

# ПРОЕКТИРОВАНИЕ БИБЛИОТЕКИ ВИЗУАЛИЗАЦИИ ГРАФОВЫХ СТРУКТУР НА ОСНОВЕ ТРЕБОВАНИЙ ПОЛЬЗОВАТЕЛЯ

*Рассматриваются основные аспекты в ходе проектирования библиотеки для визуализации графовых структур на основе шаблона визуализации, заданным пользователем на некотором декларативном языке.*

## ВВЕДЕНИЕ

Графовые структуры часто используются в разных сферах науки. Их визуализация упрощает последующий анализ. В зависимости от предметной области и целей анализа ставятся различные требования для их графического отображения. Среди таких требований может быть указана необходимость визуализации определенных уровней структуры некоторым конкретным способом. Однако существует мало инструментов графического отображения графов с заданной гибкой шаблона визуализации.

## I. ПРОЕКТИРОВАНИЕ

Библиотека визуализации графовых структур будет позволять отображать требуемый граф на основе шаблона визуализации, заданного пользователем данного программного решения. Пользователь сможет задавать шаблон визуализации с помощью декларативного языка. Данный подход был выбран с целью отделить описание визуализации от описания графовой структуры.

Существует много готовых решений, которые могут качественно отображать граф, применяя какой-то определенный алгоритм визуализации. Поэтому решено было в качестве основы библиотеки использовать шаблон проектирования «Адаптер», используя существующие решения для возможности отрисовки конкретных частей графовых структур.

Пользователь данной библиотеки сможет задавать шаблоны визуализации с помощью определенных декларативных правил. Основными правилами будут являться:

- `get(graphSubsetDescription)`. В качестве аргумента будет передаваться объект, который будет описывать подграф к которому необходимо применить определенный алгоритм визуализации.
- `useLayout(layoutDescription)`. В качестве аргумента будет передаваться объект, который будет описывать алгоритм визуализации. Обязательным является указание

названия алгоритма. Если алгоритм имеет настраиваемые параметры, то их также можно будет указать в передаваемом объекте. В противном случае будут применены параметры по умолчанию для соответствующего алгоритма.

- `useFor(graphStructure)`. В качестве аргумента будет передаваться графовая структура, которую необходимо визуализировать.
- `export(exportType)`. В качестве аргумента будет передаваться формат файла, где будет сохранено конечное изображение графа.
- `build()`. Данный метод будет запускать процесс построения визуализации графа согласно ранее описанным правилам.

Процесс построения визуализации можно описать блок-схемой на рисунке 1.

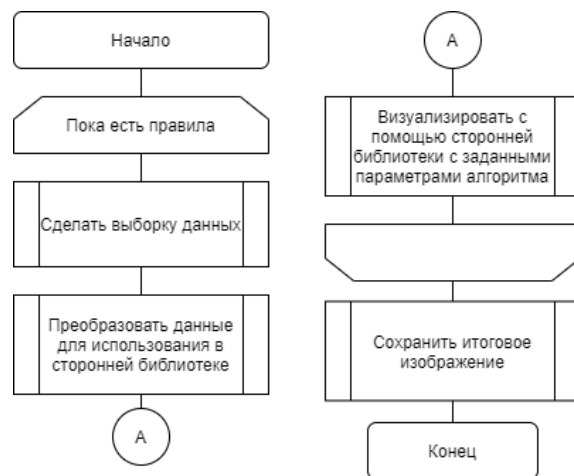


Рис. 1 – Блок-схема процесса построения визуализации

## II. ВЫВОДЫ

В качестве языка программирования для реализации данной библиотеки будет использоваться язык JavaScript. Входные данные с описанием графовых структур будут задаваться в формате JSON.

*Тарасенко Владислав Александрович, магистрант кафедры интеллектуальных информационных технологий БГУИР, u.tarasenka@yandex.by.*

*Научный руководитель: Сердюков Роман Евгеньевич, ведущий системный архитектор ООО «СтаркГеймз», кандидат технических наук, доцент, rserdyukov@gmail.com.*