|  |
| --- |
| Неформальное заведение JaggedStudio  E-mail: Jagged.N@yandex.ru  Steam: https://steamcommunity.com/id/JaggedNel/ |
| **СПРАВОЧНОЕ ПОСОБИЕ ВНУТРИИГРОВОГО ПРОГРАММНОГО КОМПЛЕКСА**    ПРЕДВАРИТЕЛЬНАЯ ВЕРСИЯ |
| Разработчик и исполнитель: JaggedNel  Дата начала реализации проекта: 23.12.2019  Дата, на которую актуально содержание пособия: 23.02.2020  Актуальная версия Nelbrus: 0.3.8 [22.02.2020] |

Санкт-Петербург 2020

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc29418055)

[ОПРЕДЕЛЕНИЯ 4](#_Toc29418056)

[1 Описание программного комплекса 5](#_Toc29418057)

[2 Установка и обновление 6](#_Toc29418058)

[2.1 Первая установка ядра 6](#_Toc29418059)

[2.2 Установка подпрограмм 6](#_Toc29418060)

[2.3 Установка сборок, дополнений, настроек и др. 6](#_Toc29418061)

[2.4 Обновление ядра 6](#_Toc29418062)

[2.5 Обновление подпрограмм, сборок дополнений и др. 6](#_Toc29418063)

[3 Принцип работы комплекса 7](#_Toc29418064)

[3.1 Базовые сведения 7](#_Toc29418065)

[3.2 Указания по взаимодействию с компонентами комплекса 7](#_Toc29418066)

[3.3 Указания по разработке подпрограммных систем 7](#_Toc29418067)

[ССЫЛКИ 8](#_Toc29418068)

ВВЕДЕНИЕ

Использование внутриигровых программ в игре Space Engineers часто становится решением прикладных задач всевозможного спектра при реализации различных проектов любых уровней. Однако, системы могут использовать неудобные или неэффективные методы настройки и работы. При этом в отдельно взятом проекте может использоваться множество программ, которые, при необходимости, трудоёмко укомплектовать в один программируемый блок.

В связи с этим, получение инструмента эффективного для разработки и взаимодействия с прикладными системами представляется одной из перспективнейших задач. Система Nelbrus призвана стать таким инструментом.

Данный свод документации создан с целью подробно ознакомить пользователей комплекса с его функционалом, помочь при разработке дополнительных компонентов и комплексов для системы. Пособие подразумевает наличие у его пользователя наличие основных знаний о внутриигровых скриптах. Если их нет, рекомендуется ознакомится с внутриигровым программированием перед началом ознакомления.

О любых ошибках и недочётах содержания или работы системы прошу сообщать через соответствующие обсуждения на страницах работ в Workshop[] или репозитории[].

ОПРЕДЕЛЕНИЯ

Компонент – набор команд, рассматриваемый как единое целое, выполняющий законченную функцию и применяемый самостоятельно или в составе комплекса.

Комплекс – набор команд, состоящий из двух или более компонентов или комплексов, выполняющих взаимосвязанные функции, и применяемый самостоятельно или в составе другого комплекса.

Ядро – комплекс системы Nelbrus, не включающий сторонние дополнения, подпрограммы и т.п.

Материнский блок – программируемый блок с системой Nelbrus.

Программа – полный перечень компьютерных инструкций и данных, записанных в программном блоке.

Подпрограмма – программный комплекс или компонент, под управлением Nelbrus в составе программы.

Сборка – программа, содержащая ядро вместе с любым набором компонентов, комплексов или подпрограмм.

Тик – минимальная внутриигровая единица времени, равная 1/60 реальной секунды при скорости симуляции 1,00.

1 Описание программного комплекса

Систему Nelbrus следует рассматривать как операционную систему, созданную для использования в среде внутриигровых скриптов игры Space Engineers, выполняющую следующие основные функции:

1. Организация среды для функционирования прикладных подпрограмм;
   1. Количество различных подпрограмм, используемых в одной программе ограничено лишь максимальной длинной программы выраженной в символах (100 000 символов).
   2. Количество подпрограмм, запущенных единовременно практически не ограничено.
2. Обеспечение доступа пользователя к ресурсам и функционалу системы и подпрограмм посредством:
   1. Базового командного интерфейса взаимодействия Nelbrus, предлагаемого к поддержке подпрограммами любых типов и назначений.
   2. Другими типами интерфейсов, встроенных в подпрограммы или добавленные в программу иными способами.

Цель разработки системы – получение эффективных методов разработки и управления внутриигровыми программными проектами.

1.1 Структура кода системы

На рисунке 1 приведена схема общего устройства сборки системы.



Рисунок 1 – Структура сборки

Программа состоит из следующих блоков:

* Стандартного конструктора Program, вызываемого при компиляции программы. Инструкции выполняемые здесь разделены на выполняющиеся один раз после компиляции программы 3 этапа подготовки «Ready, Steady, Go» (RSG):

- Ready производит первоначальную настройку ОС без чего невозможна выполнения других этапов. Представляет собой вызываемый метод.

- Steady инициализирует сторонние компоненты и комплексы (подпрограммы) в области видимости ОС. Является группой методов (SetEchoCtrl, ISP).

- Go завершает настройку ОС и переводит её в рабочий режим. Является методом.

* Ядра системы, содержащего: стандартные методы скрипта, родительские классы подпрограмм и другие компоненты системы.
* Определений сторонних подпрограмм, комплексов и компонентов. При этом они должны вноситься в область видимости ОС на этапе Steady в конструкторе программы.

1.2 Структура ядра системы

На рисунке 2 приведены комплексы и компоненты, составляющие ядро.



Рисунок 2 – Архитектура ядра системы Nelbrus

2 Установка и обновление

Есть два основных источника получения программных компонентов и комплексов:

1. Steam Workshop;
2. Репозитории.

Система Nelbrus доступна через оба источника:

- через подписку на сборку нужной версии в Steam Workshop от JaggedNel и последующую частичную или полную вставку кода программы в программируемый блок;

- частичное или полное копирование кода программы из репозитория JaggedNel [], где хранятся все доступные версии комплекса и документация.

Всегда перед установкой любого программного обеспечения тщательно ознакомьтесь с рекомендациями автора по установке или обновлению. Ниже приведены общие инструкции по установке и обновлению.

2.1 Чистая установка ядра

Для любых целей возможна чистая установка ядра системы наиболее подходящим способом с полным копированием кода сборки в программируемый блок. После этого комплекс будет доступен для использования.

2.2 Установка подпрограмм

2.4 Обновление ядра

Для обновления ядра программы необходимо полностью заменить код устаревшего ядра системы на новое. Ядро выделено в коде ключевыми фразами: «#region Core zone» перед ним и «#endregion Core zone» после. Замену удобно производить в любом текстовом редакторе, копируя в него весь код из материнского блока (CTRL + A – CTRL + C – CTRL + V).

2.5 Обновление подпрограмм

3 Работа комплекса

3.1 Командный интерфейс

3.2 Указания по разработке подпрограммных систем

ПОЛЕЗНЫЕ ССЫЛКИ

1. Steam-профиль JaggedNel: https://steamcommunity.com/id/JaggedNel/
2. Nelbrus v.1:
3. Репозиторий Nelbrus:
4. Коллекция систем для ОС Nelbrus: