PHP CLASS MANAGER

Оглавление

[Цель работы 3](#_Toc478152673)

[Обзор предметной области 4](#_Toc478152674)

[Описание применимости 7](#_Toc478152675)

[описание результатов проекта 8](#_Toc478152676)

[ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА 9](#_Toc478152677)

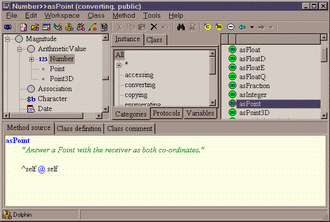
# Цель работы

Создать интегрированную среду разработки ООП систем с возможностью генерации исходного кода на его основе.

# Обзор предметной области

Рассматриваемая мною предметная область относиться к разработке веб-приложений и сайтов с использованием языка PHP.

**Основная идея проекта** – это создать интегрированную среду разработки по аналогии среды Smalltalk, для создания, изменения и проектирования ООП системы будущего веб-приложения.



Проект не несёт цели создания полного аналога/аналогов для Web`а, а берёт идею, в соответствии с которой манипулирование с классами ведётся с помощью графического интерфейса.

Аналогами являются:

* Dolphin Smalltalk
* VisualWorks
* Visual Smalltalk
* И другие реализации Smalltalk

Язык Smalltalk и его интерактивная среда разработки неразрывно связанны. Сам язык используется для решения следующих задач:

|  |  |
| --- | --- |
| * Автоматизация, робототехника * Диспетчеризация, планирование * Коммуникация, связь * Медицина | * Обработка коммерческой информации * Системы управления * Тренажёры, моделирования * Обучение программированию |

Сам подход к написанию программ в языке Smalltalk кардинально отличается от подхода современных языков.

В таких языках как C++, Java, C#, Python, Ruby и т.п. мы разрабатываем программу путём композиции библиотек/модулей/пакетов в связке с особенностями парадигмы и самого языка, а в конечном итоге мы получаем программу, которая решает поставленную задачу.

В случае с языком Smalltalk, который неразрывно связан с графической средой, разработка идёт путём декомпозиции, то есть мы из "коробки" имеем библиотеки/модули/пакеты и затем в процессе разработки мы убираем лишнее. Так же мы занимаемся реализацией своих решений и создания своих классов, объекто и т.д.

Как правило такая среда разработки предоставляет разработчику следующие инструменты:

* **Workspace** – рабочее пространство, в котором можно писать исходный код и выполнять его
* **Inspector** – инструмент для просмотра содержимого инспектируемого объекта
* **Class** **Browser** – основной инструмент, в котором происходит создание/удаление и изменения классов
* **Debugger** – отладчик

Помимо этого, вся система является единым целым и её можно изменять "на лету". Так же, среда разработки Smalltalk является самой продуктивной с точки зрения программирования.

В качестве основного языка для реализации проекта, был выбран PHP, так как он имеет общие черты с языком Smalltalk:

* Интерпретируемость
* Динамическая типизация
* Объекто-ориентированная парадигма
* Имеет виртуальную машину (Zend Engine)

Ниже приведены

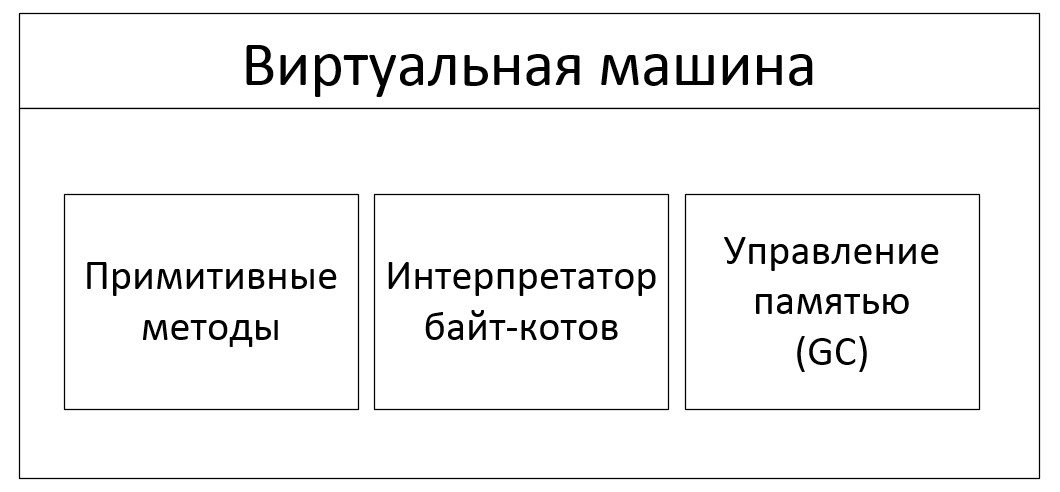


Рисунок A. Устройство VM Smalltalk

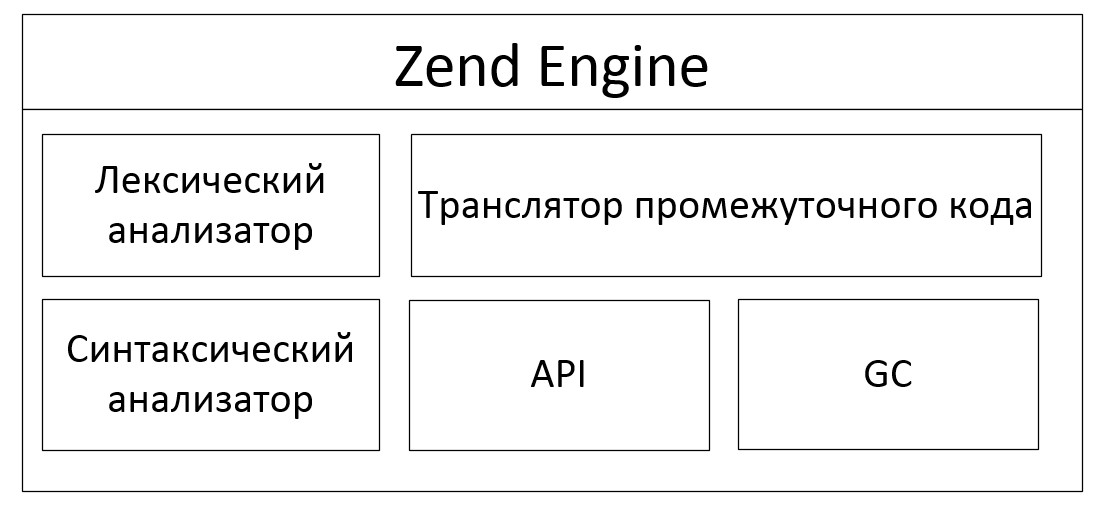


Рисунок B. Устройство VM Zend Engine

Так же есть и недостатки:

* Язык не является чисто объекто-ориентированным
* Язык не предназначен для реализации огромных комплексных систем и приложений

# Описание применимости

Разрабатываемая среда должна решать следующие задачи:

* Управления ООП системой (образом)
  + Создание классов
  + Изменения реализации классов
* Оперативный доступ к частям системы
* Возможность корректировки генерируемого исходного кода
* Экспорт/импорт образов в виде упакованных объектов

Лично я считаю, что данная среда позволит больше сосредоточиться на следующих этапах разработки:

1. Проектирование
2. Быстрое тестирование отдельных классов
3. Исправлении ошибок в классах

Во время работы со средой, все выполняемые действия сохраняются во временном образе. После создания основного образа уже можно будет по нему сгенерировать код или сразу испытать работу отдельных классов прямо в среде. Таким образом мы не работает с файлами, как таковыми, а работаем с целым образом, что позволяет нам легко получить доступ к реализации кого-либо класса и при необходимости внести корректировки в код.

# описание результатов проекта

# дополнительные материалы

В этом разделе я оставлю ссылки на некоторые статьи о языке Smalltalk и PHP.

* Smalltalk
  + [История языка](http://www.smalltalk.ru/articles/smalltalk.html)
  + [Почему Smalltalk?](http://www.smalltalk.ru/articles/why.html)
* PHP

# ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА

1. История языка Smalltalk (статья) URL: <https://goo.gl/uc7v3q>
2. Неизвестный Smalltalk (статья) URL: <https://goo.gl/X8QaX2>
3. Smalltalk: Концепция объекто-ориентированного программирования (статья) URL: <https://goo.gl/L8A38Q>
4. «Профессиональное программирование на PHP» (книга) URL: <https://goo.gl/7mCwsM>