# Tools,
# tools everywhere.

**@dcarral**
dcarral.org

github.com/dcarral/ruby-tools

# Agenda

➜ **Intro** (2 min)
Who are you?
Why tools?

➜ **(Ruby) Tools** (15-20 mins)
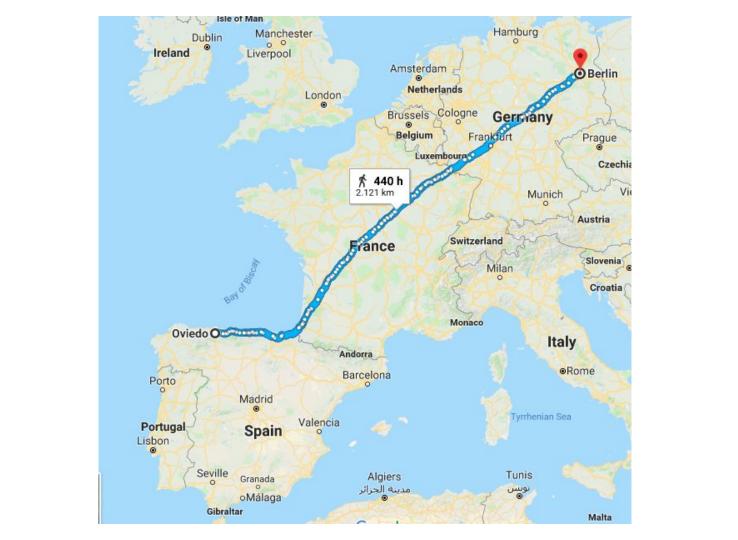Happy hacking! :)

➜ **Outro** (2 mins)
Using the source
And now?

# 1. Intro

**Who are you?**
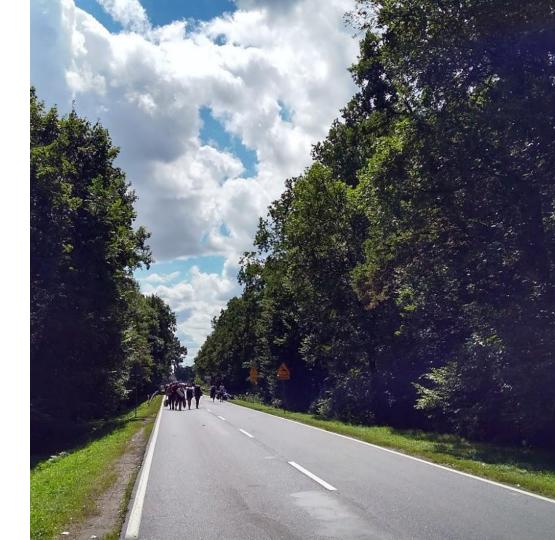
**Why tools?**

# Who are you?

Babbel

RUG::B

Softwerkskammer Berlin

# On a journey



Corey Haines

**And now...**

# Why tools?

# The
# Pragmatic
# Programmer

from journeyman
to master

## Andrew Hunt
## David Thomas

Foreword by Ward Cunningham

*"Tools amplify your talent. The better your tools, and the better you know how to use them, the more **productive** you can be."*

*David Thomas & Andrew Hunt*
*The Pragmatic Programmer*

# eXtreme Programming explained

## Programming

## eXplained

### EMBRACE CHANGE

Kent Beck

## Coding Standards

If you are going to have all these programmers changing from this part of the system to that part of the system, swapping partners a couple of times a day, and refactoring each other's code constantly, you simply cannot afford to have different sets of coding practices. With a little practice, it should become impossible to say who on the team wrote what code.

The standard should call for the least amount of work possible, consistent with the Once and Only Once rule (no duplicate code). The standard should emphasize communication. Finally, the standard must be adopted voluntarily by the whole team.

# 2. (Ruby) Tools

# First one?

# RuboCop

*"A Ruby static code analyzer, based on the community Ruby style guide."*
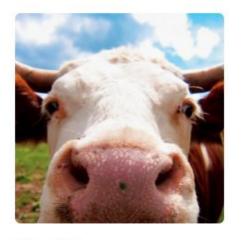
# RuboCop: tips & tricks

# Brakeman

*"A static analysis security vulnerability scanner for Ruby on Rails applications."*

**By the way...**

## Justin
presidentbeef

I work on Brakeman and do security stuff.

**Follow**

Block or report user

Overview    Repositories **55**    Stars **122**    Followers **173**    Following **13**

## Pinned repositories

### brakeman
A static analysis security vulnerability scanner for Ruby on Rails applications

● Ruby   ★ 4.6k   ⑂ 457

### inject-some-sql
Have fun injecting SQL into a Ruby on Rails application!

● Ruby   ★ 154   ⑂ 36

### brat
Brat is a little language for people who don't like to be told what to do.

● C   ★ 50   ⑂ 5

### melon
"Message Exchange Language Over the Network" is a terrible acronym

● Ruby   ★ 1   ⑂ 1

### space_dodge_100
Model 100-style arcade game (GitHub GameOff 2017)

### github-auto-locker
Automatically lock old, closed GitHub issues

*"Patch-level verification for Bundler"*

# Reek

*"Reek is a code smell detector for Ruby."*

*"Reports the most tortured code in an easy to read pain report. The higher the score, the more pain the code is in."*

# By the way... (2)

Code Responsibly

Ruby Edition

# 99 Bottles of OOP

A Practical Guide to Object-Oriented Design

Sandi Metz & Katrina Owen

*"Metrics are fallible but human opinion is no more precise. Checking metrics regularly will **<u>keep you humble</u>** and **<u>improve your code</u>**."*

*Sandi Metz & Katrina Owen*
*99 bottles of OOP*

Baruco 2013: Rules, by Sandi Metz

# Just those?

- *Our text editor / IDE*
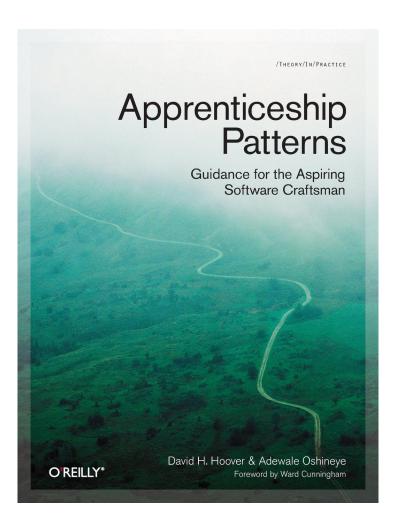- *Command line*
- *Bundler*
- *Rake*

- *Our testing framework*
- *Mutant*
- *Approvals*
- *...*

- *Travis*
- *Code Climate*
- *Danger*
- *Guard*

- *Flay*
- *Rubrowser*
- *Bundler-audit*
- *Peek*

# 3. Outro

# Using the <u>source</u>

# Apprenticeship Patterns

## Guidance for the Aspiring Software Craftsman

David H. Hoover & Adewale Oshineye

Foreword by Ward Cunningham

*"The best way to prepare [to be a programmer] is to write programs, and to __study__ ~~great~~ __programs__ that other people have written.*

*I went to the garbage cans at the CS center and fished out listings of their operating system."*

Bill Gates
*Programmers at work*

RailsConf 2017: Perusing the Rails Source Code - A Beginners Guide by Alex Kitchens

4,106 views

👍 92    👎 2    ➔ SHARE    ≡+    •••

# And <u>now</u>?

*"Start with a **<u>basic set</u>** of generally applicable tools. As you gain experience, and as you come across special requirements, you'll add to this basic set."*

*David Thomas & Andrew Hunt*
*The Pragmatic Programmer*

- Open spaces
- Deliberate practice
- Code katas / dojos
- Code retreats
- Journey-crafter tours

- *XP  explained*
- *The Pragmatic Programmer*
- *99 bottles of OOP*
- *Apprenticeship Patterns*

- RuboCop
- Brakeman
- Reek
- Flog
- SandiMeter
- bundler-audit

- Enforcing standards
- Code smells
- Software metrics
- Using the source
- ..

**@dcarral**

dcarral.org