

Инструментальная среда для анализа программных систем

А.М. Полоцев, гр. 63501/13
научный руководитель: к.т.н. доцент В.М. Ицыксон

Санкт-Петербургский Политехнический Университет

XLII Неделя науки

- В различных методах анализа часто решаются похожие задачи:
 - Построение моделей программы (AST, CFG и т.п.)
 - Построение метрик
 - Реинжиниринг программного обеспечения (оптимизация, рефакторинг и т.п.)
 - Визуализация свойств системы

- В различных методах анализа часто решаются похожие задачи:
 - Построение моделей программы (AST, CFG и т.п.)
 - Построение метрик
 - Реинжиниринг программного обеспечения (оптимизация, рефакторинг и т.п.)
 - Визуализация свойств системы

Обычно эти задачи решаются вручную

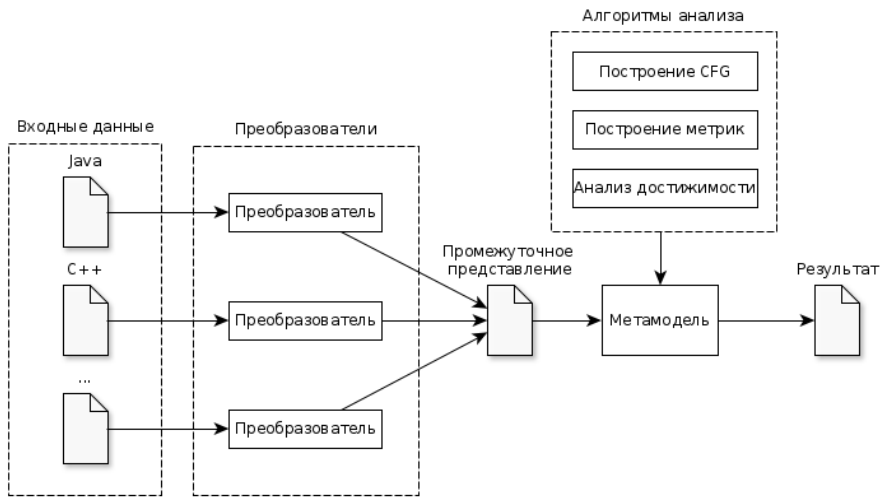
- Цели

- Автоматизация анализа
- Визуализация свойств системы

- Требования

- Поддержка анализа объектно-ориентированных языков
- Извлечение различных моделей анализируемой системы
- Встроенные средства отображения
- Модульная расширяемая структура
- Экспорт разных артефактов системы во внешние отчеты
- Предоставление API к разработанным методам анализа

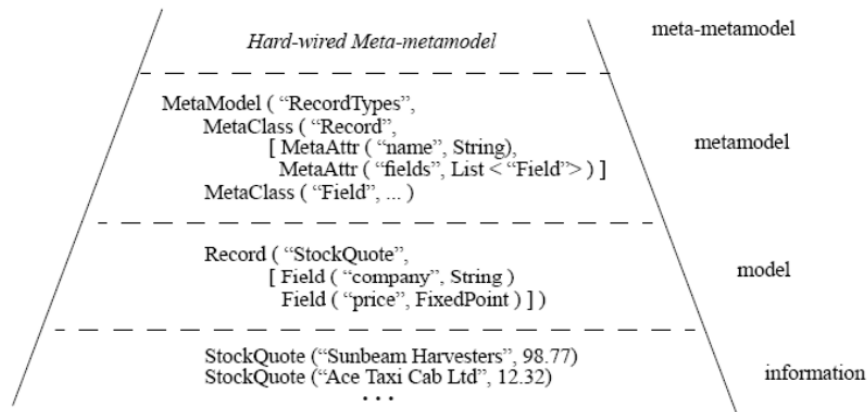
Архитектура инструментального средства




- Цели:
 - Унификация представления системы для проведения анализа и трансформаций
- Требования:
 - Независимость от языка описания анализируемой системы
 - Достаточная мощность для извлечения различных моделей
 - Расширяемость путем добавления новых метаданных

- Стандарт, разработанный Object Management Group (OMG)
- Метамета модель для описания метамodelей
- Поддержка объектно-ориентированной парадигмы
- Четырехуровневая архитектура
 - Уровень метамета модели (M3)
 - Уровень метамodelей (M2)
 - Уровень моделей (M1)
 - Информационный уровень (M0)

Meta Object Facility (MOF)





Missing
figure

Диаграмма классов

Преобразование исходного кода в метамодель



