch, Sec-CH-UA-Bitness, Sec-CH-UA-Full-
Version, Sec-CH-UA-Full-Version-List, Sec-CH-UA-Model, Sec-CH-
UA-Wow64, Sec-CH-UA-Form-Factors, Sec-CH-UA-Platform, Sec-CH-UA-
Platform-Version
Permissions-Policy: ch-ua-arch=, ch-ua-bitness=, ch-ua-full-
version=, ch-ua-full-version-list=, ch-ua-model=, ch-ua-wow64=,
ch-ua-form-factors=, ch-ua-platform=, ch-ua-platform-version=
reporting-endpoints: default="/_/_/TranslateApiHttp/web-reports?
context=eJzj4tDikmLw1pBiE0LmeLNm7Rk2gReb79kqKSXlF8aXFCXmFecklqQW
pxaVpRbFGxkYmRkYGVjoGVjEFxgAABtyEsI"
Access-Control-Allow-Origin: *
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Alt-Svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
```

So if we got an error the result will be by default html and not json or txt, we can convert the stream console output :

404. That's an error.

The requested URL `/translate_a/tclient=dict-chrome-ex&sl=auto&tl=ja&q=Time%20and%20Temps%20and%20` was not found on this server. That's all we know.

For the official Cloud Translation API, a 403 almost never depends on "magic" parameters, but on quota, auth, or project setup. The key things that avoid 403s are:

Unofficial web endpoint (what maXbox Starter uses):

- Some community libraries found that using `client=gtx` against `https://translate.googleapis.com/translate_a/single` works without the `tk` token; other `client` values (`t`, etc.) can trigger 403 or captchas if tokens are missing or invalid.
- So for the *unofficial* endpoint, the typical safe pattern is:
 - `https://translate.googleapis.com/translate_a/single?client=gtx&sl=auto&tl=de&dt=t&q=...` (no `tk`).
- Avoid mixing in undocumented params like incorrect `tk` tokens; broken tokens are a known cause of 403 on the web endpoint.
- To minimise 403s:
- Keep requests small and throttled:
 - Respect character quotas per 100 seconds and overall daily quota, and add a delay or queue if you approach the limits.
- Batch logically:
 - Use the official API's batch features instead of spamming many single-character or single-word calls.

```

373 // writeln(utf8ToAnsi(Text_to_translate_API2(TBASE_URLS,'dict-chrome-ex','auto','es',atext)));
374 //writeln(Ansiutf8(jsonunescape(Text_to_translate_API2(TBASE_URLS,'dict-chrome-ex','auto','ja',atext),#10#13)));
375
376 writeln((Text_to_translate_API52(TBASE_URLS,'dict-chrome-ex','auto','ja', atext1)));
377
378 sleep(100)
379
380 //writeln((Text_to_translate_API54(TBASE_URLS,'dict-chrome-ex','auto','ja', atext)));
381
382 writeln((Text_to_translate_API55(TBASE_URLS,'dict-chrome-ex','auto','ja', atext)));
383
384 writeln((Text_to_translate_API57(TBASE_URLS,'dict-chrome-ex','auto','ja', atext)));
385 //writeln((Text_to_translate_API59(TBASE_URL_TestError,'dict-chrome-ex','auto','ja', atext)));
386
387 writeln((Text_to_translate_API59(TBASE_URLS,'dict-chrome-ex','auto','ja', atext)));
388

```

```

[[["時間と温度と暗号","en"]]
[[["時間と温度と暗号","en"]]
[[["時間と温度と暗号","en"]]
GetDefaultTextEncoding cp1252
GetWinEncoding : cp1252
mX5 executed: 11/02/2026 11:12:05 Runtime: 0:0:3.40 Memload: 73% use
RemObjects Pascal Script. Copyright (c) 2004-2026 by RemObjects Software & maXbox5

```

https://sourceforge.net/projects/maxbox5/files/examples/1465_GoogleTranslate_API1_5.txt/download



Several Endpoints to travel with code

Advertisements

Occasionally, some of your visitors may see an advertisement here, as well as a [Privacy & Cookies banner](#) at the bottom of the page. You can hide ads completely by upgrading to one of our paid plans.

[UPGRADE NOW](#) [DISMISS MESSAGE](#)

Posted in Code, Data Science, Languages, Machinelearning, maXbox, Tools

Leave a comment


Write a comment...

maxkleiner - Logged in via WordPress.com - Log out ×

Notify me of new posts
Receive web and mobile notifications for posts on this site.

Email me new posts
[Instantly](#) [Daily](#) [Weekly](#)

Email me new comments



[← Free Google Translator API](#)