Taking Iceland by storm

"Senior" developer at Kóði ehf

Building financial & banking software for too long

- Tiny

- Young

- Focused on trading, finance, banking

- Based in sunny 101 Reykjavík, Iceland

- Staff in Iceland and Ireland

- Also a brewery

- Market Data servers and data streamers

- Trading applications

- Order Management Systems

- REST services for market and financial data
- Clients and APIs for all of the above

Before

- C# server

- C# clients

- Java components

- C++ components

- C components

- BAT and PowerShell hell

- Hundred of thousands LOC

- DB bottlenecks

- Performance issues

- Scalability issues

Why change?

- It worked great before

- But then the economy started growing

- And our customers went mental

Spikes!

- Erlang + node.js

- And ended up with Erlang

After

- Erlang servers

- C#/JS clients

- C components

- Python components

- Tens of thousands LOC

- No DB bottleneck

- Easier to scale

- Distributed

Getting there

- Different mental model

- 6 months of coding

- 3 months of refactoring

- Develop deployment procedures

- ... getting it audited & deployed

Obstacles

- .NET is big in Iceland

- Erlang is unknown

- Community is small and "geeky"

- Businesses hadn't heard about it

- Deployment on customer premises

- Limited pool of developers

What we heard

- "Erlang, isn't that a dead language?"
- "Erlang is a toy, not fit for actual work"
- "Erlang is slow"
- "Nobody uses it"
- "There isn't a big backer"
- "You're using emacs?!"

The antidote

- Erlang Solutions & Ericsson
- Erlang usage stories
- Code sharing
- Simple deployment
- Better software
- Lots and lots of meetings

So now

- Great uptime

- Deployed at our data centers and on customer premises

- More services being rewritten in Erlang

- Lots of interest in Iceland in Erlang

Deployment & monitoring

- Coding is the easy part!

- Support for RHEL & CentOS

- RPM packages and Puppet

- RPM app-ups

- Monitor hooks into nagios, hobbit, AMQP

- Detailed manuals

Deployment 2.0

- "GitHub Enterprise" model

- Web & REST interfaces for applying updates

- Easier to deploy clusters

- Still RPM & Puppet based

Done!

- @omarkj Twitter, Github, IRC
- Email omar@kodi.is