bet365

goto; conference

Click ‘engage’
to rate sessions
and ask questions

Follow us on Twitter @GOTOber

www.gotober.com

bet365

Maximising Side Effects When Adopting Functional Programming

Dan Macklin Head of R&D

Background – A Brief History of bet365

So what does that feel like?

How? – What's the Secret Sauce?

It Comes Down to Our Culture

- We take on hard problems
- We innovate
- We have great desire to be the best
- We don’t water things down when the going gets tough
- We allocate capital and resources well

bet365

Our Historic Tech Stack

bet365

Our Historic Tech Stack

- Mass market appeal with proven delivery models
- Traditional / Pragmatic
- Focused around rapid delivery
- A few key ingredients that are well seasoned
- Mostly .NET with Java Middleware
- Lots and lots of SQL Server
- The foundation of a multi £billion business

But there were some fairly large problems on the horizon

Productivity was declining

Which was a Huge Problem

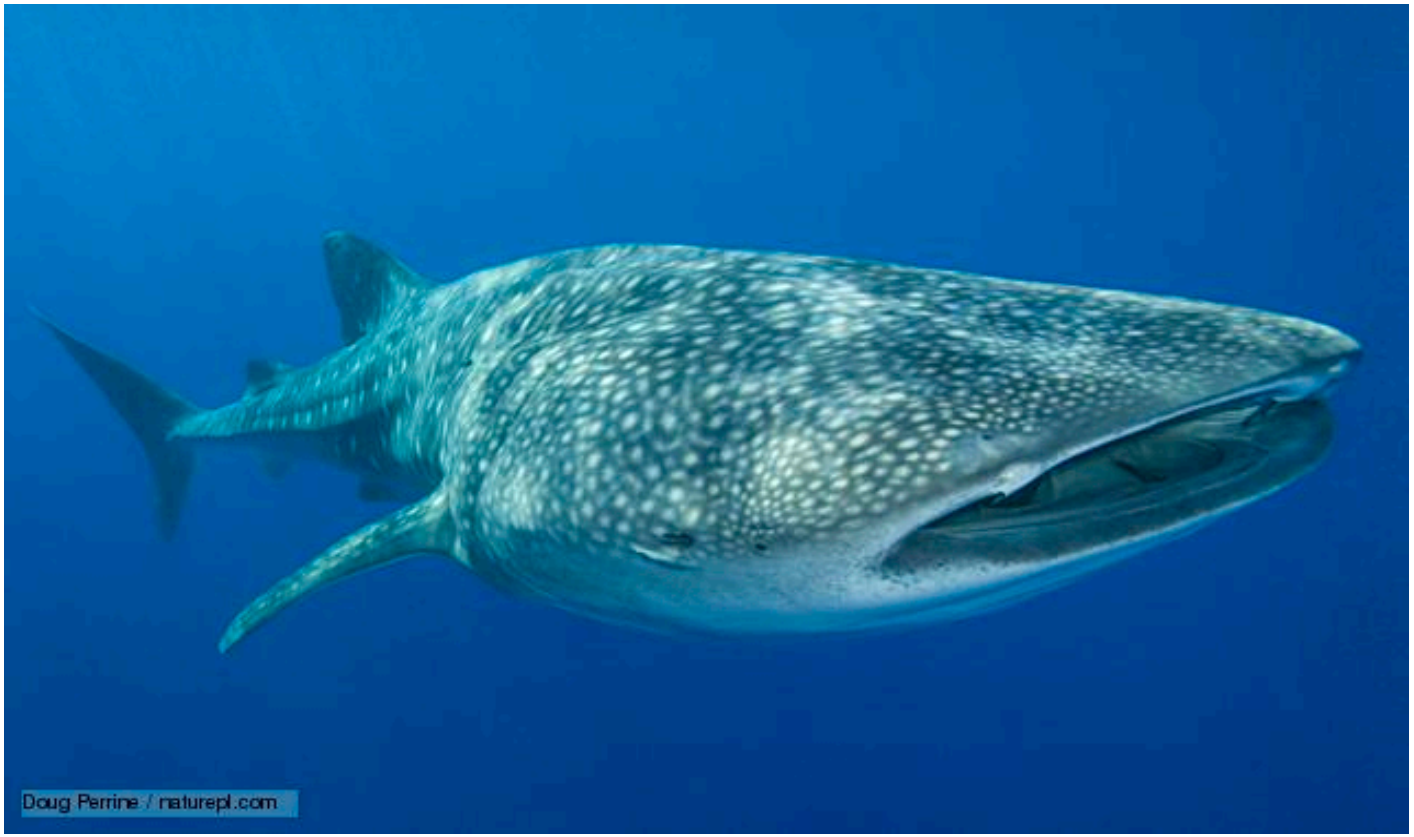
- We'd always led from the front and set the agenda
- We have very high expectations
- We operate in a fiercely competitive market
- Where it is relatively easy to change suppliers

Our systems were becoming unreliable

Our code was becoming overly complex due to all of the scalability hacks

Parallelism rather than clock speed began to drive compute performance

Our software had reached its limits



We needed to Reinvent

Going forward we wanted to improve our customer experience by

“Orders of Magnitude”.

We needed to be more like this

bet365

We wanted to create a culture where it was ok to take big technical risks

We wanted to experiment

We wanted to get hands on and try lots of stuff

We wanted to replace
“We think that”
with
“We know that”

So we set up an R&D team

- Free thinking, curious people from varied backgrounds
- Outside of day-to-day development concerns
- Given space and time
- Flexible and entrepreneurial
- Not scared of big problems
- Well resourced with good sponsorship

Our Mission

“Explore promising methodologies, seek out novel technology and languages, to pragmatically go where nobody at bet365 has gone before.”

So what's the link to Functional Programming?

bet365

And the answer is.....

- To achieve our aims we would need to build Distributed Systems
- This is challenging and can become extremely complex
- We needed to express these difficult concepts as simply as possible
- Erlang with its functional heritage seemed to offer some answers
- We wanted to stand on the shoulders of giants

We needed some solid foundations

bet365

Erlang - Overview

- Designed for highly concurrent - fault tolerant systems
- Not new – Developed in the 90s by Ericsson
- Greater than the sum of its parts. It's features work well together
- Soft real time
- Lightweight processes, "Let it Crash" supervised error recovery, Actor Model – Immutable Message Passing, Garbage Collection, OTP, Hot Code Upgrade and of course Functional Programming

bet365

Erlang – The Functional Advantage

- Computation is the result of evaluating functions
- Every function should do one thing well
- Output should only depend on input. Side Effects - eliminated
- Immutability – Once assigned variables cannot be changed
- Well written functional code is inherently simple, easy to understand and test
- Interesting syntax takes a bit of time to get your head around

It's my belief that complex systems should be designed in the small

This matches the ethos of Functional Programming

bet365

So theoretically Functional Programming makes it easier to build reliable distributed systems

How did we prove it?

We went through three stages of adoption

Stage One
“Good Idea?”

Where to start? - Initially expect Chaos

bet365

Proof of “Good Idea”

- Start with a well defined project and a clear definition of success
- Make sure that you have good sponsorship
- Keep things simple
- Measure everything that matters (qualitative and quantitative)
- Run repeated performance tests throughout the process
- Use existing tests / benchmarks to prove the case

bet365

Proof of “Good Idea”

- Remember you are doing something new
- Get something to work then iterate on it
- You will be learning on the job
- To avoid accidental complexity solve problems as they come up
- Keep journals. Document the journey

bet365

Our First Project - RealGood

- Business Critical Pub Sub System
- Pushes sports content out to our customers
- Currently 3.5 million topics with over 100k updates per second
- Objectives:
  - Can it be done?
  - How will an Erlang system perform in turns of performance and availability?
  - What does development feel like?

bet365

RealGood - Results

- Erlang worked
- Our first POC was up and running within a couple of months
- It could handle 4* the throughput along with substantially more clients
- The POC was less susceptible to failure
- The code was smaller, simpler and easier to understand

bet365

RealGood - Results

- We could learn Erlang on the Job and enjoyed using it
- The results looked good
- We passed on our POC to a small Dev team
- With our support they moved it to a full Erlang implementation

Stage Two
Widening Involvement

AKA The First Big Sales Pitch

Ok the last slide was a joke....

You need to get people to buy-in not just buy!

bet365

Widening Involvement

- Picked a challenging new problem
- Asked the development teams to propose a solution using existing techniques
- We built a POC to validate an Erlang approach
- Formulated a strategy to help the development teams take our POC and deliver a production system
- Wrote extensive documentation around the problem domain and our POC

Widening Involvement – Key Considerations

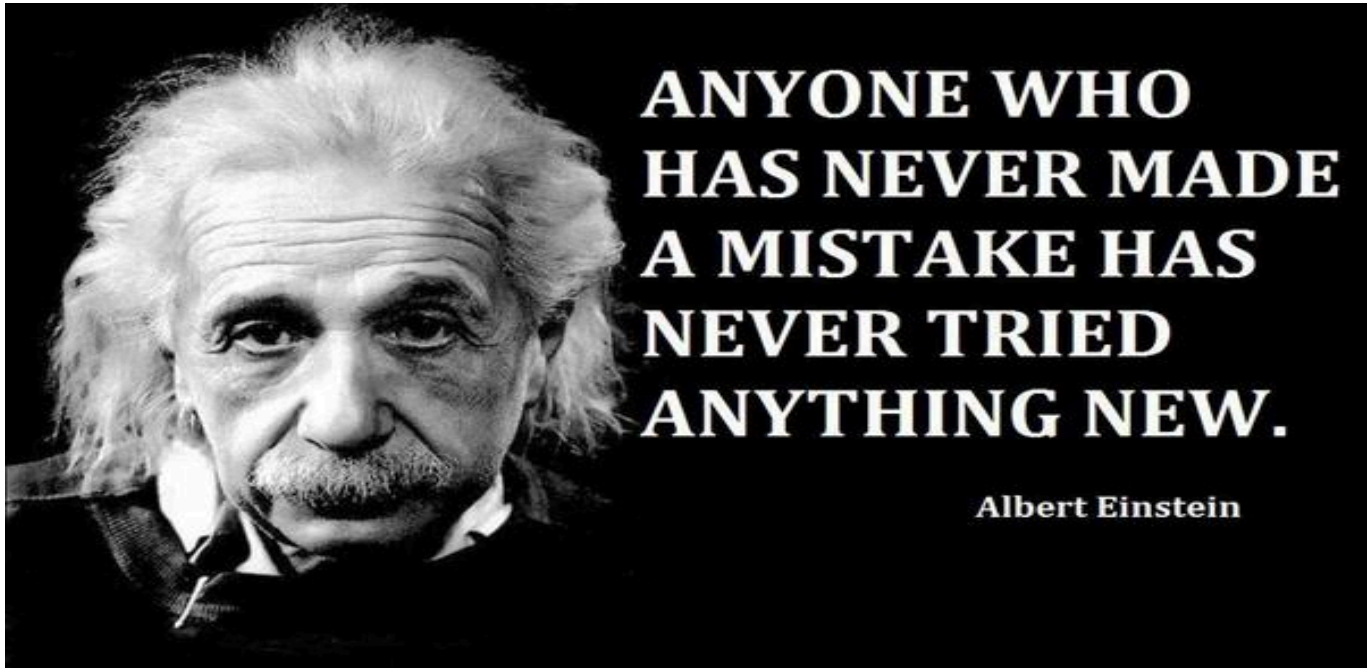
- You can't force enthusiasm / buy-in
- It's challenging to move people from their comfort zones
- People worry about skills, delivery and their future careers
- We needed to listen, empathise, address concerns and be positive

Adoption Requires a Journey

Adoption Requires Balance

Give space to learn but be close enough to step in and help

I can Guarantee that mistakes will Happen



Failures have great value if you learn from them

Widening Involvement – Cash Out

- A very difficult problem
- For many millions of open bets
- Calculate a Cash Out value given the current state of the market
- At peak times markets move quickly (100K changes per second)
- Must be low Latency
- Must support many sports and bet types
- Must work intuitively, have a good flow and excite our customers

Widening Involvement – What happened?

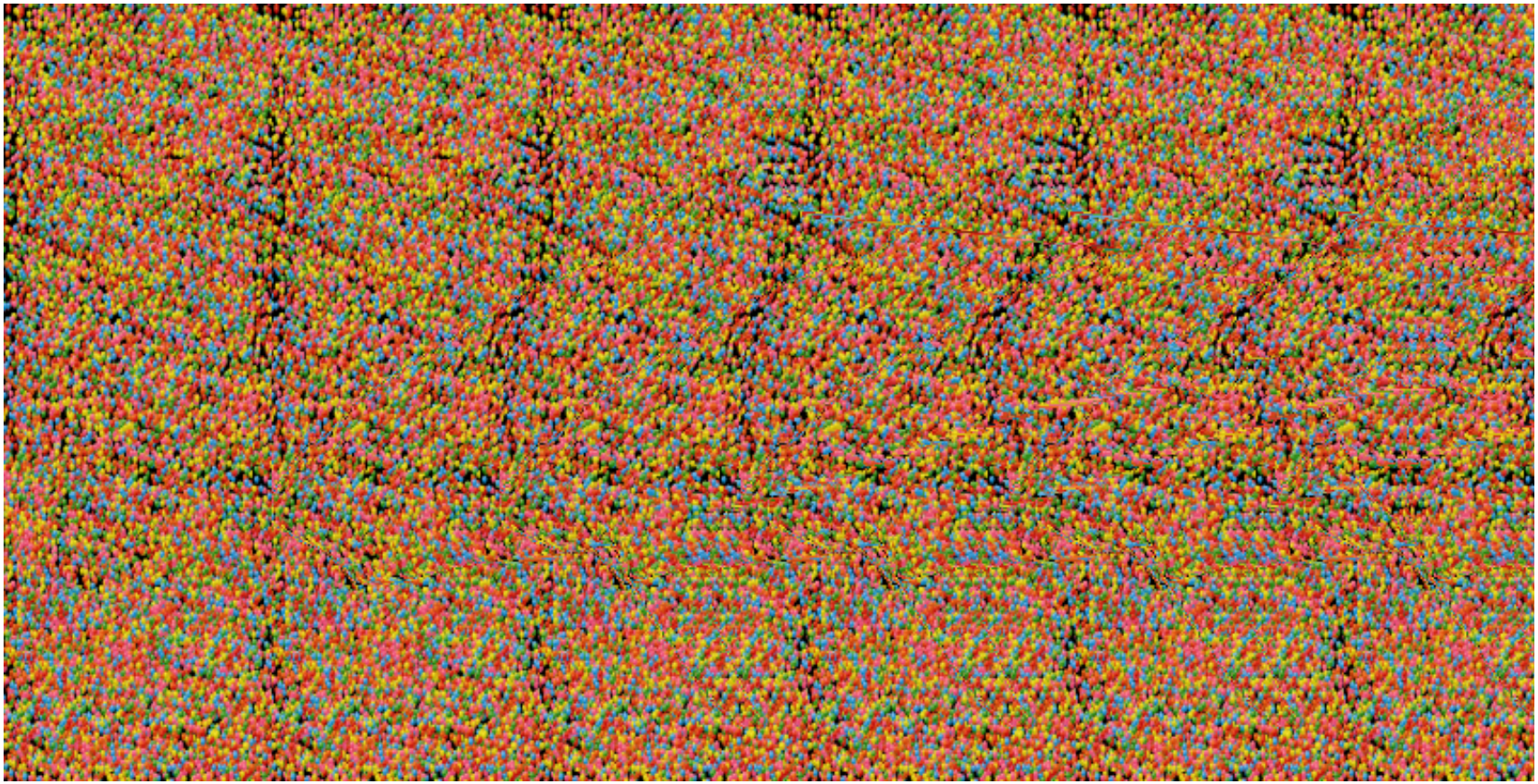
- R&D built a POC that showed the suitability of Erlang
- The initial performance figures looked encouraging
- Our strategy – Help people to understand the benefits
  - We put significant effort into getting everyone to fully understood the problem
  - We gave space to learn
  - We prepared presentations about the perceived benefits
  - We organised training and wrote guides
  - We made ourselves available
  - Above all we tried to get people “hands on”
  - We started people on their learning journeys

Initially Erlang was disliked

There was significant resistance!

Why?

Erlang – Takes Time to Learn



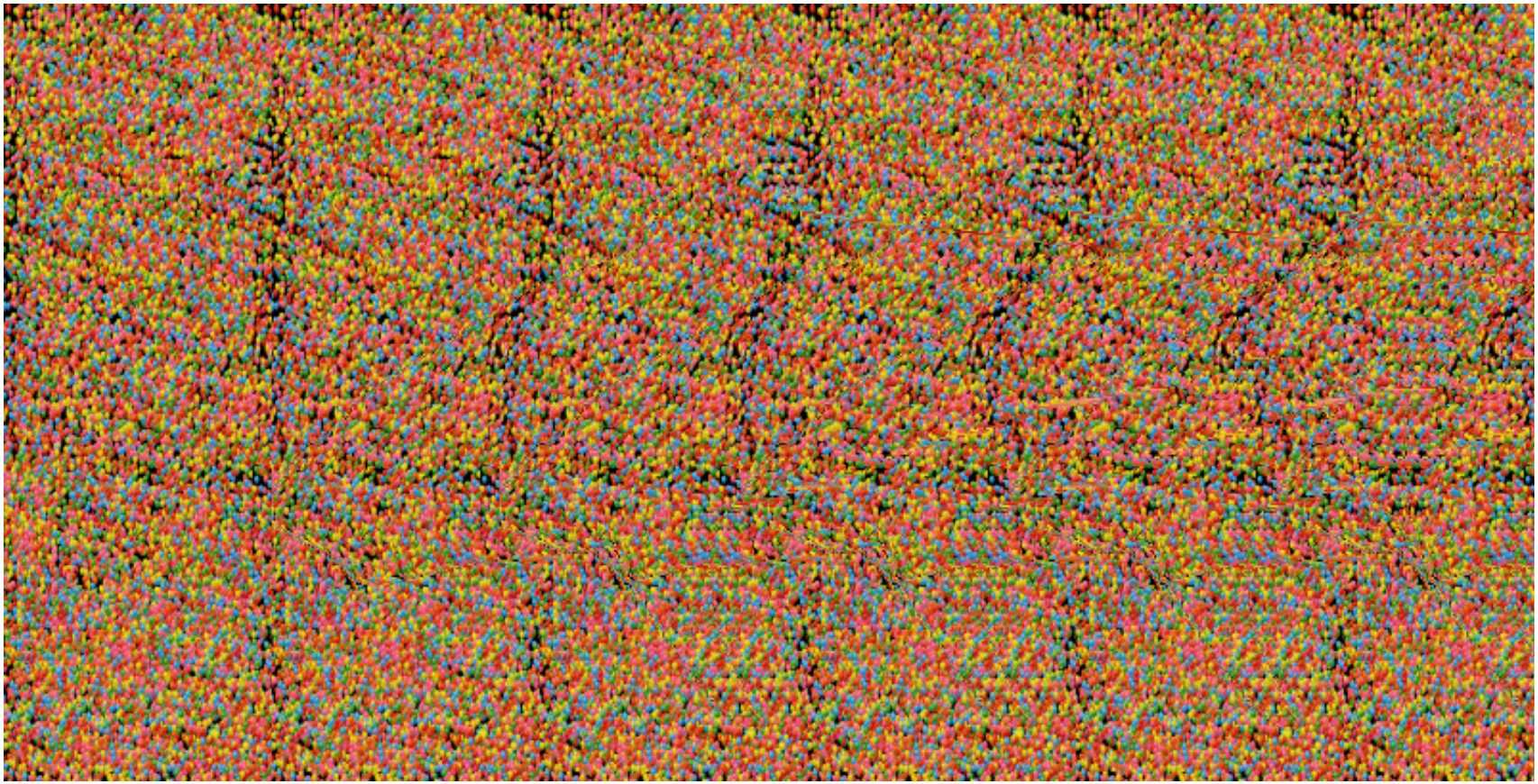
bet365

There was significant resistance - Why?

- We chose a really hard problem
- This was a huge paradigm and language shift
- Our Developers were new to Functional but highly proficient with OO
- Learning was getting in the way of delivery
- However strategically we believed that this was the right thing to do
- So we persevered…..

Widening Involvement – A Few Weeks Later

Look it's a Shark!



bet365

Widening Involvement – A Few Weeks later

Once we got our people engaged

- They began to appreciate the subtleties
- They saw that complex concurrency problems could be solved with elegant simplicity
- They saw new ways to deal with errors and failure
- They realised that they could attack more challenging problems
- They began to see the benefits and understand the rationale for change

Widening Involvement – Many weeks later

Many of our developers started to love Erlang.
Job well done (so we thought….)

Widening Involvement – Many Months later

“Erlang I’m flying. I feel free to solve any problem in the world!”

bet365

Widening Involvement – Many Months later

- One level of complexity can be replaced with another
- We were really into Erlang. We channelled our enthusiasm beyond the core problem
- We came up with some really good solutions for the wrong problems
- We lost time and our design became unnecessarily complex
- That said we got a new product “out of the door” and on the whole it worked well
- Our mistake - We invested too much energy into building love for Erlang, with limited thought on how to manage success

We needed to find a path to a more mature relationship with Functional Programming

Stage Three Mainstream – Business as Usual

bet365

Business as Usual

- With the foundations in place. We began a period of intense research into Distributed Systems
- We moved on to more detailed and difficult problems such as:
  - Synchronisation free computing
  - Eventual Consistency
  - CRDTs
- A number of these systems are now live
- Many more are in the pipeline

So what were the Side Effects?

We Found our Solid Foundations

The Side Effects

- We opened minds
- We increased the possibilities
- We found that learning motivates
- We now think about using the right tool for the right job
- We have shown that we can adopt new things
- We made it easier to solve hard problems
- We gained the confidence needed to attack seriously difficult ones

bet365

The Side Effects

- Learning Functional Programming is good preparation for dealing with the multi-headed Hydra of Distributed Computing
- We have available and responsive systems that continue to keep our customers happy
- We have a pipeline of innovation that ensures this continues
- We have significantly less downtime

Maximising The Side Effects

- Make sure that you have real reasons for change
- Validate and communicate them all the time
- People will want to know “Why – is this worth the pain?”
- Have the “complexity” debate
- Involve , support, coach and mentor
- Let it be known that failure is a distinct possibility, and that it’s ok
- If you get it right the first time you are probably doing it wrong

Maximising The Side Effects

- Recognise that not everyone is going to be happy
- Go the extra mile to get people on-board
- Make it fun. Humour can go a long way
- Adoption will take time and there will be difficult days
- Persevere, be positive and think long term

bet365

What Would I Change ?

- I wish that I'd tried this sooner
- We thought that the adoption would be an uphill struggle
- Some people surprised us and I wish we'd been better prepared
- There were some excellent solutions to the wrong problems
- We could have done a better job at explaining the positive long term career implications of learning Functional Programming
- Being a pragmatic problem solver trumps X years of blah any day

Conclusions

- There are numerous side effects
- Some were easier to measure than others
- From our perspective this was a very worthwhile endeavour
- It opened the floodgates to a series of big improvements
- It formed the basis of our new platform

But Most Importantly

We Grew as People

We Grew as Teams

We Grew as an Organisation

So get started on your Functional Journey tomorrow and see how you Grow

bet365

goto; conference

Please

Remember to rate session

Thank you!

Follow us on Twitter @GOTOber

www.gotober.com