Análise comparativa de produtividade entre Groovy e Java, aplicado no desenvolvimento web

Vandir Fernando Rezende

Orientado por Marcel Hugo

Roteiro

- Introdução
- Objetivos
- Fundamentação Teórica
- Desenvolvimento
- Resultados
- Conclusão
- Extensões

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Introdução

Linguagem A é mais produtiva que linguagem B.

- Como comprovar que uma linguagem é mais produtiva que outra?
- A afirmação está baseada no que?
- Em qual segmento de software?
- Quais os critérios de avaliação?

Objetivos

- Comparar a produtividade, no desenvolvimento web, entre as linguagens Groovy e Java
  - Definir os critérios de avaliação
  - Analisar o aplicativo, estudo de caso
  - Verificar as diferenças entre Groovy e Java
  - Implementar o estudo de caso em ambas as linguagens
  - Estabelecer os resultados da comparação

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Fundamentação Teórica

- NBR-13596
  - — Funcionalidade
  - — Confiabilidade
  - — Usabilidade
  - — Eficiência
  - — Manutenibilidade
  - — Portabilidade
- Características das linguagens
  - — Ortogonalidade
  - — Simplicidade Global
  - — Legibilidade
  - — Tipos de Dados e Estrutura
  - — Sintaxe
  - — Capacidade de Escrita
  - — Abstração
  - — Expressividade
  - — Confiabilidade
  - — Verificação de Tipos
  - — Tratamento de Exceção

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Fundamentação Teórica

- Groovy
- Grails
- Java
- JCompany

Trabalhos Correlatos

- NBR-13596
  - Avaliação da qualidade de sites acadêmicos (Rossi, 2002)
- Groovy
  - RunGroovy: extensão do BlueJ (Müller, 2007)
- Scrum
  - PRONTO! Software para gestão de projetos ágeis (Gomes, 2009)
  - Ambiente web para gestão de processo de software (Pereira, 2005)

Desenvolvimento

- Aplicação da NBR-13596 nos critérios de avaliação
- Correlação entre as características das linguagens com a norma
- Meio de avaliação dos critérios
- Especificação dos casos de uso
- Cálculo dos UCPs
- Diferenças entre Groovy e Java

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Desenvolvimento

- Implementação do aplicativo em Groovy
- Implementação do aplicativo em Java
- Resultado do questionário de avaliação
- Produtividade por UCP
- Comparação de desempenho dos aplicativos

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

NBR-13596

- Produtividade
- Usabilidade
  - Inteligibilidade
  - Apreensibilidade
- Eficiência
  - Tempo
  - Recurso
- Manutenibilidade
  - Analisabilidade
  - Modificabilidade
- Confiabilidade

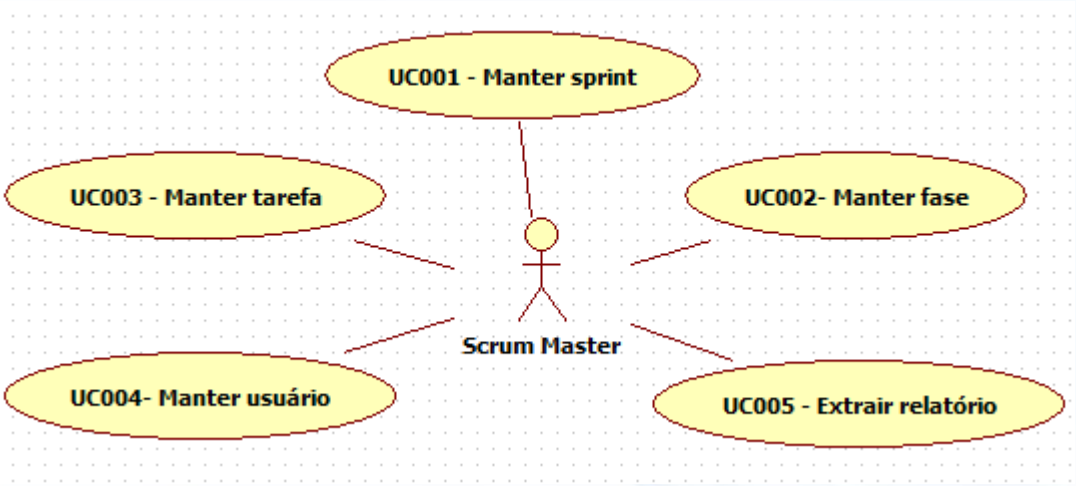
Correlação

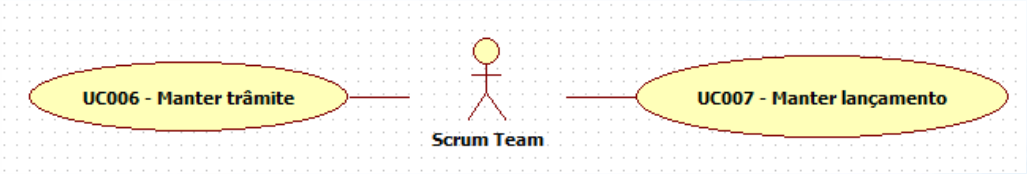
	Custo	Usabilidade	Eficiência	Manutenibilidade	Confiabilidade
Ortogonalidade	X	X		X	X
Simplicidade global		X			X
Legibilidade	X			X	X
Tipo de dados e estrutura		X			
Sintaxe		X			
Capacidade de escrita	X		X	X	
Abstração		X	X	X	X
Expressividade	X		X	X	
Verificação de tipos	X				X
Tratamento de exceção	X		X		X

Meio de Avaliação

- Estático
  - Usabilidade
  - Manutenibilidade
  - Confiabilidade
- Dinâmico
  - Produtividade
  - Eficiência

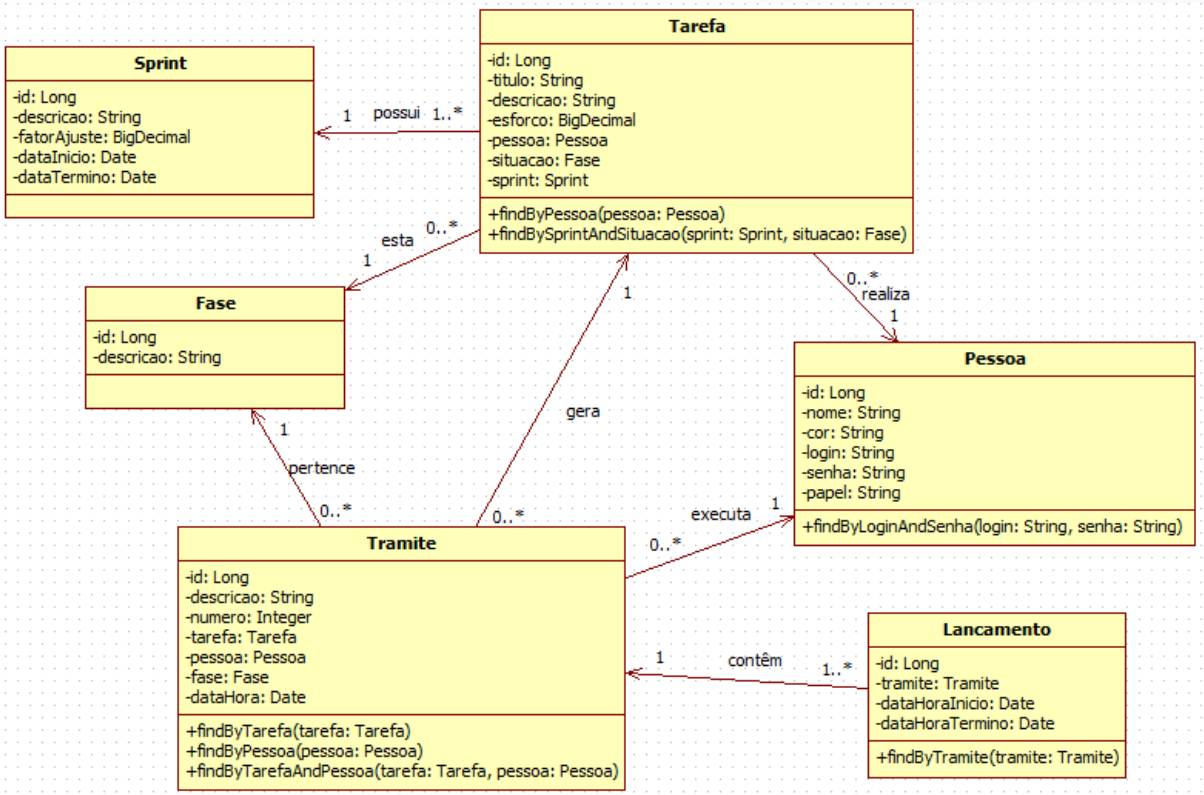
Casos de Uso





FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Diagrama de Classes



Cálculo dos UCPs

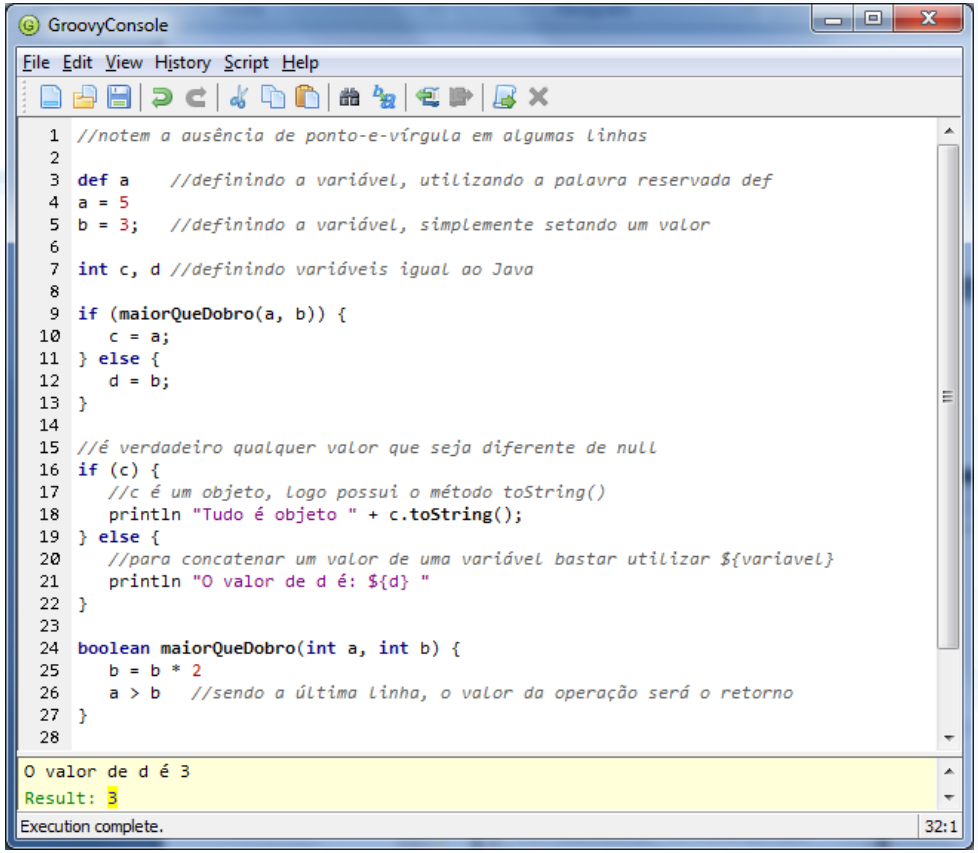
CASO DE USO	UCP
UC001 - Manter sprint	13,6
UC002 - Manter fase	13,6
UC003 - Manter tarefa	19,7
UC004 - Manter usuário	19,7
UC005 - Extrair relatório	25,9
UC006 - Manter trâmite	19,7
UC007 - Manter lançamento	25,9

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Diferenças

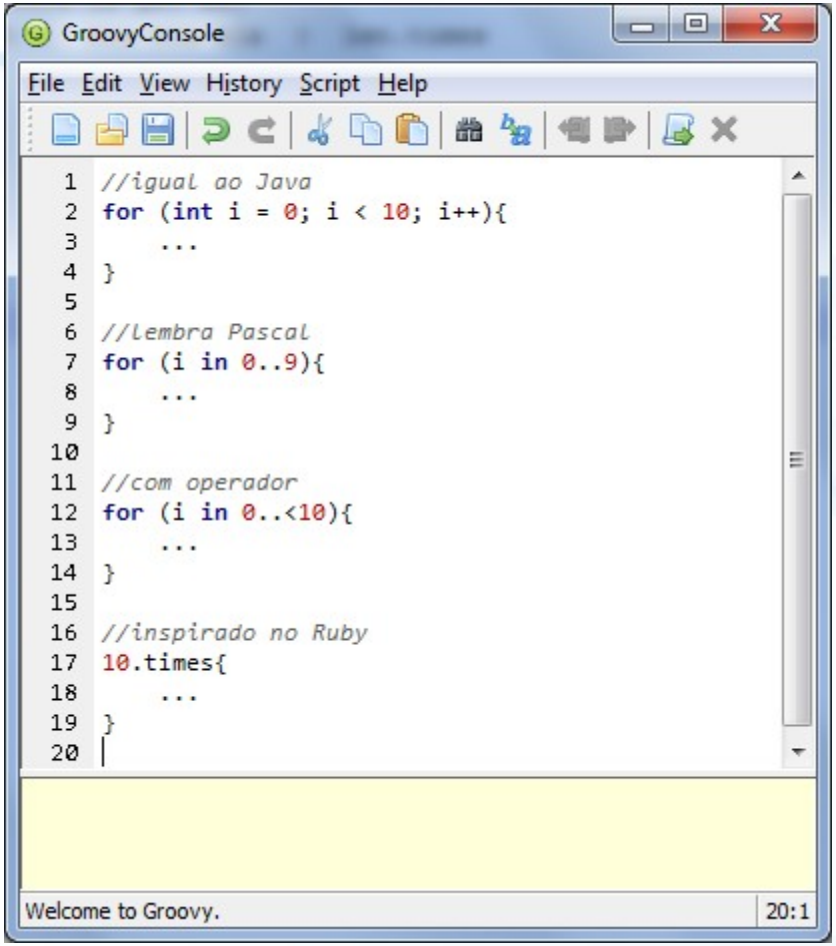
- Objetos
- Tipagem dinâmica / *Duck typing*
- Ponto-e-vírgula
- *Return*
- Igualdade
- Conceito de verdade
- Concatenação de *String*

Diferenças


GroovyConsole
File Edit View History Script Help
1 //notem a ausência de ponto-e-vírgula em algumas linhas
2
3 def a   //definindo a variável, utilizando a palavra reservada def
4 a = 5
5 b = 3;   //definindo a variável, simplemente setando um valor
6
7 int c, d //definindo variáveis igual ao Java
8
9 if (maiorQueDobro(a, b)) {
10   c = a;
11 } else {
12   d = b;
13 }
14
15 //é verdadeiro qualquer valor que seja diferente de null
16 if (c) {
17   //c é um objeto, logo possui o método toString()
18   println "Tudo é objeto " + c.toString();
19 } else {
20   //para concatenar um valor de uma variável bastar utilizar ${variavel}
21   println "O valor de d é: ${d} "
22 }
23
24 boolean maiorQueDobro(int a, int b) {
25   b = b * 2
26   a > b   //sendo a última linha, o valor da operação será o retorno
27 }
28
O valor de d é 3
Result: 3
Execution complete.
32:1

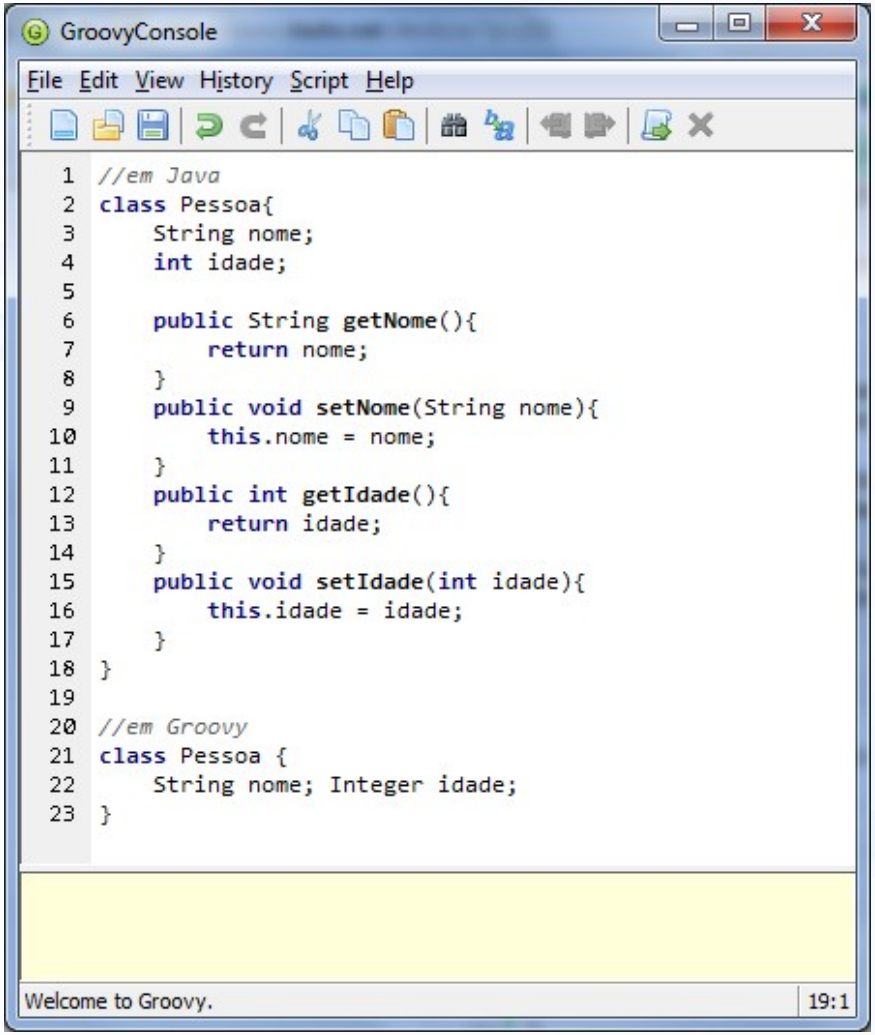
Diferenças

- Laços de repetição



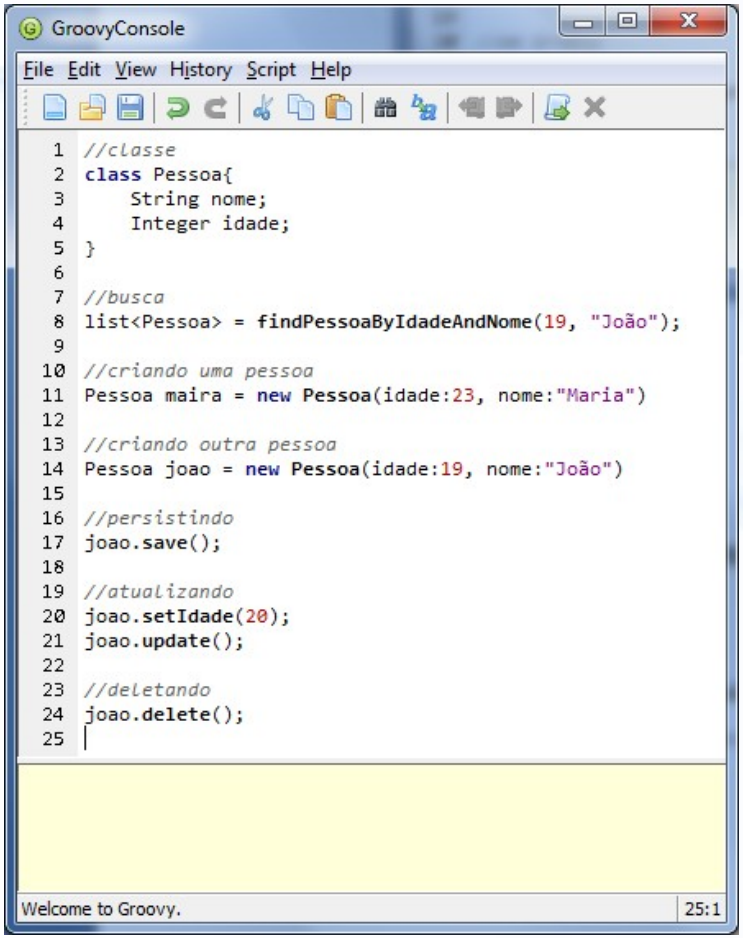
Diferenças

- Groovy Beans



Diferenças

- GORM (*Groovy Object Relational Mapping*)

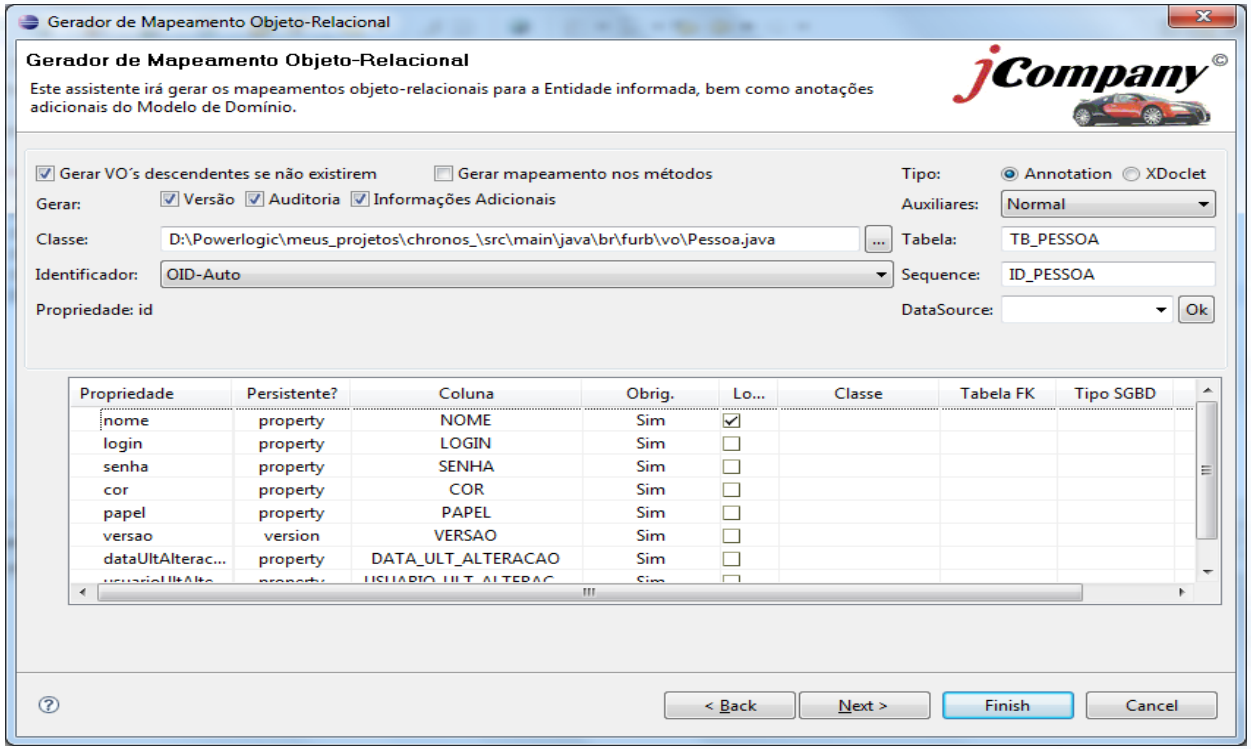


Implementação (Java)

- JCompany
  - *Struts*
  - *Tiles*
  - *Hibernate*
  - *Log4j*
  - XML
  - *Wizards*

Implementação (Java)

- JCompany Wizard



FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Implementação (Java)

- JCompany classe de modelo

```java
package br.furb.vo;

import java.math.BigDecimal;

@MappedSuperclass
public abstract class Tarefa extends PlcBaseVO {

    @Id @GeneratedValue(strategy=GenerationType.AUTO, generator = "ID")
    @Column (name = "ID", nullable=false)
    private Long id;

    @Column (name = "TITULO", nullable=false)
    private String titulo;

    @Column (name = "DESCRICAO", nullable=false)
    private String descricao;

    @Column (name = "ESFORCO", nullable=false)
    private BigDecimal esforco;

    @SuppressWarnings("unchecked")
    @ManyToOne (targetEntity = PessoaVO.class, fetch = FetchType.EAGER)
    @JoinColumn (name = "ID_PESSOA", nullable=false)
    private Pessoa pessoa;

    @SuppressWarnings("unchecked")
    @ManyToOne (targetEntity = SituacaoVO.class, fetch = FetchType.EAGER)
    @JoinColumn (name = "ID_SITUACAO", nullable=false)
    private Situacao situacao;

    @SuppressWarnings("unchecked")
    @ManyToOne (targetEntity = SprintVO.class, fetch = FetchType.EAGER)
    @JoinColumn (name = "ID_SPRINT", nullable=false)
    private Sprint sprint;
```

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Implementação (Groovy)

- Grails
  - *Spring*
  - *SiteMesh*
  - *JUnit*
  - GORM
  - Configuração por convenção

Implementação (Groovy)

- Grails (instalação)
  - Download em grails.org
  - Descompactar em pasta base
  - GRAILS_HOME
  - PATH

```
Microsoft Windows [versão 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Todos os direitos reservados.

C:\Users\vandir.rezende>grails -version
Welcome to Grails 1.3.7 - http://grails.org/
Licensed under Apache Standard License 2.0
Grails home is set to: C:\grails
```

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Implementação (Groovy)

- Grails (criação de projeto)

```
grails create-app nome_projeto
```

```
%PROJECT_HOME%
+ grails-app
   + conf              Mantém as configurações gerais da aplicação
      + hibernate         Mantém as configurações de acesso a banco de dados
      + spring            Mantém as configurações opcionais do Spring
   + controllers       Mantém as classes de controle
   + domain            Mantém as classes de domínio
   + i18n              Mantém as classes de internacionalização
   + services          Mantém as classes de serviços (equivalente a local
                       session bean)
   + taglib            Mantém as bibliotecas de tags
   + views             Mantém os templates de visão, com subdiretórios para cada
                       classe de controle.
      + layouts           Mantém os layouts de templates
   + grails-tests      Mantém as classes de testes unitários
   + lib               Mantém as bibliotecas extras necessárias para a aplicação
   + src               Mantém arquivos fontes adicionais
      + groovy            Mantém arquivos Groovy adicionais
      + java              Mantém arquivos Java Adicionais
   + web-app
      + css               Mantém as folhas de estilo da aplicação
      + images            Mantém os arquivos de imagens
      + js                Mantém as bibliotecas javascrips
      + WEB-INF           Mantém os arquivos de deploy da aplicação
```

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Implementação (Groovy)

- Grails (criação de rotina)

```
cd nome_projeto
grails create-domain-class br.furb.NomeClasse
```

Definir atributos

```
grails create-controller br.furb.NomeClasse
```

```
package br.furb

class SprintController extends BaseController{
    def scaffold = true
}
```

```
meClasse
grails run-app
```

http://localhost:8080/nome_projeto

Resultados

- Características estáticas

CARACTERÍSTICA	GROOVY	JAVA
Ortogonalidade		X
Simplicidade global		X
Legibilidade		X
Tipos de dados e estrutura	X	
Sintaxe	X	
Capacidade de escrita	X	
Abstração	X	
Expressividade	X	
Confiabilidade		X
Verificação de tipos		X
Tratamento de exceção	X	

Resultados

- Características dinâmicas

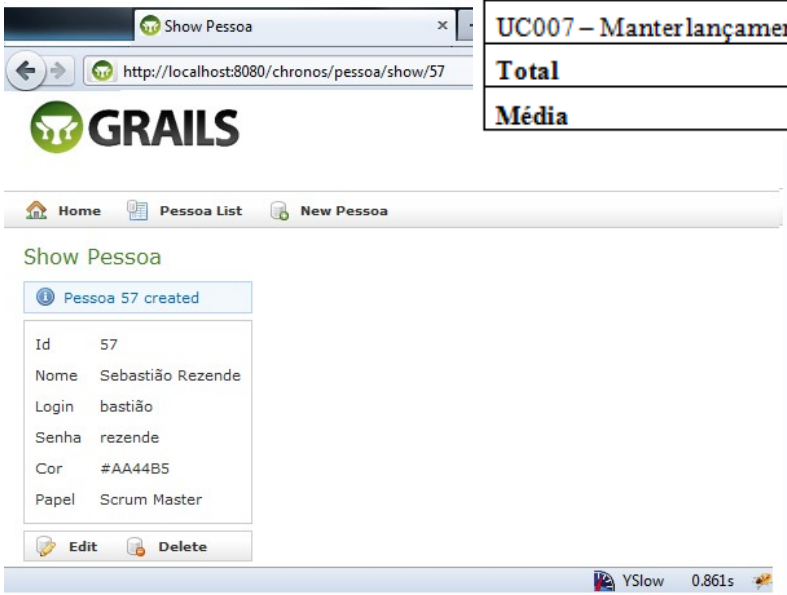
CASO DE USO	GROOVY (min)	JAVA (min)	UCP	f(GROOVY)	f(JAVA)	G - J (min)	%
UC001 - Manter *sprint*	80	135	13,6	5,88	9,93	-55	40,74
UC002 - Manter fase	40	75	13,6	2,94	5,51	-35	46,67
UC003 - Manter tarefa	180	265	19,7	9,14	13,45	-85	32,08
UC004 - Manter usuário	130	210	19,7	6,60	10,66	-80	38,10
UC005 - Extrair relatório	240	45	25,9	9,27	1,74	195	-433,33
UC006 - Manter trâmite	150	210	19,7	7,61	10,66	-60	28,57
UC007 - Manter lançamento	235	360	25,9	9,07	13,90	-125	34,72
Total	**815**	**1255**	**112,20**	**41,24**	**64,11**	**-440**	-
Média	**137,83**	**209,17**	**18,70**	**6,87**	**10,64**	**-73,33**	**36,81**

FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Resultados

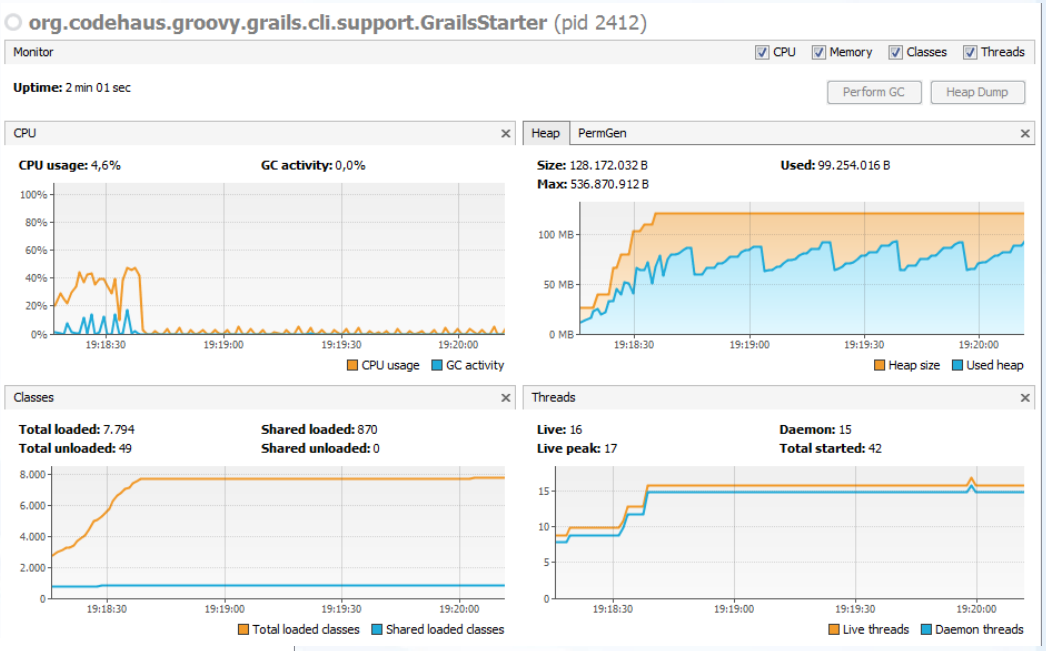
- Características dinâmicas

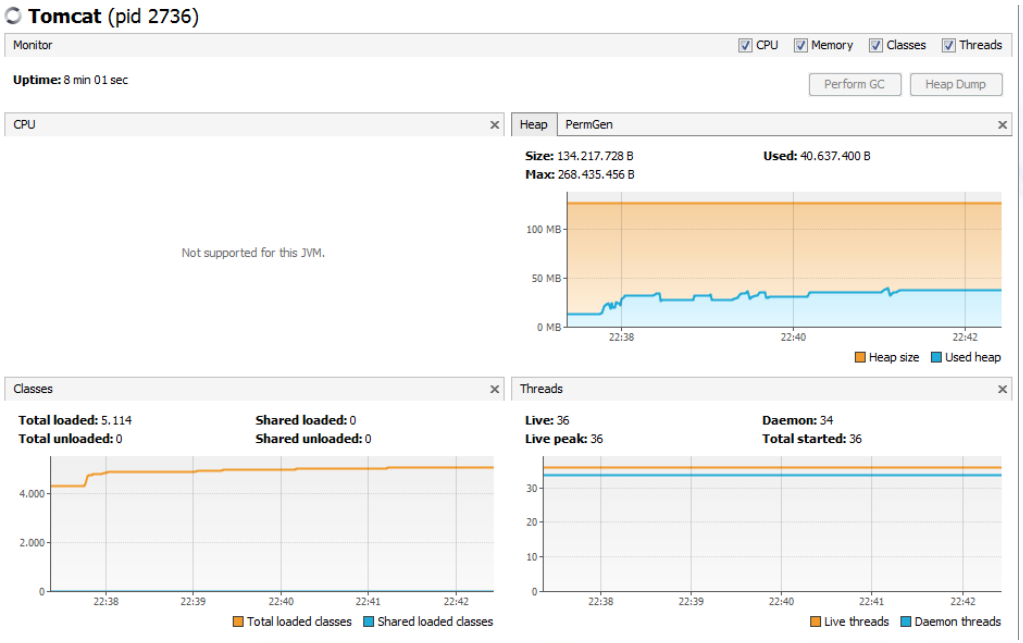
CASO DE USO	Groovy (ms)				Java (ms)				%
	Gerar	Listar	Apagar	Média	Gerar	Listar	Apagar	Média	
UC001 – Manter *sprint*	1078	652	669	**799,67**	972	576	595	**714,33**	**10,67**
UC002 – Manter fase	809	536	521	**622,00**	744	487	479	**570,20**	**8,33**
UC003 – Manter tarefa	956	217	263	**478,67**	887	206	242	**444,59**	**7,12**
UC004 – Manter usuário	847	228	279	**451,33**	779	235	247	**420,33**	**6,87**
UC005 – Extrair relatório	1843	416	-	**1129,50**	1807	398	-	**1102,50**	**2,39**
UC006 – Manter trâmite	412	233	398	**347,67**	379	204	356	**313,01**	**9,97**
UC007 – Manter lançamento	389	207	364	**320,00**	338	187	315	**280,00**	**12,50**
Total	**6334,00**	**2489,00**	**2494,00**	**4148,83**	**5906,00**	**2293,00**	**2234,00**	**3844,97**	-
Média	**904,86**	**355,57**	**415,67**	**592,69**	**843,71**	**327,57**	**372,33**	**549,28**	**8,26**



Resultados

- Características dinâmicas





FURB
UNIVERSIDADE REGIONAL DE BLUMENAU

Conclusão

- Groovy é 35% mais produtivo
- Java é 10% performático
- Java consome 50% menos memória
- Groovy aloca 2500 classes a mais

- Groovy possui características para produtividade
- Java destacam-se características para maturidade

Conclusão

Produtividade X Performance

Extensões

- Analisar a arquitetura (performance)
- *Profilers* (memória)
- Comparar Groovy com demais linguagens
- Usar Groovy/Grails em futuros trabalhos
- Integrar o estudo de caso com o PRONTO! (Gomes, 2009)

Obrigado!

FIM.