KAZAKH KBTU BRITISH
TECHNICAL
UNIVERSITY

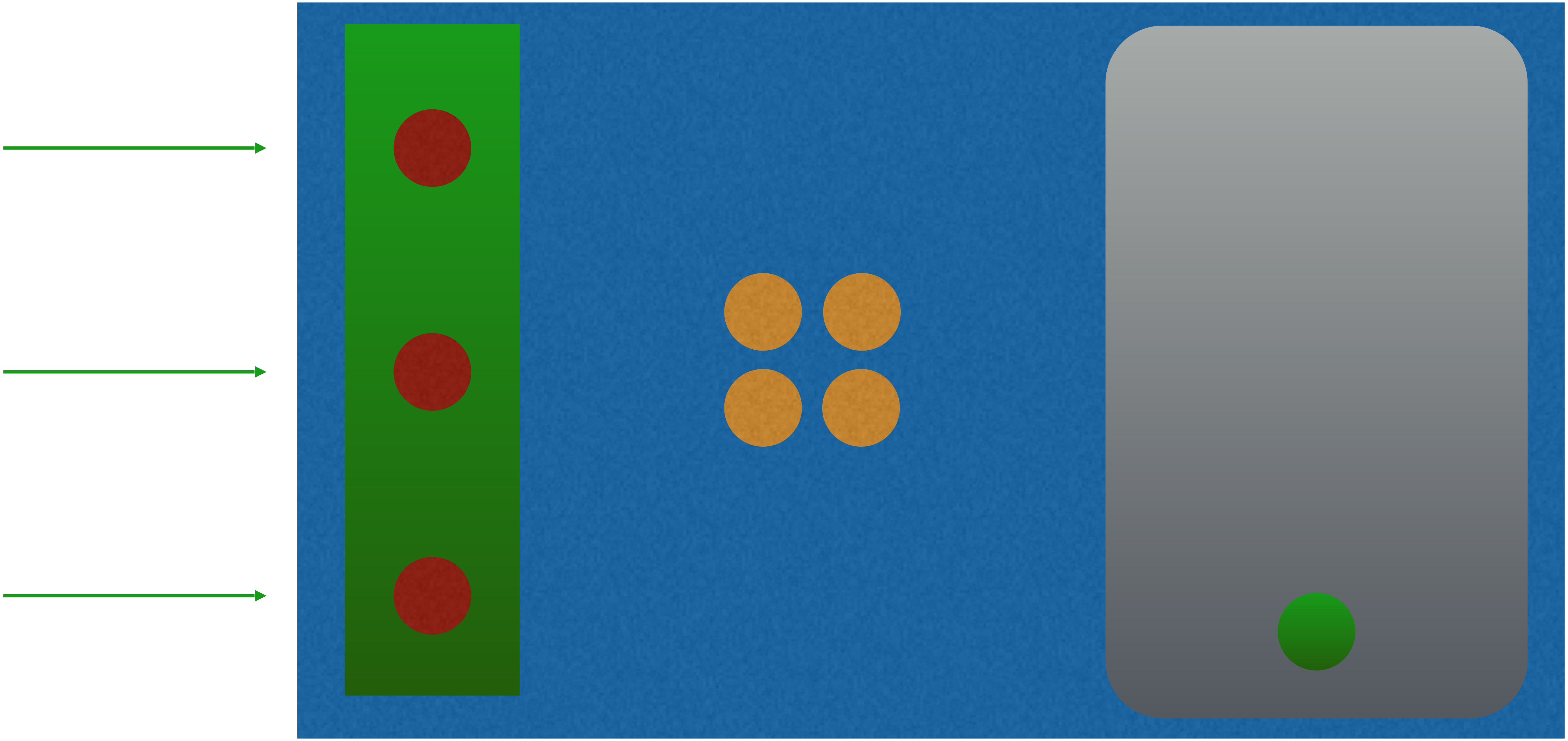
WEB DEVELOPMENT

Lesson 1

What is the web site?

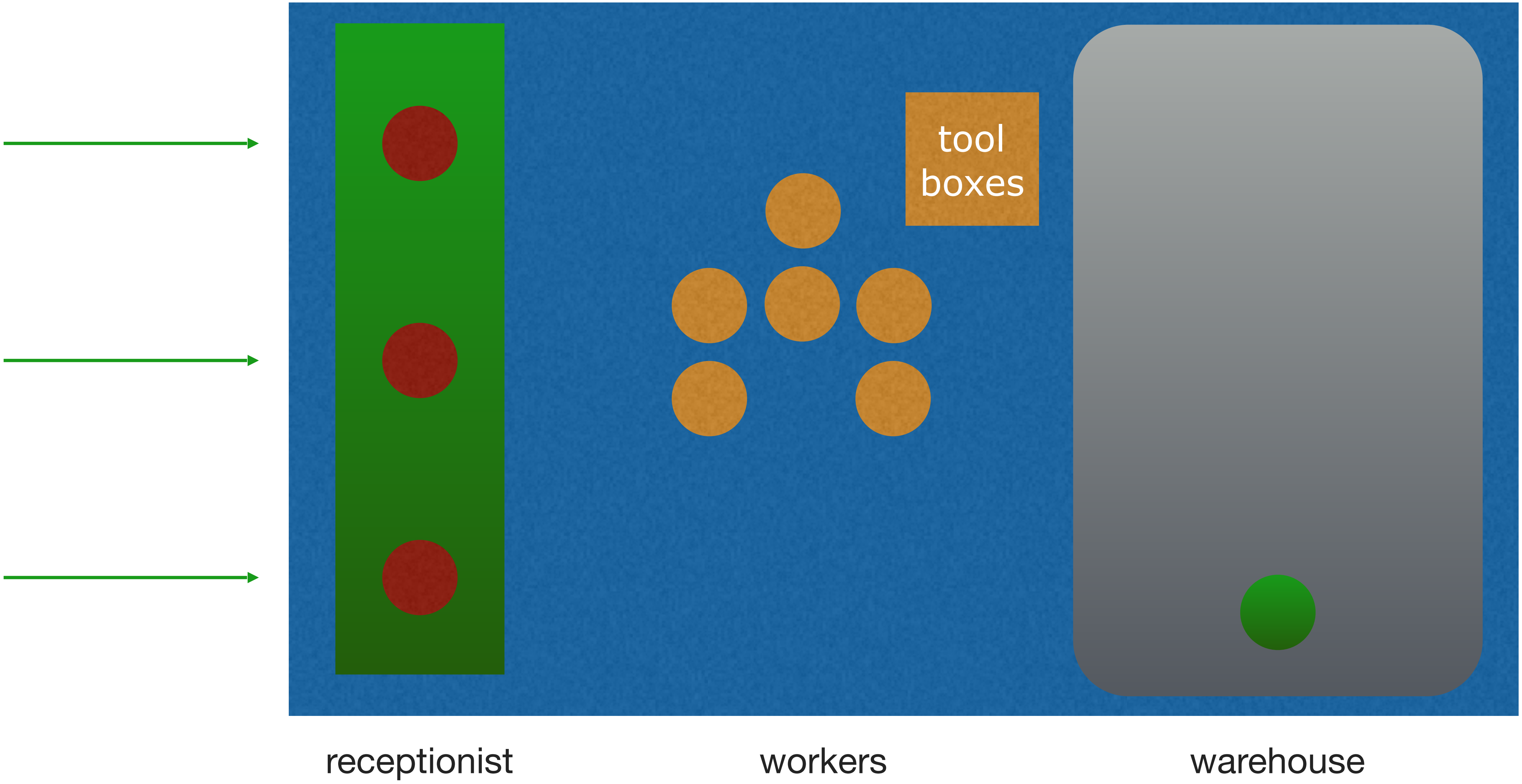
How the Web works?

Furniture store

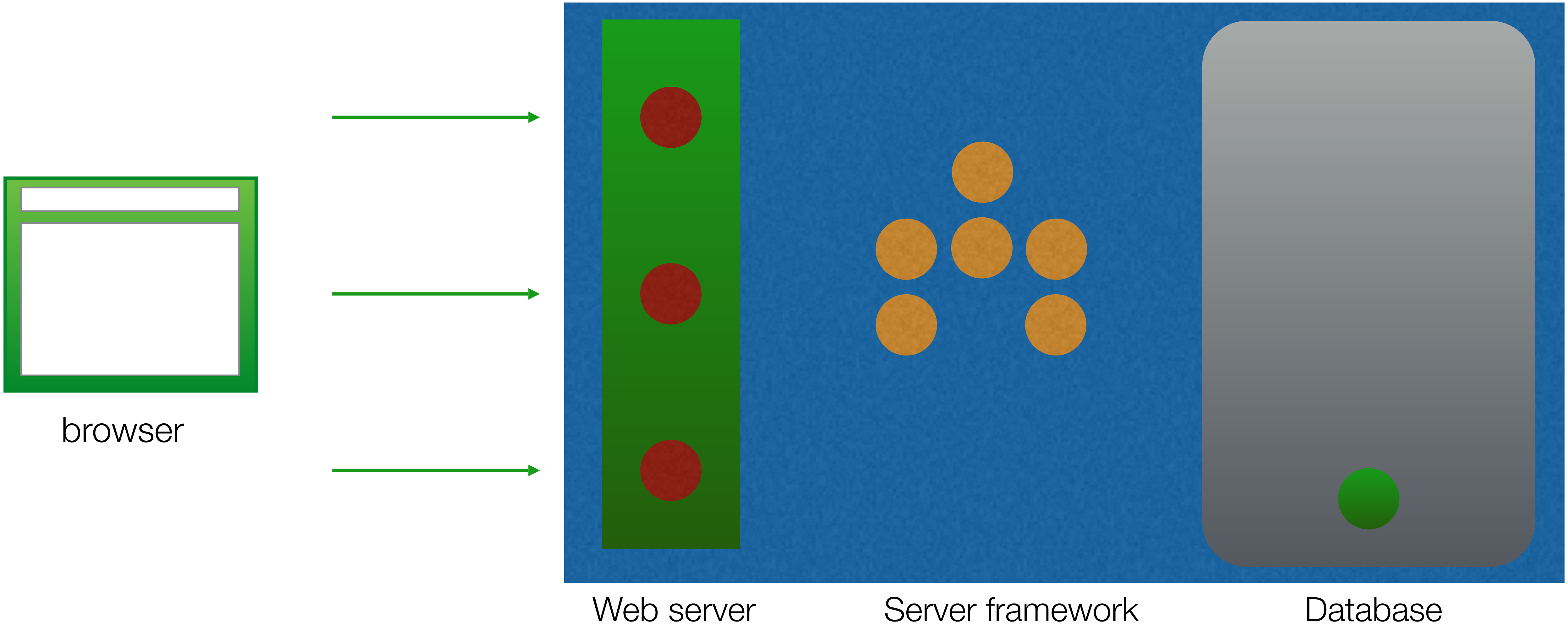


receptionist workers warehouse

Making the store more responsive



Web development terminology



browser

Web server

Server framework

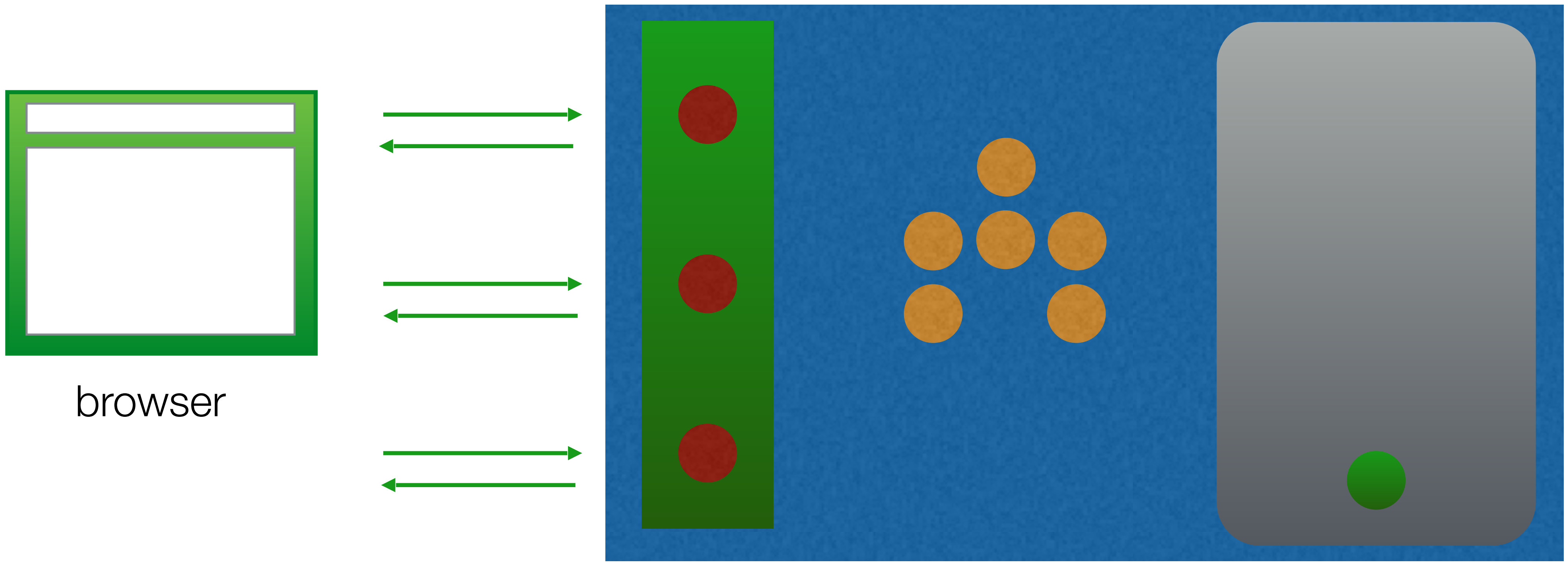
Database

HTML
CSS
JavaScript

NGINX
Apache
IIS

Django
RoR
Laravel
MVC

Postgres
MongoDB
Oracle



Hosting

AWS
Microsoft Azure
Heroku
ps.kz
hoster.kz

HTML
CSS
JavaScript

HAML
LESS SASS
CoffeScript
TypeScript

jQuery
React
Angular
Ember
Vue

Web server

NGINX
Apache
IIS
Unicorn
Thin
Puma

Server framework

Python
Django, Flask

Ruby
Rails, Sinatra

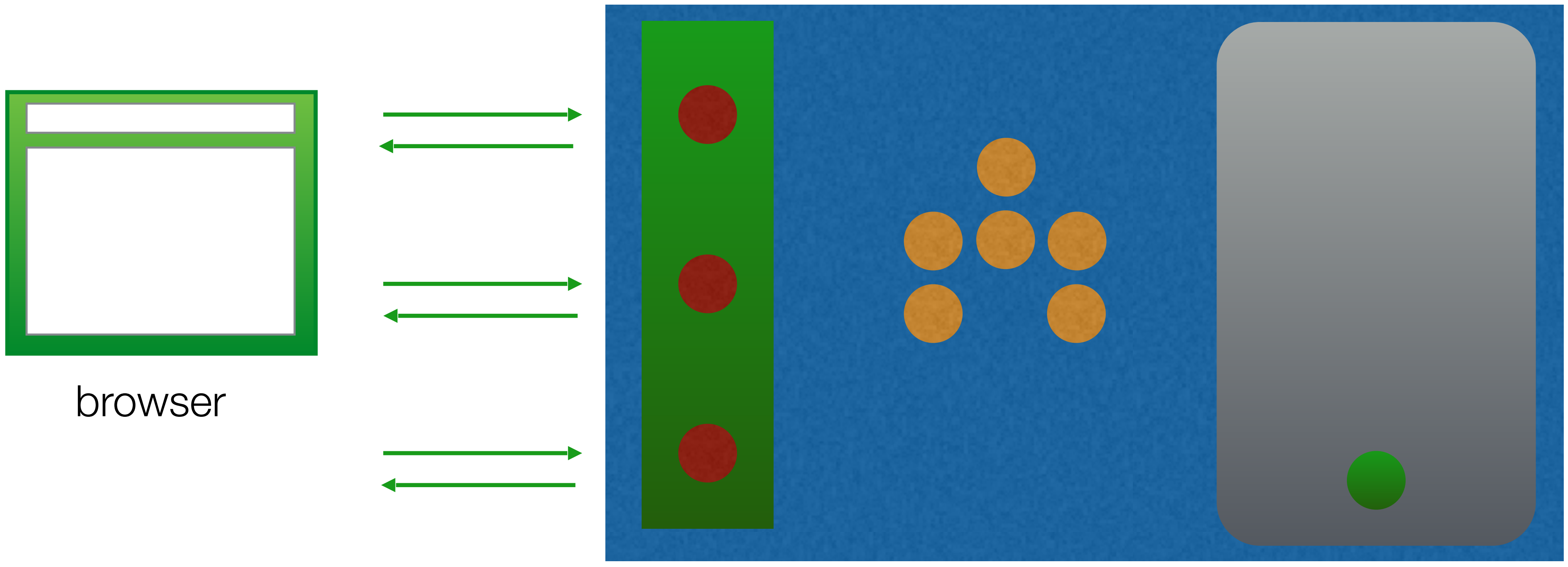
PHP
Laravel, Yii

Java
Spring, Play

C#
ASP.NET

Database

MySQL
Postgres
Microsoft SQL
SQLite
MongoDB
Oracle
Redis
Memcached
Cassandra



Hosting

AWS
Microsoft Azure
Heroku
ps.kz
hoster.kz

HTML
CSS
JavaScript

HAML
LESS SASS
CoffeScript
TypeScript

jQuery
React
Angular
Ember
Vue

Web server

NGINX
Apache
IIS
Unicorn
Thin
Puma

Server framework

Python
Django, Flask

Ruby
Rails, Sinatra

PHP
Laravel, Yii

Java
Spring, Play

C#
ASP.NET

Database

MySQL
Postgres
Microsoft SQL
SQLite
MongoDB
Oracle
Redis
Memcached
Cassandra

front end | back end | server

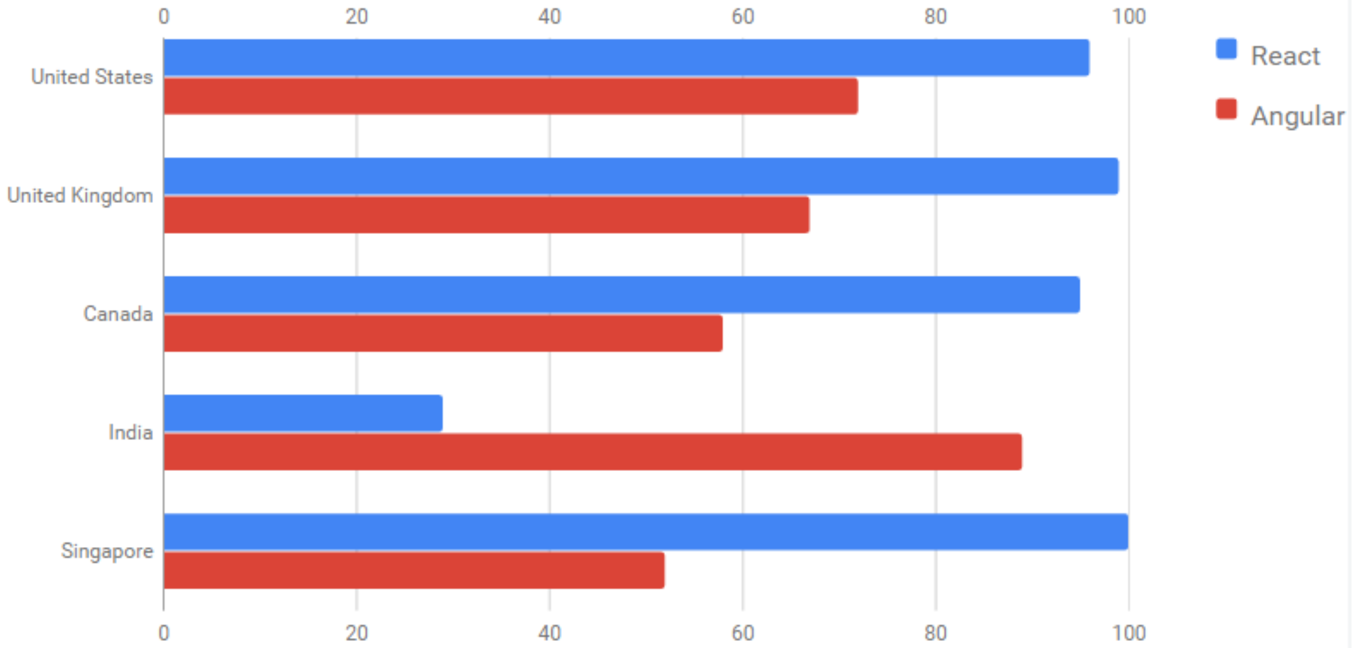
client side | server side

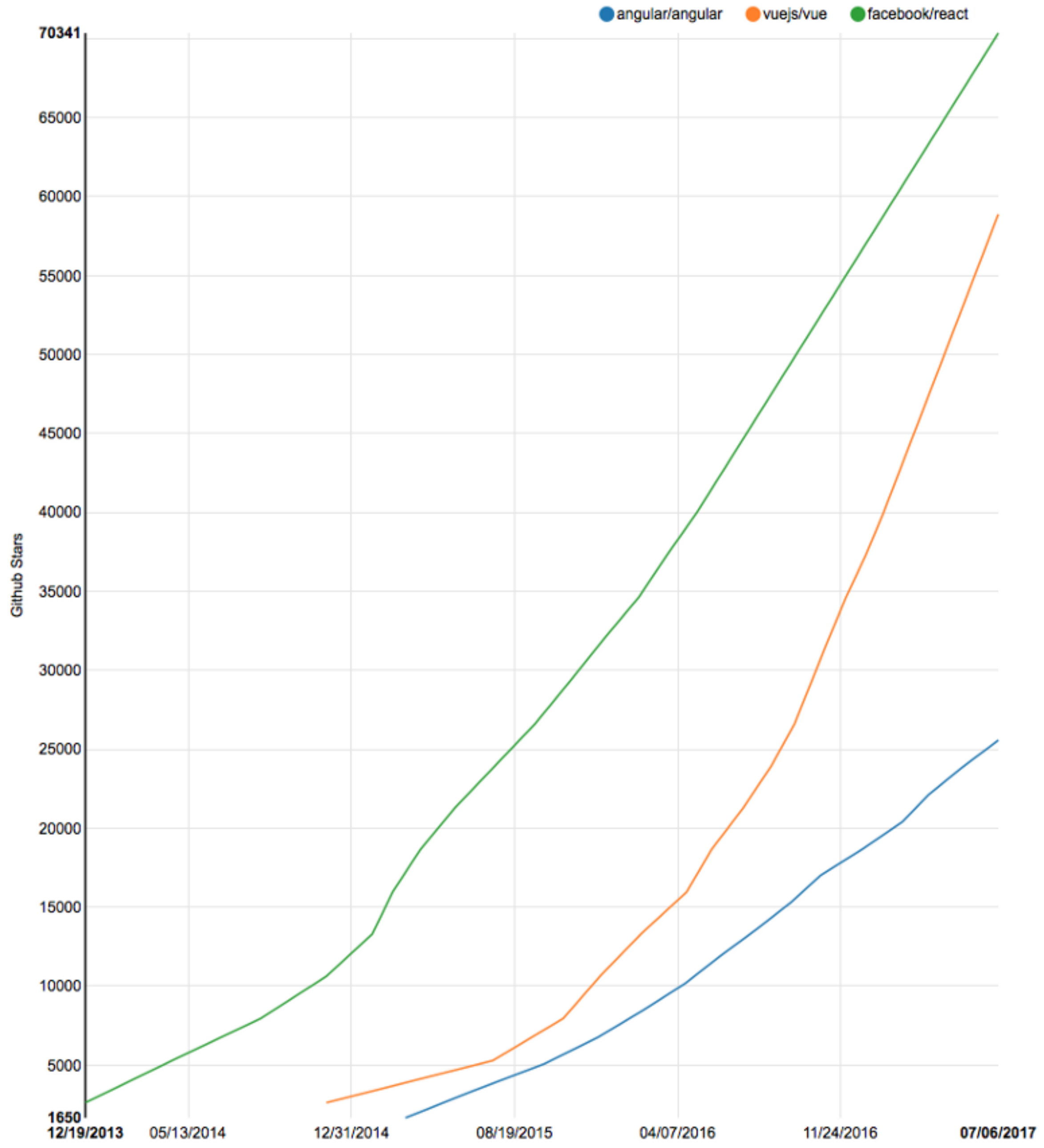
Technologies

Front end frameworks

1. React — Facebook
2. Angular — Google
3. Vue - Google employee

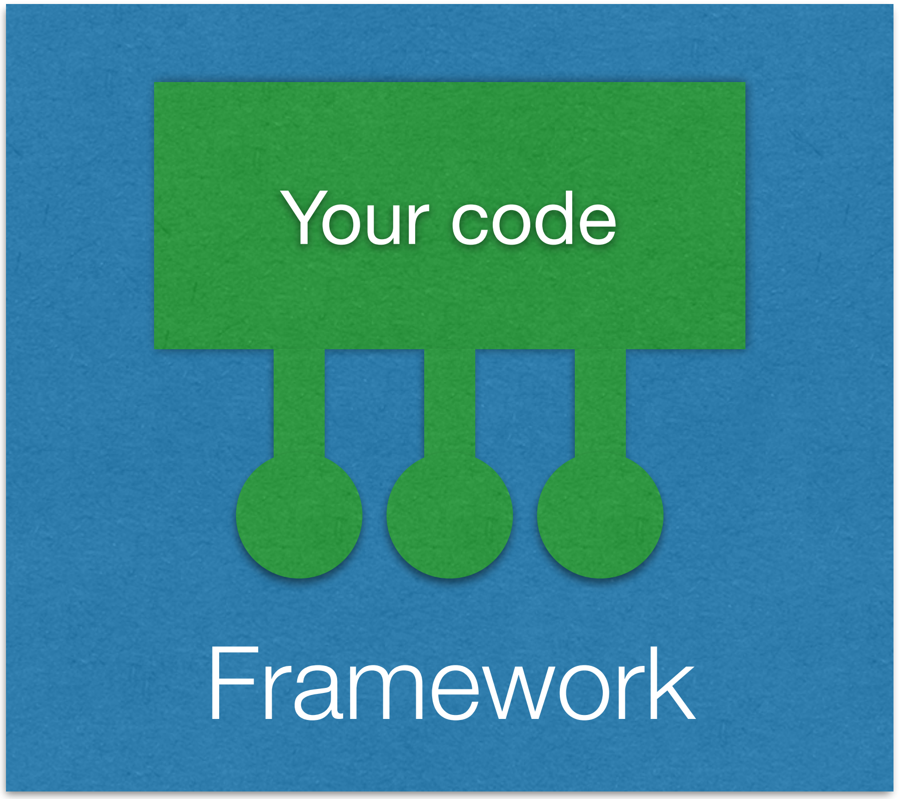
International interest:

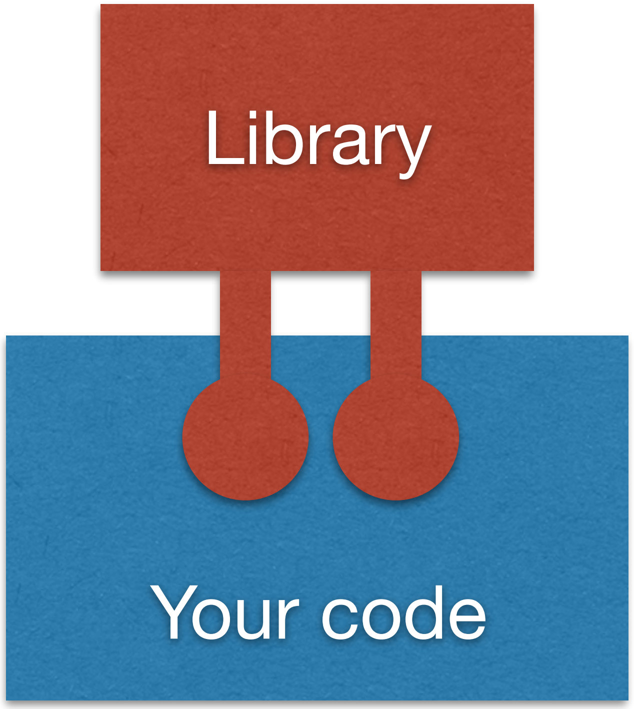




HARDWAR

Framework & Library





Back end frameworks

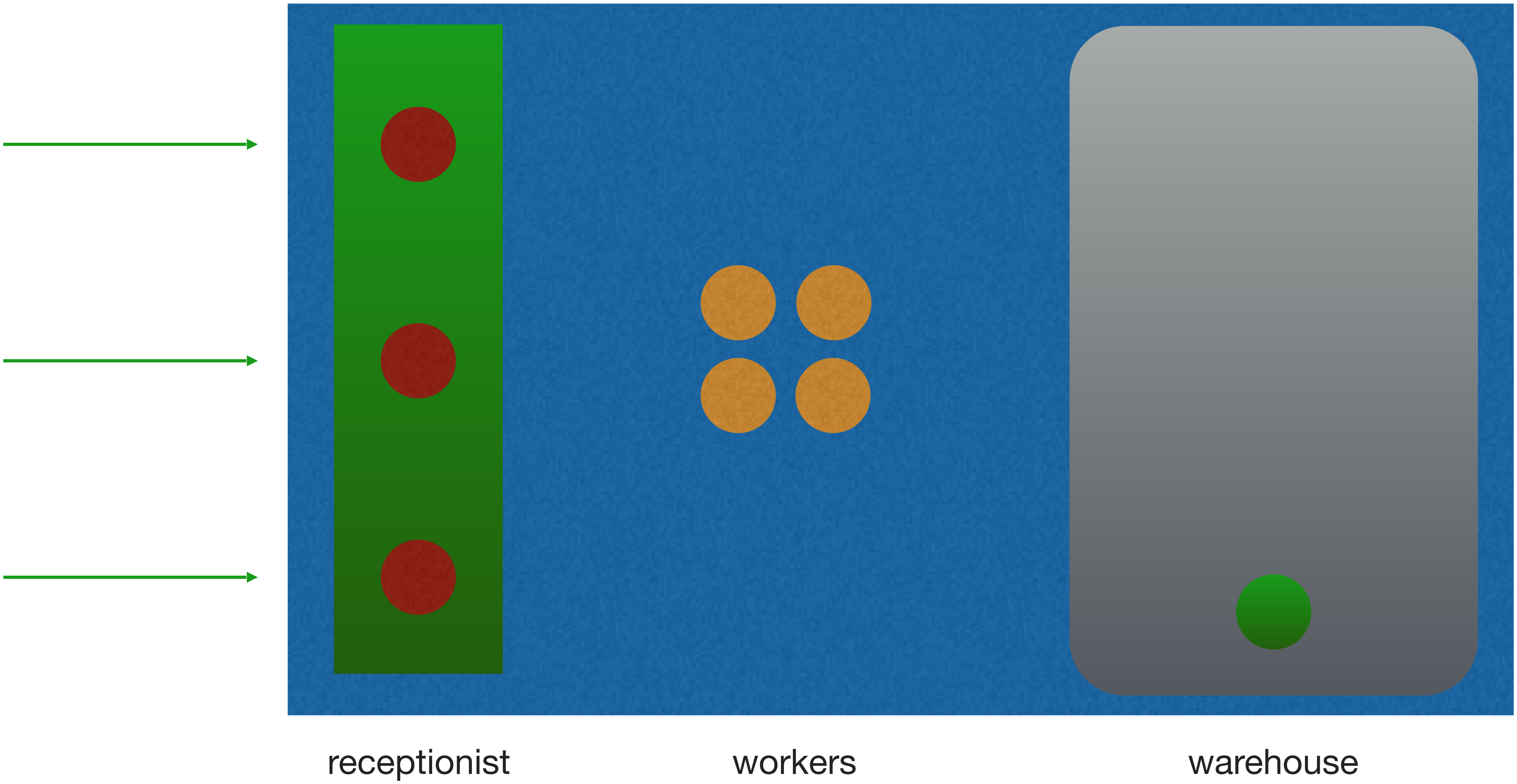
	Pros	Cons
PHP	Designed for beginners. Easy to build functional web apps.	Some PHP apps could lack structure.
Ruby	OOP and shorter codes (with Rails). Great TDD.	OOP and shorter codes. Lots of things happening in the background.
Python Django	Easy to learn Python. Great community. Lots of code "already written".	Need to learn the framework.
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

Back end frameworks

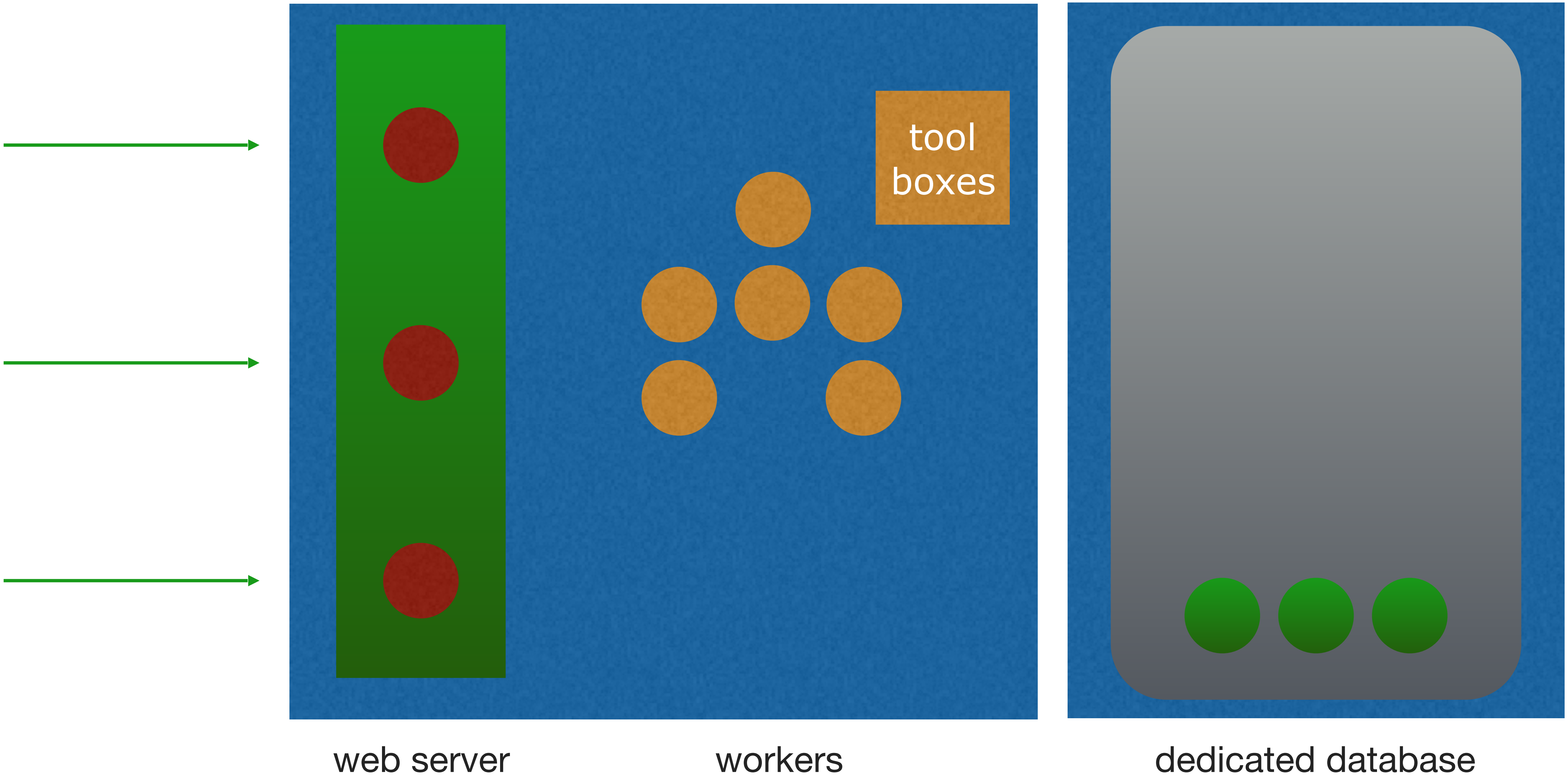
Django	**Rails**
Python	Ruby
MVT	MVC
Explicit is better than implicit	Convention over Configuration
beginners	seasoned professionals

Scaling concepts

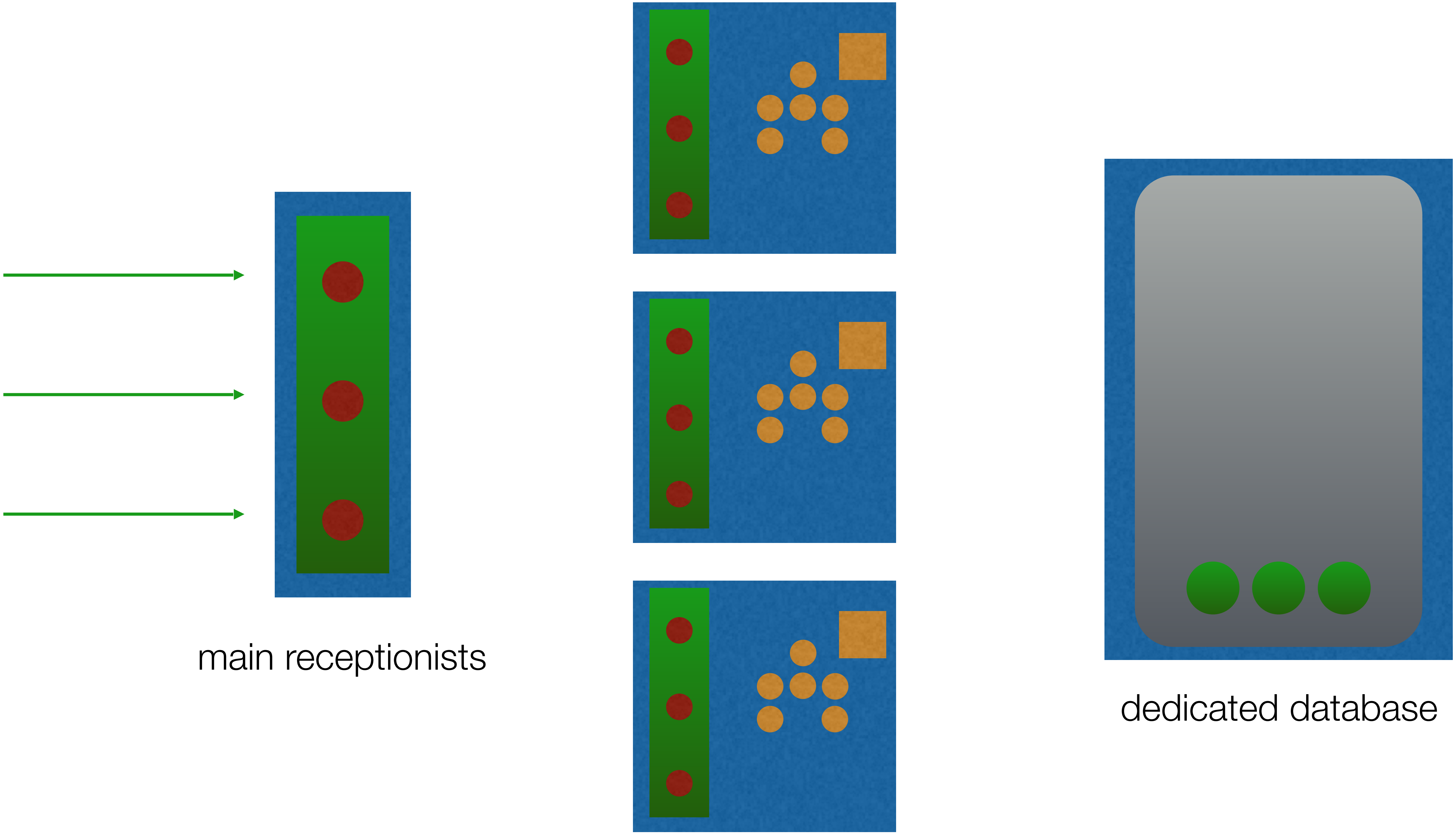
Furniture store



Separating the data store



Scaling the server

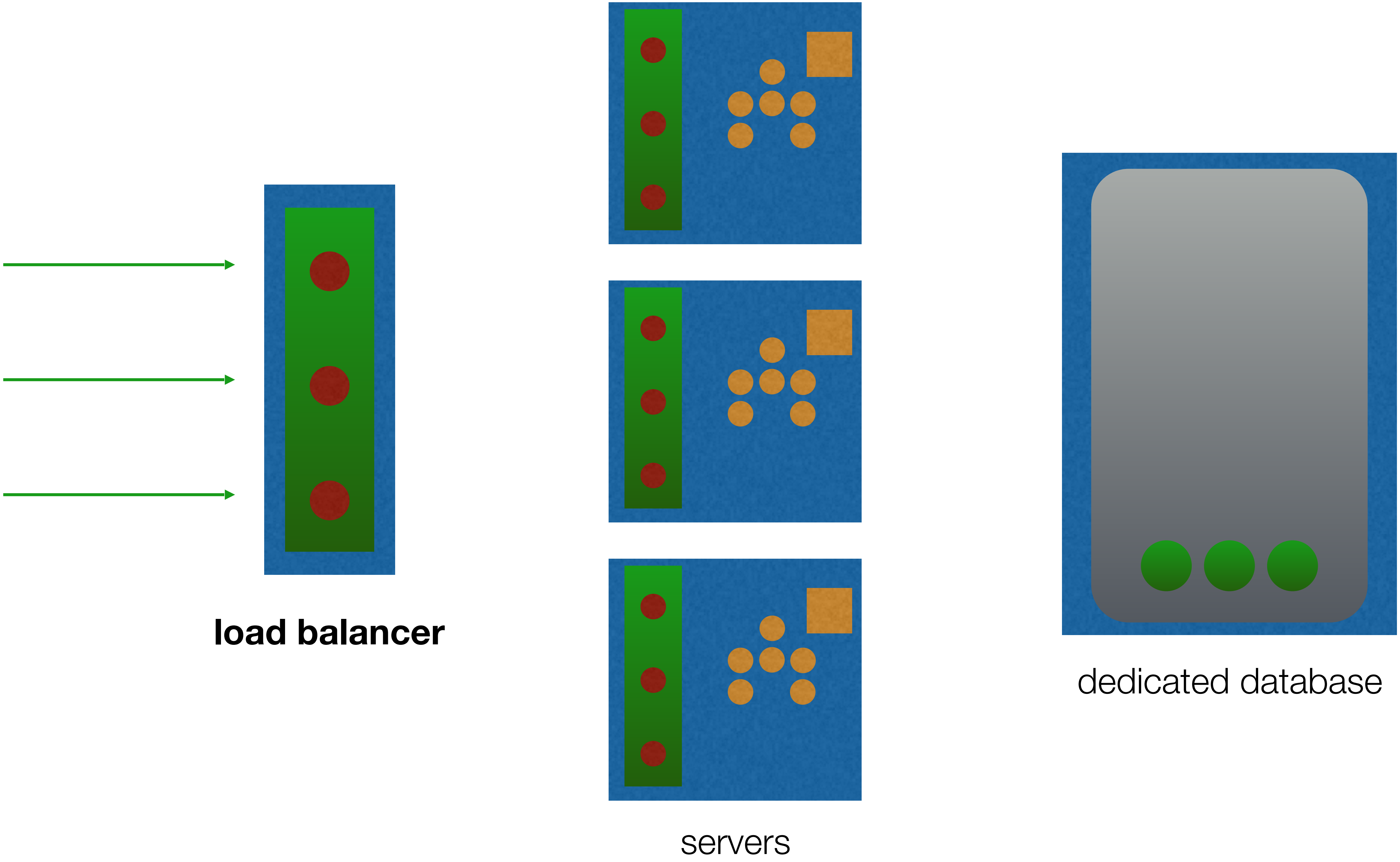


main receptionists

servers

dedicated database

Scaling the server

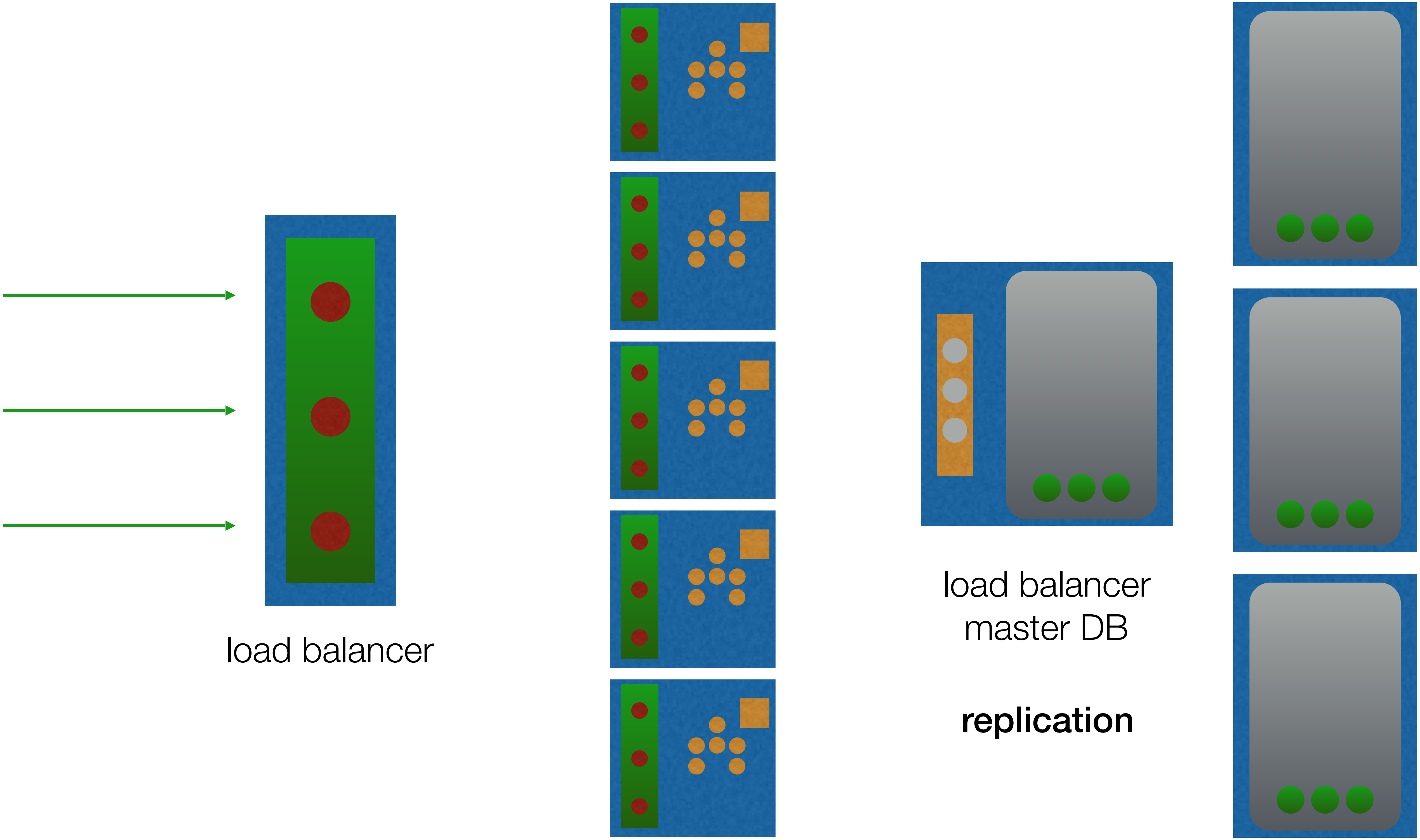


load balancer

servers

dedicated database

Scaling the data store



load balancer

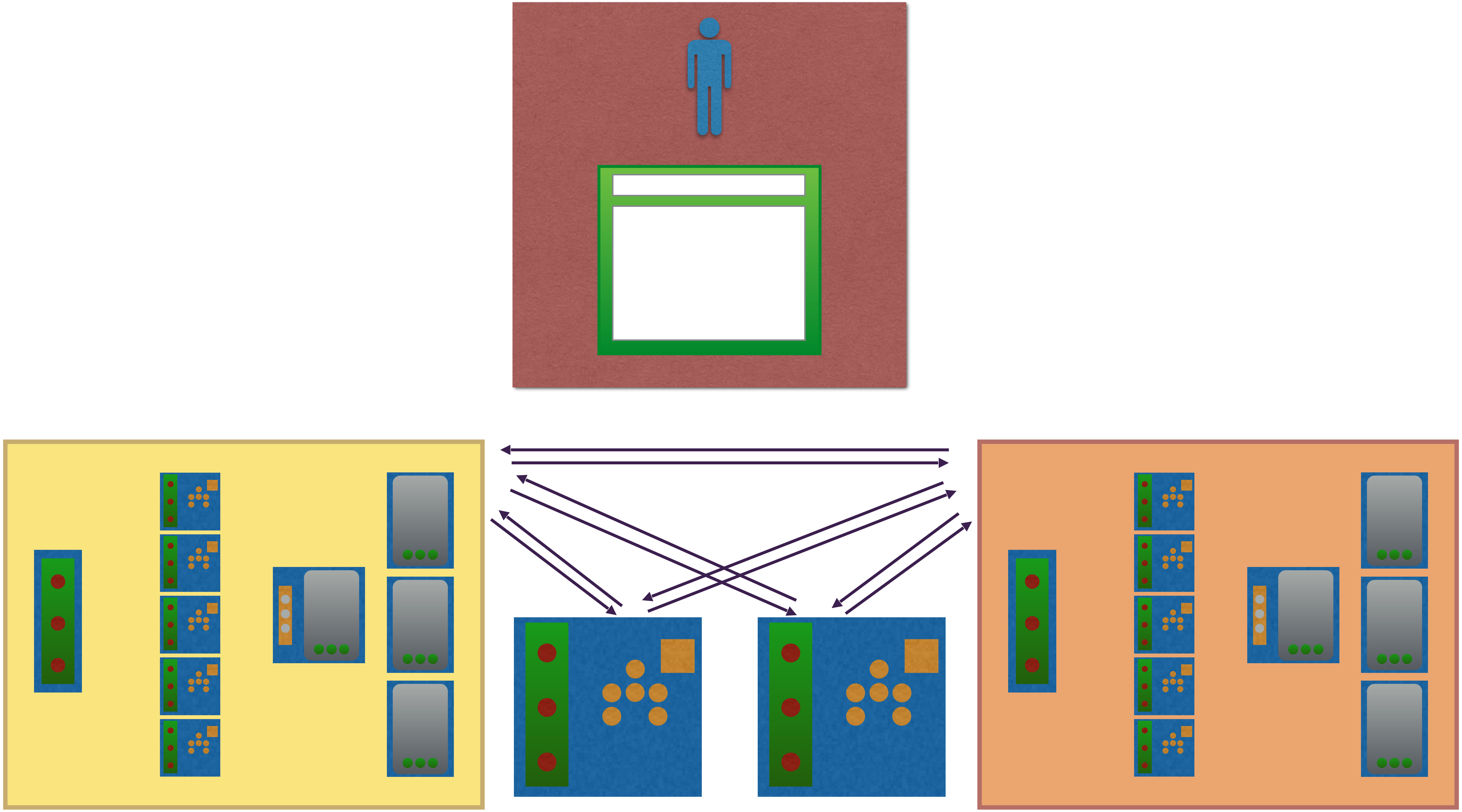
servers

load balancer
master DB

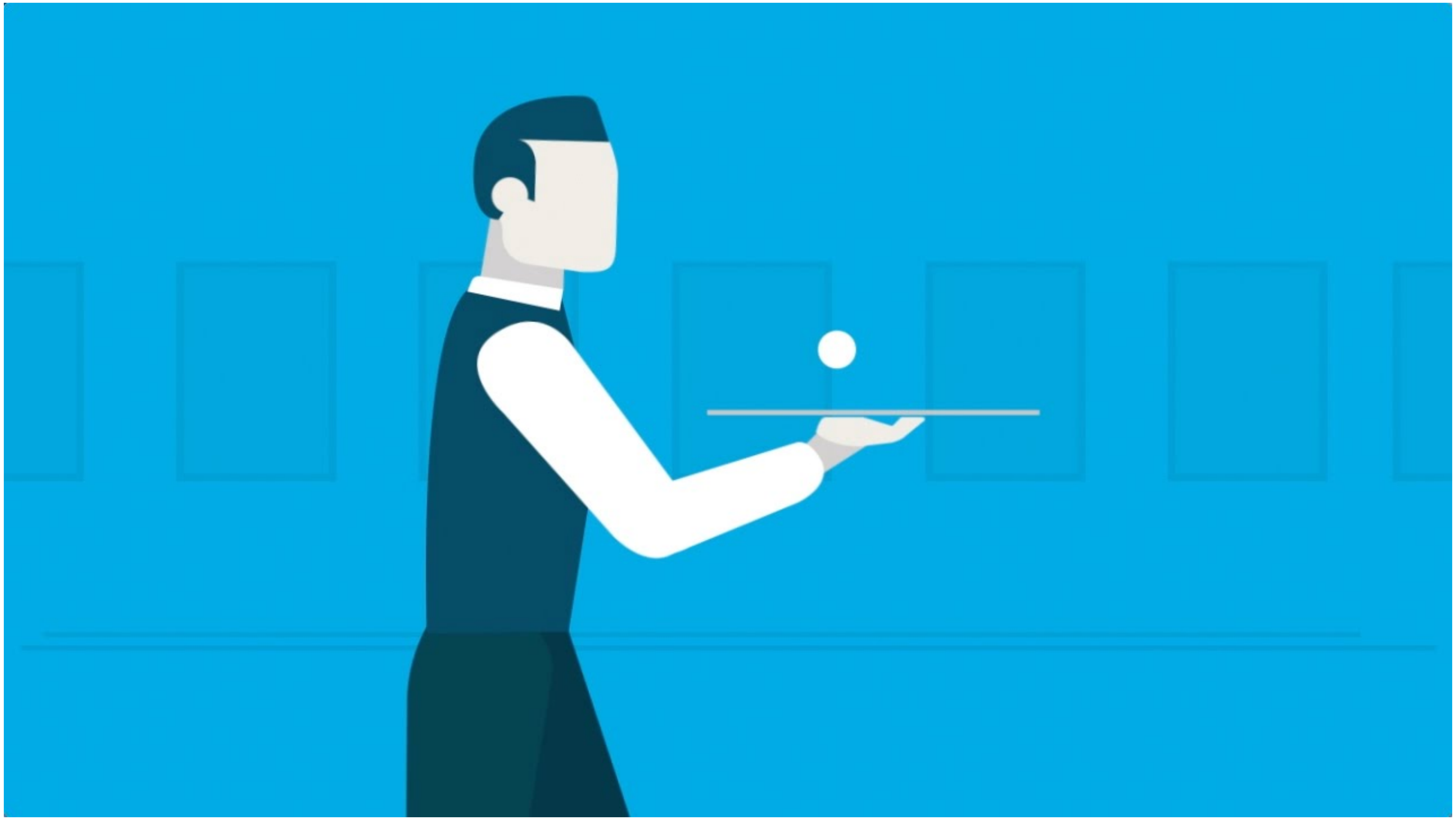
replication

slave DBs

Modularization and APIs



What is API?



API — is like an artist performing on stage, and its users are the audience

RESTful API

1. REST (REpresentational State Transfer) — is an architectural style for developing web services
2. API (Application Program Interface) — is code that allows two software programs to communicate with each other

API endpoint for Companies

```
1./getAllCompanies
2./addNewCompany
3./showCompanyDetail?id=23
4./deleteCompany?id=23
```

✗

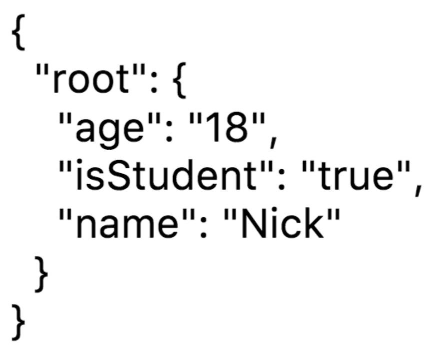
The URL is a sentence, where resources are nouns and HTTP methods are verbs.

```
1./companies            (GET)
2./companies            (POST)
3./companies/23         (GET)
4./companies/23         (DELETE)
```

✓

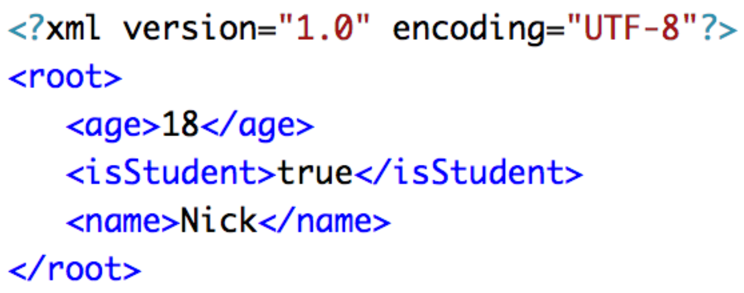
Data formats

- JSON



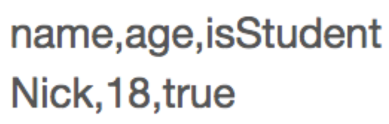
```
{
  "root": {
    "age": "18",
    "isStudent": "true",
    "name": "Nick"
  }
}
```

- XML



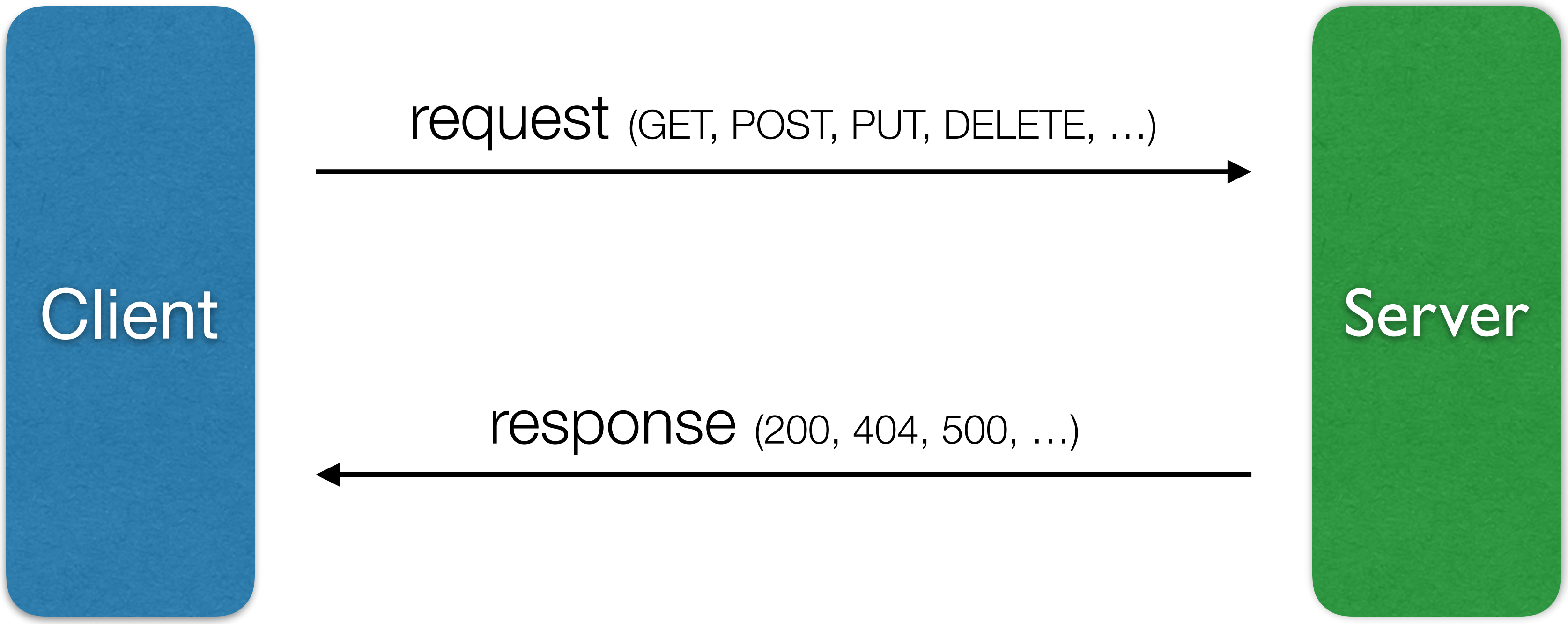
```
<?xml version="1.0" encoding="UTF-8"?>
<root>
   <age>18</age>
   <isStudent>true</isStudent>
   <name>Nick</name>
</root>
```

- CSV



```
name,age,isStudent
Nick,18,true
```

Client Server Communication



Client

request (GET, POST, PUT, DELETE, …)

response (200, 404, 500, …)

Server

Protocols

- TCP/IP — Transmission Control Protocol / Internet Protocol
  - communication among computers on Internet
    - HTTP — Hyper Text Transfer Protocol
      - Communicates with browsers to send web page packets
    - HTTPS — Hyper Text Transfer Protocol Secure
      - HTTP with Secure Sockets Layer (SSL)
    - FTP — File Transer Protocol
      - Used by FTP Clients to transfer file packets

HTTP response status codes

- 2xx — Success category
  - 200 Ok
  - 201 Created

- 3xx — Redirection Category
  - 304 Not Modified

- 4xx — Client Error Category
  - 400 Bad Request
  - 401 Unauthorized
  - 403 Forbidden
  - 404 Not Found

- 5xx — Server Error Category
  - 500 Internal Server Error
  - 503 Service Unavailable

Questions?