Архитектура интерпретатора для исполнения программ на языке PostScript в JVM

Рустам Макулов

444 группа

29 мая 2014 г.

Научный руководитель: Д.Ю. Булычев

Введение

JVM

- Виртуальная машина для исполнения байт-кода
- Универсальная платформа для исполнения программ на многих языках программирования

Введение

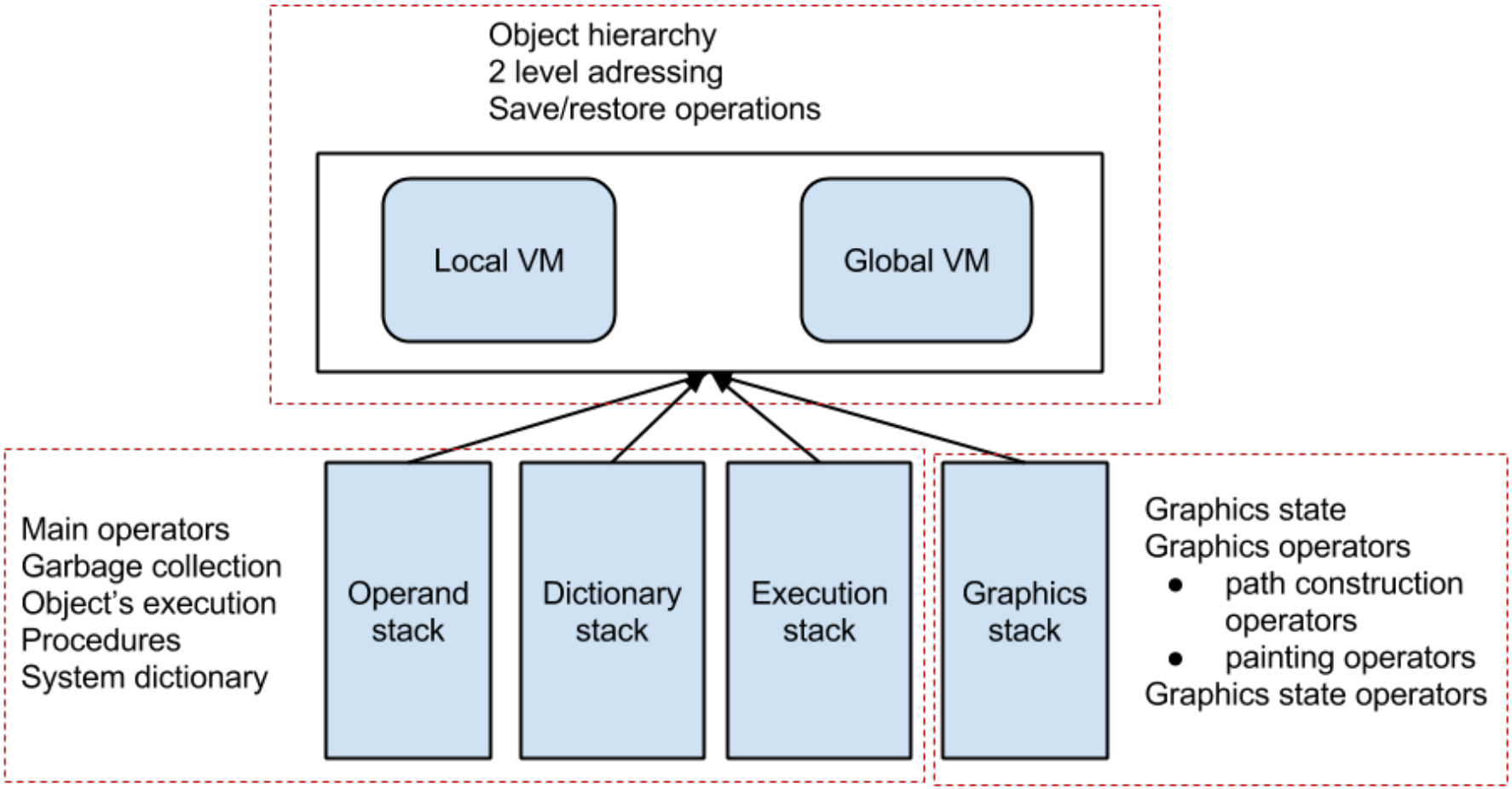
PostScript

- Мощные графические средства в рамках легко расширяемого интерпретируемого языка программирования с широким спектром возможностей для управления выполнением программ
- Абстрактность модели синтеза изображений позволяет не зависеть от характеристик устройств отображения

Цели работы

- Цель курсовой – описание и реализация правильным образом всех основных компонент языка для работы переносимого интерпретатора
- Предполагаемые результаты:
  - корректная иерархия объектов
  - реализация модели памяти языка
  - реализация операторов save и restore

Общая архитектура



Виртуальная память

- Объекты делятся на простые и сложные
- Виртуальная память языка предназначена для хранения значений сложных объектов
- Разделяется на локальную и глобальную памяти
- Значения объектов, расположенных в локальной памяти, сохраняются и восстанавливаются операторами save и restore

Операторы save и restore

- save сохраняет состояния локальной памяти и возвращает сохраненный объект в виде снимка
- restore возвращает локальную память к снимку, сгенерированному предшествующим оператором save
- Более эффективный и простой, но менее универсальный способ управления памятью, чем сборка мусора

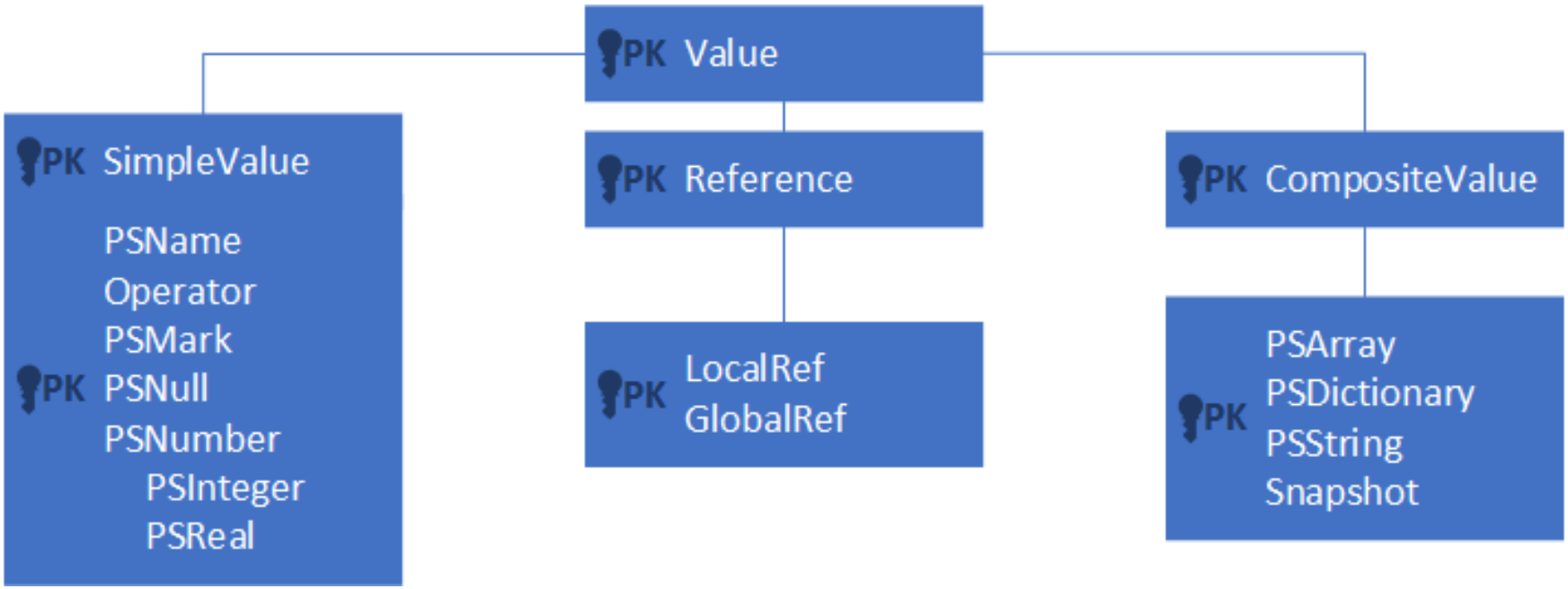
Этапы реализации

- Простая иерархия объектов без типов и атрибутов
- Среда исполнения: стек операндов и локальная память, представляющая собой ArrayList из объектов
- Юнит-тесты для save/restore для проверки всех дальнейших реализаций среды исполнения
- reference для значений всех объектов, включая простые
- reference только для значений сложных объектов
- Поддержка сборки мусора, ArrayList заменен на HashMap

Реализация операторов save и restore

- save -> сохранение состояния локальной памяти
  - создание снимка состояния локальной памяти
  - дальнейшее изменение объектов не затрагивает снимок
- restore -> восстановление состояния локальной памяти
  1. проверка: на стеке операндов не должно быть сложных объектов, созданных после сохранения состояния
  2. обновление значения всех строк со снимка
  3. восстановление значений всех объектов локальной памяти

Реализация иерархии объектов

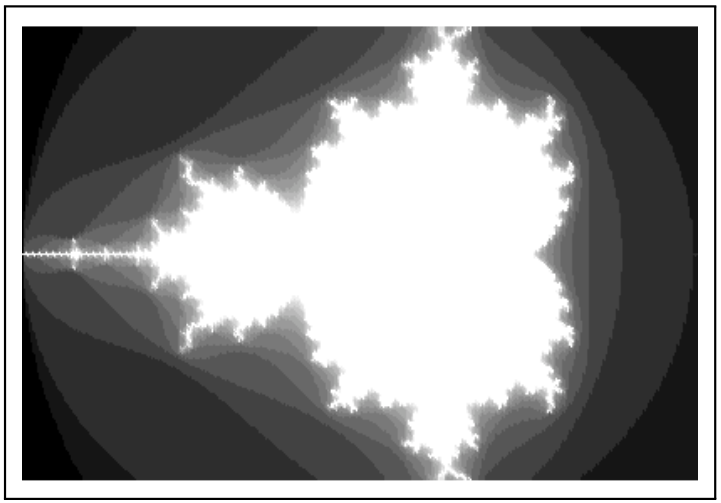


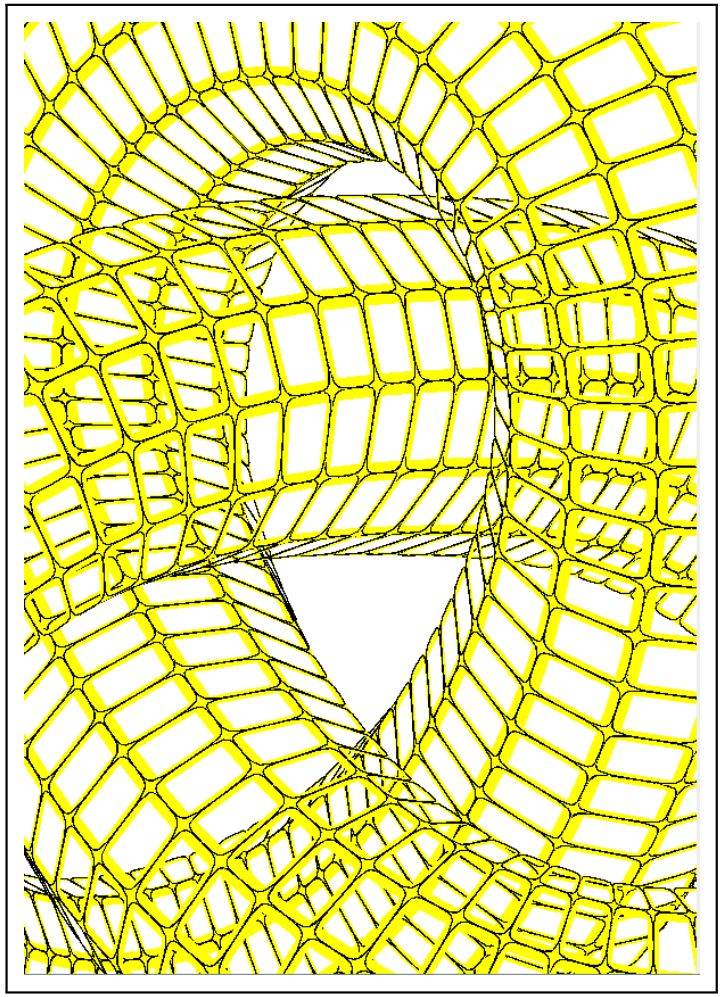
- Для restore значения неизменяемые

Реализация локальной памяти

- Двухуровневая адресация:
  - объекты содержат ссылки на память
  - память содержит ссылки на значения
- restore подменяет значение по ссылке, не меняя саму ссылку

Примеры





Итоги

- Построена и реализована архитектура интерпретатора
- Реализованная архитектура правильным образом описывает все основные компоненты
- Реализована иерархия объектов
- Реализованы операторы сохранения и восстановления состояния локальной памяти