

Bachelor Thesis

Declarative API for Defining Visualizations in Envision

Andrea Helfenstein

2012-10-24

Background

Envision is a visual programming environment for object-oriented languages. One of the core ideas of visual programming is to use graphical components to complement and partially replace the textual representation of the program.

In Envision we want to enable software developers to define their own visualizations for embedded Domain Specific Languages or libraries. Also, in order to improve existing visualizations, we need to be able to determine which visualizations work best for which elements, and therefore be able to quickly adjust and compare them.

This means, that visualizations in Envision need to be easy to add and modify.

Envision needs to be able to display thousands of visualizations at the same time. This is why a custom framework based on Qt is used, and not a standard GUI framework like Qt Widgets. Such a framework is designed for displaying a few objects at a time, and does not scale well.

However, using the current Envision approach, every visualization needs to implement fine-grained control of the rendering process, which makes adding or modifying even trivial visualizations a tedious task due to the need to write a lot of boilerplate code.

Goal

The goal of this project is to design and implement a visualization API on top of the already existing framework, providing a higher level of abstraction. This abstraction should allow standard visualization components (e.g. a container with a header and a body) to be written and combined in a declarative way, with good readability and minimal amount of code.

Nonetheless, it should still be possible to implement non-standard visualizations, avoiding the higher level abstraction and using the underlying framework.

The design of other GUI frameworks (e.g. Qt Widgets) will be taken into consideration when designing the API.

Core Tasks

- Look at other GUI frameworks (e.g. Qt Widgets) and identify which techniques can be reused for this project.
- Identify the standard visualization types and components for Envision.
- Define an API that enables declarative composition of these components, using suitable C++ language constructs.
- Implement the functionality of the API.
- Re-implement some existing visualizations using the new API and compare their complexity and readability to the previous implementation.

Possible Extensions

- Define an XML specification language for visualizations. This should make it possible to add new visualizations without having to write C++ code.
- Re-implement all existing visualizations and/or add new visualizations.
- Implement a graphical visualization editor, where visualizations can be created and modified, and then be saved as an XML file.