Building a Highly Scalable Service that Survived a Super Bowl

Keith Elder – Sr. Technology Evangelist
@keithelder
http://keithelder.net
Today’s Talk

Some stories
Some demos
Some technologies we used and why
Some lessons learned
Some Facts

• Started 6 years ago
• Launched Feb 7th, 2016
• 50 states, 3,143 counties
• Worked on by over 500+ team members
• 34 seconds, 9 Minutes
• 32nd largest
What would you do?
Buy? Build?
Super Bowl is 302 Days Away
We only had 90+ days to figure out a solution
Other Requirements

• Hard deadline
• 1-2 millisecond response time
• 30K transactions / second
• Always on
• Vast array of languages (C#, Java, Python, PHP)
• Hybrid cloud app
• Encryption during transit and at rest
• Easy for developers
What would you do?

Buy? Build?
What did we do?

We chose to build
Let’s go backwards before we go forwards
Innovation Time

Quicken Loans
Seven Languages in Seven Weeks

A Pragmatic Guide to Learning Programming Languages

Bruce A. Tate

Edited by Jacquelyn Carter
Popular Blog Architecture

Web Request

Web Server

Database

Cache
Enterprise Blog Architecture

Database Cluster

Web Request

Load Balancer

Fetch Data

Web Farm

Cache Cluster
It is his fault
Queued Services Architecture

Database -> Get angle brackets (html) -> Web Farm

Database -> Get curly braces (data) -> REST API

Web Farm <- Get curly braces (data) <- REST API

REST API -> Fetch Data

Fetch Data -> Database

REST API -> Queue

Queue -> Pub/Sub

REST API <- Pub/Sub

Cache -> REST API

REST API <- Cache

REST API <- Backend Service
MESSAGING
Erlang
Erlang in a tweet

Erlang: a battle-hardened, X-platform, functional language that makes writing reliable, concurrent, distributed systems a joy.
Where Erlang is Used

Automotive, Gaming, Finance, Robots, Cars, Drone software (updates drone while flying), Health Care, Infrastructure, Social Networks, Media (Huffington Post, Boston Globe), Messaging
<table>
<thead>
<tr>
<th>Game</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SKYLANDERS: SUPERCHARGERS</strong></td>
<td>Skylands is in peril! Kaos has unleashed his most evil weapon yet - the Doom Station of Ultimate Doomstruction.</td>
</tr>
<tr>
<td><strong>CALL OF DUTY: BLACK OPS 3</strong></td>
<td>Call of Duty: Black Ops 3 is a dark, gritty future where a new breed of Black Ops soldier emerges.</td>
</tr>
<tr>
<td><strong>GUITAR HERO</strong></td>
<td>Unleash your inner rock star with Activision's series of Guitar Hero games for all platforms.</td>
</tr>
<tr>
<td><strong>CALL OF DUTY: ADVANCED WARFARE</strong></td>
<td>Call of Duty: Advanced Warfare envisions the powerful battlegrounds of the future with evolved tech and tactics.</td>
</tr>
<tr>
<td><strong>DIABLO III: ULTIMATE EVIL EDITION</strong></td>
<td>The Prime Evil rages within the Black Soulstone, its essence screaming for vengeance and release.</td>
</tr>
<tr>
<td><strong>DESTINY</strong></td>
<td>From the creators of Halo and the company that brought you Call of Duty. In Destiny you are a Guardian of the last city on Earth.</td>
</tr>
</tbody>
</table>
.NET 4.0 Thread (1MB)  →  * One Erlang Process (allocates one kilobyte)
NODE1

NODE2

NODE3

NODE4
A few demos
Erlang Problem Domains

• Can’t fail
• Distributed
• Fault tolerance
• Upgrade while the app is running
• Performed without stopping the system
• Large number of concurrent activities
“My team has created a very innovative solution, but we’re still looking for a problem to go with it.”
Houston, we have a problem
Super Bowl is 302 Days Away
(or really our deadline is 90 days away)
April 12th 2015 – June 21st 2015

POC
April 12th 2015 – June 21st 2015
Interesting Tidbits

- 2000 lines of code
- Two Sr. Engineers
- The most used service in Rocket Mortgage
The Stack
I'M SO EXCITED
I'M SO SO SCARED
Moment of Truth

Load test dev

12,881 transactions per second
How it works
1) I’d like to send something to App B

2) Okay! Here is a key...

3) Here’s some encrypted data

4) I need to decrypt App A’s message

5) Here is the key
Hybrid Cloud – Each node is self contained
Hybrid Cloud – data is stored locally
Hybrid Cloud – kept in sync on all nodes
How did we do?
Report Card

- Hard deadline
  - Completed in about 2 months
- 1–2 millisecond response time
  - 1 millisecond response times
- Always on
  - Zero production downtime since first deployment
- Vast array of languages (C#, Java, Python, PHP)
  - Libsodium provides things for today plus future
- Hybrid cloud app
- Support encryption during transit and at rest
- Make it easy for developers
  - ?
```csharp
static void Main(string[] args)
{
    string json = SerializingCrypto.ToJson(new Foo { Bar = 123, Baz = 456 });
    Console.WriteLine(json);
    Foo fooFromJson = SerializingCrypto.FromJson<Foo>(json);
    Console.WriteLine();

    string xml = SerializingCrypto.ToXml(new Foo { Bar = 123, Baz = 456 });
    Console.WriteLine(xml);
    Foo fooFromXml = SerializingCrypto.FromXml<Foo>(xml);

public class Foo
{
    public int Bar { get; set; }

    [Encrypt]
    public int Baz { get; set; }
}
```

Lessons Learned
Erlang (BEAM) is a viable platform for performant API'S
Erlang is legit. We did load tests in production. Couldn’t break it.
In a distributed system keep each node as independent as possible (local data, etc)
Load test to pinpoint bottlenecks
And with each release
Thank You Innovation Time

[Image of a logo with the text "QL Bullet Time" and a Quicken Loans logo]
Building a Highly Scalable Service that Survived a Super Bowl

Keith Elder – Sr. Technology Evangelist
@keithelder
http://keithelder.net