

A german explanation can be found at <<http://wiki.openstreetmap.org/wiki/Sym2name>>

What is sym2name for?

A gpx file might contain the following tags:

```
<name>051</name>
```

```
<sym>House</sym>
```

So this waypoint is the 51st House. But programs like JOSM ignore the <sym>-Tag, So you do not see, if it is House 51 or Bridge 51. There comes sym2name. It copies the sym-information to the name-tag, Either directly or according to a translation table. So without a translation Table this example will produce:

```
<name>House / 051</name>
```

and if the translation-table contains for example the line:

```
House,building
```

it will produce

```
<name>building / 051</name>
```

The translation table is in a csv-File, called by default: sym2name.csv. For the exact format of that file see below.

Parameters of sym2name

```
sym2name [infile] [-i:infile] [-o:outfile] [-c:configurationfile]
```

You need to provide at least the path of the original file, the infile, either with or without '-i:'

If you do not provide a outfile a '_s2n' will be added to the infilename.

If you do not provide a configuration filename, sym2name.csv will be used

this way you can just drop the infile on the program

"example 1: sym2name example.gpx

example.gpx will be transformed into example_s2n.gpx using sym2name.csv

example 2: sym2name -i:example.gpx -o:output.gpx -c:my.csv

example.gpx will be transformed into output.gpx using my.csv

Format of the csv-File for sym2name Version 2

Detailed explanation

you might go straight to the next chapter.

csv stands for comma separated values, it is a text format used by spreadsheet programs.

So you should be able to write it with any such program, but if you don't have one an editor will OK too.

Each line in the spreadsheet corresponds to a line in the csv .

Each cell in the spreadsheet corresponds to a string in the csv.

To separate the strings any character can be used, originally it was a comma. Sym2name accepts either comma or semicolon.

To be able to include this separator in a String, the strings can be included in quotation marks.

So any of the following lines contains 2 strings:

```
String1,string2
```

```
String1;string2
```

```
“String, 1”,”string2”
```

sym2name only uses the first 2 strings of each line. Any thing after this is a comment.

A line starting with a semicolon (or a quotation mark followed by a semicolon) is also ignored as a comment.

In the first line sym2name expects the version number. Here the first string has to be #Version and the second the number, so any thing like:

```
#Version,2
"#Version", "2"
```

This way a new version of sym2name can read old csv. The version number is not the version number of the program, but that of the format → it will be changed less often. After this line sym2name will only use the separator used in this line. If there is no separator both comma or semicolon will be used. So

```
#Version,2
string,1,string2
```

will read 3 strings in the second line, but

```
#Version;2
string,1;string2
```

will recognize 2 strings separated by semicolon, so the comma is not used as a separator.

After this first line each line must contain 2 strings:

The first string is the symbol-name

the second is what should be inserted into the name of the waypoint

Blanks and tabs outside of quotation marks will be ignored. Only at the end of the second string they will be conserved.

Easy explanation

This may all sound difficult, but it's difficult to make it easy for the user ;-)

If you use Excel or OpenOffice.Calc just write "#Version" (without the quotation marks) in the top left cell and in the one right of it the version-number (should be 2), on each line thereafter write in the first cell the symbol-name to look for, in the second cell the string to insert into the name and if you want a comment in the third. If there is any '.' or ';' in the first or second column you should activate quotation marks. In any case use either '.' or ';' as separator..

Program history

Version 1.03 from 23.Nov.2008:

this supports quotation marks in the csv file and some small bugfixes

Version 1.04 from 1.Dec.2008

the default sys2name.csv was not searched for in the folder where the exe is.

Version 1.05 from 9.Dec.2008

A log File is written, which might help if there is a problem.

This helped to find, that by mistake sym2link looked for comma or colon instead of comma or semicolon as separators

Version 1.06 from 11.Dec.2008

Some csv-files were read with mistakes

gpx-files with long parts without waypoints took very long to process. A 1.1MB file without waypoint took 7min, now less than 7s!!!