

Troper

- ▶ O sistema
- ▶ O plugin
- ▶ As ferramentas
- ▶ As metodologias
- ▶ Diga NÃO!
- ▶ Diga SIM!

Troper - O Sistema Hospedeiro

- ▶ Baseado nos modelos já existentes no sistema
- ▶ Usa as regras do negócio já existentes

Troper - O Sistema Hospedeiro

Listing 1: Modelo ActiveRecord

```
1 # app/models/person.rb
2 class Person < ActiveRecord::Base
3   has_many :addresses
4 end
5 # app/models/address.rb
6 class Address < ActiveRecord::Base
7   belongs_to :person
8 end
```

Troper - A template Liquid padrão

- ▶ Usa as regras de negócio dos modelos
- ▶ O plugin entende estas regras através das *reflections* do modelo
- ▶ Extensibilidade das tags para liquid
- ▶ Criação de formatadores e outros de forma simples

Troper - Um exemplo Liquid

Listing 2: Template do Liquid

```
1 <table>
2   {% for person in people %}
3     <tr>
4       <td>{{ person.name }}</td>
5       <td>{{ person.email | link_to_mail }}</td>
6       <td>{{ person.phone | format_phone }}</td>
7     </tr>
8   {% endfor %}
9 </table>
```

Troper - Um exemplo do Troper

Listing 3: Usando o Troper

```
1 report = Troper::Report.new("people")
2 report.columns.add "addresses"
3 puts report.template_to_resource
```

Troper - Consultando a template

Listing 4: Resultado do puts anterior

```
1 <table id="people">
2   <tr>
3     <td>Name</td><td>Email</td><td>Phone</td><td>Address</td>
4   </tr>
5   {% for person in people %}
6   <tr><td>{{ person.name }}</td>
7     <td>{{ person.email }}</td>
8     <td>{{ person.phone }}</td>
9     <td>
10       <table id="addresses">
11         <tr><td>Description</td><td>City</td><td>State</td>
12         {% for address in person.addresses %}
13           <tr><td>{{ address.description }}</td>
14             <td>{{ address.city }}</td>
15             <td>{{ address.state }}</td></tr>
16         {% endfor %}
17       </table>
18     </td>
19   </tr>
20 </table>
```

Troper - Consultando a saída do relatório

Listing 5: report.output

```
1 >> puts report.output
2 <table id="people">
3   <tr><td>Name</td><td>Email</td><td>Phone</td><td>
4   <tr><td>Jonatas Davi Paganini</td>
5   <td>jonatasdp@gmail.com</td>
6   <td>46 9911 7879</td><td>
7   <table id="addresses">
8     <tr><td>Description</td>
9       <td>City</td><td>State</td></tr>
10    <tr><td>rua pernambuco, 123, fco beltrao - pr</td>
11    <tr><td>avenida julio assis, fco beltrao - pr</td>
12 </table>...
```


Troper - Cadê os filtros?

Listing 6: Adicionando filtros

```
1 report.filters << "person.addresses != empty"
```

Na template sericionada um **if**:

Listing 7: Exibindo na template

```
1 {% for person in people %}  
2     {% if person.addresses != empty %}  
3         <tr><td>{{ person.name }}</td> #...
```

Listing 8: Adicionando formatadores

```
1 col = report.columns[:description]  
2 col.formats << "truncate:_50"
```

Na template sericionada um **mdo**:

Listing 9: Exibindo na template

```
1 # ...  
2 <td>{{ record.description | truncate: 50 }}</td>  
3 #...
```

Troper - Ferramentas

- ▶ Rails
- ▶ ExtJS
- ▶ Liquid Templates
- ▶ Terminal Unix
- ▶ GIT
- ▶ Vim
- ▶ Cucumber
- ▶ Rake
- ▶ GitHub
- ▶ LaTeX

Troper - As metodologias

- ▶ Behaviour Development Driven - Cucumber
- ▶ DRY - Don't repeat yourself
- ▶ Convention Over Configuration
- ▶ Prototipação

Troper - Padrões de Projeto

- ▶ Forwardable
- ▶ Command Pattern

Troper - Não fez uso das tecnologias

- ▶ Windows
- ▶ IDE
- ▶ Banco de dados para armazenamento de objetos

Troper - Apoia!

- ▶ Ruby
- ▶ Open Source
- ▶ Linux
- ▶ GIT