

Choosing the right web technology

Michael Choi, CodingDojo

Christopher Harrison, Microsoft





Michael Choi

- Chief Instructor and CTO - CodingDojo
- Degree in Engineering from CalTech and Brigham Young Univ.
- Entrepreneur - Zurple, Village88, CodingDojo, SpotTrender.com
- Husband and father of three (soon to be four)

@twitter_handle



Christopher Harrison

- Senior Content Producer - Microsoft Virtual Academy
- Web guy
- <3 OSS
- Long-time Microsoft Certified Trainer
- Periodic blogger (blog.geektrainer.com)
- Marathoner, husband, father of one four-legged child

@geektrainer

Agenda

1	Choosing the right web technology	5	Where do I get started?
2	Understanding the big picture	6	How do I find engineers for my project?
3	Pros and cons of different platforms	7	Additional resources
4	Scaling basics		



What technologies are available?

Michael Choi
Christopher Harrison



Languages

C#

Ruby

Java

JavaScript

Python

PHP

CSS

LESS

SASS

HTML

Backend frameworks and libraries

Ruby on Rails

PHP

Node.js

Django

Flask

ASP.NET Web Forms

ASP.NET MVC

Frontend frameworks and libraries

Bootstrap

jQuery

Knockout

Angular

Ember

Backbone

React

Data stores and technologies

MongoDB

SQL Server

MySQL

NoSQL

Oracle

Postgres

Hadoop

Communication standards

REST

HTTP APIs

Web sockets

Where do we get started?

How does all of this fit into the big picture?

What technology should I choose for my startup?

Pros and cons of various frameworks

What happens if we get big?

Where do I start if I want to learn?

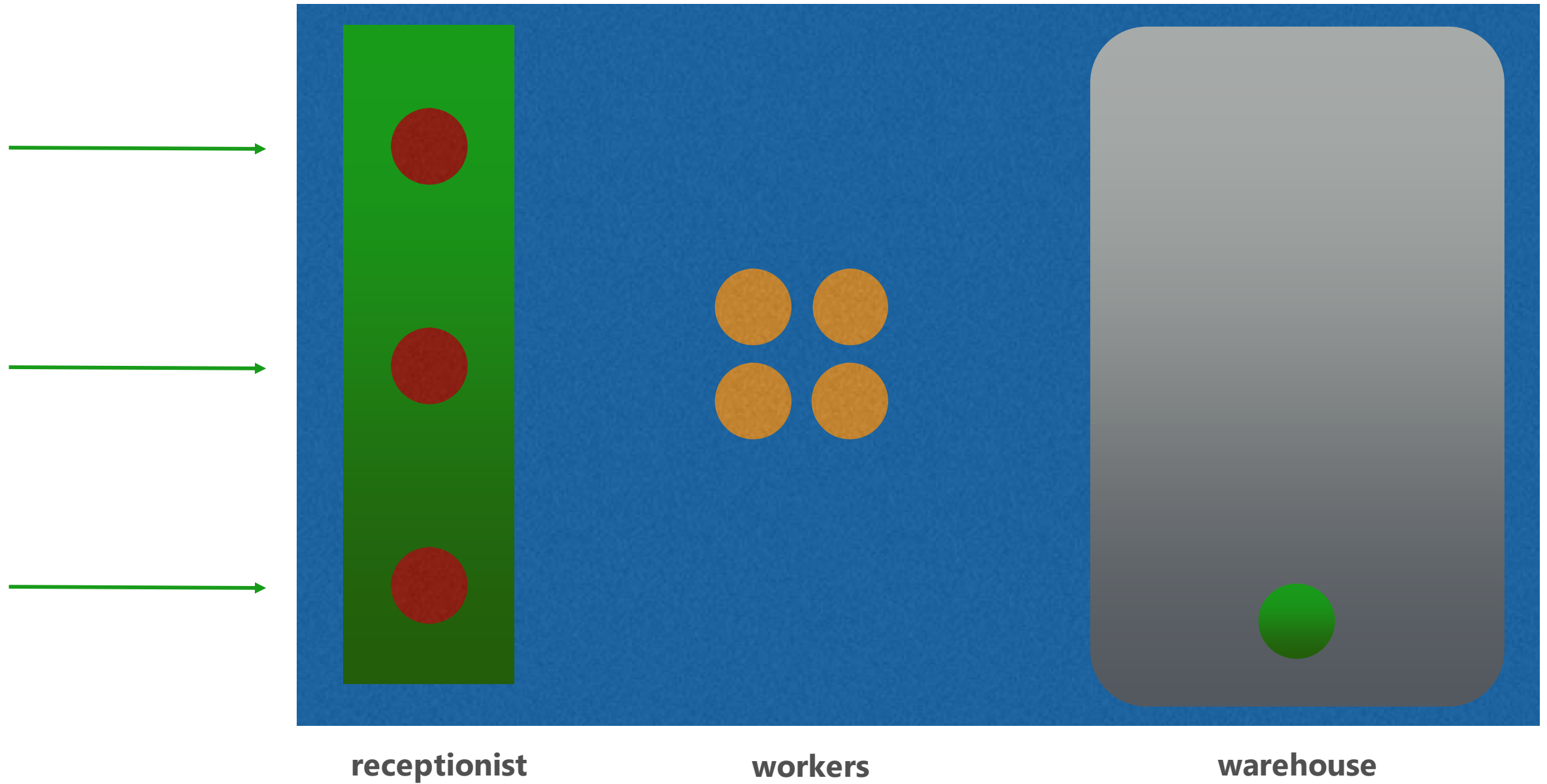
How do I find engineers who have these skillsets?

Understanding the big picture

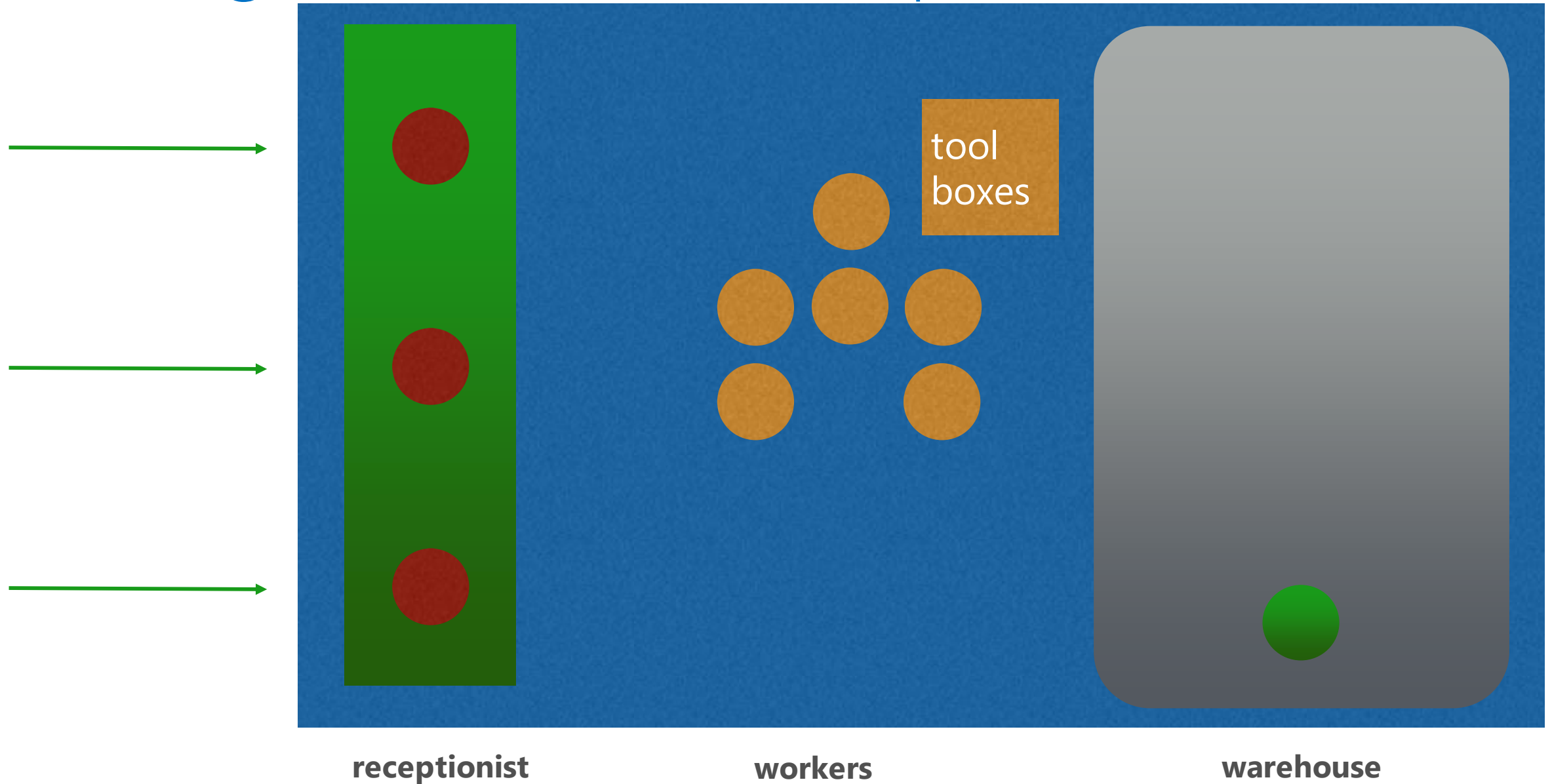
Michael Choi
Christopher Harrison



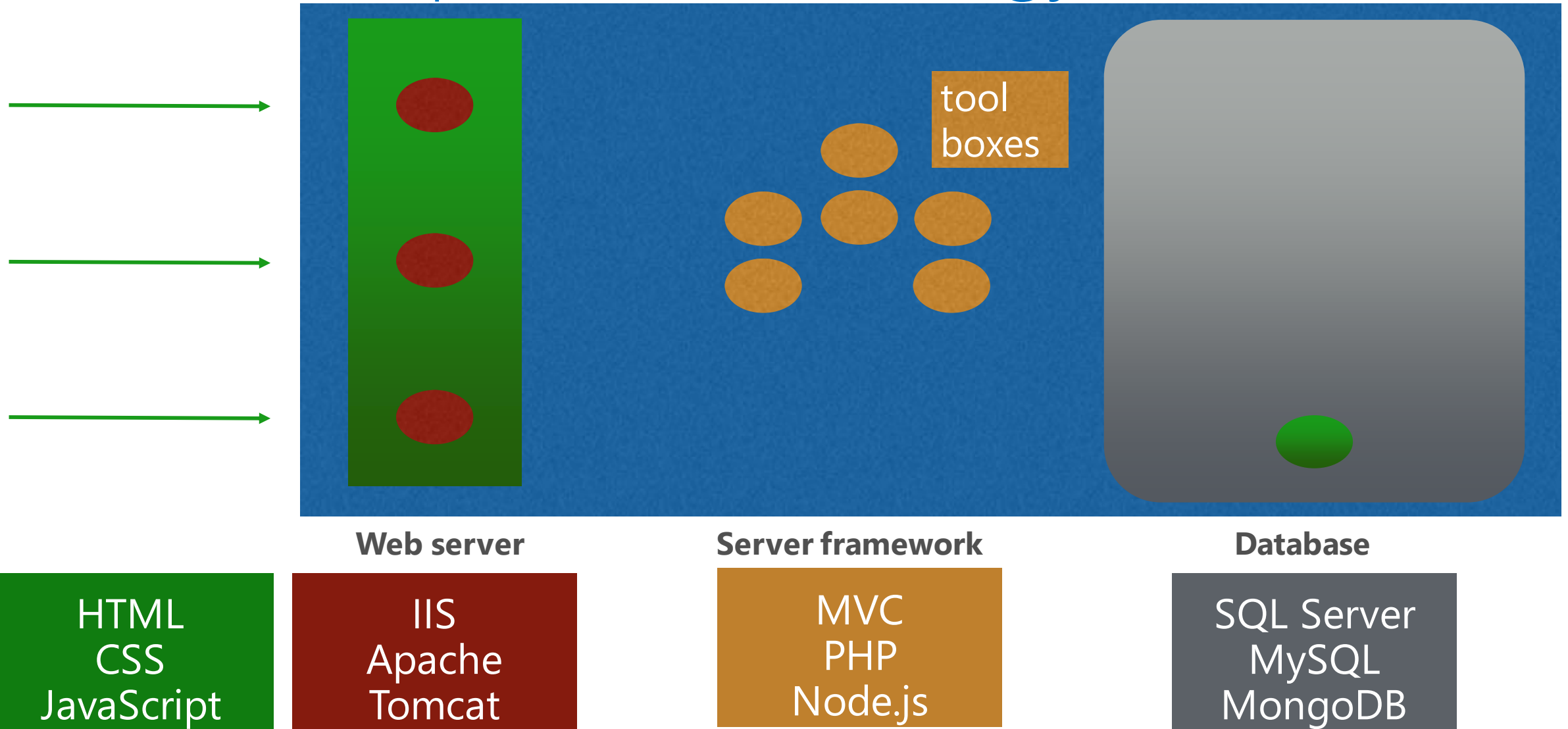
Furniture store



Making the store more responsive



Web development terminology



Comparing the server frameworks

Michael Choi
Christopher Harrison



The comparison

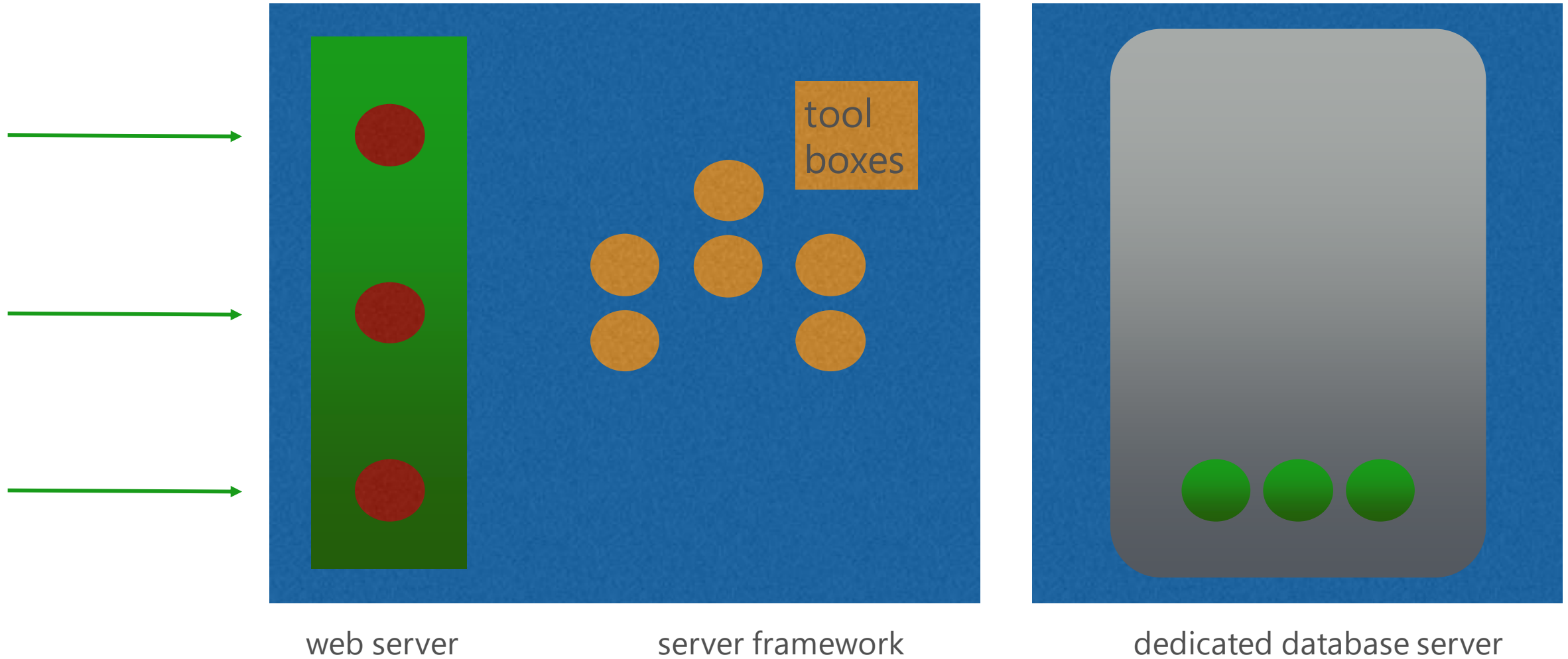
	Pros	Cons
PHP	Designed for beginners. Easy to build functional web apps.	Some PHP apps could lack structure.
Ruby	OOP and shorter codes (with Rails). Great TDD.	OOP and shorter codes. Lots of things happening in the background.
Python & Django	Easy to learn Python. Great community. Lots of code "already written".	Need to learn the framework. Lots happening in the background.
ASP.NET MVC	Uses common design patterns. Flexible. Runs compiled code.	Compiled. Need to learn C# or VB.
Node.js	Uses sockets.	Not as mature

Scaling concepts

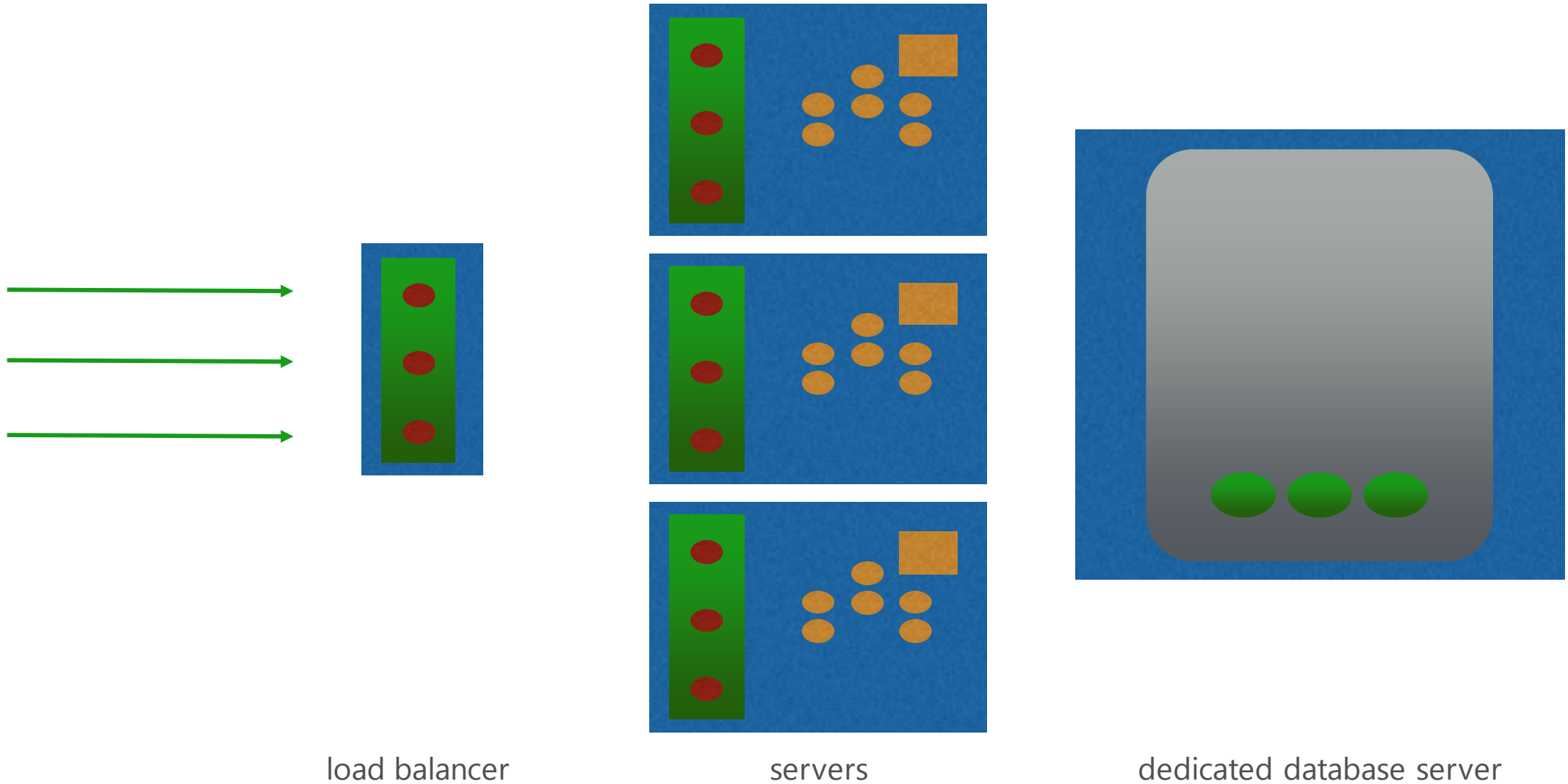
Michael Choi
Christopher Harrison



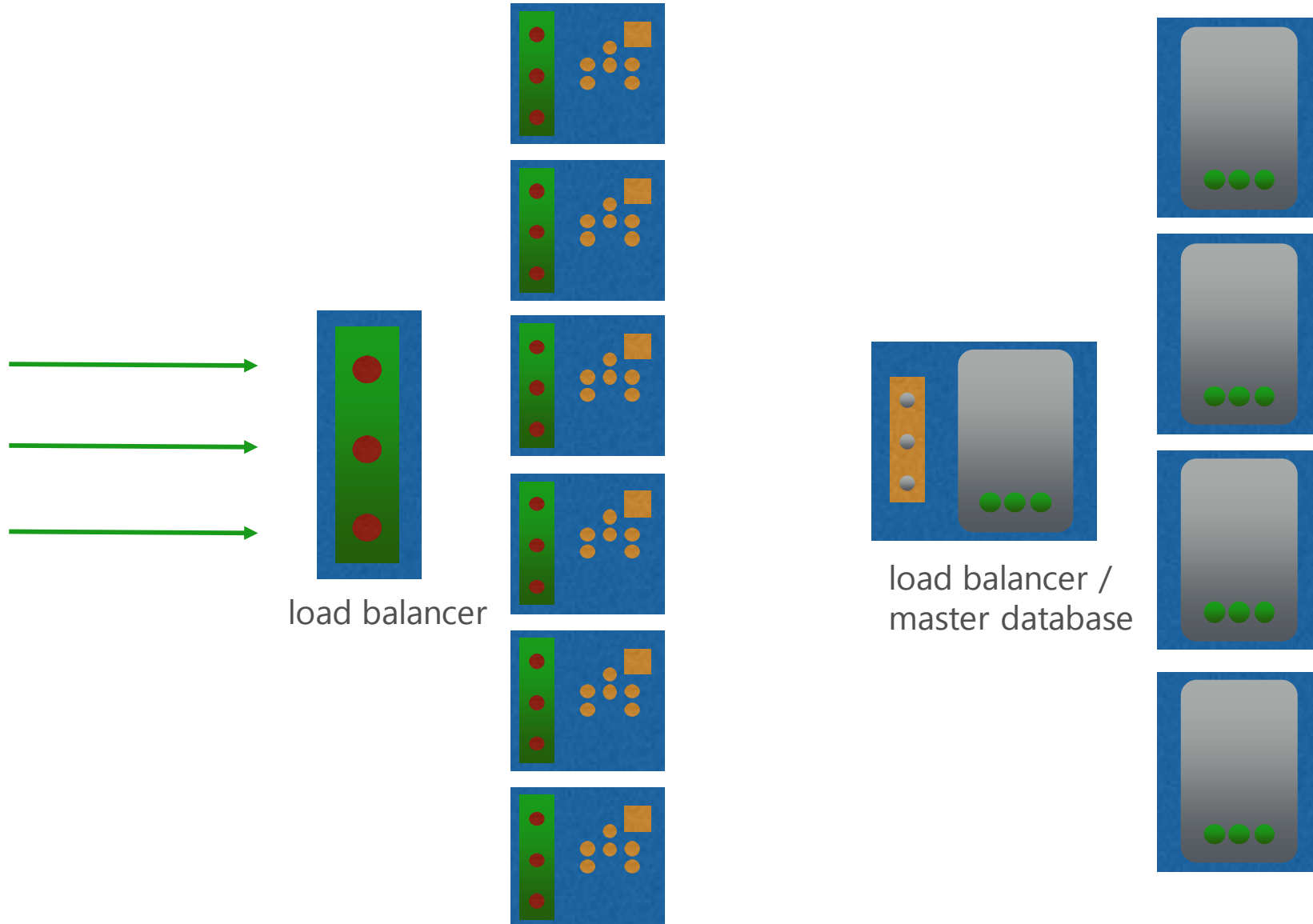
Separating the data store



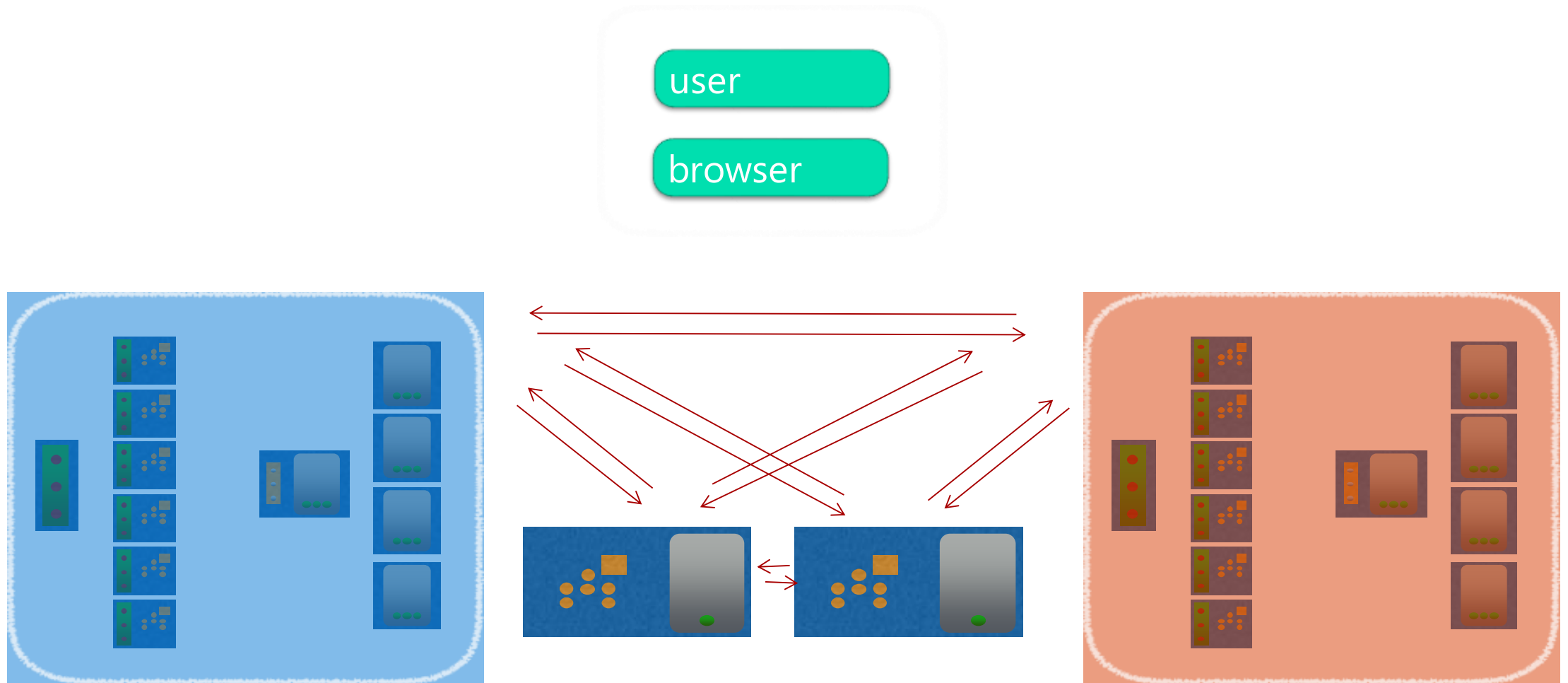
Scaling the server



Scaling the data store



Modularization and APIs



Beginning your learning journey

Michael Choi
Christopher Harrison



The path

Perfect for instructions with just a few steps



Step 1: Learn the front end



Step 2: Learn the database



Step 3: Learn the backend



Learning the front end

HTML/HTML5

CSS

For right now, focus on CSS

JavaScript

For right now, focus on JavaScript

jQuery

Resources

<http://mva.microsoft.com/>

<http://algorithm.codingdojo.com/>

<http://codepen.io/>

Learning the database

Learn SQL querying

Learn a database product

SQL Server

MySQL

Learn a backend framework

Python & Django

PHP

C# & MVC

Taking the next steps

Ajax & REST for client/server communication

JavaScript frameworks

Angular

Knockout

Gain exposure to other tools

Client side

LESS

SASS

CoffeeScript

HAML

Learn other backend frameworks

Finding engineers

Michael Choi
Christopher Harrison



How to find engineers

Friends

Networking events

User groups

Code camps

Conferences

Job ads

Finding the right engineers

Traditional CS degree

Coding schools

Self taught

Where do we go from here?

Michael Choi
Christopher Harrison



What to learn more?

CodingDojo

<http://codingdojo.com>

Algorithm app

Microsoft Virtual Academy

<http://mva.microsoft.com>

edX

<https://www.edx.org/school/microsoft>