



# elixir

## **From Perl to Elixir**

Ruben Amortegui -  
@ramortegui  
<http://rubenamortegui.com>  
<https://github.com/ramortegui>

# Agenda

- How I meet Elixir
- Personal needs to be productive
- Case: Porting EcCart from Perl to Elixir

# Meeting Elixir

- Background
  - Java
  - PHP + JavaScript
  - PHP and JAVA (joomla and XML-RPC)
  - Ruby on Rails
  - Perl 5
  - Meteor

# Meeting Elixir

- New challenges
  - IOT
  - RealTime apps
- Which options do I have?
  - Ruby
  - JavaScript
  - Perl5
  - Php
  - Elixir?

# Elixir

- Pros
  - Ready to scale
  - Syntax like Ruby
  - Has a Web Framework
  - Battletest Virtual Machine BEAM
- Cons (opportunities)
  - No working experience on Functional programming
  - Lack of packages
  - Jobs ( ? )

# Trying elixir, what do I need?

- Database abstraction
- Testing
- Web Framework
- Package manager
- Version Manager

# Trying elixir, what do I need?

- Database abstraction
  - Javascript: Bookshelf.js
  - Java: Hibernate
  - Ruby: Active Record
  - Perl: DBIx
  - Elixir: Ecto

# Trying elixir, what do I need?

- Testing
  - Javascript: Jasmine
  - Java: Junit
  - Ruby: Rspec, minitest
  - Perl: Test::More
  - Elixir: Test::Unit



# Trying elixir, what do I need?

- WebFramework
  - Javascript: Express.js
  - Java: Spring
  - Ruby: Rails, Sinatra
  - Perl: Catalyst, Dancer
  - Elixir: Phoenix

# Trying elixir, what do I need?

- Package manager
  - Javascript: npm
  - Java: maven
  - Ruby: rubygems
  - Perl: cpan
  - Elixir: hex

# Trying elixir, what do I need?

- Version Manager
  - Javascript: nvm
  - Java: Switchng env variables?
  - Ruby: rvm
  - Perl: perlbrew
  - Elixir: asdf

# Where to start?

- How can I do the things that I'm doing with my current programming language?
- `Dancer2::Plugin::Cart`

# Dancer2::Plugin::Cart

- It's a plugin for Dancer2 apps to help the developer to manage a e-cart and guide the checkout process on webstores.
  - Routes
  - Data Structure
  - Functions
  - Hooks to grow and add features
  - Config setup

# Dancer2::Plugin::Cart

- LIVE DEMO

# How I started

- Phoenix
  - Easily to modify existing code
  - It work's in a different way as the frameworks that I've been known.  
Eg. Plug, |> , Session
- Elixir
  - Elixir In Action
    - Get familiar with funcitonal programming, Elixir, and OTP concepts.

# Trying Elixir

- iex ( irb on ruby, re.pl )
- Defining variables
- Pattern matching
- Control Flow
- Data Abstractions
  - Maps, Structs



# Implementing EcCart

- The first version:

```
--- /dev/null
+++ b/lib/ec_cart.ex
@@ -0,0 +1,2 @@
+defmodule EcCart do
+end
```

# Implementing EcCart

- Define EcCartItem
- Add function to add Items
- Add function to calculate subtotals
- Define EcCartAdjustment
- Add EcCartAdjustments
- Add function to calculate totals
- Add server to preserve states by process id
- Add cache to manage multiple servers by id
- Add supervisor OTP

# Implementing EcCart

```
├── build
│   ├── dev
│   │   ├── consolidated
│   │   │   ├── Elixir.Collectable.beam
│   │   │   ├── Elixir.Enumerable.beam
│   │   │   ├── Elixir.IEx.Info.beam
│   │   │   ├── Elixir.Inspect.beam
│   │   │   ├── Elixir.List.Chars.beam
│   │   │   └── Elixir.String.Chars.beam
│   │   └── lib
│   │       ├── ec_cart
│   │       │   └── ebin
│   │       │       ├── ec_cart.app
│   │       │       ├── Elixir.EcCartAdjustment.beam
│   │       │       ├── Elixir.EcCart.beam
│   │       │       ├── Elixir.EcCartItem.beam
│   │       │       └── Elixir.EcCartServer.beam
│   └── config
│       └── config.exs
├── lib
│   └── ec
│       ├── cart
│       │   ├── adjustment.ex
│       │   ├── cache.ex
│       │   ├── item.ex
│       │   ├── server.ex
│       │   └── supervisor.ex
│       └── cart.ex
├── mix.exs
├── mix.lock
├── README.md
└── test
    ├── ec_cart_test.exs
    └── test_helper.exs
```

11 directories, 23 files

# Implementing EcCart

- Published:

[https://hex.pm/packages/ec\\_cart](https://hex.pm/packages/ec_cart)

- Source Code:

[https://github.com/ramortegui/ec\\_cart](https://github.com/ramortegui/ec_cart)

# Implementing EcCart

- LIVE DEMO

# Summary

- Elixir is fascinating.
  - Love the approach of functional programming.
- A lot of tools ready to be productive
- EcCart is available on Hex.io
- I'm working to make easier the development of webstores in elixir.

Thanks!

**Q & A?**

@ramortegui

<http://rubenamortegui.com>

<https://github.com/ramortegui>