Typesafe

Bringing Reactive Applications
to the Java Virtual Machine

# Delivering on The Guardian’s digital-first vision

**The Guardian was embarking on a major transformation that sees the newspaper changing from a print-based organization to one that is digital-first in philosophy and practice. This new strategy involves a series of bold changes not only to the way they do things, but with the tools and software they used to accomplish these tasks. The Guardian chose technologies from Typesafe to provide a solid, scalable foundation on which to start building these new services.**

## About The Guardian

The Guardian is a unique news organization whose journalism can be found across a range of different platforms. With a highly distinctive, open approach to publishing on the web, part of Guardian News & Media’s digital-first strategy, it now reaches a global audience of tens of millions.

Guardian writers are free to present the truth as they see it, without interference by shareholders, a proprietor or a political party. As a result the Guardian is a byword for serious, trusted, independent journalism.

The Guardian grew from regional roots 192 years ago in Manchester to become a major national newspaper. Now, with a digital-first philosophy and an international audience, it is among the leading news organizations in the world.

## The Problem

When Alan Rusbridger, Editor-in-Chief of Guardian News and Media revealed plans to become a digital-first organization, placing open journalism on the web at the heart of its strategy, the pressure was on to quickly modernize The Guardian’s existing web infrastructure.

Historically, The Guardian had had an off-the-shelf, albeit heavily customized, Content Management System (CMS) that required programming in Tcl and PL-SQL to extend its functionality. This solution proved hard to maintain and move forward and consequently The Guardian began to move off this is platform in 2006.

Graham Tackley – now Head of Architecture for The Guardian – was part of a team that selected a new platform that consisted of standard open source components including Java, Spring, Hibernate and Velocity with an underlying Oracle database.  Indeed, this platform has served The Guardian well over the years, and even today still runs the main website at www.guardian.co.uk.  However, since it is a monolithic application, concern grew over how long it took to add new features. Graham and his team were creating a full new release every two weeks, which is excellent compared with Enterprise Java applications, but was comparatively slow compared with other websites.

By this time, Graham was now leading the Platform Team, which consisted of four Java developers.  This team was responsible for building the backend of the Content API, as well as the MicroApp Framework, which allowed them to gradually break functionality out of the Java/Spring/Hibernate monolithic application that had grown over time.

Also during this time, the team started experimenting with different languages; an early consideration was Python with Django.  While this appeared to be tenable solution, a huge disadvantage was that it couldn't leverage any existing code assets, so it was essentially a case of throwing the baby out with the bathwater and starting from scratch.

They wanted to innovate and make changes faster, and that was going to require some bigger changes.

## The Solution

The team began to spend time in earnest looking at tools, approaches and languages that would enable them to reach their goals.

Graham's team had been playing with Scala for some time, but had been primarily using it with ScalaTest - a testing DSL that exercised their Java code.  Scala is a general purpose programming language inspired by the long tradition of functional programming, which makes it easy to avoid shared state.  Scala ensures the ready distribution of computation across cores on a multicore server, and across servers in a datacenter.  This makes Scala an especially good match for modern multicore CPUs and distributed cloud-computing workloads that require concurrency and parallelism.  Since they had gained valuable experience with Scala in this context, they decided that it would be the perfect language with which to build their new Content API.  They could leverage their existing skills, and significantly ratchet up productivity, focusing their time writing code that solved a business problem rather than writing code that solved frameworks problem.

Since Graham and his team had already started to write their Content API in Java, they decided that they couldn't just stop and rewrite everything in Scala, so they gradually converted existing Java code to Scala a class at a time which worked tremendously well; Graham put it very succinctly – "it just worked."

An important initiative for The Guardian is their mobile site, and while Scala covered the server tier well, something Graham also needed was a web framework.  Graham mentions that they tried every Scala web framework that they could find, however none fit the bill quite right.  An early web framework candidate was the Play 1.0 Web Framework for Java; while it did show some promise, it lacked some key features they were looking for.  Subsequently, the Lift Web Framework appealed to Graham and he built some applications with it, however it failed to gain traction with the other developers and was later sidelined.

By now, Graham had become the Head of Architecture and needed to make a commitment to a web framework.  Play 2.0 had come along and hit all the right criteria.  The Play 2.0 Web Framework is based on a lightweight, stateless, web-friendly architecture that features predictable and minimal resource consumption (CPU, memory, threads) for highly scalable applications.

*"Play 2.0 is designed expressly for web application developers rather than back end developers."*

*Grant Klopper*
*Technical Lead for the mobile site*

There are two major features of Play that Grant enjoys:

- Play Templating; "It's very nice, people like it a lot.  Implicit methods to help them do stuff quickly and easily.  Fully compiled and type-checked templates allow them to discover errors at compilation time, not in production!  Grant would have a hard time moving to anything else at this point".

- Akka Foundation; Akka is a toolkit and runtime for building highly concurrent, distributed, and fault tolerant event-driven applications on the JVM.  Akka vastly simplifies the concurrency problem by providing developers with a higher level of abstraction in its Actor-based model.  By utilizing Actors, developers can focus on writing business logic, instead of worrying about low-level tasks like state management, memory locking and thread management.  Grant's team has a goal of having no more than one blocking call per request, however when you look at the main page, there are a lot of calls.  These calls are made concurrently and subsequently they can build an asynchronous cache in memory, simply by leveraging the built in Actor model provided with Akka.

The Guardian's mobile site - m.guardian.co.uk - is built with a responsive design, with the aim that the site will grow and evolve to replace the current desktop site and will also be valid across all devices.  The site was launched on 26th November 2012 and regularly has over 2 million daily page views.  It is *well past* 100 million page views since it's launch, has been highly performant and has had *zero* downtime.

Grant comments "It just runs".

From a device standpoint, currently all non-tablet traffic is redirected to the mobile site, which accounts for 10-20% of all web traffic.  The site is deployed on Centos Linux servers on Amazon EC2 instances.

The Guardian also has an extensive GitHub repository that is totally public source; the mobile site code can be seen at https://github.com/guardian/frontend.

## Strategies, Lessons Learned

Moving an entire team to Scala didn't happen overnight, but Scala protects your investment in existing Java libraries, tools, and developer programming skills. Scala programs are compiled directly to Java bytecode that runs on the extremely mature Java Virtual Machine, and leverages the JVM's robust just-in-time compilation, garbage collection, and well-understood deployment techniques. The operations team won't see a difference. Developers keep working in their familiar tools, begin writing Scala in a Java format, however they're writing code that's shorter, faster, more scalable, more correct, and even more fun.

Recruiting a development team is always a tough task, but even more so when you are looking to recruit talent with skills in one of today's hottest technologies.  To that end, Graham does not even look for Scala developers, just open-minded people who are eager to engage and want to learn something new, and ultimately, resume-worthy.  Graham feels that the investment to get someone new up to speed more than pays off in the long term, especially with the plethora of material available. Graham cites Martin Odersky's Functional Programing course delivered via Coursera as being hugely popular and eagerly attended by his whole staff.

As a result of the amazing success that The Guardian has seen utilizing Scala and Play across multiple projects, their architectural standards now dictate Scala as the de facto language, and Play as the web front-end. Ultimately, The Guardian's bet on the Typesafe technologies has proven to be a good one that has served them well so far and has put them in a great position to deliver on their Editor in Chief's "digital first" strategy.

*The **Typesafe Platform** is a modern software platform that makes it easy for developers to build scalable software applications. It combines **Play Framework**, **Akka** runtime, the **Scala** programming language, and robust developer tools in a simple package that integrates seamlessly with existing Java infrastructure. Commercial support and maintenance is available for the Typesafe Platform through the **Typesafe Subscription.***