

Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Факультет математики, механики и компьютерных наук

Направление подготовки 010400 — «Информационные технологии»

Поддержка PascalABC.NET в Eclipse IDE

Выпускная квалификационная работа
на степень бакалавра
студента
А. Ю. Тулякова

Научный руководитель:
ассистент каф. ИВЭ
А. М. Пеленицын

Ростов-на-Дону
2014

Содержание

Введение.....	3
Постановка задачи.....	5
1 Средства Eclipse для работы с плагинами	6
2 Реализация.....	8
2.1 Редактор кода.....	8
2.2 Взаимодействие с компилятором	11
2.3 Пользовательский интерфейс	14
Заключение	17
Список литературы	18
Приложение	19

Введение

Eclipse – это кроссплатформенная среда разработки, написанная на Java, развиваемая и поддерживаемая Eclipse Foundation, участниками которого являются такие компании, как IBM и Oracle. Первая версия вышла в 2001 году, а в 2003 с версии 3.0 Eclipse перестал быть монолитной IDE [1], поддерживающей расширения, а сам стал набором расширений. Все доступные в Eclipse IDE средства являются плагинами, созданными на основе данной платформы [2]. Архитектура времени выполнения основана на спецификации сервисной платформы OSGI, которая описывает способы создания и работы модульных приложений. Составляющие ядро платформы плагины, как и большинство сторонних, распространяются под лицензией Eclipse Public License. Существуют различные комплекты поставки Eclipse, в частности дистрибутивы с профессиональными средами разработки для наиболее популярных языков программирования, в числе которых Java, C++, PHP и многие другие.

Особый интерес представляет написание собственных плагинов на основе уже существующих, что позволяет настроить среду разработки необходимым образом для решения широкого спектра задач [3]. К их числу можно отнести различные по масштабу и используемым средствам проекты, от создания простого текстового редактора с подсветкой синтаксиса до обеспечения полноценной поддержки определенного языка программирования. Для реализации подобных задач существуют различные подходы, концепция каждого из которых определяется выбором языка программирования, для которого требуется создать среду разработки посредством Eclipse IDE, и набором используемых для этого средств, в число которых могут войти как плагины Eclipse по умолчанию, так и дополнительные инструменты, специально предназначенные для создания таких проектов и позволяющие упростить процесс разработки [4].

Целью данной работы является реализация кроссплатформенной поддержки языка программирования PascalABC.NET в среде разработки Eclipse. Особое внимание уделяется вопросам совместимости с операционными системами семейства Linux, т.к. для них пока не создана среда разработки языка PascalABC.NET.

Данная работа состоит из введения, постановки задачи, двух глав, заключения, списка литературы и приложения. В первой главе рассматриваются средства Eclipse для работы с плагинами. Во второй главе представлены методы реализации поставленной задачи. В приложении приведены инструкции по развертыванию и использованию полученного плагина.

Постановка задачи

- 1) Создание плагина с редактором кода языка PascalABC.NET для Eclipse IDE.
- 2) Организация взаимодействия Eclipse с консольным компилятором PascalABC.NET.
- 3) Реализация пользовательского интерфейса для работы с программами на языке PascalABC.NET в Eclipse.
- 4) Тестирование.
- 5) Публикация результатов.

1 Средства Eclipse для работы с плагинами

Среда разработки Eclipse предоставляет широкие возможности для настройки при помощи написания плагинов (см. рис. 1).

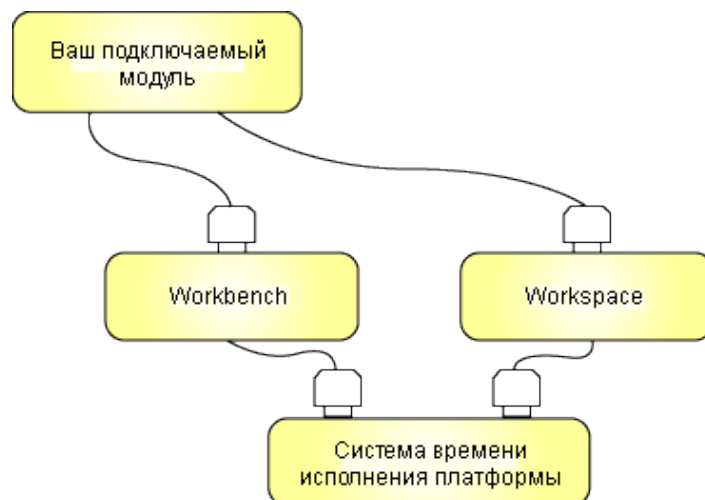


Рис. 1. Структура Eclipse.

Подключаемый модуль (плагин) представляет из себя экспортированный проект типа `Plug-in Project`. `Workbench` и `Workspace` (как и ряд других элементов, в том числе `Debug`) – обязательные компоненты платформы Eclipse.

В зависимости от нужд пользователя к плагину может быть добавлен базовый набор необходимых инструментов, например, стандартный редактор или страница настроек. В корне плагина находится файл `plugin.xml`, в котором содержатся объявления точек расширения (`Extension points`) – элементов, описывающих те места системы, которые можно расширять. В платформе Eclipse определены точки расширения для регистрации своих редакторов, перспектив, пунктов меню, кнопок на панели быстрого запуска и т.д. Само расширение описывается внутри тега `extension`, который содержит xml-код, удовлетворяющий схеме, заданной при создании точки расширения. К одной точке расширения может быть подключено несколько расширений. Eclipse предоставляет не только возможность добавления новых инструментов, но и средства для изменения

любых из уже имеющихся, включая те, что входят в стандартный установочный пакет [5]. В зависимости от типа расширения добавляемый компонент может быть описан целиком в файле `plugin.xml`, либо может потребоваться подключение уже существующих или написание своих собственных классов на Java.

Плагин может быть экспортирован как подключаемый элемент и установлен на другой экземпляр среды разработки. Eclipse поддерживает установку софта с репозиториях при помощи механизма Update Site. Этот механизм представляет собой совместное хранение (как в сети, так и на локальных устройствах) набора плагинов с их детальным описанием и инструкциями по установке. Установка осуществляется в меню Help -> Install New Software, где указывается адрес репозитория и после считывания его данных выбирается набор плагинов для последующей установки. После завершения установки среда разработки перезапускается и установленные компоненты становятся доступными для использования.

2 Реализация

2.1 Редактор кода

Для того, чтобы добавить в среду разработки Eclipse редактор кода языка PascalABC.NET был создан плагин с расширением `org.eclipse.ui.editors`, в котором был реализован новый редактор и указан тип файлов (*.pas), для которых он предназначен. Реализация нового редактора представляет собой описание правил определения принадлежности символов, входящих в файл типа *.pas, к подмножествам символов, соответствующим структуре языка PascalABC.NET и инициализацию текстовых атрибутов элементов этих подмножеств. В соответствии со структурой языка PascalABC.NET было определено десять правил и восемь типов подмножеств элементов (см. табл. 1).

Тип	Правила
1) Ключевые слова	Слова begin , end и т.д.
2) Комментарии	Последовательность символов: 1) После символов // и до конца строки. 2) Между фигурными скобками { и }. 3) Между наборами символов (* и *).
3) Документирующие комментарии	Последовательность символов после символов /// и до конца строки.
4) Строки	Последовательность символов, находящихся между парой символов ' .
5) Типы данных	Слова integer , real и т.д.
6) Числа	Числовые значения.
7) Директивы компилятора	Последовательность символов между наборами символов {\$ и }.
8) Текст	Остальные символы.

Табл. 1. Правила подсветки синтаксиса PascalABC.NET.

Для описания этих правил в плагине был создан шаблонный проект Plug-in Project with an editor типа Plug-in Project, в котором указанные выше правила были описаны при помощи средств соответствующих классов из `org.eclipse.jface.text.rules.*` следующим образом [6]:

- 1) Для ключевых слов использовался класс `WordRule`:

```
WordRule(IWordDetector detector, IToken defaultToken,  
boolean ignoreCase)
```

Это класс реализует правило определения ключевых слов, позволяя при помощи метода `addWord` добавлять строковые переменные, к которым будет применен набор текстовых атрибутов `defaultToken`.

Параметр `ignoreCase` отвечает за зависимость от регистра.

- 2) Для комментариев предназначались `SingleLineRule` и `MultiLineRule`:

```
SingleLineRule(String startSequence, String endSequence,  
IToken token)
```

```
MultiLineRule(String startSequence, String endSequence,  
IToken token)
```

Эти правила позволяют применить набор текстовых атрибутов `token` к символам, находящимся между последовательностями `startSequence` и `endSequence`. Первое правило используется для однострочных комментариев, второе – для многострочных.

- 3) Для документирующих комментариев использовались те же средства, что и для обычных комментариев, приведенные в предыдущем пункте.
- 4) Для строк предназначался класс `SingleLineRule` аналогично п.2.
- 5) Для типов данных использовался такой же механизм с `WordRule`, как и для ключевых слов в п.1.
- 6) Для чисел предназначался класс `NumbersRule`:

```
NumberRule(IToken token)
```

Этот класс определяет числа в коде и применяет к ним набор текстовых атрибутов `token`.

- 7) Для директив компилятора использовался класс `MultiLineRule` аналогично механизму работы с многострочными комментариями в п.2.
- 8) Текст представляет собой последний тип подмножеств. Считается, что текст – это те символы, которые не попадают под определение ни одного из предыдущих правил.

По умолчанию определено восемь типов текстовых атрибутов `token`, соответствующих подсветке синтаксиса, принятой в среде разработки `PascalABC.NET` для операционных систем семейства `Windows`, поставляемой в комплекте с компилятором на официальном сайте проекта.

Применение текстовых атрибутов осуществлялось при помощи средств класса `Token`:

`Token(Object data)`, где в качестве `data` использовался объект класса `TextAttribute`:

`TextAttribute(Color foreground, Color background, int style, Font font)`, в котором `foreground` и `background` отвечают за цвет текста и фона соответственно, `style` и `font` – за свойства шрифта.

При подобной реализации набора правил подсветка синтаксиса в полученном редакторе совпадает с используемой по умолчанию в среде разработки, поставляемой вместе с системой `PascalABC.NET` под `Windows`.

2.2 Взаимодействие с компилятором

Для обеспечения взаимодействия с компилятором с целью компиляции программ и запуска скомпилированных исполняемых файлов в консоли Eclipse по команде из среды разработки в созданный на предыдущем этапе плагин была добавлена реализация нового типа конфигураций запуска при помощи подключения к плагину следующих расширений:

- 1) `org.eclipse.debug.core.launchConfigurationTypes;`
- 2) `org.eclipse.debug.ui.launchConfigurationTabGroups;`
- 3) `org.eclipse.debug.ui.launchConfigurationTypeImages.`

Полученный тип конфигураций принадлежит к типу внешних инструментов (External Tools) и отображается в меню External Tools Configurations. Средства, предоставляемые данным типом конфигураций, практически идентичны имеющемуся в Eclipse по умолчанию типу Program, который позволяет запускать различные программы и скрипты и отображать результат их выполнения в консоли среды разработки. Разница состоит в том, что для типа Program по умолчанию не определены настройки (адрес запускаемой программы и т.д.), поэтому при создании экземпляра данной конфигурации пользователь вынужден каждый раз указывать их вручную. Для решения этой проблемы был создан новый тип конфигураций запуска, имеющий отображаемое имя PascalABC.NET. Для него были определены настройки по умолчанию, часть которых инициализируется в зависимости от типа операционной системы, определяемого по значениям переменных окружения, что обеспечивает кроссплатформенность плагина. Для этого используется метод `setDefault` класса `AbstractLaunchConfigurationTabGroup`, который устанавливает необходимые значения для определенных атрибутов при помощи конструкции `setAttribute`.

Компиляция и запуск программ по команде из среды разработки осуществляются при помощи скриптов командных оболочек (cmd.exe для Windows, bash для Linux), поставляемых вместе с плагином. Плагин определяет, какой из скриптов необходимо подключить и при создании нового экземпляра конфигурации запуска прописывает адрес к нему в поле Location. Несмотря на использование различных командных оболочек в зависимости от типа операционной системы алгоритмы работы скриптов не отличаются друг от друга. Каждый скрипт имеет два аргумента, указанных в поле Arguments в настройках экземпляра конфигурации запуска. Это путь к компилятору и путь к файлу с кодом программы. Эти аргументы по умолчанию определены следующим образом:

- 1) "\${eclipse_home}\plugins\PascalABC.NET_Eclipse_Plugin_1.0.0/PABCNETC/pabcnetcclear.exe", что подразумевает наличие распакованного дистрибутива с консольным компилятором в папке с плагином. По умолчанию данный дистрибутив включен в плагин. \${eclipse_home} в пути к компилятору является переменной окружения среды Eclipse, возвращающей путь к каталогу запущенного экземпляра среды разработки. Эта переменная инициализируется непосредственно во время запуска экземпляра соответствующей конфигурации.
- 2) "\${selected_resource_loc}" является адресом файла с исходным кодом, инициализируется во время запуска экземпляра конфигурации.

Алгоритм работы скрипта состоит из двух шагов, на первом запускается компилятор и ему на вход передается адрес файла с кодом. В случае успешной компиляции в консоль среды разработки ничего не выводится, в случае ошибки выводится соответствующее сообщение. На втором шаге проверяется код, с которым завершилось предыдущее действие, и если он не превышает 0 (успешная компиляция), то запускается полученный в качестве результата исполняемый файл. В том случае, если

этот файл осуществляет какие-то операции ввода-вывода все эти действия производятся в консоли Eclipse.

Процесс работы с программами на языке PascalABC.NET в Eclipse после реализации новой конфигурации запуска и подключения полученного плагина отражен на следующей диаграмме деятельности (см. рис. 2).



Рис. 2. Взаимодействие с компилятором.

Скрипты командных оболочек выполняют все действия, указанные на данной диаграмме между этапами редактирования кода и возврата в среду разработки. Подобная модель взаимодействия с компилятором аналогична используемой по умолчанию в стандартной среде разработки PascalABC.NET для Windows.

2.3 Пользовательский интерфейс

Создание элементов пользовательского интерфейса предназначено для настройки добавленных компонентов и предоставления пользователю дополнительных средств для работы с программами на PascalABC.NET в Eclipse. На этом шаге в плагин были использованы следующие расширения:

- 1) `org.eclipse.ui.preferencePages;`
- 2) `org.eclipse.core.runtime.preferences;`
- 3) `org.eclipse.help.toc;`
- 4) `org.eclipse.ui.navigator.navigatorContent;`
- 5) `org.eclipse.ui.navigator.viewer;`
- 6) `org.eclipse.ui.bindings;`
- 7) `org.eclipse.ui.perspectives.`

Расширения из первых двух пунктов предназначены для реализации новой страницы настроек (`Window -> Preferences -> PascalABC.NET`), создания на ней необходимых инструментов и средств управления ими. Для этого используется метод `addField`, позволяющий добавить на страницу настроек поле определенного типа, указать его имя и переменную, с которой он связан [7]. Те компоненты плагина, инициализация которых осуществляется при помощи добавленной страницы настроек, считывают необходимые значения в случае их изменения. В плагине реализован механизм присваивания переменным значений по умолчанию, которые будут инициализированы при сбросе настроек на странице. Добавленная страница настроек позволяет указать путь к компилятору, который будет использоваться как путь по умолчанию для создаваемых экземпляров конфигурации запуска и изменять настройки подсветки синтаксиса. Пользователю не предоставляется возможность изменять правила, по которым множества символов, входящих в файл типа `*.pas`, разделяются на подмножества, соответствующие языку PascalABC.NET (комментарии,

строки и т.д.). При этом имеется полный доступ к изменению текстовых атрибутов элементов заданных подмножеств. Пользователь может изменить визуальные свойства элементов любого типа – цвет, тип шрифта, размер.

Расширение `org.eclipse.help.toc` используется для добавления необходимых справочных страниц по PascalABC.NET в Eclipse и оформления их общей структуры. Реализация этого компонента представляет собой справочные страницы в формате .html с официального сайта PascalABC.NET [8], включенные в состав плагина. Эти страницы структурированы при помощи xml-описания в файле `toc.xml` (`toc` – `table of contents`), входящем в плагин. Для каждой справочной страницы указывается отображаемое имя и ссылка на .html-файл, например:

```
<topic
  href="html/LangGuide/_ProgrStruct_Defs/vardef.htm"
  label="Описание переменных">
</topic>
```

Eclipse позволяет отображать справку по PascalABC.NET на одной из панелей среды разработки по нажатию F1, либо делать это в отдельном окне (`Help -> Help Contents -> PascalABC.NET`). Подобный подход к реализации позволяет разделить справочные материалы на главы (оглавление идентично странице на официальном сайте PascalABC.NET) и использовать интерактивный поиск по их содержанию.

Расширения `org.eclipse.ui.navigator.navigatorContent` и `org.eclipse.ui.navigator.viewer` предназначены для изменения параметров отображения файлов в навигаторе `Project Explorer` [9]. Плагин добавляет два новых фильтра, которые скрывают скомпилированные программы и модули (расширения `*.exe` и `*.psu` соответственно).

Расширение `org.eclipse.ui.bindings` используется для автоматического добавления «горячих клавиш» в настройки среды по умолчанию в случае установки данного плагина. Клавиша F9, как и в

стандартной среде разработки PascalABC.NET для Windows, предназначена для запуска программы (в терминологии Eclipse эта клавиша предназначена для запуска последнего использованного экземпляра конфигурации запуска). F10 обращается к окну настроек внешних инструментов (External Tools Configurations).

Расширение `org.eclipse.ui.perspectives` предназначено для добавления перспективы языка PascalABC.NET, состоящей из необходимых элементов среды, состав и расположение которых описаны в плагине. Данная перспектива состоит из навигатора, консоли и основной области, предназначенной для редактирования кода программ.

Заключение

Результатом данной работы является плагин, обеспечивающий базовые функции среды разработки для PascalABC.NET в Eclipse:

- 1) Разработан редактор кода языка PascalABC.NET.
- 2) Реализовано взаимодействие Eclipse с консольным компилятором языка PascalABC.NET.
- 3) Добавлены элементы пользовательского интерфейса для работы с языком PascalABC.NET в Eclipse.
- 4) Проведено тестирование плагина и процесса его установки под различные операционные системы, в числе которых Ubuntu 14.04, Mac OS X Mavericks, Windows 7, Windows 8. Проверена многофайловая компоновка консольных приложений с использованием модулей на языке PascalABC.NET на всех поддерживаемых платформах. Под Windows протестирована работа программ с графическим пользовательским интерфейсом, использующих стандартные модули GraphABC, ABCObjects, Исполнителей (Робота и Чертежника) и задачника Programming Taskbook (при использовании консольного компилятора из стандартного установочного пакета для Windows).
- 5) Все компоненты плагина вместе с дистрибутивом консольного компилятора PascalABC.NET версии 2.2 от 02.06.2014 собраны в единый для всех поддерживаемых платформ установочный пакет, доступный на странице проекта под названием *PascalABC.NET Eclipse Plugin* на сервисе SourceForge [10]. Исходные коды размещены на странице указанного проекта в разделе Source Code [11].

Данная работа объединяет набор средств, необходимых для программирования на PascalABC.NET в Eclipse, при этом она не вносит никаких изменений в процессы компиляции и запуска программ, а служит для отображения результатов этих действий в среде разработки.

Список литературы

1. Eric Clayberg. Eclipse Plug-ins. Addison-Wesley Professional, 2008.
2. Разработка простого Eclipse RCP приложения.
<http://habrahabr.ru/post/139340/>
3. Туляков А. Ю. Применение среды разработки Eclipse для создания плагина с редактором кода языка программирования PascalABC.NET. // Курсовая работа. Мехмат ЮФУ. 2013.
4. Wikipedia: Eclipse (software).
http://en.wikipedia.org/wiki/Eclipse_%28software%29
5. Документация по Eclipse Kepler на официальном сайте Eclipse.
<http://help.eclipse.org/kepler/index.jsp>
6. Implementing support for your own language.
http://wiki.eclipse.org/The_Official_Eclipse_FAQs#Implementing_Support_for_Your_Own_Language
7. Eclipse Preferences – Tutorial.
<http://www.vogella.com/tutorials/EclipsePreferences/article.html>
8. Справка по языку программирования PascalABC.NET.
http://pascalabc.net/downloads/pabcnethelp/PABC_Help_Web.html
9. Creating a Custom Navigator.
<http://cvalcarcel.wordpress.com/2009/10/18/writing-an-eclipse-plug-in-part-7-creating-a-custom-navigator/>
10. Репозиторий плагина для онлайн установки.
http://sourceforge.net/projects/pabceclipse/files/upd_site/
11. Исходные коды проекта на сервисе SourceForge.
<http://sourceforge.net/p/pabceclipse/git/ci/master/tree/>
12. Архив с плагином (ручная оффлайн установка).
http://sourceforge.net/projects/pabceclipse/files/plugin_offline_manual_installation.zip/download

Приложение

Данный раздел включает в себя указания по развертыванию и использованию полученного плагина. Для его установки в системе должны присутствовать следующие компоненты:

- 1) Java Runtime Environment.
- 2) NET Framework или Mono, в зависимости от типа операционной системы (NET под Windows, Mono под Linux).
- 3) Стандартный пакет Eclipse IDE (не ниже версии Kepler).

Рекомендуется устанавливать плагин на «чистую» версию Eclipse. Установка может осуществляться с репозитория (Help -> Install New Software) [10] или посредством ручной распаковки архива с плагином [12] в папку с плагинами (<корневой каталог Eclipse>/plugins).

После установки рекомендуется выбрать в меню пункт Window -> Show Perspective -> Other и указать перспективу PascalABC.NET в качестве перспективы по умолчанию. Данная перспектива уже содержит консоль и навигатор, фильтрующий (не отображающий в содержимом проекта) скомпилированные программы и модули, необходимые для работы с программами на языке PascalABC.NET в Eclipse.

Для запуска программы следует создать новый проект общего типа (New Project -> General -> Project) и добавить в него файл с кодом на языке PascalABC.NET (с расширением *.pas). При открытии данного файла будет использоваться редактор с подсветкой синтаксиса, входящий в установленный плагин. После набора кода и нажатия на клавишу запуска (F9) Eclipse предложит выбрать конфигурацию запуска. В появившемся меню необходимо дважды кликнуть по PascalABC.NET, чтобы создать экземпляр конфигурации данного типа. Когда экземпляр будет создан, можно осуществить первый запуск, нажав на кнопку Run, находящуюся в правой нижней части этого окна.

В дальнейшем запуск может осуществляться нажатием на кнопку `Run Last Tool` на панели, либо клавишей `F9`. Экземпляры конфигурации запуска могут создаваться в любом количестве, одновременно запускаться может только один.

В установочный пакет включен дистрибутив с консольным компилятором `PascalABC.NET`, но при необходимости в дальнейшем можно будет указать (`Window -> Preferences -> PascalABC.NET -> Compiler location`) путь к другому компилятору `PascalABC.NET`, если он имеется в системе. В таком случае этот путь будет адресом компилятора по умолчанию для всех создаваемых с этого момента экземпляров конфигурации запуска.

Если пользователь не сохранил код программы, ему будет предложено сделать это перед запуском. Такое поведение среды разработки можно изменить в `Window -> Preferences -> Run/Debug-> Launching -> Save required dirty editors before launching`. По умолчанию среда запрашивает подтверждение (вариант `Prompt`), но можно отключить запрос (вариант `Never`) или активировать автоматическое сохранение при запуске (вариант `Always`).

В меню `Window -> Preferences -> PascalABC.NET` можно настроить подсветку синтаксиса.

В меню `Help -> Help Contents -> PascalABC.NET` (или нажав `F1`) можно просматривать справку по языку `PascalABC.NET`.

Порядок установки, настройки и использования данного плагина идентичен для всех поддерживаемых типов операционных систем.