

TYPETR-Assistants

This is our new Helpers and Assistant environment. All sources for the BaseAssistant classes and additional functions will be kept here, instead of having copying them inside project repositories.

It is a fast and flexible collection of helpers for specific type projects: this repository contains library of assistants and helpers for type project.

This collection of tools offers a fast and flexible collection of helpers and assistant examples for specific type projects.

Documentation is at docs/. There may be a website later.

Github

<https://github.com/petrvanblokland/TYPETR-Assistants>

Environment

TYPETR-Assistants is developed for RoboFont only.

Installing

Install for your own use with pip (@@@ Does not work yet):

```
$ pip install typetr_assistants
```

Or install from a cloned repository in your local Python version by (@@@ Does not work yet)

```
$ python setup.py install
```

Naming

Helpers

In the context of this repository, "Helpers" are scripts that can be run without any coding. They are generic tools that perform a specific task, such as the editing of curves or building PDF proofs. "Helpers can best be compared with existing tools that live inside the RoboFont extension menu, maintained by Mechanic.

With the pip-installer not working yet, it's possible to run the helpers directly from this repository. Open the current version of GlyphBrowser.py in RoboFont and run the code. If there is a UFO open, the browser will show a list of all other UFO's in the same directory, filtering on romand/italic and showing the superset of all glyphs that exist in all masters.

Double click on one of the glyphs will open an EditorWindow on that glyph for the selected UFO.

The glyphset can easily be filtered by various levels of name patterns.

Operational helpers

- GlyphBrowser.py

Planned helpers

- CurvePalette (will soon be implemented, see docs/CurvePalette.md)
- Text samples connection with FontGoggles
- InterpolationFixer (probably part of GlyphBrowser)
- GuidesBuilder
- Quadratics → Bezier → Quadratics conversions
- Measures (inside the EditorWindow)
- Overlays and point snapping
- KerningEditor (inside the EditorWindow)
- Live Auto-spacing and Auto-kerning, based on trained neural networks.
- Python source generators for build.py, MasterData and GlyphData dictionaries
- Exporting OT feature code, design space files,
- Proofing PDF

Assistants

“Assistants” are applications that are written for a specific project. They develop alongside the different phases of a project, supporting the automation of tasks as they are needed.

They inherit from BaseAssistant classes that take care of the base level of “event” plumbing between tools, UFO’s and RoboFont. Assistants may contain very specific information about the type project, such as glyphset, design space, shape of serifs, position of guide lines and spacing dependencies. Assistant code is likely to live inside the directories of type projects.

In general Assistants will be a mixture of functions that are specific for a design project and a range of adapted helpers.

Documentation

All markdown documents (“.md” extension) in this repository can best be opened with MacDown <https://macdown.uranusjr.com>

Usage

- Some generic helpers (scripts with a specific task, such as the CurvePalette and the GlyphBrowser) will be supplied in this repository, ready to use without any coding. Note that these are “simple” scripts that will develop overtime, without specific knowledge about a particular design project.
- Helper names carry a version number. While developing them further, we’ll have some older versions available. “beta” is the latest, adding functions but use it with care. It is safer to use one of the older versions.
- The TYPETR-Assistant assistantLib/ library exists as a set of evolving base classes that help to create your own local Project Assistants. It is supplied as a

toolkit without warrenty or support.

- We'll try to keep everything backward compatible as much as we can, but there is no guarantee that this will always be possible.
- Since TYPETR-Assistants are published as open source under MIT licensing, written from scratch without the inclusion of proprietary code, you are free to use all tools for your own projects as much as you like.

Dependencies

TYPETR-Assistants imports open source type libraries, such as:

- vanilla/ (by Tal Leming)
- fontmake/ (by Google and Just van Rossum)
- FontGoggles (by Just van Rossum)
- drawBot/ (by Frederik Berlaen and Just van Rossum)
- merz/ and ezui/ (by Tal Leming)
- neuralNetworkKerning/ (by Lars van Blokland)
- ... more here.

Special thanks to all designers, coders and companies who published the sources that TYPETR-Assistants are based on.

Contributions

Any contribution, reporting code issues or extending documentation from others is appreciated.

Helper and Assistant wishes

- Bring functions under a single key stroke.
- Show rendered text samples live through FontGoggles, including OT-features.
- Save all and close all.
- Color marker on the glyphs you just did.

TYPETR and TYPE-TRY are registered trademarks by Buro Petr van Blokland + Claudia Mens