



# MARY HTTP Interface: Documentation by example

This is a simple plain-HTML page showing the various queries that the MARY HTTP server supports. This is intended as a live documentation for developers who want to use the HTTP interface to build MARY clients.

All examples below use relative URLs. That is, if this page is shown as `http://localhost:59125/`, you can refer to the relative URL `version` as the absolute URL `http://localhost:59125/version` etc.

## Information requests

### Server version

`version` requests the version of the MARY server:

[version](#)

### Available data types

`datatypes` requests the list of available data types:

[datatypes](#)

### Available voices

`voices` requests the list of available voices

[voices](#)

### Example texts

`exampletext?datatype=(datatype)&locale=(locale)` requests the example text for the given datatype and locale, e.g.:

[exampletext?datatype=RAWMARYXML&locale=en\\_US](#)

`exampletext?voice=(voicename)` requests the example text for the given voice. This makes sense in case of limited domain voices, e.g.:

[exampletext?voice=dfki-bundesliga](#)

### Available target features

Much processing in MARY uses the concept of a feature vector, abstracting a broad range of linguistic and acoustic analyses into byte-valued, short-valued and continuous features which can be computed for every phone or every halfphone in an input sentence.

`features?locale=(locale)` requests the list of available features that can be computed for the given locale, e.g.:

[features?locale=en](#)

`features?voice=(voicename)` requests the list of available features that can be computed for the given voice, which may or may not be the same as for the voice's locale, e.g.:

[features?voice=slt-arctic](#)

### Audio formats

`audioformats` requests the list of supported audio file format types.

[audioformats](#)

### Audio effects

MARY TTS can use audio effects to modify the synthesis output.

`audioeffects` requests the list of available audio effects:

[audioeffects](#)

`audioeffect-default-param?effect=(effectname)` requests the default parameters of the given audio effect, e.g.:

[audioeffect-default-param?effect=Robot](#)

`audioeffect-full?effect=(effectname)&ms=(parameter-settings)` requests a full description of the given audio effect, including effect name, parameters and help text, e.g.:

[audioeffect-full?effect=Robot&ms=amount:100.0](#)

`audioeffect-help?effect=(effectname)` requests a help text describing the given audio effect, e.g.:

[audioeffect-help?effect=Robot](#)

`audioeffect-is-hmm-effect?effect=(effectname)` requests a boolean value (plain text "yes" or "no") indicating whether or not the given effect is an effect that operates on HMM-based voices only, e.g.:

[audioeffect-is-hmm-effect?effect=Robot](#)

## Vocalizations

Some voices can generate non-verbal or listener vocalizations such as "(laughter)", "yeah", "hmm" etc. The list of vocalizations that can be generated depends on the voice. An empty list indicates that the voice does not support vocalizations.

`vocalizations?voice=(voicename)` requests the list of vocalizations available for a given voice.

For example:

[vocalizations?voice=dfki-poppy](#)

[vocalizations?voice=dfki-prudence](#)

[vocalizations?voice=dfki-spike](#)

[vocalizations?voice=dfki-obadiah](#)

The following is an example document in WORDS format that requests a vocalization:

```
<maryxml version="0.5" xmlns="http://mary.dfki.de/2002/MaryXML" xml:lang="en-GB">
<voice name="dfki-poppy">
<p>
<vocalization name="yeah" meaning="uncertain" intonation="falling" voicequality="modal"/>
</p>
</voice>
</maryxml>
```

## Synthesis requests

`process` requests the synthesis of some text.

It can be called as a GET or a POST request, using the following parameters:

- `INPUT_TEXT` (required) is the text to be processed.
- `INPUT_TYPE` (required) is the data type of the input text. It must be one of the input [data types](#).
- `OUTPUT_TYPE` (required) is the data type to be generated as output. It must be one of the output [data types](#).
- `LOCALE` (required) is the locale of the input text -- either a language (e.g., `en`) or a language and country (e.g., `en_US`).
- `AUDIO` (required only if `OUTPUT_TYPE=AUDIO`) is the format in which to send the synthesized audio. It must be one of the available [audio formats](#).
- `OUTPUT_TYPE_PARAMS` (optional) can be used to provide additional information regarding the requested output format. The only use at the moment is in connection with the output types `TARGETFEATURES` and `HALFPHONE_TARGETFEATURES`, where it can list the selection of [features](#) to compute.
- `VOICE` (optional) is the default voice to use for generating output. If absent, the locale's default voice will be used for producing audio.

- `STYLE` (optional) can be used for requesting a given speaking style for voices supporting this feature (none yet).
- `LOG` (optional) can be used for logging some information in the server's log file.
- `effect_(effectname)_selected` (optional) can be used to indicate whether the named effect should be applied (value is `on`) or not (value is `off`).
- `effect_(effectname)_parameters` (optional) can be used to transport the parameters to use for the named effect.

## GET examples

Simple text to speech for the text `Hello world` using the default US english voice to produce WAVE output data:

[process?INPUT\\_TEXT=Hello+world&INPUT\\_TYPE=TEXT&OUTPUT\\_TYPE=AUDIO&AUDIO=WAVE\\_FILE&LOCALE=en\\_US](process?INPUT_TEXT=Hello+world&INPUT_TYPE=TEXT&OUTPUT_TYPE=AUDIO&AUDIO=WAVE_FILE&LOCALE=en_US)

Partial processing of data: convert `Hello world` into ACOUSTPARAMS format, the full XML format used inside MARY TTS:

[process?INPUT\\_TEXT=Hello+world&INPUT\\_TYPE=TEXT&OUTPUT\\_TYPE=ACOUSTPARAMS&LOCALE=en\\_US](process?INPUT_TEXT=Hello+world&INPUT_TYPE=TEXT&OUTPUT_TYPE=ACOUSTPARAMS&LOCALE=en_US)

Computation of feature vectors for `Hello world` for the features `phone stressed accented` for every phone:

[process?INPUT\\_TEXT=Hello+world&INPUT\\_TYPE=TEXT&OUTPUT\\_TYPE=TARGETFEATURES&LOCALE=en\\_US&OUTPUT\\_TYPE\\_PARAMS=phone+stressed+accented](process?INPUT_TEXT=Hello+world&INPUT_TYPE=TEXT&OUTPUT_TYPE=TARGETFEATURES&LOCALE=en_US&OUTPUT_TYPE_PARAMS=phone+stressed+accented)

Simple form containing only the required fields, as a GET request:

INPUT\_TEXT:   
 INPUT\_TYPE:   
 OUTPUT\_TYPE:   
 LOCALE:   
 AUDIO:

## POST example

Simple form containing only the required fields, as a POST request:

INPUT\_TEXT:   
 INPUT\_TYPE:   
 OUTPUT\_TYPE:   
 LOCALE:   
 AUDIO: