**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл |  |

СОГЛАСОВАНО  
Профессор департамента программной инженерии, кандидат технических наук  
  
\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.М. Гринкруг  
«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

УТВЕРЖДАЮ  
Академический руководитель образовательной программы «Программная инженерия»  
  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  
«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

**ПРОГРАММНЫЙ КОМПОНЕНТ OPENGLVIEWER БИБЛИОТЕКИ JAVABEANS-КОМПОНЕНТ ДЛЯ 3D-ГРАФИКИ**

**Руководство программиста**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.01-01 33 01-1-ЛУ**

**Исполнитель**

Студент группы БПИ163

\_\_\_\_\_\_\_\_\_\_\_\_ / Д.Е. Крайнов /

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

**Москва 2019**

**УТВЕРЖДЕН**

**RU.17701729.04.01-01 33 01-1-ЛУ**

**ПРОГРАММНЫЙ КОМПОНЕНТ OPENGLVIEWER БИБЛИОТЕКИ JAVABEANS-КОМПОНЕНТ ДЛЯ 3D-ГРАФИКИ**

**Руководство программиста**

**RU.17701729.04.01-01 33 01-1**

**Листов 11**

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл |  |

**Москва 2019**

**АННОТАЦИЯ**

В данном документе приведено руководство оператора по эксплуатации программы «Программный компонент OpenGLViewer» (далее “OpenGLViewer”), назначением которой является работа с трехмерной графикой.

В разделе «Назначение и условия применения программы» указаны сведения о назначении программы и информация, достаточная для понимания функций программы и её эксплуатации.

В разделе «Характеристика программы» приведено описание основных характеристик и особенностей программы.

В разделе «Обращение к программе» приведено описание процедур вызова программы.

В разделе «Входные и выходные данные» приведено описание организации используемой входной и выходной информации.

В разделе «Сообщения» указаны тексты сообщений, выдаваемых программисту или оператору в ходе выполнения программы.

Настоящее руководство распространяется исключительно на программу и не заменяет учебную, справочную литературу, руководства от производителя операционной системы и прочие источники информации, освещающие работу с графическим пользовательским интерфейсом операционной системы.

СОДЕРЖАНИЕ

1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ 5

2. ХАРАКТЕРИСТИКА ПРОГРАММЫ 7

3. ОБРАЩЕНИЕ К ПРОГРАММЕ 8

4. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ 9

5. СООБЩЕНИЯ 9

СПИСОК ЛИТЕРАТУРЫ 10

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 11

**1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ**

* 1. **Функциональное назначение**Функциональным назначением компонента является предоставление возможности графического рендеринга моделей, представленных в виде графа сцены, с помощью технологии OpenGL и графического процессора пользовательского компьютера.
  2. **Эксплуатационное назначение**Программный компонент предлагается к эксплуатированию программистами в области компьютерной графики и 3D-моделистами в качестве инструмента для тестирования созданных графических моделей.
  3. **Состав функций**Приложение реализовано в виде набора JavaBeans-компонент [3] с возможностью автономного запуска в виде оконного приложения. Оконное приложение выполняет следующий набор функций:
* отрисовка переданных на вход компоненту OpenGLViewer графических моделей;
* загрузка модели из файла в формате json (десериализация модели);
* сохранение модели в файл в формате json (сериализация модели);
* включение/выключение отрисовки;
* включение/выключение отображения осей координат;
* включение/выключение отображения отладочного текста;
* добавление новых объектов в сцену;
* выбор текущего объекта;
* изменение объектов.
  1. **Климатические условия эксплуатации**

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

* 1. **Минимальный состав технических средств**
  + Минимальная тактовая частота процессора – 1 Ггц;
  + Минимальный объем ОЗУ – 512 Мб;
  + Минимальное свободное место на жёстком диске для хранения приложения – 10 МБ;
  + Видеокарта с минимальным размером видеопамяти 512 Мб;
  + Монитор с минимальным разрешением 800х600;
  + Клавиатура и мышь.
  1. **Минимальный состав программных средств**
  + ОС Windows XP или новее;
  + Java SE Runtime Environment 8 или новее;
  + Драйвер видеокарты с поддержкой OpenGL 2.1 или новее;
  + **Для использования набора компонентов в качестве библиотеки требуется также наличие установленной среды Java SE Development Kit 8 или новее.**
  1. **Требования к оператору**  
     Компонент эксплуатируется как составная часть библиотеки компонент. Требуемая минимальная классификация пользователя для работы в режиме просмотра – базовый оператор, для работы с библиотекой – программист.

1. **ХАРАКТЕРИСТИКА ПРОГРАММЫ**

**2.1. Временные характеристики, средства контроля правильности выполнения и самовосстанавливаемости программы**

Временные характеристики в данном документе не фиксируются, поскольку производительность компонентов сильно варьируется в зависимости от установленного технического обеспечения. В частности, главный критерий производительности компонентов – мощность установленной видеокарты, поскольку отрисовка сцен производится аппаратным путём с помощью технологии OpenGL 2.1.

Главным средством контроля правильности выполнения служат программные ограничения на свойства, которые можно задавать компонентам, представляющим графические примитимы. Например, нельзя создать сферу с отрицательным радиусом. В случае попытки присвоения некорректных значений свойствам примитивов библиотека будет выдавать исключение java.lang.IllegalArgumentException.

**2.2. Прочие характеристики и особенности**

Для отрисовки используется OpenGL версии 2.1. Данная версия выбрана в силу попытки поддержки наибольшей совместимости с более старым техническим обеспечением.

Большинство классов библиотеки следует спецификации JavaBeans [1], что делает их более пригодными для использования в визуальных редакторах графических интерфейсов.

1. **ОБРАЩЕНИЕ К ПРОГРАММЕ**

**3.1. Описание структуры набора компонентов OpenGLViewer и главных классов набора**

Структура набора компонентов состоит из нескольких пакетов с классами:

* **objects**, содержащий компоненты, представляющие собой графические примитивы;
* **viewer**, содержащий компонент OpenGLViewer для просмотра сцен;
* **window**, содержащий компонент окна для запуска набора в автономном режиме;
* **operations**, содержащий компоненты, представляющие трансформации сцен или объектов;
* **math**, содержащий математические объекты для описания и работы с примитивами.

**3.2. Вызовы методов программы и параметры данных**

Конструирование графа сцены – процесс, состоящий из создания графических примитивов из пакета **objects** и добавления к ним потомков с помощью метода **addChild()**, либо через геттер **getChildren()** для получения ссылки на список. Удаление потомков производится через методы списка, получаемого через геттер.

Для отрисовки сцены необходимо инстанцировать экземпляр компонента **GLViewerCanvas**, находящегося в пакете **viewer**, и добавить его как составную часть любого компонента, наследующегося от компонента **javax.swing.JFrame**, являющегося частью стандартной библиотеки Java.

Иерархия классов отрисовки:

* **GLViewerCanvas** содержит поле – экземпляр **OpenGLViewer**;
* **OpenGLViewer** содержит поле – экземпляр **GLObject**;
* **GLObject** может содержать или не содержать ссылки на потомков такого же класса в списке **children.**

Поскольку данные классы являются «бинами» (JavaBeans) [1], основные их поля являются доступными через сеттеры и геттеры.

Для удобства отрисовки всей сцены в классе **GLObject** присутствует рекурсивный метод **drawTree()**, отрисовывающий данный объект и все его потомки.

Подробная информация о функциональном назначении всех классов набора и их полей и методов содержится в Приложениях 1 и 2 прилагаемой к документации Пояснительной записки.

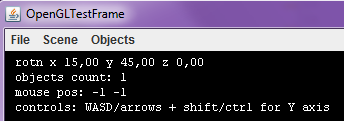
1. **ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ**

Помимо вывода отрисованной сцены на экран в качестве выходных данных и использования графа сцены в качестве входных данных для компонента OpenGLViewer как таковых входных и выходных данных нет – пользователь библиотеки волен получать и изменять в пределах доступности любые данные компонентов набора.

1. **СООБЩЕНИЯ**

Во время выполнения компоненты библиотеки могут возбудить исключительные ситуации в случае, если произошла попытка присвоить некорректное значение свойствам графических объектов (см. п. 2.1 данного документа). Исключения имеют тип java.lang.IllegalArgumentException и текст исключения: «Invalid {field}: {value}».

На экране отрисовки при включенном свойстве отображается отладочный текст, содержащий информацию о количестве объектов, текущем угле поворота камеры, позиции мыши при перетаскивании и клавишах управления сценой.



*Рисунок 1. Пример отладочного текста.*

**СПИСОК ЛИТЕРАТУРЫ**

1. JavaBeans Spec [Electronic resource]. // Oracle [Official website]. URL: <https://www.oracle.com/technetwork/articles/javaee/spec-136004.html> (accessed: 14.04.2019)

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подпись | Дата |
| измененных | замененных | новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |