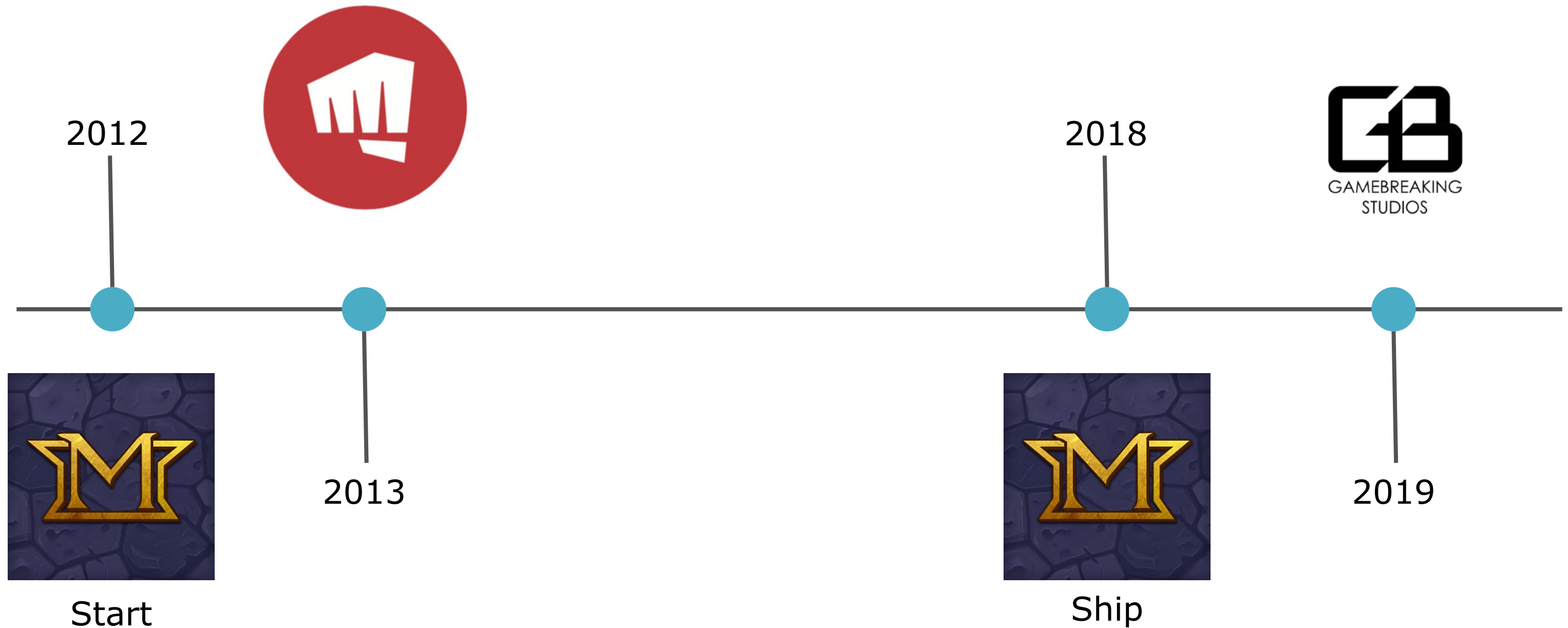


Scaling to 10 Concurrent Users: Online Services for Indies

Dru Erridge
Engineer & Cofounder @ Gamebreaking Studios

Why do Indies Need Online Services?

My Background & Project Timeline



“We’re Making an RTS”

-- No rational 21 year old, ever



Featuring

- 3v3 Online Multiplayer
- Dedicated Servers
- Built in Unreal 3 “UDK”

Team Size

Phase 1

9 months

Full Team: 45

Engineers: 15

Service Engineers: 2

Phase 2

4 years

Full Team: 7

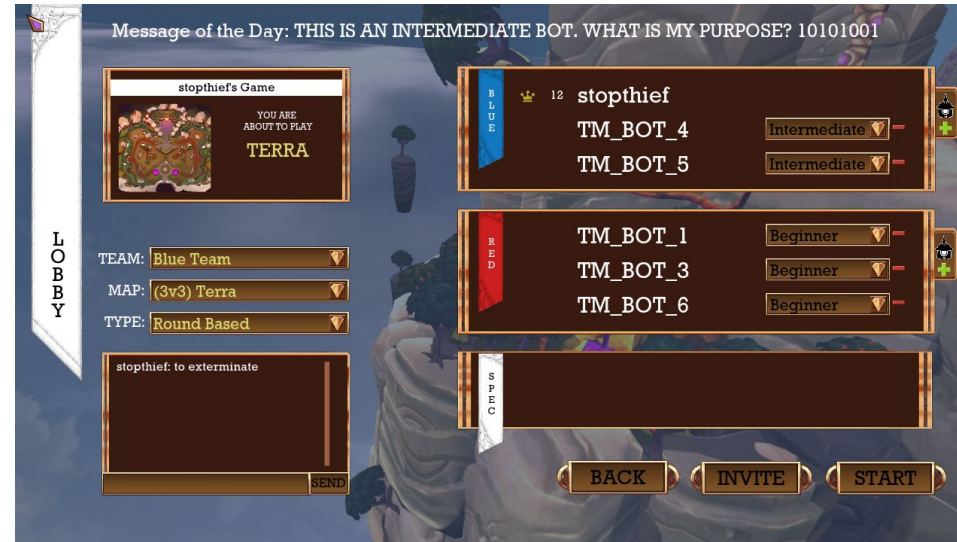
Engineers: 4

Service Engineers: 2

We need to support...



Login + Accounts



Lobbies + Chat



Dedicated Server
Management

And some other stuff...

Analytics

Player Stats

Admin Pages

Load Balancing

Health Monitoring

Multi-box logging

In-memory caching

Player Progression

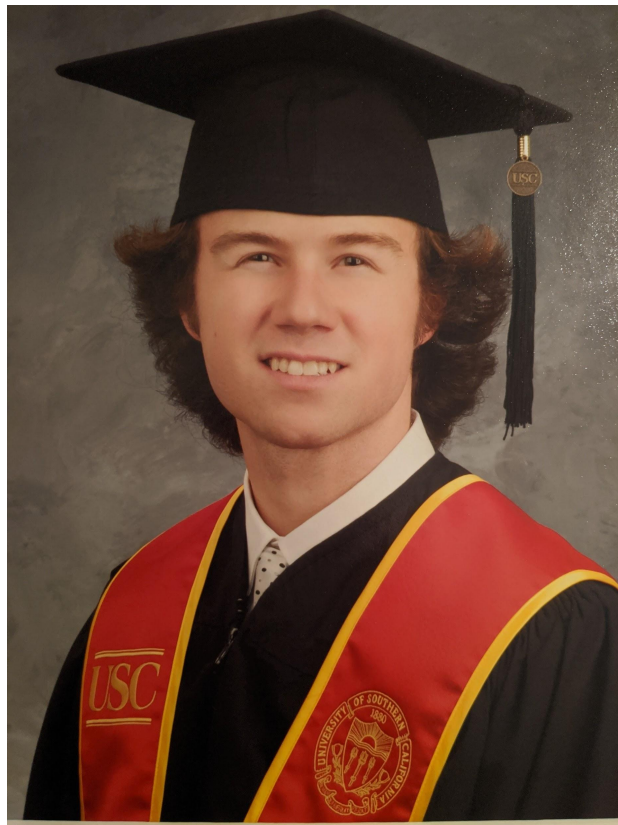
Database/Persistence

Deploying Servers

Installing & Patching

Rewind to 2013

Dru Erridge



Sean Saleh



Rewind to 2013

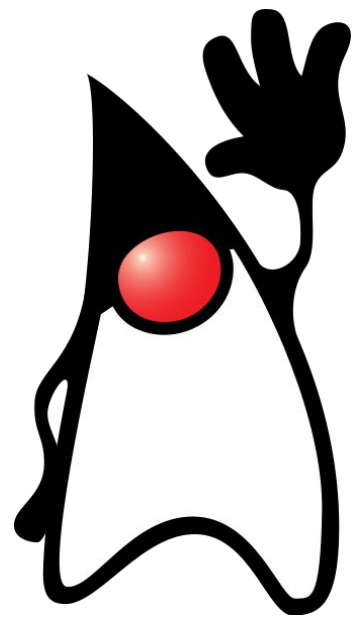
Game Dev

- Unity is **hot**
- No Unreal 4
- No Playfab / Gamesparks
- Pre-Steam Greenlight
- No Itch.io downloader

Services Dev

- Node.js is **hot**
- Cloud Wars heating up
- No Lambda
- No Kubernetes

“What should we use?”



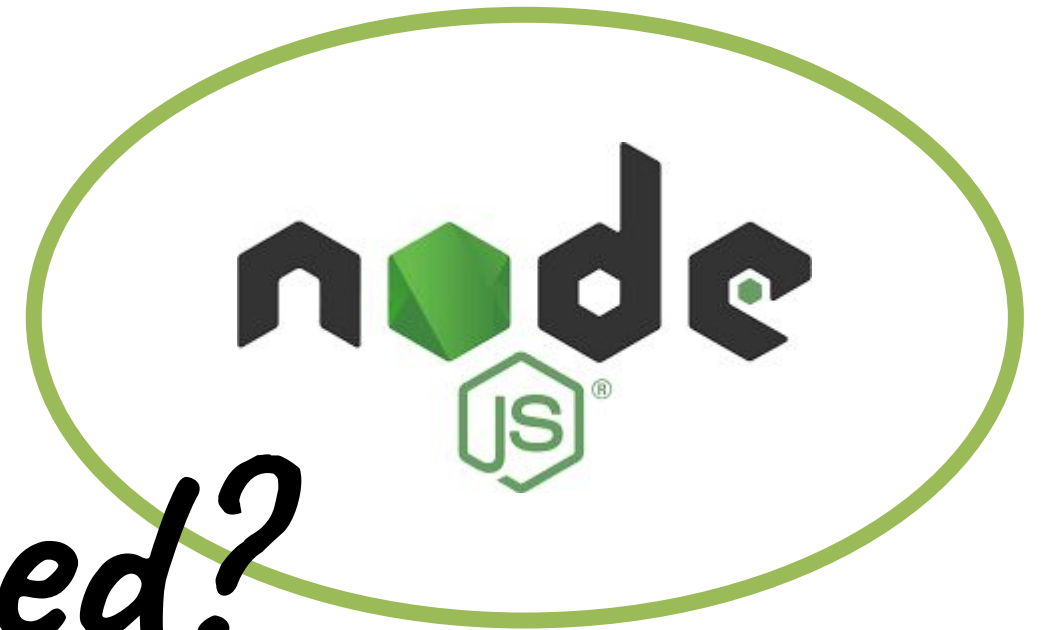
Java



“What should we use?”



“What should we use?”



Indie Approved?

“What should we use?”

die approved?



Why Node Worked

Express - easy web framework

Fewer hard problems (Multithreading, Async I/O)

We were excited, so we worked harder!

Some obstacles with Node

Very immature at the time (Pre-1.0)

Lack of static typing (we added TypeScript)

Awesome - Let's do some work!

Step 1 - Log in

Let's make an HTTP call
For security, make it HTTPS!

HTTP POST

<https://maestrosgame.com/game/>

```
{  
  "name" : "stopthief"  
  "password" : "hunter2"  
}
```



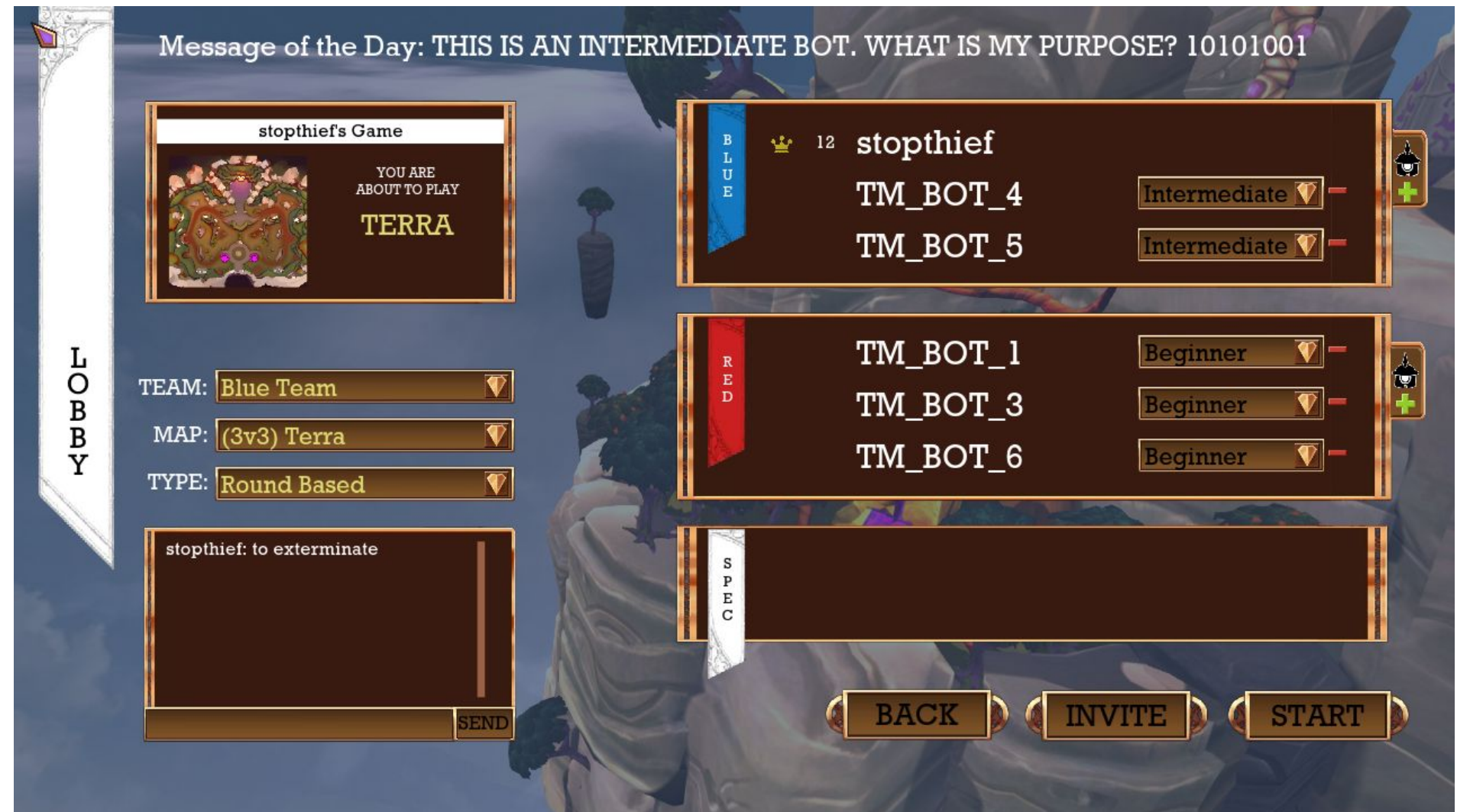
Step 2 - Host a game

Easy, another HTTP call

HTTP POST

maestrosgame.com/game/

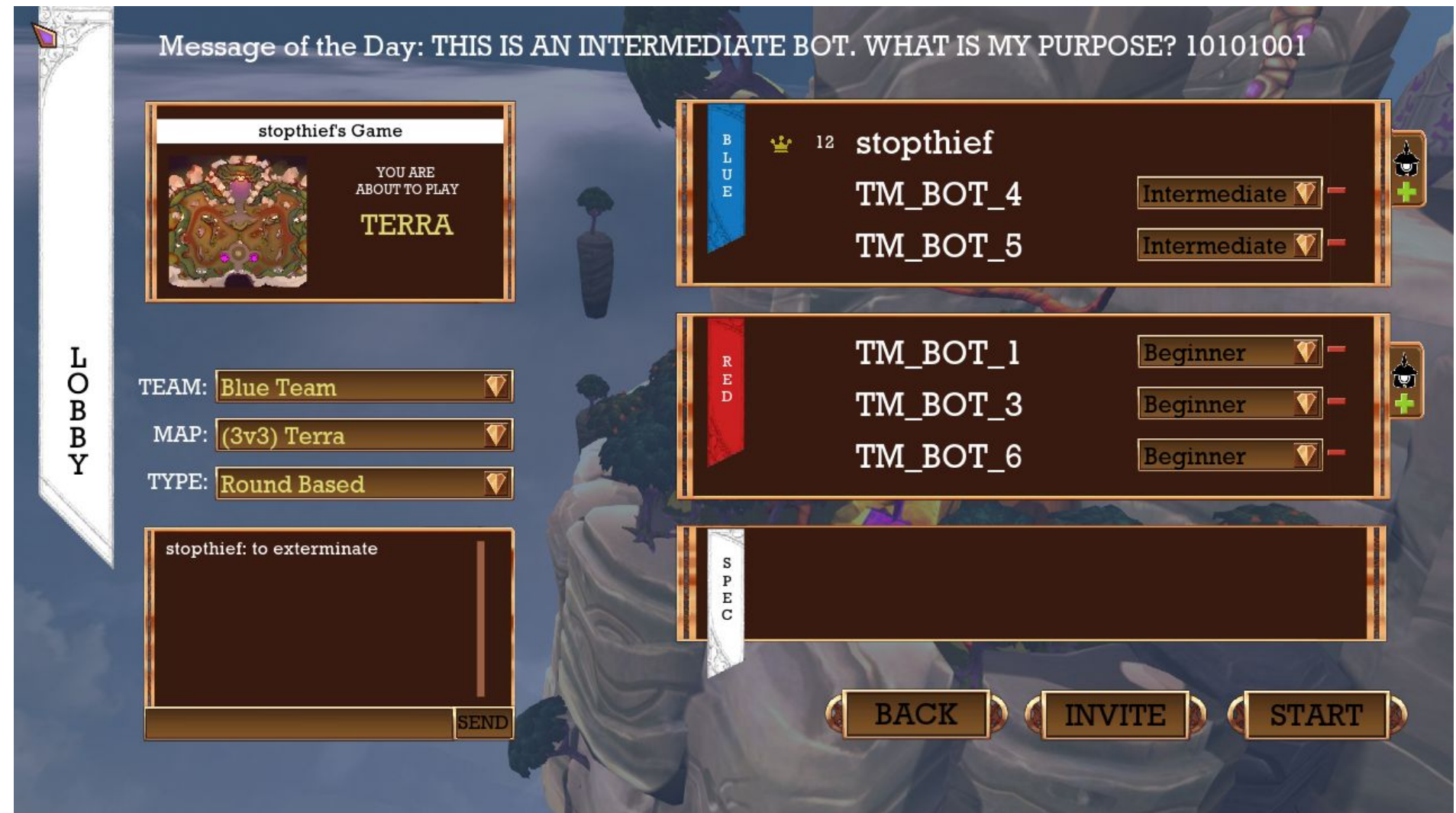
{“name” : “stopthief’s Game”}



Step 3 - Notify Map was Changed

Shoot...

How do we “notify” people?



HTTP POSTs & GETs Aren't Enough

Unidirectional

Polling for updates (inefficient and latent)

Concept of a “session” is weak & non-native

HTTP POSTs & GETs Aren't Enough

You need another protocol for sending updates w/o request

How do we talk to our services?

UDP

WebSockets

gRPC

TCP

HTTP
Polling

How do we talk to our services?

UDP

WebSockets

gRPC

TCP

HTTP

How do we talk to our services?

UDP

WebSockets

Indie Approved?

gRPC

TCP

HTTP

How do we talk to our services?



How should we talk to our services?

UDP

WebSockets

gRPC

TCP

HTTP
Polling

Criteria

Bidirectional

supports out-of-band communication

Easy to set up

native support in common services languages

Native Sessions

Are you connected? You've got a session!*

*it's definitely more complicated, but easier than HTTP "connectivity"

How should we talk to our services?

UDP

WebSockets

gRPC

TCP

HTTP

What do we put in each message?

Only the
changes

A hybrid

The entire
state

Message Contents - Only Changes

HTTP POST to maestrogame.com/game/

```
{  
  "action": "MapUpdated"  
  "newMap" : "Tera"  
}
```


Message Contents - Entire State

HTTP POST to maestrosgame.com/game/

```
{  
  "gameType": "RoundBased"  
  "map": "Tera"  
  "players": [  
    "player1",  
    "bot1"  
  ]  
}
```

What do we put in each message?



The entire
state

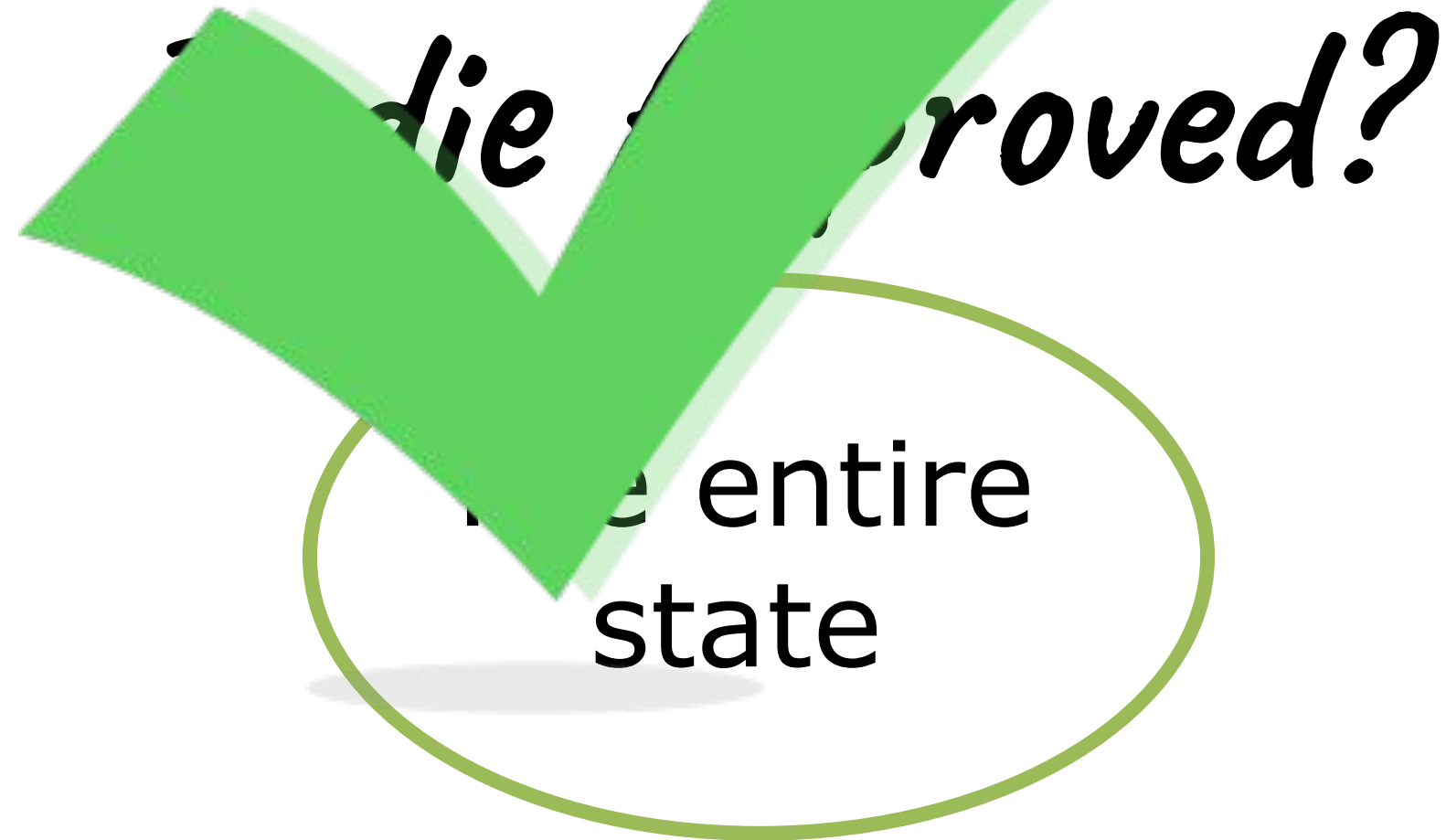
What do we put in each message?

Indie Approved?



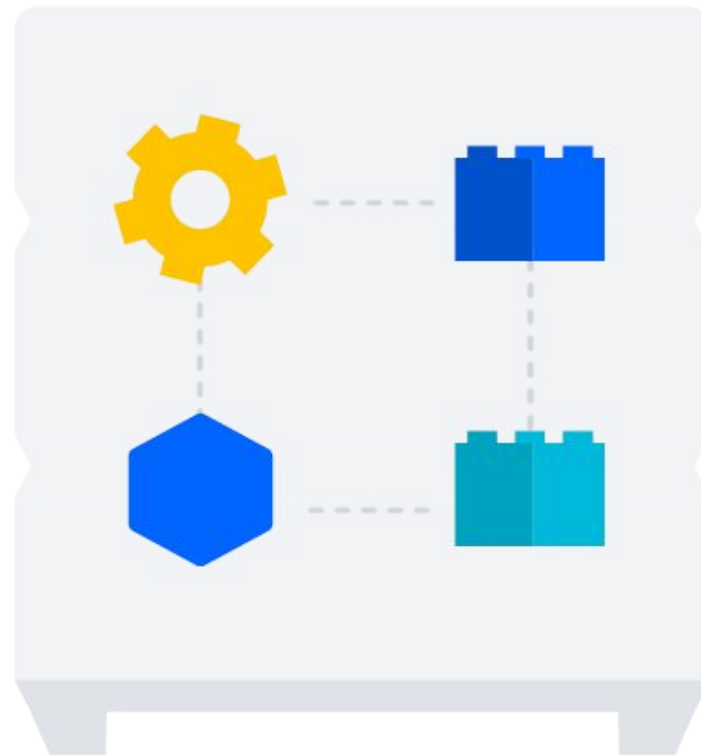
The entire
state

What do we put in each message?



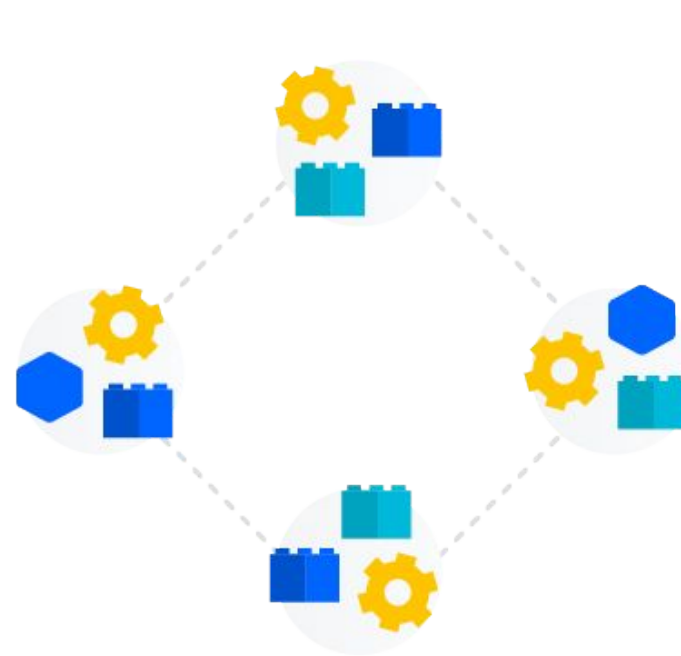
What should our architecture be?

Monolith



VS

Microservices

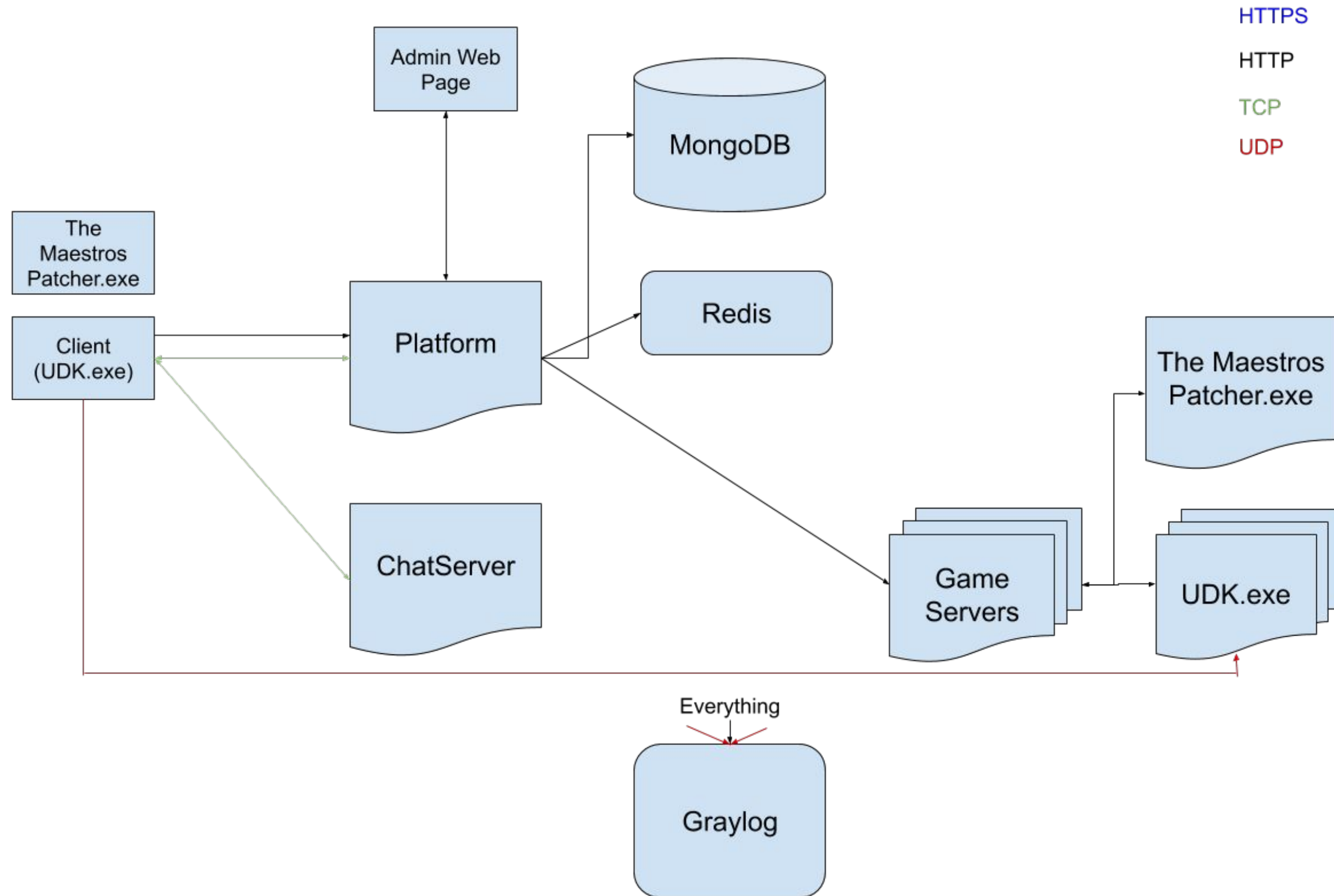


VS

Serverless



Architecture - organic first pass



Reviewing our Architecture

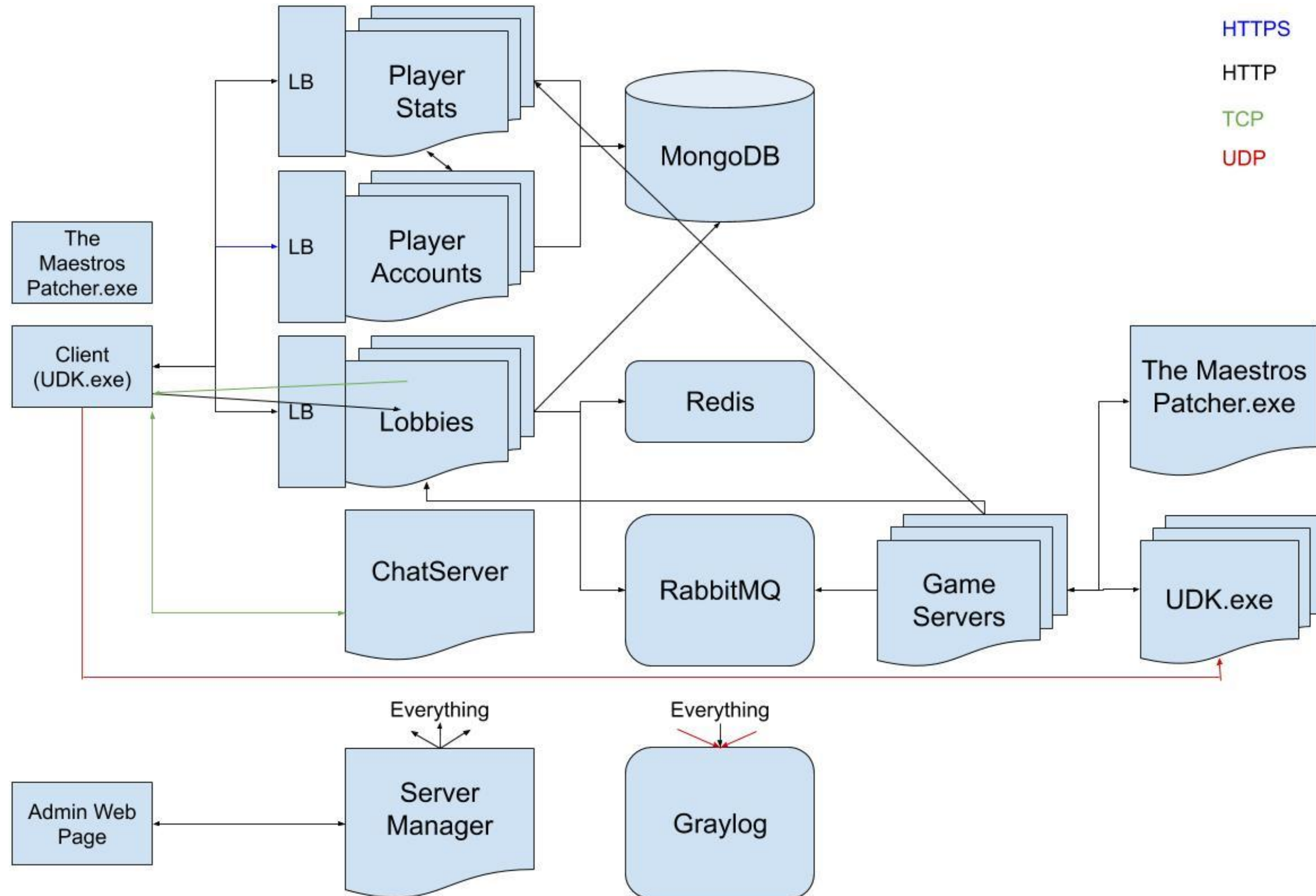
Not enough redundancy

Buggy game server management

Very manual deployment (copying folders)

Probably won't scale (single box)

Solution? Microservice Clusters!



Or was it?

- Months of tooling
 - Unifying deployment & logging
 - Unifying project structure
 - Sharing code (DB connections, HTTP calls, etc.)
- Complicated Code
 - DB race conditions
 - Data residency issues

Microservices vs Monolith vs Serverless

Which is Indie Approved?

Microservices vs Monolith

Which is Indie Approved?

I'm not sure.

Microservices vs Monolith

Listen to some smart people

Martin Fowler's MicroservicePremium

<https://martinfowler.com/bliki/MicroservicePremium.html>

Paul Hammant's Cookie-Cutter Scaling

<https://paulhammant.com/2011/11/29/cookie-cutter-scaling/>

Take-aways

Your most limited resource is **time**

Motivation is an important decision-making criteria

Beware the hype train, there be dragons