Tools of the Trade

Web Development at Fairfax Media

Who am I?

George Wright

Solution Lead - Intelligent Systems

Fairfax Media

A Solution Lead is a senior developer that is responsible for the **delivery** of a **business function** (called the solution).

I apologise in advance

This presentation has a lot of:

Acronyms, Buzzwords & Brands

- Software architecture is evolving rapidly and is extremely diverse.
- There are no right answers!

Software Developers

The CRAFT of software

- part engineering - accuracy and efficiency
- part designing - elegance
- part project managing - accountable
- part poetry - expressive

Web is Client/Server
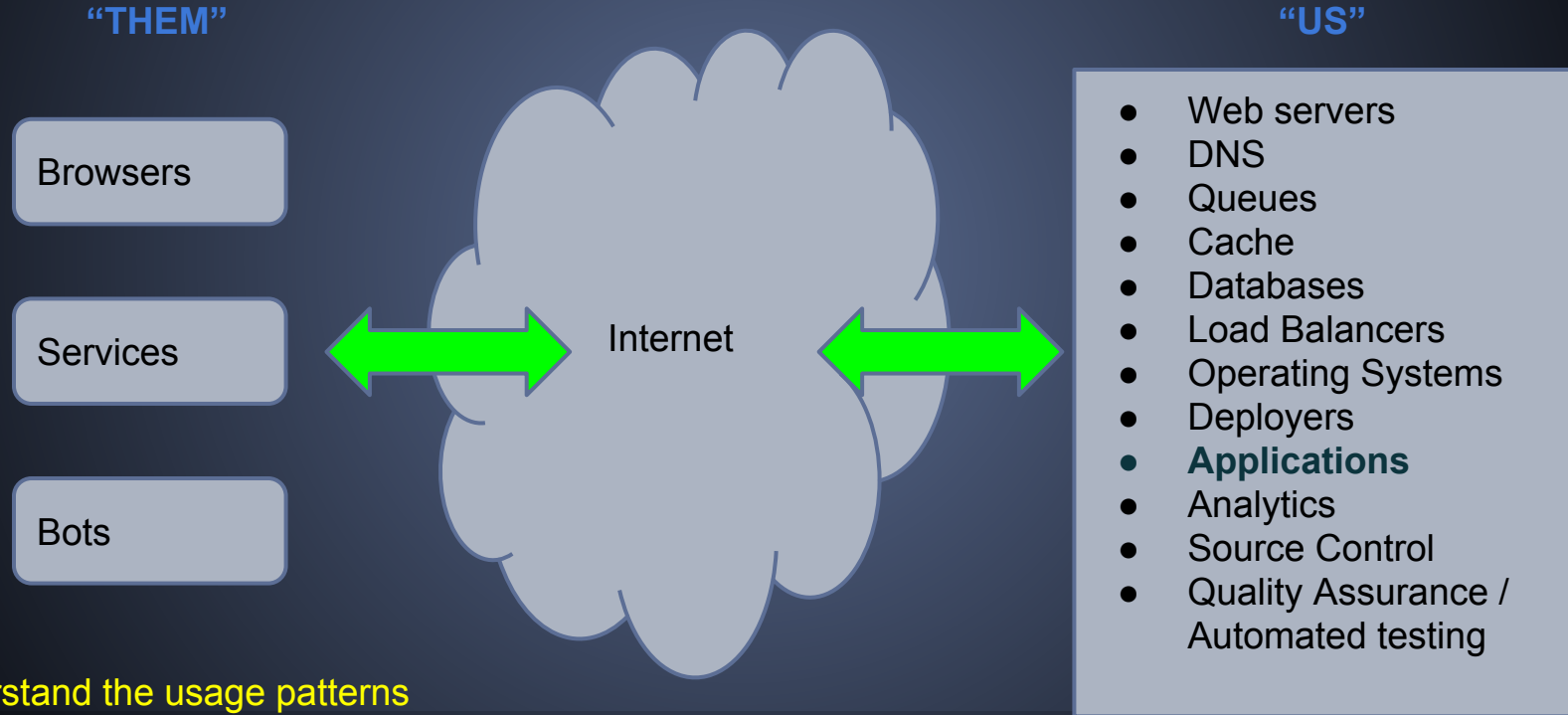
It's actually more like Client/Service(s).

- Engineering for interoperability is key.
- No longer a monoculture approach.
- No one technology is the silver bullet.

the "Stack"

All of the systems and services that fulfil a request (end-to-end) - in this case a request from a web client.

The stack is the largest box you can draw around the things you can control/influence.

Stack basics

Browsers

Services

Bots

Internet

- Web servers
- DNS
- Queues
- Cache
- Databases
- Load Balancers
- Operating Systems
- Deployers
- **Applications**
- Analytics
- Source Control
- Quality Assurance / Automated testing

Understand the usage patterns

The "Corporate" stack

Proxies & Cache (MS Forefront, Java Caching System, Squid)

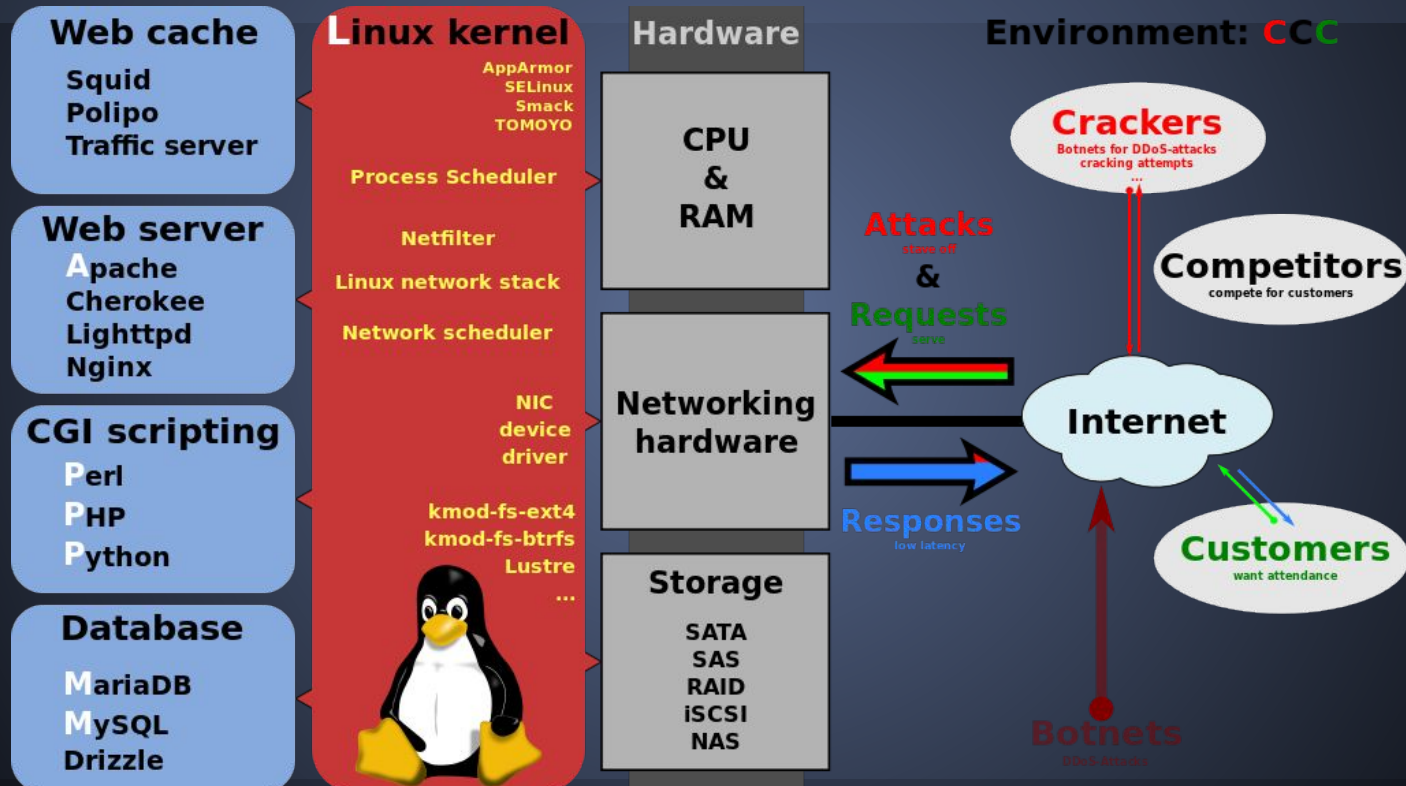DNS & Load Balancing (Windows Server NLB, Apache Camel)

Web Server (IIS, Apache)

Application Server (Tomcat, JBoss, GlassFish, IIS .Net)

Application (Java, .Net)

Corba, WSDL, RPC, SOAP

Database (SQLServer, Oracle)

The LAMP stack

Web cache
- Squid
- Polipo
- Traffic server

Web server
- **A**pache
- Cherokee
- Lighttpd
- Nginx

CGI scripting
- **P**erl
- **P**HP
- **P**ython

Database
- **M**ariaDB
- **M**ySQL
- Drizzle

Linux kernel
- AppArmor
- SELinux
- Smack
- TOMOYO
- Process Scheduler
- Netfilter
- Linux network stack
- Network scheduler
- NIC device driver
- kmod-fs-ext4
- kmod-fs-btrfs
- Lustre
- ...

Hardware
- CPU & RAM
- Networking hardware
- Storage
  - SATA
  - SAS
  - RAID
  - iSCSI
  - NAS

Environment: CCC

Crackers
Botnets for DDoS-attacks
cracking attempts
...

Competitors
compete for customers

Internet

Customers
want attendance

Botnets
DDoS attacks

Attacks stave off
&
Requests serve

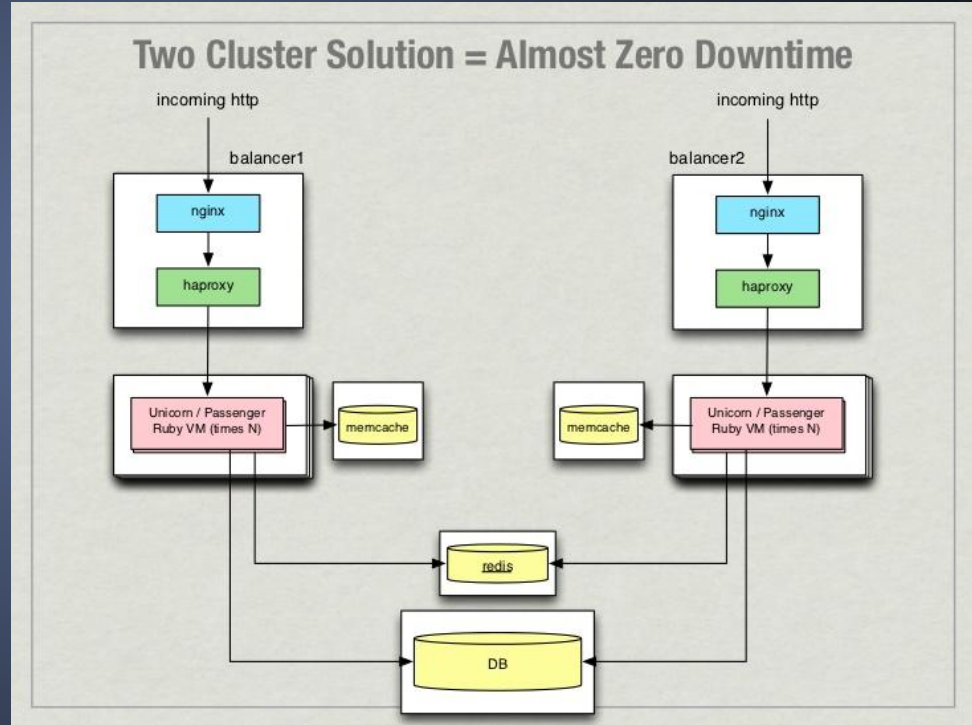**Responses** low latency

New wave stack - Ruby on Rails

Excellent presentation on RoR Enterprise Arch.:

www.slideshare.net/kigster/
enterprise-architectures-with
-ruby-and-rails

Redis, Solr, Capistrano,

Unicorn, Puppet, Postgres,

ActiveRecord, nginx,
NewRelic

The "hipster" stack

MEAN stands for:

MongoDB is the leading NoSQL database, empowering businesses to be more agile and scalable.

Express is a minimal and flexible node.js web application framework, providing a robust set of features for building single and multi-page, and hybrid web applications.

AngularJS lets you extend HTML vocabulary for your application. The resulting environment is extraordinarily expressive, readable, and quick to develop.

Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications.

http://mean.io/#!

Where is the rest?

In
The
Cloud

Amazon AWS ftw

Rethinking the architecture at scale:

- Compute (EC2) - processing
- Elastic Load Balancer
- Storage (S3/Glacier)
- Database (DynamoDB/RDS/Redshift)
- In Mem. Cache (Elasticache)
- DNS (Amazon Route53)
- Web Cache (CloudFront)
- Queues (SQS)

- Notification (SNS, SES)
- Analytics (EMR)
- Applications (ElasticBeanStalk)

+ Many, many more.

Alternatives

Application Hosting - Heroku, EngineYard, OpenShift, Google AppEngine, Microsoft Azure, Joyent + new entrants every day.

Check out DigitalOcean tutorials

https://www.digitalocean.com/community/tutorial_series

The Browser "Basics"

- HTML5
- CSS3
- Javascript - JQuery(?), Modernizr
- Debugging - FireBug, Chrome Dev Tools
- JSON - CORS and JSONP
- REST

Frontend Stack provides:

- Package Management
- Templating
- Testing framework
- Common widgets
- Databinding
- Server communication

But wait there's more..

Frontend JS frameworks (just a sample)

Yeoman: **Package manager:** yeoman, **Module system:** RequireJS, AMD, **Build system:** Grunt, **Widgets:** Bootstrap

Angular/Closure: Module system: Closure Library modules, Build system: ClosureBuilder, Templates: AngularJS,

Data binding: AngularJS, Sync: AngularJS services, Widgets: Closure Library, Test framework: Closure Library testing

Foundation: Package manager: Component, Module system: CommonJS, Build system: Component, Data binding:

Reactive or Rivets or Knockout or Backbone, Sync: component/model, Test framework: test/assert, Mocha

Developer stack

Every developer should have their own personal productivity stack including:

- Editor - Notepad++, Atom, Sublime, vim etc.
- IDE - Visual Studio, JetBrain, Netbeans, Eclipse
- Version Control - git, mercurial, bzr, cvs
- Repository - gitorious, github, bitbucket, launchpad, googleCode
- Issue tracking - JIRA, Redmine, Bugzilla
- Agile/SCRUM - Greenhopper, Trello

Quality assurance stack

- Automated testing eg. Mocha, Jasmine, Selenium etc.
- User acceptance testing tools
- Production monitoring
- Disaster recovery plan
- Peer reviewed documentation
- Stakeholder management
- Legal signoff(?)

Thank you UNSW

Questions?