

# Интеграция Java и Ruby/Python

# Зачем?

- Ускорение процесса производства ПО
- Повышение комфорта при написании кода
- Возможность модификации кода «на лету»
- Доступ к библиотекам стороннего языка

# Jython

- Jython – реализация Python 2.1  
(август 2001 г)
- Стабильная версия хорошо реализующая Python и его библиотеки.
  - Есть компилятор в Java байткод.
  - Прежнее название: JPython

# JRuby

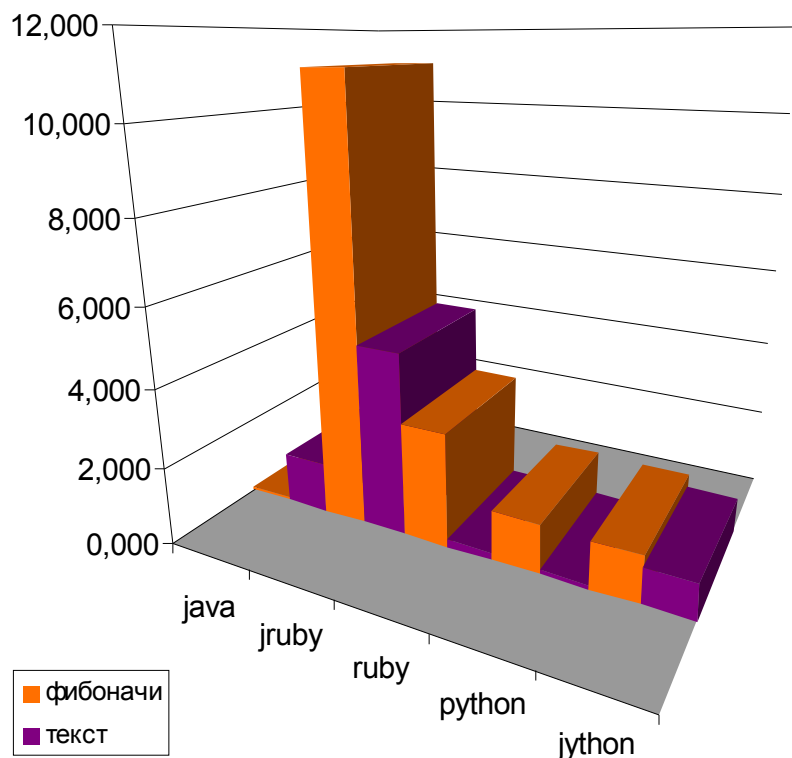
Реализация языка Ruby 1.8.4  
(декабрь 2005)

- Молодой развивающийся проект
- Современная реализация
- Возможен запуск RoR

# Поддержка в Java IDE

- Поддержка подсветки, дополнения и пр. есть в IntelliJ IDEA и Eclipse через дополнительные плагины.
- Есть проблемы при интеграции разных языков, т.е. дополнения и пр. при связи языков работают некорректно.
- Проблемы при отладке, когда часть кода на одном а часть на другом языке.
- Плохо совместимы с такими вещами как CodeCoverage, Unit-тестами, профайлингом.

# Производительность



Была произведена проверка языков на двух алгоритмах:  
- Подсчет вхождений слова в текстовый документ  
- Вычисление числа из ряда Фибоначи.

Данный тест на производительность довольно субъективный, но должен отображать общую тенденцию.

По его результатам видно что jruby очень слаб для задач где множество простых арифметических действий.

# Текущие проблемы - Jython

- Очень устаревшая версия базового языка, в результате чего большинство реальных программ не могут быть под ним запущены.
- Очень слабое развития проекта, лишь мелкие и редкие коммиты в SVN

# Текущие проблемы - JRuby

- Не полная реализация Ruby и его библиотек, некоторые вещи ведут себя по разному в JRuby и Ruby
- Отсутствие компилятора в Java байткод.
- Проблема при использовании в Java классов расширенных в Ruby
- Низкая производительность

# Заключение

- JRuby можно выбрать как язык облегчающий написание дополнительного функционала и инфраструктуры проекта
- Jython можно выбрать как язык реализующий в том числе часть важной функциональности, благодаря своей производительности и надежности, но нужно смириться с тем что невозможно будет использовать многие привычные техники.
- В любом случае данные языки нерационально использовать в основе реализации основных алгоритмов
- Для полноценной работы требуется большая работа по улучшению существующих IDE и дополнительных инструментов