

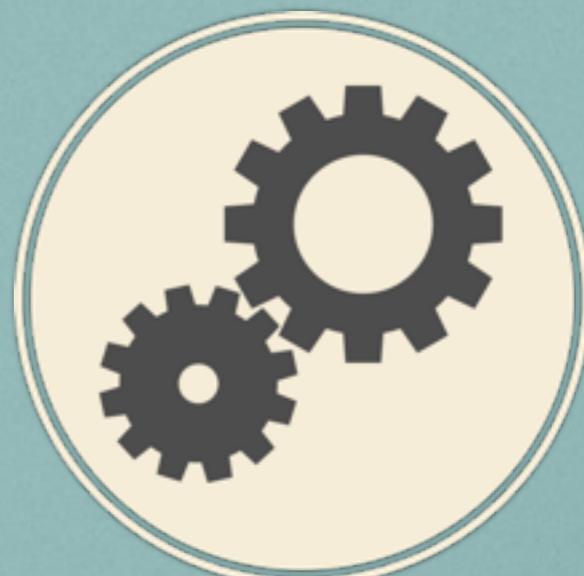
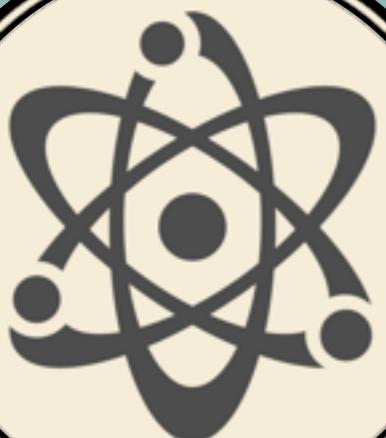
DAY 1

## COURSE OVERVIEW

## SOFTWARE LIFE CYCLE

## EXPLORE THE COMMAND LINE

## SOFTWARE REQUIREMENTS



# COURSE SYLLABUS

WHO IS ELIZA?

My experience and background.

COURSE GOALS

From starting point to end goal.

SYLLABUS

An overview of the course

WHY RUBY?

Diving into the approach that we'll be taking.

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?



# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?



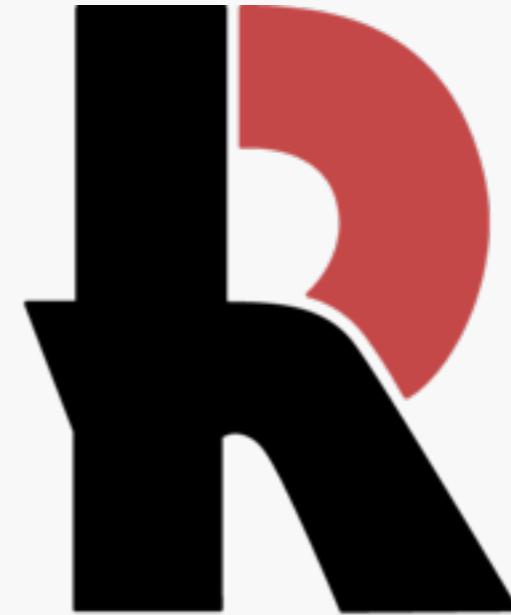
# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?



**ROSE-HULMAN**

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?



# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?



**Eliza Brock**  
SOFTWARE

# COURSE SYLLABUS

WHO IS ELIZA?

(START)

COURSE GOALS

15 students with a wide **variety** of  
**experiences** and **education** levels.

SYLLABUS

(FINISH)

WHY RUBY?

Hirable **Junior Developers** that can  
maintain and develop Ruby/Rails  
applications.

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?

← ( START ) →

15 students with a wide **variety** of  
**experiences** and **education** levels.

At least some experience with:

- HTML/CSS
- Javascript
- Client work

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?

← ( START ) →

And, we have a wide variety of:

- Ages
- Education
- Experience
- Professional Backgrounds

This brings us some unique (good!) challenges.

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?

← (FINISH) →

Hirable **Junior Developers** that can maintain and develop Ruby/Rails applications.

You will learn to:

- build on existing Rails applications
- work on cross-disciplinary teams
- grow your skills independently
- understand the depth and breadth of the field

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?

SYLLABUS

4 UNITS  
over  
12 WEEKS

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?

SYLLABUS

UNIT 1: TDD of Ruby data structures

UNIT 2: Explore the Ruby Ecosystem

UNIT 3: Simple Rails Applications

UNIT 4: Advanced Topics

...

# COURSE SYLLABUS

WHO IS ELIZA?

COURSE GOALS

SYLLABUS

WHY RUBY?

← (RUBY?) →

This is a Rails course, right?

So, why are we doing Ruby first-and for so very long?

Fundamentals are damn important.



# INTRODUCTIONS

- Name
- Profession/Background
- What you hope to get from this course



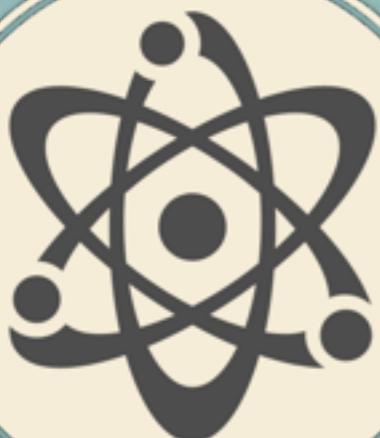
DAY 1

## COURSE OVERVIEW

## SOFTWARE LIFE CYCLE

## EXPLORE THE COMMAND LINE

## SOFTWARE REQUIREMENTS



# SOFTWARE LIFECYCLES

WATERFALL

Much maligned, but much used

AGILE

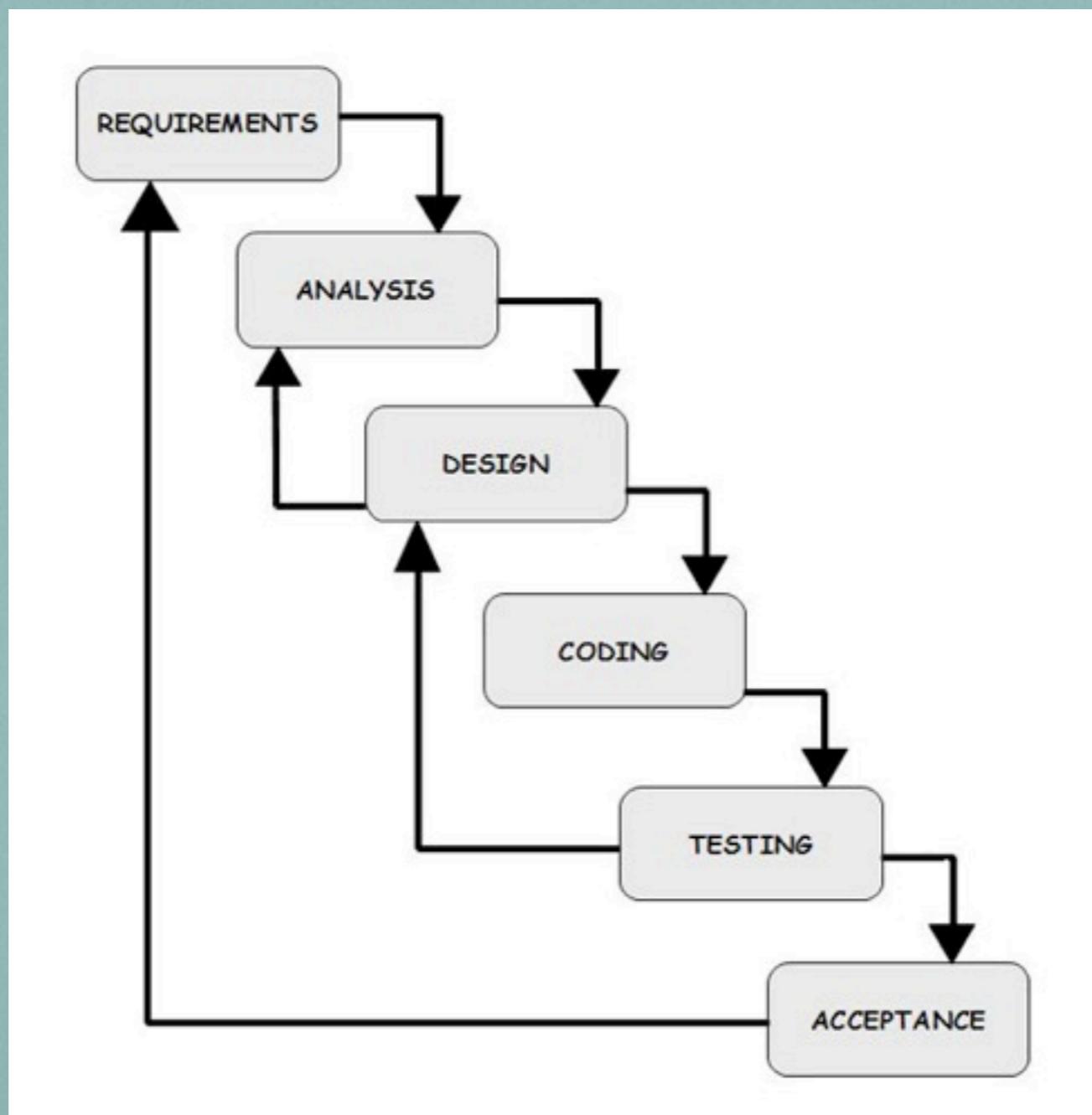
The new hotness, since 2001

WHY MANY?

There's actually a reason.  
Who knew?

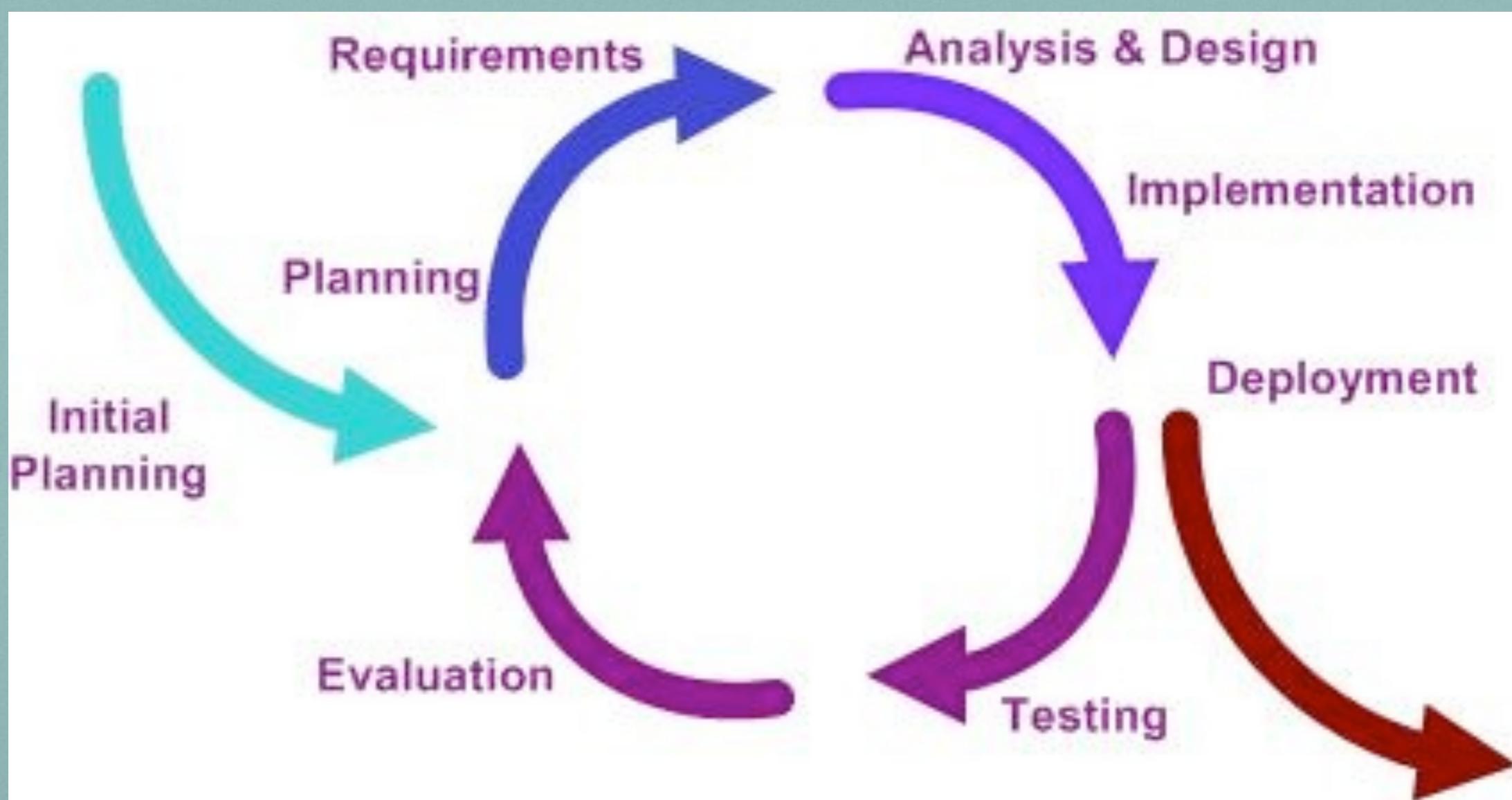
# SOFTWARE LIFECYCLES

## 1970 - WATERFALL



# SOFTWARE LIFECYCLES

## WELCOME TO 1957

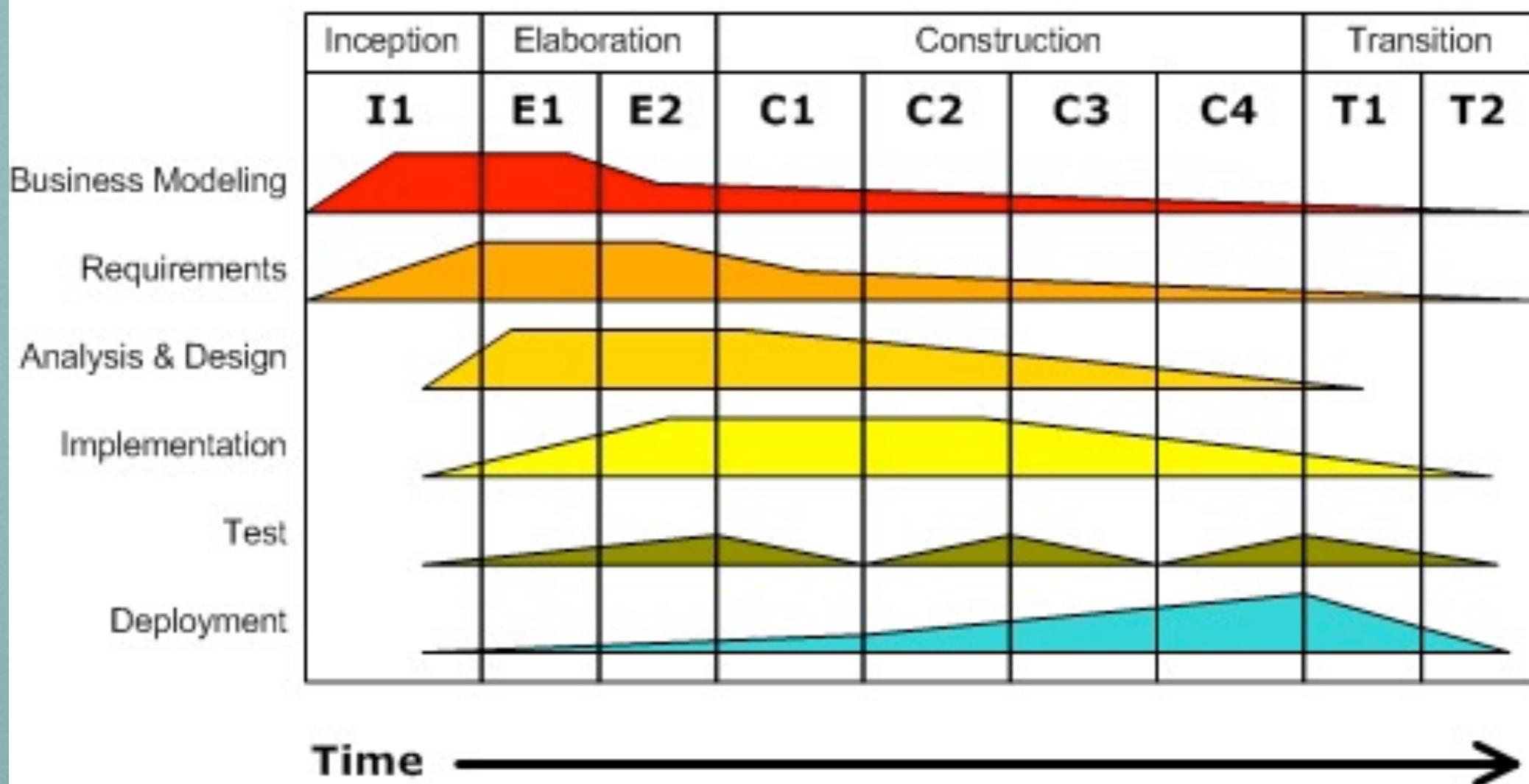


# SOFTWARE LIFECYCLES

## ITERATIVE DEV.

### Iterative Development

Business value is delivered incrementally in time-boxed cross-discipline iterations.



# SOFTWARE LIFECYCLES

## AGILE MANIFESTO

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

**Individuals and interactions** over processes and tools

**Working software** over comprehensive documentation

**Customer collaboration** over contract negotiation

**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

# SOFTWARE LIFECYCLES

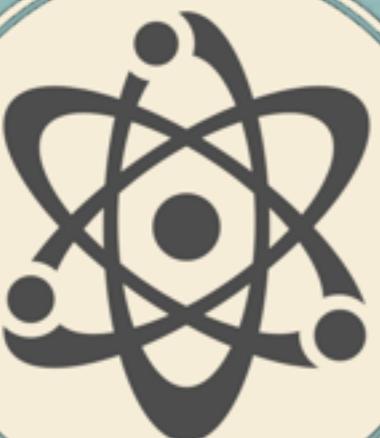
## WHY BOTH?

<b>Agile</b>	<b>Plan-driven</b>	<b>Formal Methods</b>
Low Criticality	High Criticality	Extreme Crit.
Sr. Devs	Jr. Devs	Sr. Devs
Requirements Churn	Static Requirements	Limited Req/Features
Small Dev Team	Large Dev Team	Model-able Req.
Culture that responds to change	Culture that demands order	Extreme quality

**Suitability of different development methods**  
from the Wikipedia article on Agile

DAY 1

## COURSE OVERVIEW



## SOFTWARE LIFE CYCLE



## EXPLORE THE COMMAND LINE



## SOFTWARE REQUIREMENTS



EXPLORE CLI

TIME TO EXPLORE!



# EXPLORE CLI ASSIGNMENT

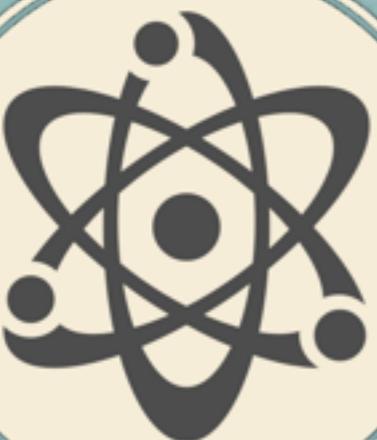
Let's talk about TDD for a brief moment.

Then, start going through the in-browser Ruby Koans:

<http://koans.herokuapp.com/>

DAY 1

## COURSE OVERVIEW



## SOFTWARE LIFE CYCLE



## EXPLORE THE COMMAND LINE



## SOFTWARE REQUIREMENTS



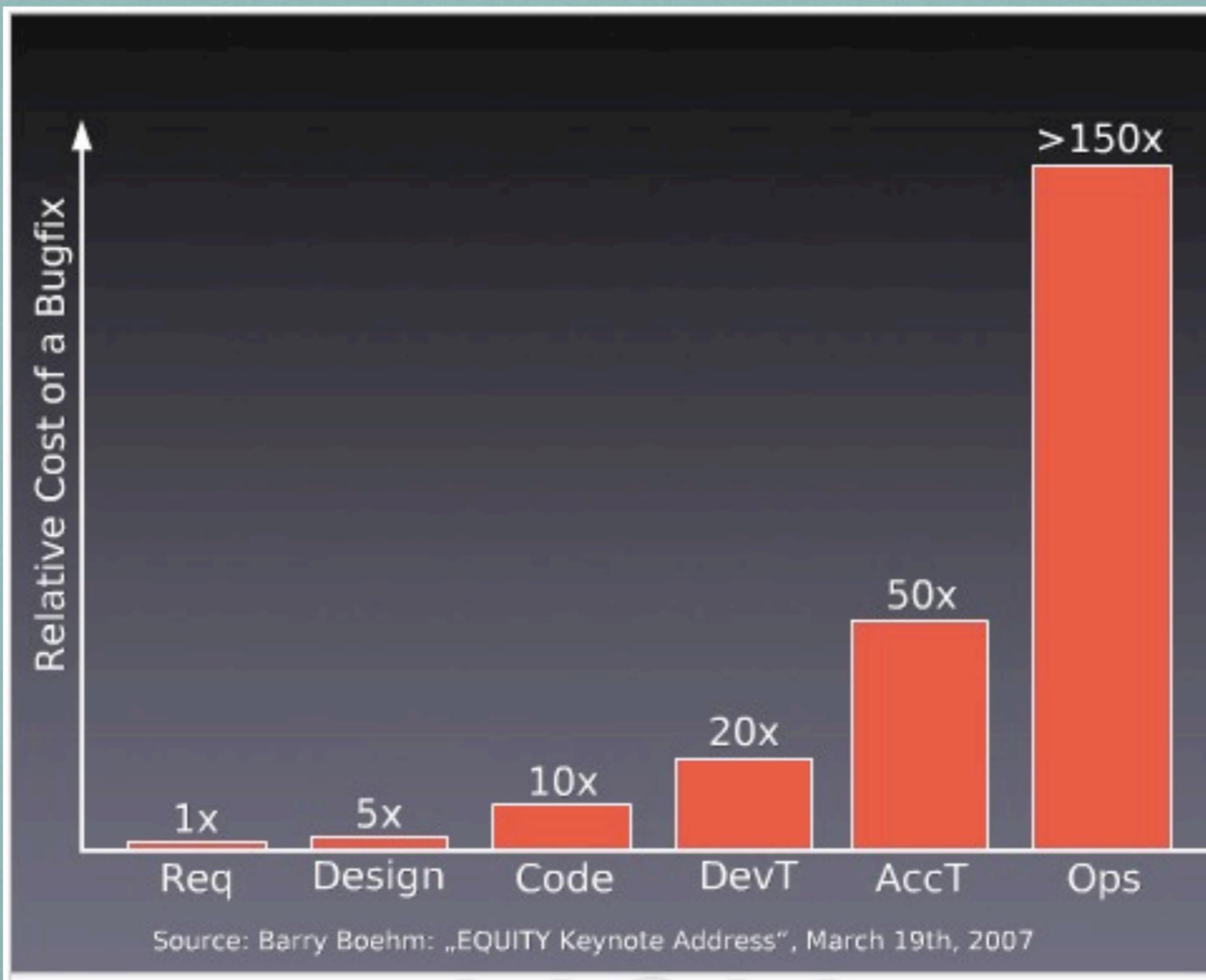
# SOFTWARE REQUIREMENTS

## EXERCISE 1

Pink Elephants

# SOFTWARE REQUIREMENTS

## WHY IT MATTERS



# SOFTWARE REQUIREMENTS

## EXERCISE 2

What now, **calendars?**

# SOFTWARE REQUIREMENTS

## ASSIGNMENT

Write down, in painstaking detail, exactly **how to change a tire.**

Assume an absolute asshole\* is going to try to fuck it up, while still following your instructions to the letter.

\* This is a good pet name for your compiler/interpreter