Pysilfont User Documentation

Installation

Pull the repo from https://github.com/silnrsi/pysilfont.git

In the pysilfont directory, install by running:

```
sudo python setup.py install --record installed-files.txt
```

This will install some scripts (currently just ufoConvert) as well as the python libraries needed by all scripts.

Look in INSTALL.txt for information on uninstalling Pysilfont.

If running on a Mac, there is extra information in scripts/actionosx for setting up so that ufoConvert via right-click on a ufo.

If you have downloaded from the github repository, there will be many more scripts available which by be run using:

```
python <path>/<script name>.py ...
```

Basic usage

Most commands provide basic help with the "-h" option, eg

```
$ UFOconvert -h
usage: UFOconvert [-h] [-d] [-v VERSION] [-p PARAMS] ifont [ofont]

Convert/normalise a UFO.
- If no options are chosen, the output font will simply be a normalised version
of the font.

positional arguments:
  ifont                 Input font file
  ofont                 Output font file

optional arguments:
  -h, --help            show this help message and exit
  -d                    Display help with info on default values
  -v VERSION, --version VERSION
                        UFO version to output
  -p PARAMS, --params PARAMS
                        Font output parameters

Version: 1.0.0
```

Default values

Font/file names

Once the initial input file (eg input font) had been given, most other font and file names will have defaults based on those.

This applies to other input font names, output font names, input file names and output file names and is done to minimise retyping repeated information like the path the files reside in.   For example, simply using:

```
python FFcolourGlyphs.py  /home/data/fonts/SILcharis-r.ufo
```

will:

open	/home/data/fonts/SILcharis-r.ufo
read its input from	/home/data/fonts/SILcharis-rcolourGlyphs.csv
write its log to	/home/data/fonts/SILcharis-rcolourGlyphs.log
and output to	/home/data/fonts/SILcharis-rnew.ufo

If just part of a file name is supplied, other parts will default, eg if just 'test' was supplied for the output font name, the font would be output to /home/data/fonts/test.ufo.

Other parameters

Other parameters will just have standard default value

Displaying defaults for a command

Use the "-d" option to see what defaults are for a given command, eg "FFcolourGlyphs -d" will output its help text with the following appended:

```
Defaults for parameters/options

  Font/file names
    ofont                 new
    -i                    colourGlyphs.csv
    -l                    colourGlyphs.log
```

Specific scripts

UFOconvert

```
UFOconvert [-h] [-d] [-v VERSION] [-p PARAMS] ifont [ofont]
```

This will convert between UFO 2 and UFO3 (if -v is used to specify the alternative version) or otherwise simply normalize the UFO by 'converting' to the existing version.

Current default normalization behaviours include:

- XML formatting
  - Use 2 spaces as indents
  - Don't indent the <dict> for plists
  - Sort all <dicts>s in ascending key order
  - Where values can be "integer or float", store integer values as <integer>
  - Limit <real> precision to 6
- glif file names - use the UFO 3 suggested algorithm
- order glif elements and attributes in a predetermined order

Most of the above can be overridden by parameters supplied using -p:

- indentFirst The indent for first level elements (default 2 spaces)
- indentIncr The increment for xml indents (default 2 space)
- indentML Should multi-line string items be indented (default False)
- plistIndentFirst Indent for first level plist items to override indentFirst (default 0 spaces)
- sortDicts Sort dict elements (default True)
- precision Max precision for decimal numbers (default 6 decimal places)
- renameGlifs Standardize glif file names (default True)

Reporting is both to screen and a log file, with different levels of reporting set for each via loglevel (default W) and scrlevel (default P) parameters which can be set to one of:

- P Progress - Reports basic progress messages and all errors
- W Warning - As P but with warning messages as well
- I Info - As W but with information messages as well
- V Verbose - even more messages!

Current Limitations

The following are known limitations that are due to be addressed in the future

- UFO 3 specific folders (data and images) are not copied
- Converting from UFO 3 to UFO 2 only handles data that has a place in UFO 2 - but does include converting UFO 3 anchors to the standard way of handling them in UFO 2