THE 'BUSINESS' BEHIND MICROSERVICES:

(LET'S NOT REPEAT THE MISTAKES OF SOA)

ORGANISATIONAL, ARCHITECTURAL AND OPERATIONAL CHALLENGES

Daniel Bryant

@danielbryantuk

www.opencredo.com

OpenCredo

delivering emerging technology today

Business and People == “Fluffiness”?

“Micro-fluffiness?”

Key Messages

The transition to implementing ‘microservices’ cannot be completed successfully with technology alone

- Think organisation, architecture, operations -

We require a holistic viewpoint for change, but we often resist. Challenge yourself, your team and your company

Who Am I?

- Chief Scientist at OpenCredo
  - ✓ Digital/technical transformations
  - ✓ Java, Golang, CI/CD, DevOps
  - ✓ Microservices, cloud, containers
  - ✓ **Maintainer of muservicesweekly.com**

- London Java Community Associate
- Adopt OpenJDK and JSR
- InfoQ Editor, DZone MVB, Voxxed

LJC London Java Community

Java Community Process

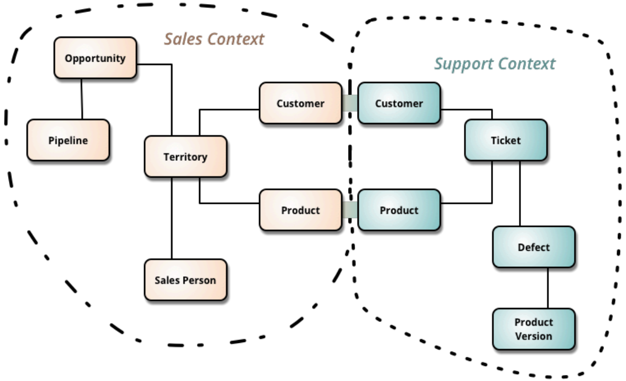
InfoQ

SHARE THE KNOWLEDGE

DZone MVB MOST VALUABLE BLOGGER

OpenCredo delivering emerging technology today

So, What are microservices?



"Loosely coupled service oriented architecture with bounded contexts"

Adrian Cockcroft

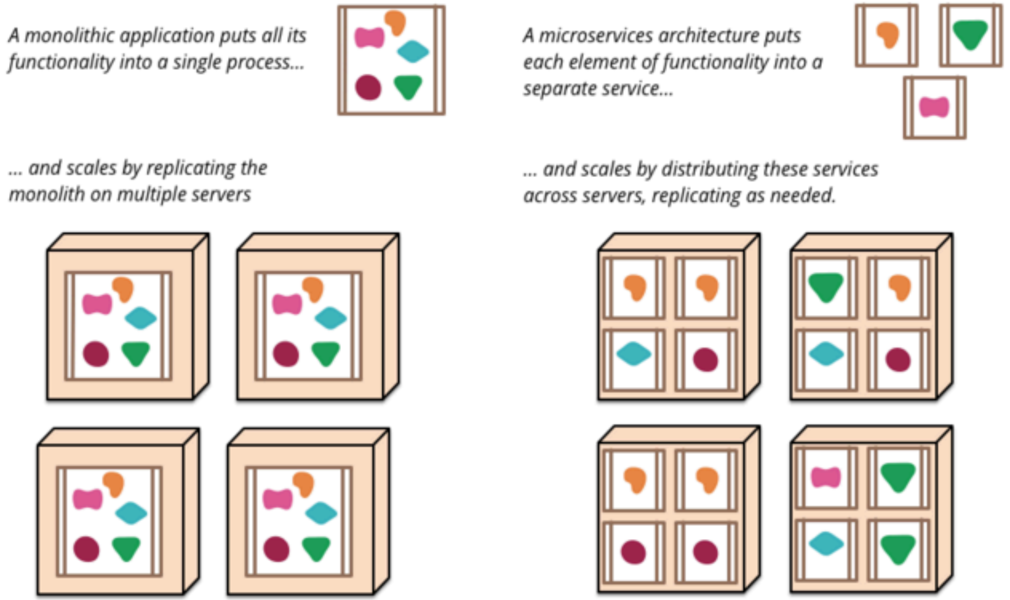
"Applications that fit in your head"

James Lewis

So, What are microservices?

- Architectural style - build services around biz capability
- Single app composed of multiple services
- Services are individual processes, individually deployable
- Lightweight external communication (e.g. REST over HTTP)
- Potentially polyglot Language and Data stores
- Minimum centralised management

Monolith vs microservices



A monolithic application puts all its functionality into a single process...

... and scales by replicating the monolith on multiple servers

A microservices architecture puts each element of functionality into a separate service...

... and scales by distributing these services across servers, replicating as needed.

Figure 1: Monoliths and Microservices

martinfowler.com/articles/microservices.html

So, this appears familiar (Corba, SOA, etc)…

Oh, Yes...

But this time it's Different...

Hopefully...

The Perfect Storm?

- Collaborative processes (requirement for speed)
  - Agile, lean and Devops
- Programmable infrastructure
  - Config management, Cloud and containers
- Open source
  - Download, consume and contribute

Over the last few years of working with clients at OpenCredo
We have come to value:

products over projects
small inter-discipline teams over large homogeneous teams
multiple cohesive services over highly coupled monoliths
goal-driven technical leadership over command and control
automated continuous deployment over manual big bangs
individuals and interactions over processes and tools

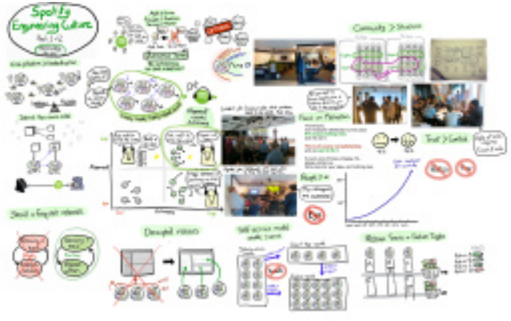
Organisation, Architecture, (Dev)Operations

ORGANISATION: CONWAY WAS TELLING THE TRUTH…

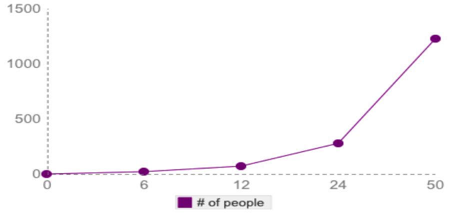
Conway’s Law

Cross-functional Teams

- Spotify (bit.ly/1C46ZKo)
  - Culture



- Amazon (bit.ly/1F3Dgkm)
  - Communication

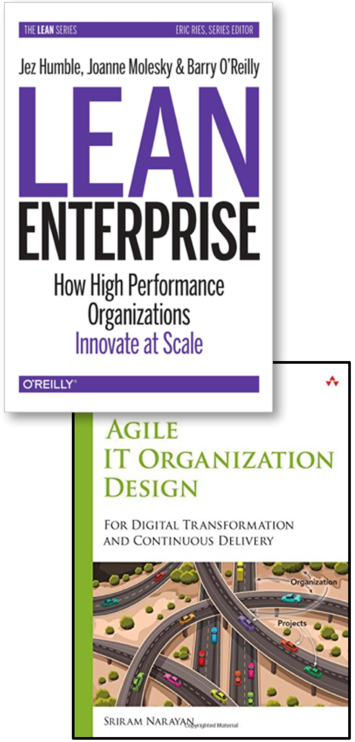


- Gilt (gi.lt/1rgyWvO)
  - Strategic alignment

Team 1
Team 2
Team 3
Repo
Initiative

Portfolio Planning

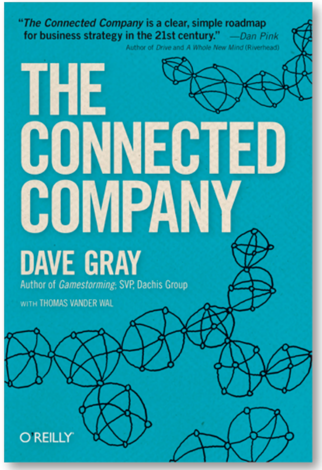
- Budget for capacity (#NoProjects)
- Predictable software development?
  - Chase verifiable value instead (KPIs)
- Business cases secure funding…
  - Switch to “value-driven” projects
  - “What problems can we solve by X?”


THE LEAN SERIES
ERIC RIES, SERIES EDITOR
Jez Humble, Joanne Molesky & Barry O'Reilly
LEAN ENTERPRISE
How High Performance Organizations Innovate at Scale
O'REILLY
AGILE IT ORGANIZATION DESIGN
FOR DIGITAL TRANSFORMATION AND CONTINUOUS DELIVERY
Organization
Projects
SRIRAM NARAYAN

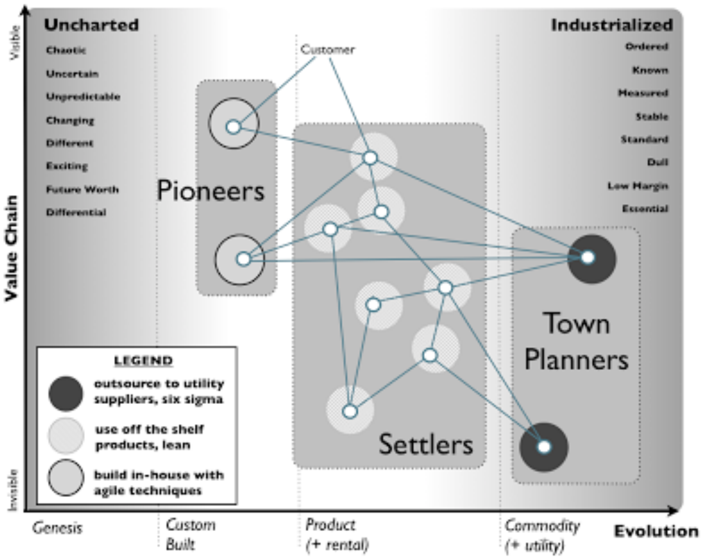
A Word of Caution

- Divided Companies
  - Traditional ‘enterprise’ organisation
  - Command and control, specialised, division of labour
  - Predictable in stable environments
  - Six sigma, ESBs, and classical SOA

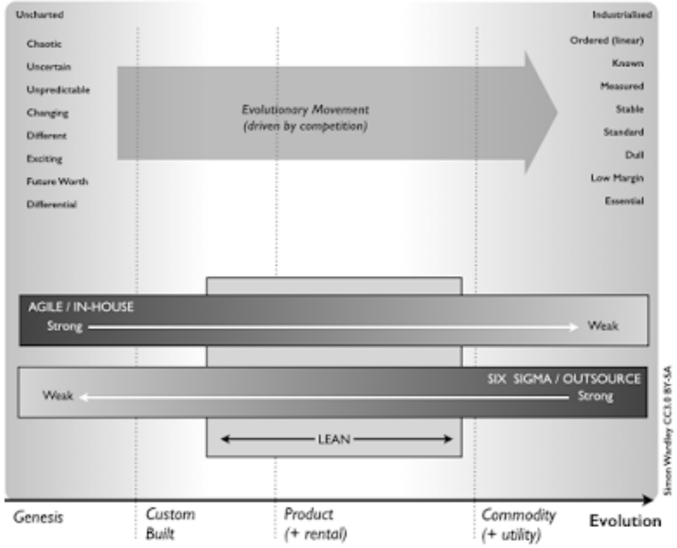
- Connected Companies
  - Startups and forward-thinkers
  - Autonomous, fractal, service-focused
  - Adaptive in uncertain environments
  - Agile/lean, REST, and microservices


"The Connected Company is a clear, simple roadmap for business strategy in the 21st century." —Dan Pink
Author of Drive and A Whole New Mind (Riverhead)
THE CONNECTED COMPANY
DAVE GRAY
Author of Gamestorming; SVP, Dachis Group
WITH THOMAS VANDER WAL
O'REILLY

Bi-Modal / Tri-Modal IT



Simon Wardley blogs extensively in this space…



blog.gardeviance.org/2015/06/why-agile-lean-and-six-sigma-must-die.html

Organisational Values

small inter-discipline teams over large homogeneous teams

products over projects

user impact over customer requirements

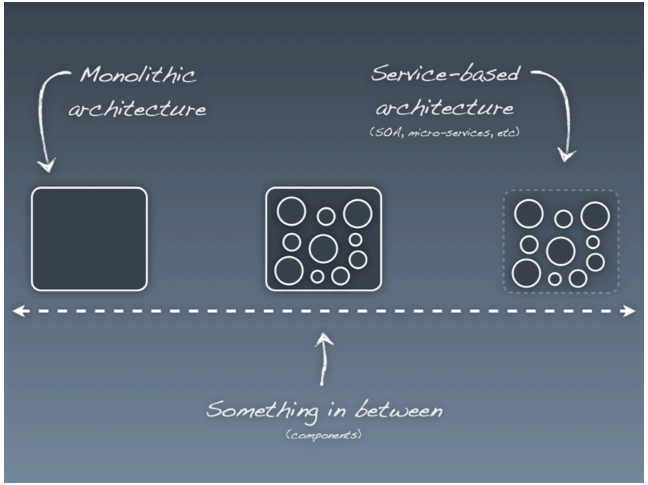
creating verifiable incremental value over adding features

incremental transformation over big-bang “re-org”

ARCHITECTURE: THINK MICRO, THINK MACRO, THINK LEADERSHIP

Architectural/Design Skills

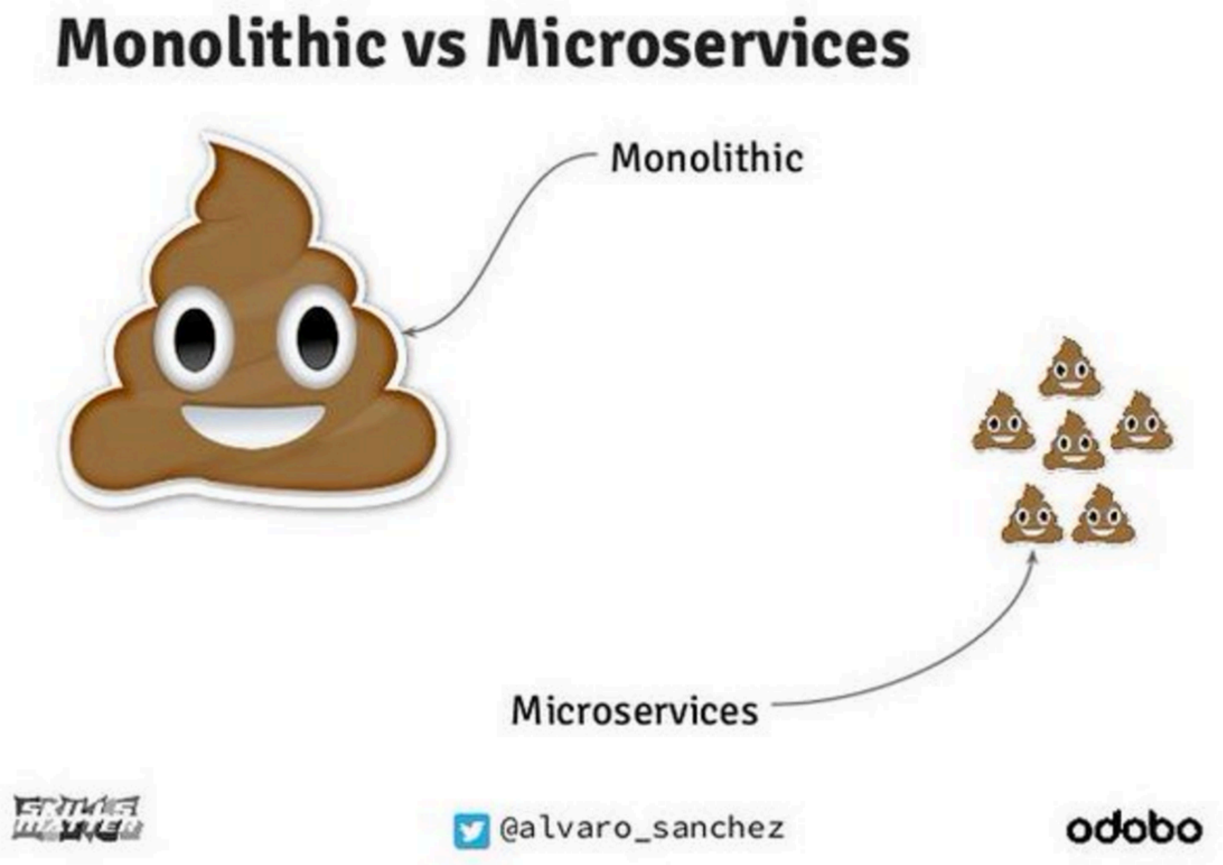
"If you can't build a [well-structured] monolith, what makes you think microservices are the answer?"



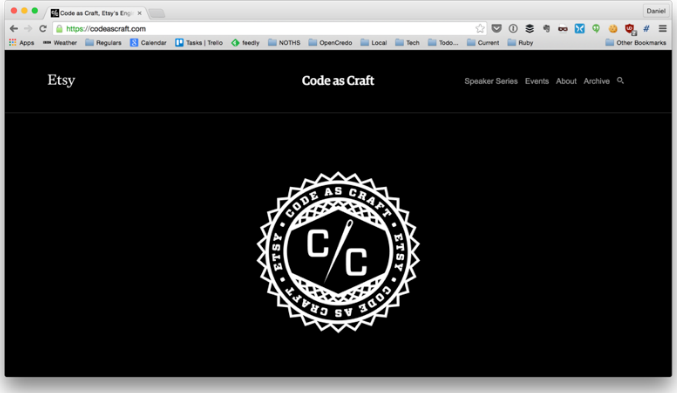
Simon Brown

(bit.ly/1n7D0vp)

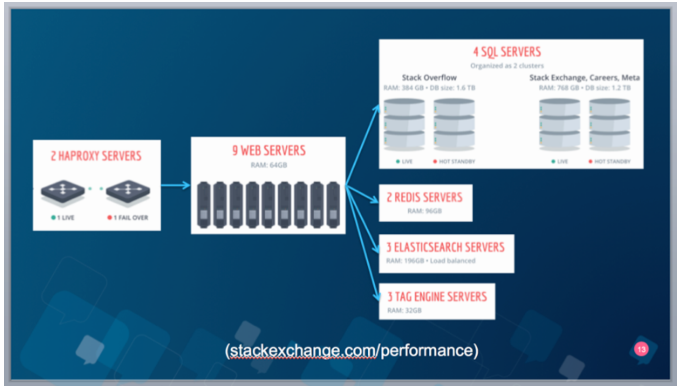
Microservices in a nutshell



Architectural/Design Skills



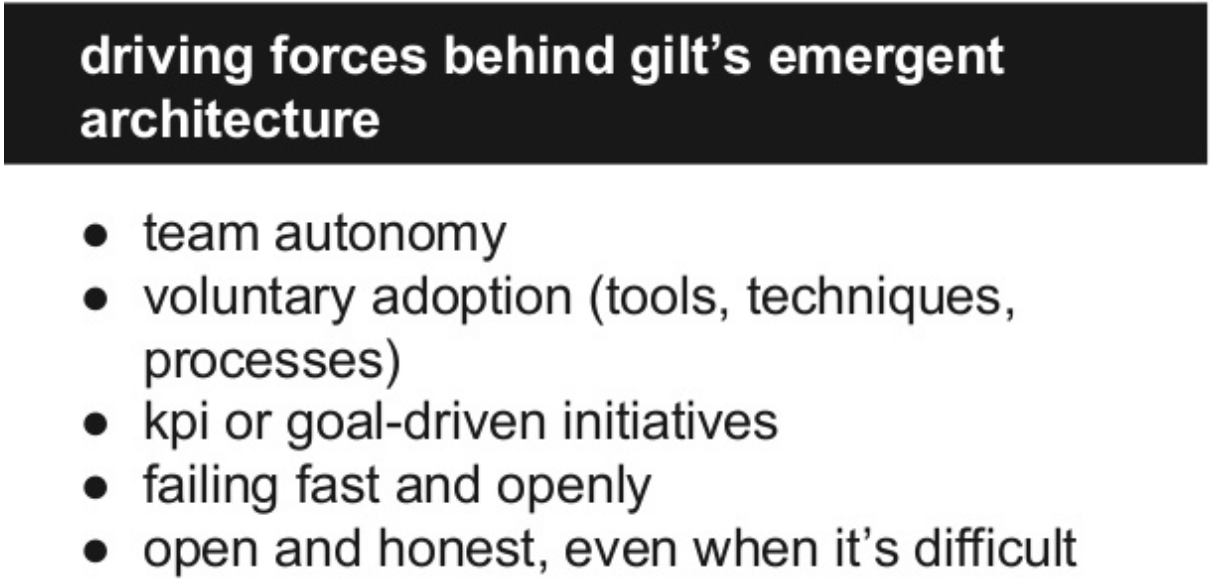
www.etsy.com



(stackexchange.com/performance)

stackexchange.com

gilt.com (Are Awesome!)



driving forces behind gilt’s emergent architecture

- team autonomy
- voluntary adoption (tools, techniques, processes)
- kpi or goal-driven initiatives
- failing fast and openly
- open and honest, even when it’s difficult

www.slideshare.net/trenaman/javaone-2015-scaling-micro-services-at-gilt

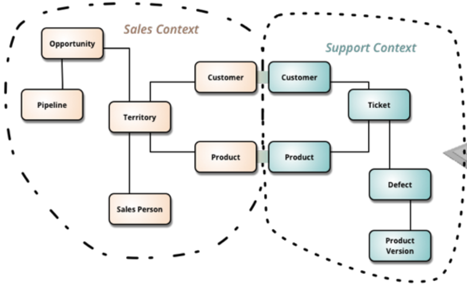
Architecture, Minus the Ivory Towers

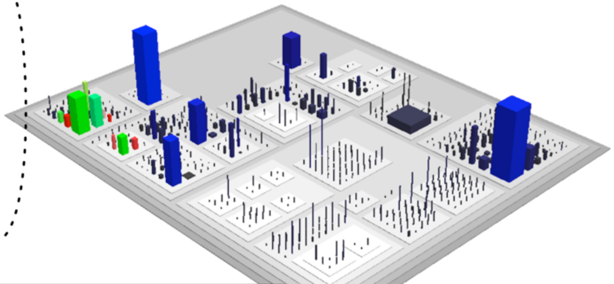
- Technical leadership (bit.ly/1EUwpLl)
- Communication (bit.ly/1Ia3u8o)
- Empathy
- ‘Just enough’ up front design

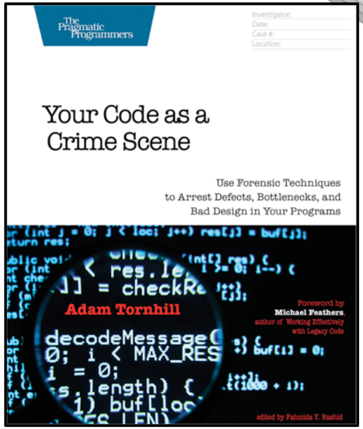
coding architecture
Software Architecture for Developers
Technical leadership by coding, coaching, collaboration, architecture sketching and just enough up front design
Simon Brown
Talking with Tech Leads
From Novices to Practitioners
Patrick Kua
Foreword by Jim Webber

Migrating Architecture

- Find seams
  bit.ly/1SwHryU
- Measure toxicity of code
- Standardise what matters (glue/infra)
- Patterns (bit.ly/1GRch2v)
  - Big bang
  - Change via copy/extraction
  - Strangle







The Pragmatic Programmers

Investigator:
Date:
Case #:
Location:

Your Code as a Crime Scene

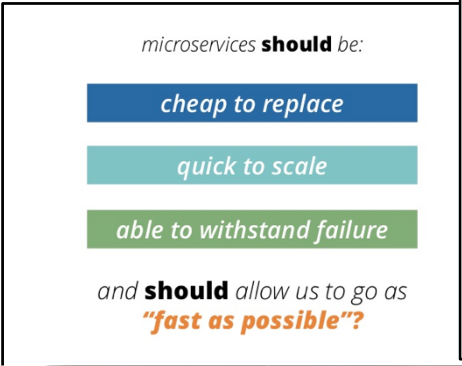
Use Forensic Techniques
to Arrest Defects, Bottlenecks, and
Bad Design in Your Programs

Adam Tornhill

Foreword by
Michael Feathers,
author of Working Effectively
with Legacy Code

edited by Fahmida Y. Rashid

Design for “Replacability”



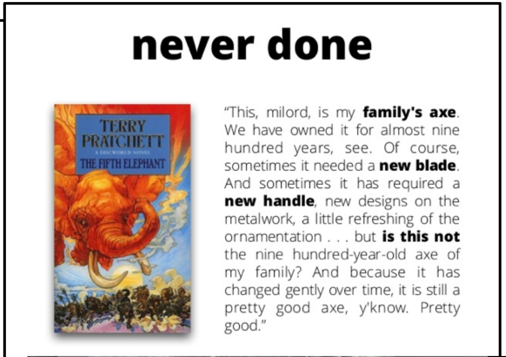
microservices **should** *be:*

cheap to replace

quick to scale

able to withstand failure

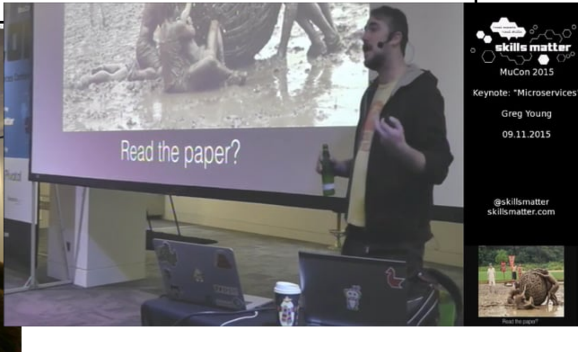
and **should** *allow us to go as* ***“fast as possible”?***



never done

TERRY PRATCHETT
A DISCWORLD NOVEL
THE FIFTH ELEPHANT

“This, milord, is my **family's axe**. We have owned it for almost nine hundred years, see. Of course, sometimes it needed a **new blade**. And sometimes it has required a **new handle**, new designs on the metalwork, a little refreshing of the ornamentation . . . but **is this not** the nine hundred-year-old axe of my family? And because it has changed gently over time, it is still a pretty good axe, y'know. Pretty good.”



James Lewis
bit.ly/1Qy4g2d

Greg Young
bit.ly/1l0ir61

Caution: Shiny Technology Ahead!

- Plenty of ‘microservice’ technology

- Emerging platforms and frameworks
  - Principles and practice should drive tooling

- How to choose?

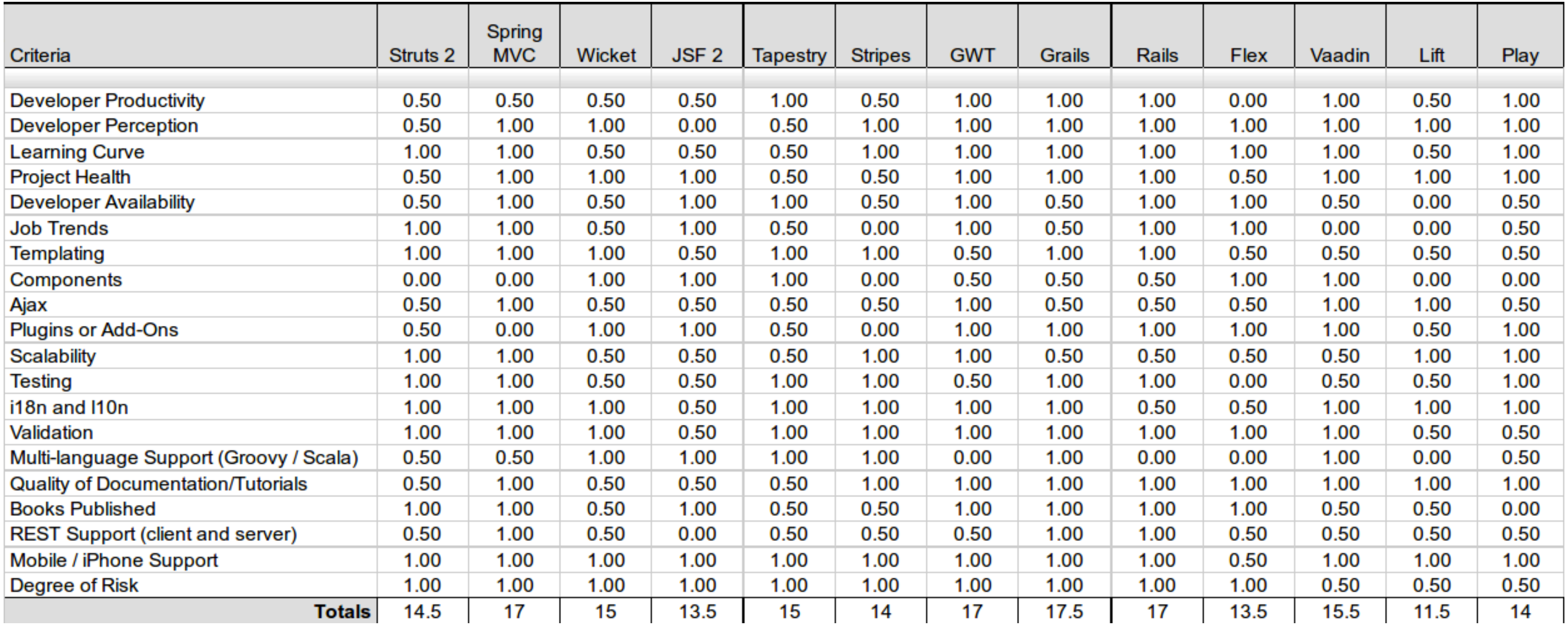
Key Skill: Evaluation

"I will postpone using this shiny new framework until my peers have validated the proposed benefits with rigorous scientific experiments"

- Said by no programmer

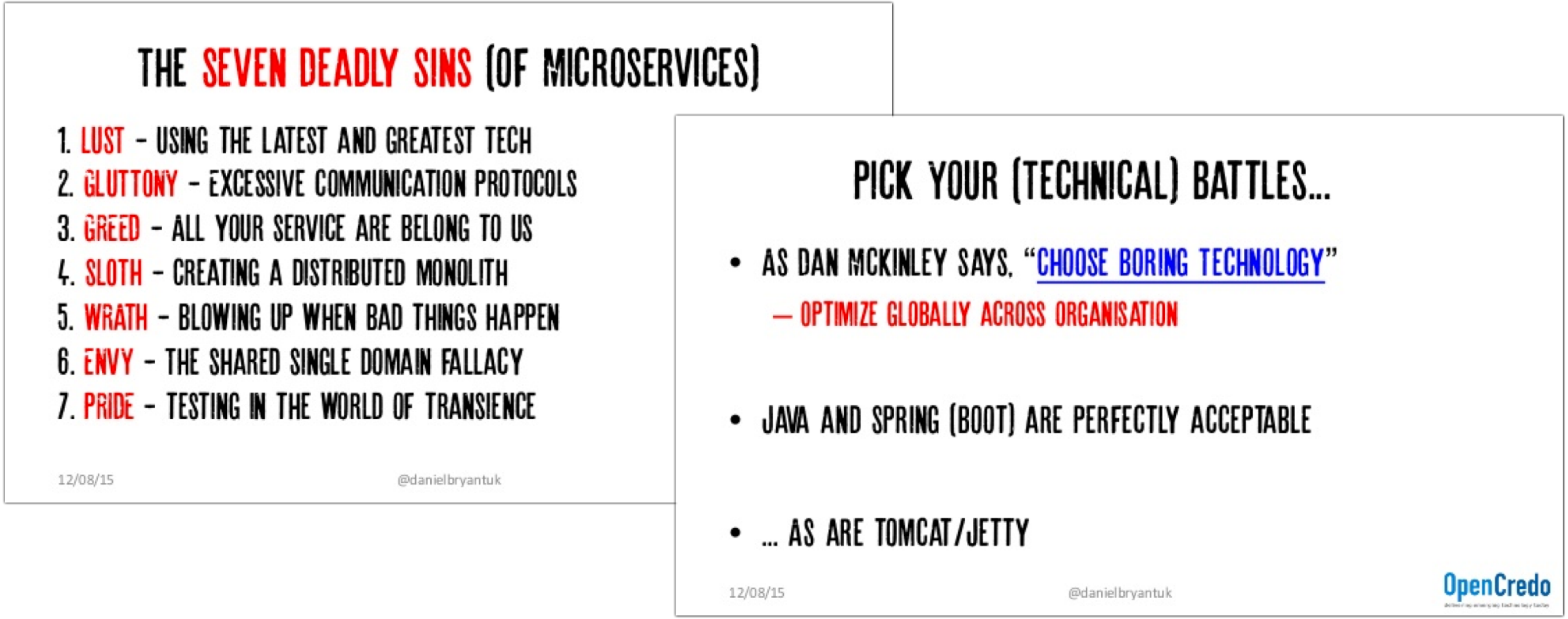
…ever

Matt Raible's Comparison Matrix



Criteria	Struts 2	Spring MVC	Wicket	JSF 2	Tapestry	Stripes	GWT	Grails	Rails	Flex	Vaadin	Lift	Play
Developer Productivity	0.50	0.50	0.50	0.50	1.00	0.50	1.00	1.00	1.00	0.00	1.00	0.50	1.00
Developer Perception	0.50	1.00	1.00	0.00	0.50	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Learning Curve	1.00	1.00	0.50	0.50	0.50	1.00	1.00	1.00	1.00	1.00	1.00	0.50	1.00
Project Health	0.50	1.00	1.00	1.00	0.50	0.50	1.00	1.00	1.00	0.50	1.00	1.00	1.00
Developer Availability	0.50	1.00	0.50	1.00	1.00	0.50	1.00	0.50	1.00	1.00	0.50	0.00	0.50
Job Trends	1.00	1.00	0.50	1.00	0.50	0.00	1.00	0.50	1.00	1.00	0.00	0.00	0.50
Templating	1.00	1.00	1.00	0.50	1.00	1.00	0.50	1.00	1.00	0.50	0.50	0.50	0.50
Components	0.00	0.00	1.00	1.00	1.00	0.00	0.50	0.50	0.50	1.00	1.00	0.00	0.00
Ajax	0.50	1.00	0.50	0.50	0.50	0.50	1.00	0.50	0.50	0.50	1.00	1.00	0.50
Plugins or Add-Ons	0.50	0.00	1.00	1.00	0.50	0.00	1.00	1.00	1.00	1.00	1.00	0.50	1.00
Scalability	1.00	1.00	0.50	0.50	0.50	1.00	1.00	0.50	0.50	0.50	0.50	1.00	1.00
Testing	1.00	1.00	0.50	0.50	1.00	1.00	0.50	1.00	1.00	0.00	0.50	0.50	1.00
i18n and l10n	1.00	1.00	1.00	0.50	1.00	1.00	1.00	1.00	0.50	0.50	1.00	1.00	1.00
Validation	1.00	1.00	1.00	0.50	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	0.50
Multi-language Support (Groovy / Scala)	0.50	0.50	1.00	1.00	1.00	1.00	0.00	1.00	0.00	0.00	1.00	0.00	0.50
Quality of Documentation/Tutorials	0.50	1.00	0.50	0.50	0.50	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Books Published	1.00	1.00	0.50	1.00	0.50	0.50	1.00	1.00	1.00	1.00	0.50	0.50	0.00
REST Support (client and server)	0.50	1.00	0.50	0.00	0.50	0.50	0.50	1.00	1.00	0.50	0.50	0.50	0.50
Mobile / iPhone Support	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	1.00	1.00	1.00
Degree of Risk	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.50	0.50	0.50
Totals	14.5	17	15	13.5	15	14	17	17.5	17	13.5	15.5	11.5	14

Choose Boring Technology



THE SEVEN DEADLY SINS (OF MICROSERVICES)

1. LUST - USING THE LATEST AND GREATEST TECH
2. GLUTTONY - EXCESSIVE COMMUNICATION PROTOCOLS
3. GREED - ALL YOUR SERVICE ARE BELONG TO US
4. SLOTH - CREATING A DISTRIBUTED MONOLITH
5. WRATH - BLOWING UP WHEN BAD THINGS HAPPEN
6. ENVY - THE SHARED SINGLE DOMAIN FALLACY
7. PRIDE - TESTING IN THE WORLD OF TRANSIENCE

12/08/15 @danielbryantuk

PICK YOUR (TECHNICAL) BATTLES...

- AS DAN MCKINLEY SAYS, "CHOOSE BORING TECHNOLOGY"
  - OPTIMIZE GLOBALLY ACROSS ORGANISATION
- JAVA AND SPRING (BOOT) ARE PERFECTLY ACCEPTABLE
- ... AS ARE TOMCAT/JETTY

12/08/15 @danielbryantuk OpenCredo

www.opencredo.com/2016/01/08/the-seven-deadly-sins-of-microservices-redux/

Architectural Values

multiple cohesive services over a highly coupled monolith

trust and empathy over command and control

autonomous goal-driven leadership over chaos and anarchy

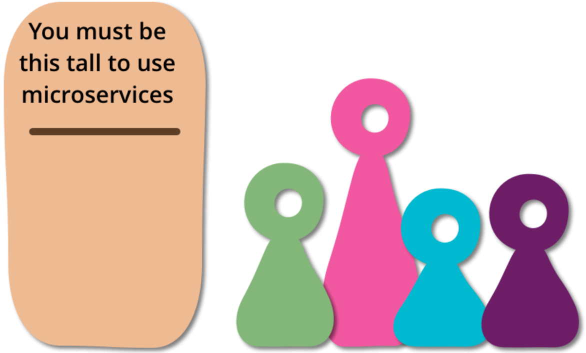
implementing signals and performance indicators over measuring vanity metrics

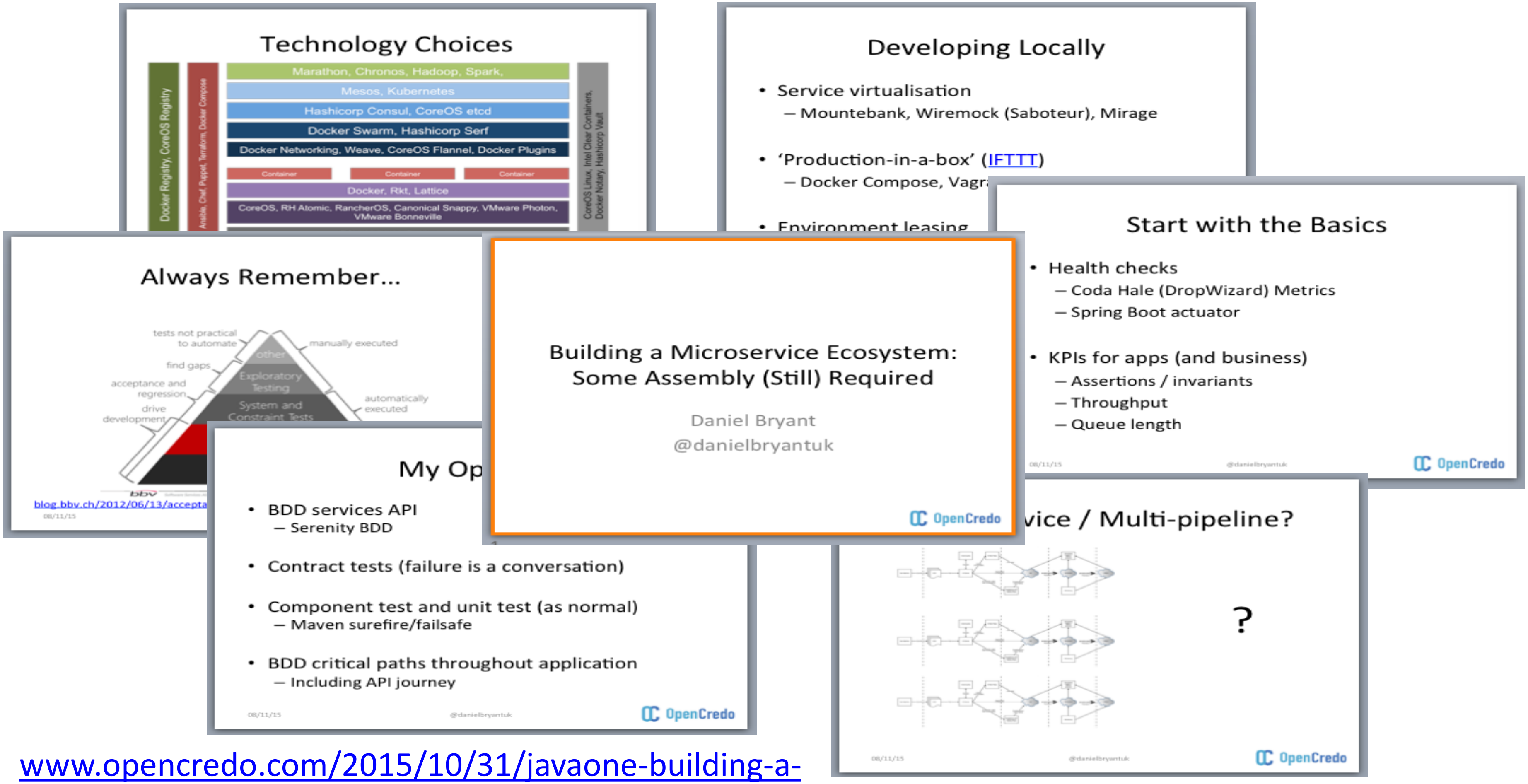
OPERATIONS: AUTOMATE EVERYTHING (EXCEPT THE PEOPLE)

Microservices… Macro Operations

- Microservice Prerequisites (bit.ly/1wIjY58)
  - Rapid provisioning
  - Basic monitoring
  - Rapid application deployment

- In a nutshell…
  - DevOps
  - CI/CD





Technology Choices

Marathon, Chronos, Hadoop, Spark,
Mesos, Kubernetes
Hashicorp Consul, CoreOS etcd
Docker Swarm, Hashicorp Serf
Docker Networking, Weave, CoreOS Flannel, Docker Plugins
Container
Container
Container
Docker, Rkt, Lattice
CoreOS, RH Atomic, RancherOS, Canonical Snappy, VMware Photon, VMware Bonneville
Docker Registry, CoreOS Registry
Ansible, Chef, Puppet, Terraform, Docker Compose
CoreOS Linux, Intel Clear Containers, Docker Notary, Hashicorp Vault

Developing Locally

- Service virtualisation
  - Mountebank, Wiremock (Saboteur), Mirage
- 'Production-in-a-box' (IFTTT)
  - Docker Compose, Vagr
- Environment leasing

Always Remember...

tests not practical to automate
manually executed
other
find gaps
Exploratory Testing
acceptance and regression
System and Constraint Tests
automatically executed
drive development
bbv

blog.bbv.ch/2012/06/13/accepta

08/11/15

Building a Microservice Ecosystem: Some Assembly (Still) Required

Daniel Bryant
@danielbryantuk

OpenCredo

Start with the Basics

- Health checks
  - Coda Hale (DropWizard) Metrics
  - Spring Boot actuator
- KPIs for apps (and business)
  - Assertions / invariants
  - Throughput
  - Queue length

08/11/15 @danielbryantuk OpenCredo

My Op

- BDD services API
  - Serenity BDD
- Contract tests (failure is a conversation)
- Component test and unit test (as normal)
  - Maven surefire/failsafe
- BDD critical paths throughout application
  - Including API journey

08/11/15 @danielbryantuk OpenCredo

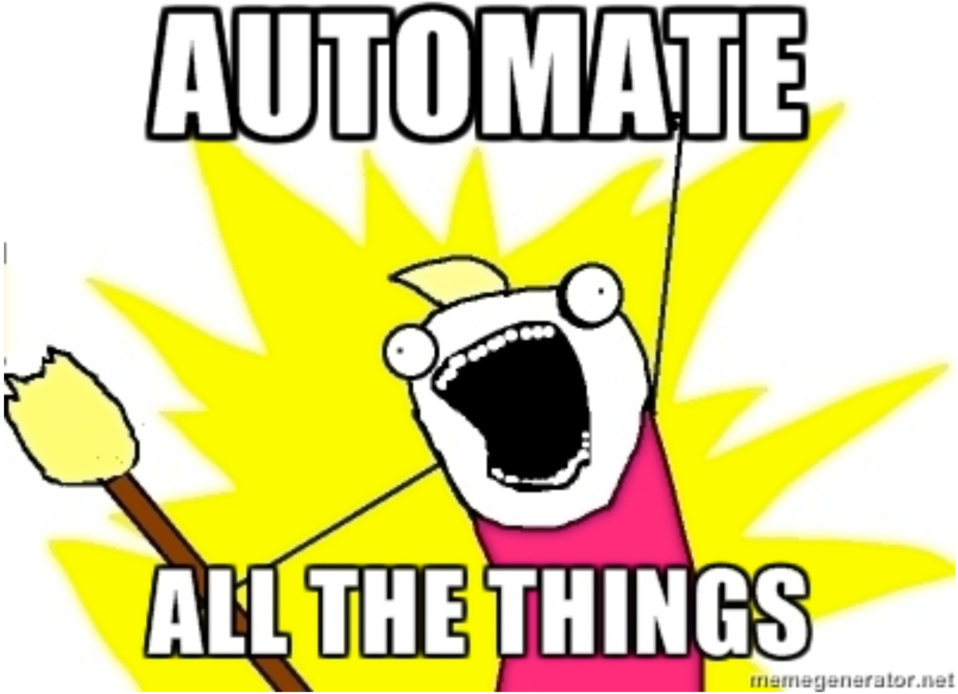
vice / Multi-pipeline?

?

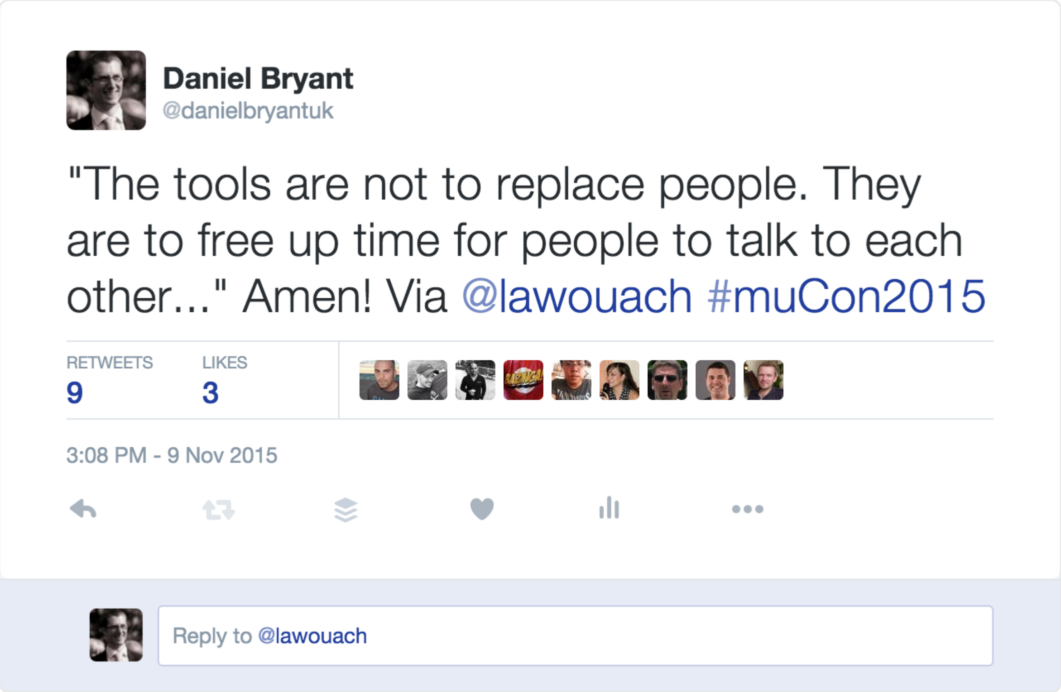
08/11/15 @danielbryantuk OpenCredo

www.opencredo.com/2015/10/31/javaone-building-a-microservice-development-ecosystem-video

In the Beginning…



Remind Others of the Benefits



Daniel Bryant
@danielbryantuk

"The tools are not to replace people. They are to free up time for people to talk to each other..." Amen! Via @lawouach #muCon2015

RETWEETS 9

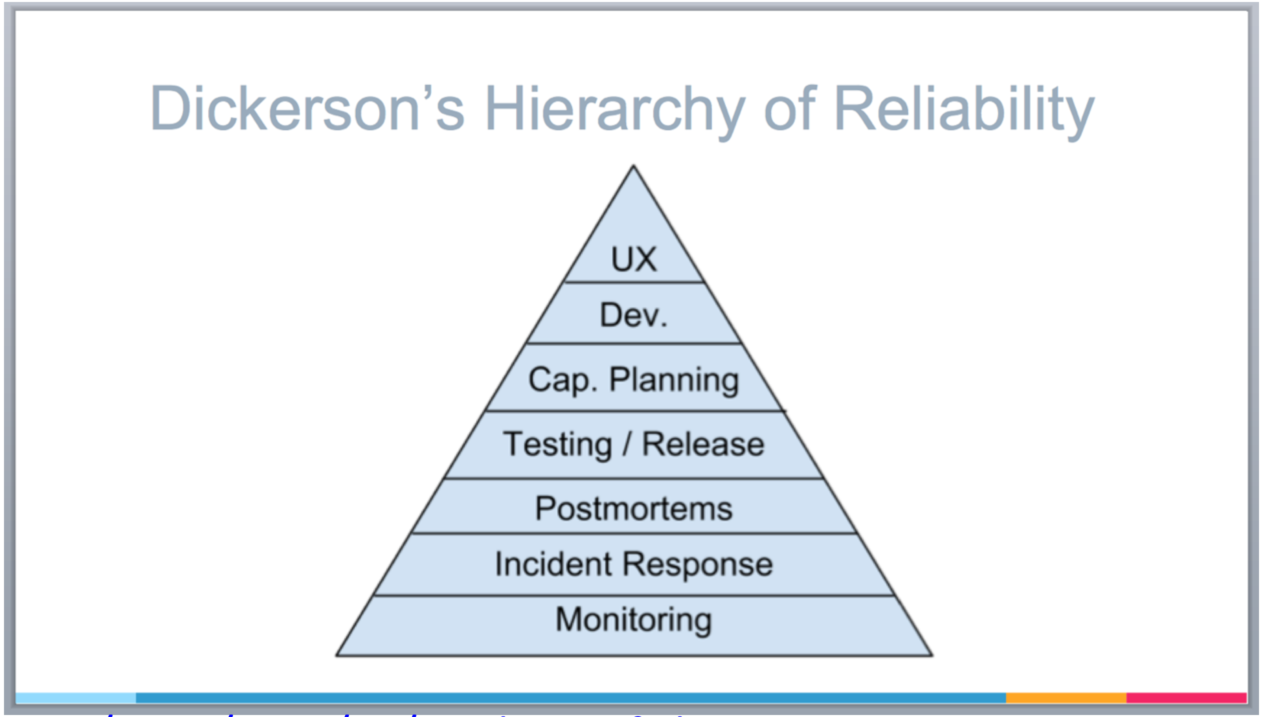
LIKES 3

3:08 PM - 9 Nov 2015

Reply to @lawouach

…and in the Middle (and the ‘End’)

…and in the Middle (and the 'End')

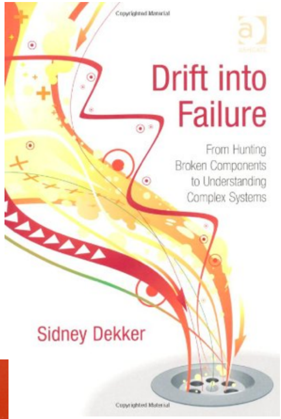


Dickerson's Hierarchy of Reliability

www.infoq.com/news/2015/06/too-big-to-fail

Failing to Prepare, is Preparing to…

- Practice
  - Chaos, Gamedays, DiRT sessions
- Accountability
  - R.A.S.C.I.
- Engage ‘system 2’ thinking


Drift into Failure
From Hunting Broken Components to Understanding Complex Systems
Sidney Dekker

'It has been years since I read a book so powerful and so thought-provoking' MALCOLM GLADWELL
BY THE BESTSELLING AUTHOR OF COMPLICATIONS AND BETTER
THE CHECKLIST MANIFESTO
HOW TO GET THINGS RIGHT
ATUL GAWANDE

Helpful Processes

- BDD user journeys across system (via APIs)
  - …and individual services
  - Don’t forget ‘ilities’ (ZAP, Jmeter)

- Brendan Gregg’s USE methodology
  - check utilization, saturation, and errors.

- Symptom-based Monitoring (Ticketmaster)

Operational Values

designing for failure over implementing extensive redundancy

independent automated continuous deployment over coordinated manual big bang releases

programmable infrastructure over manually-configured snowflakes

Individuals and interactions over processes and bureaucracy

principles and practices over tooling

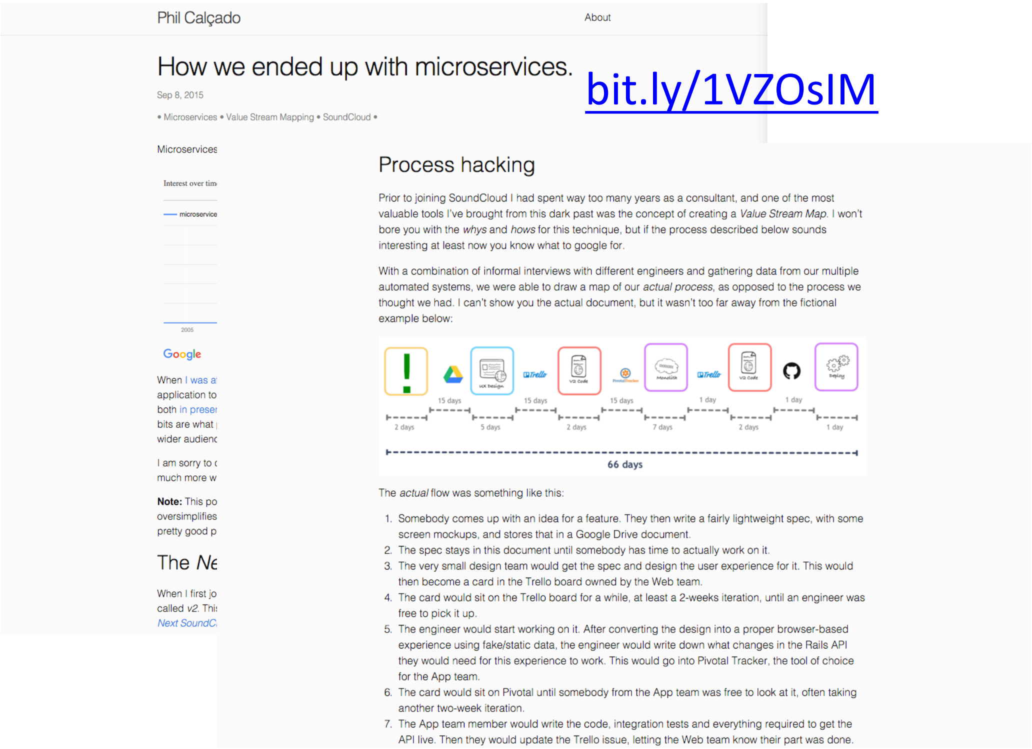
5. CHANGE MANAGEMENT WITHOUT MANAGEMENT DOUBLE-SPEAK

Change Management is Essential

- Fair process (three ‘E’s)
  - Engagement
  - Explanation
  - Expectation
- Leading change
  - Transformation is a process
  - Visualise the issues and goals
  - Communicate, plan, evaluate, learn

HBR'S 10 MUST READS
FEATURING "Leading Change" By John P. Kotter
On Change Management
If you read nothing else on change management, read these definitive articles from Harvard Business Review.
HBR'S 10 MUST READS
The Essentials
An introduction to the most enduring ideas on management from Harvard Business Review.

(Visualising) Flow



Phil Calçado About

How we ended up with microservices.

Sep 8, 2015

• Microservices • Value Stream Mapping • SoundCloud •

Microservices

When I was a
application to
both in preser
bits are what
wider audienc

I am sorry to c
much more w

Note: This po
oversimplifies
pretty good p

The *Ne*

When I first jo
called *v2*. Thi
Next SoundC

Process hacking

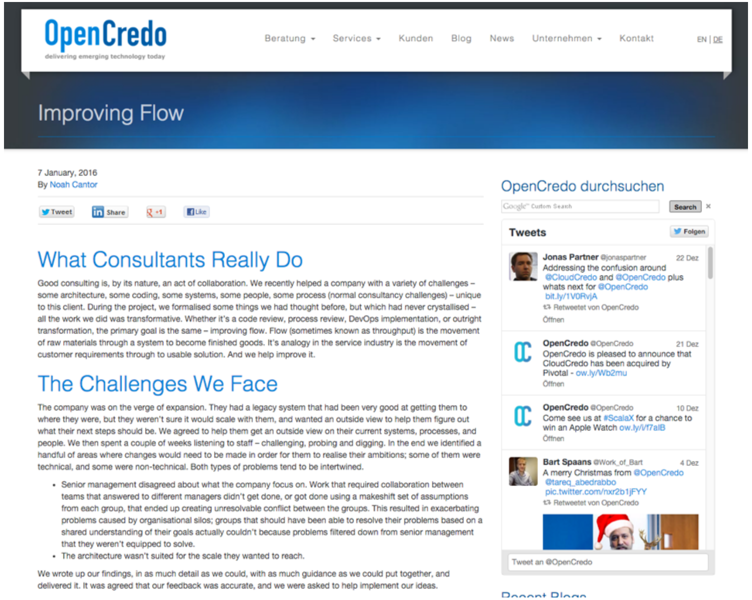
Prior to joining SoundCloud I had spent way too many years as a consultant, and one of the most valuable tools I've brought from this dark past was the concept of creating a *Value Stream Map*. I won't bore you with the *whys* and *hows* for this technique, but if the process described below sounds interesting at least now you know what to google for.

With a combination of informal interviews with different engineers and gathering data from our multiple automated systems, we were able to draw a map of our *actual process*, as opposed to the process we thought we had. I can't show you the actual document, but it wasn't too far away from the fictional example below:

The *actual* flow was something like this:

1. Somebody comes up with an idea for a feature. They then write a fairly lightweight spec, with some screen mockups, and stores that in a Google Drive document.
2. The spec stays in this document until somebody has time to actually work on it.
3. The very small design team would get the spec and design the user experience for it. This would then become a card in the Trello board owned by the Web team.
4. The card would sit on the Trello board for a while, at least a 2-weeks iteration, until an engineer was free to pick it up.
5. The engineer would start working on it. After converting the design into a proper browser-based experience using fake/static data, the engineer would write down what changes in the Rails API they would need for this experience to work. This would go into Pivotal Tracker, the tool of choice for the App team.
6. The card would sit on Pivotal until somebody from the App team was free to look at it, often taking another two-week iteration.
7. The App team member would write the code, integration tests and everything required to get the API live. Then they would update the Trello issue, letting the Web team know their part was done.

bit.ly/1VZOsIM



Improving Flow

7 January, 2016
By Noah Cantor

What Consultants Really Do

Good consulting is, by its nature, an act of collaboration. We recently helped a company with a variety of challenges – some architecture, some coding, some systems, some people, some process (normal consultancy challenges) – unique to this client. During the project, we formalised some things we had thought before, but which had never crystallised – all the work we did was transformative. Whether it's a code review, process review, DevOps implementation, or outright transformation, the primary goal is the same – improving flow. Flow (sometimes known as throughput) is the movement of raw materials through a system to become finished goods. It's analogy in the service industry is the movement of customer requirements through to usable solution. And we help improve it.

The Challenges We Face

The company was on the verge of expansion. They had a legacy system that had been very good at getting them to where they were, but they weren't sure it would scale with them, and wanted an outside view to help them figure out what their next steps should be. We agreed to help them get an outside view on their current systems, processes, and people. We then spent a couple of weeks listening to staff – challenging, probing and digging. In the end we identified a handful of areas where changes would need to be made in order for them to realise their ambitions; some of them were technical, and some were non-technical. Both types of problems tend to be intertwined.

- Senior management disagreed about what the company focus on. Work that required collaboration between teams that answered to different managers didn't get done, or got done using a makeshift set of assumptions from each group, that ended up creating unresolvable conflict between the groups. This resulted in exacerbating problems caused by organisational silos; groups that should have been able to resolve their problems based on a shared understanding of their goals actually couldn't because problems filtered down from senior management that they weren't equipped to solve.
- The architecture wasn't suited for the scale they wanted to reach.

We wrote up our findings, in as much detail as we could, with as much guidance as we could put together, and delivered it. It was agreed that our feedback was accurate, and we were asked to help implement our ideas.

OpenCredo durchsuchen

Jonas Partner @jonaspartner 22 Dez
Addressing the confusion around @CloudCredo and @OpenCredo plus whats next for @OpenCredo bit.ly/1V0RvjA
Retweetet von OpenCredo
Öffnen

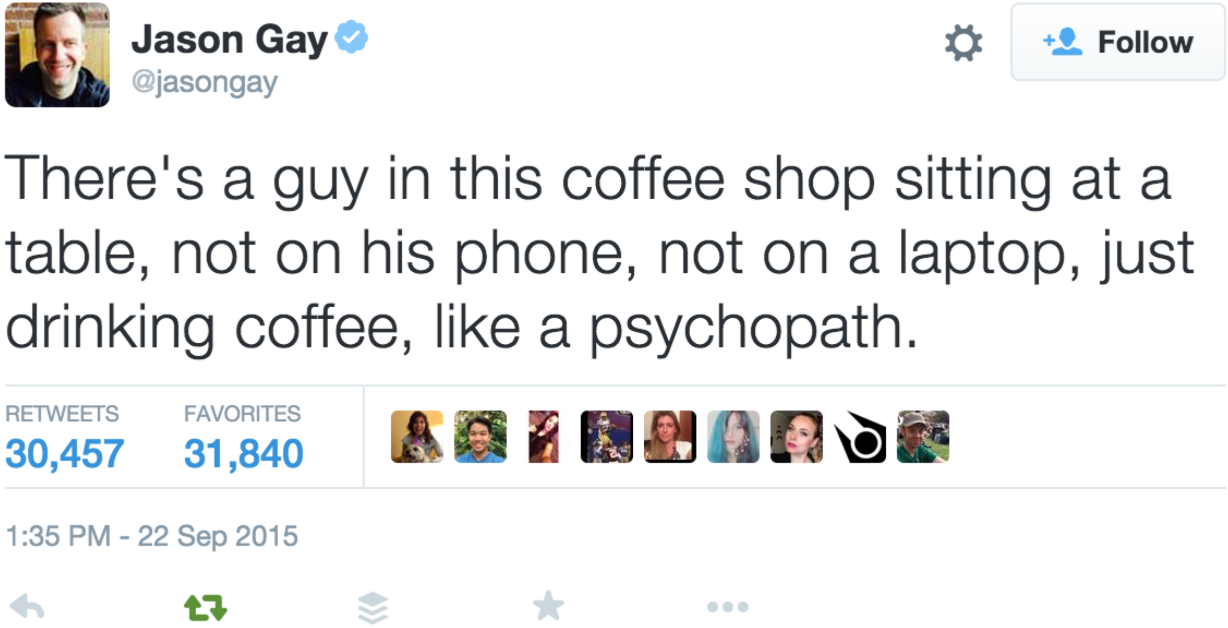
OpenCredo @OpenCredo 21 Dez
OpenCredo is pleased to announce that CloudCredo has been acquired by Pivotal - ow.ly/Wb2mu
Öffnen

OpenCredo @OpenCredo 10 Dez
Come see us at #ScalaX for a chance to win an Apple Watch ow.ly/i/f7alB
Öffnen

Bart Spaans @Work_of_Bart 4 Dez
A merry Christmas from @OpenCredo @tareq_abedrabbo pic.twitter.com/nxr2b1jFYY
Retweetet von OpenCredo

bit.ly/1OePqid

Have a Little Empathy…



Jason Gay

@jasongay

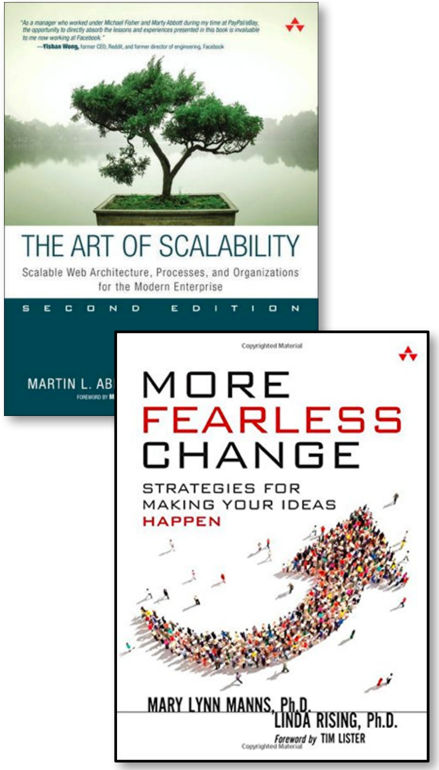
There's a guy in this coffee shop sitting at a table, not on his phone, not on a laptop, just drinking coffee, like a psychopath.

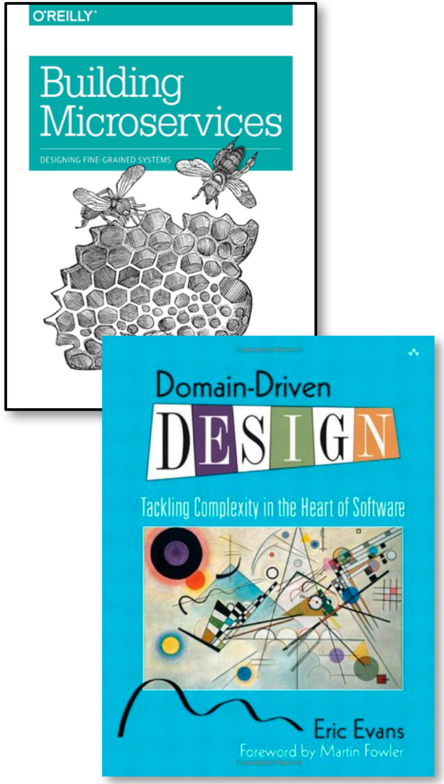
RETWEETS
30,457

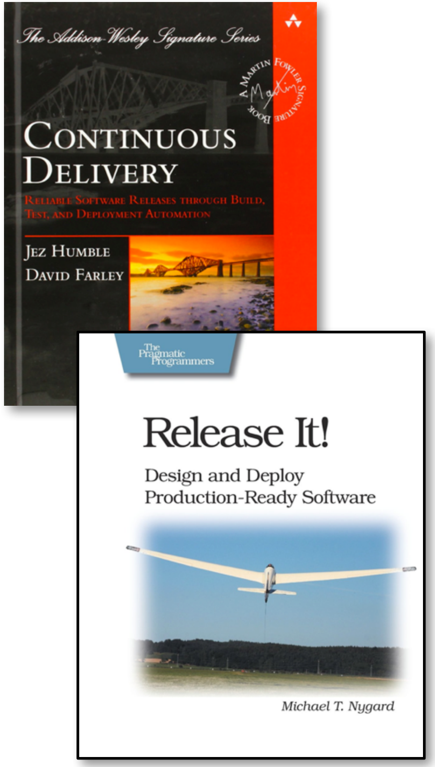
FAVORITES
31,840

1:35 PM - 22 Sep 2015

Some Bedtime Reading…




O'REILLY
Building Microservices
DESIGNING FINE-GRAINED SYSTEMS
Domain-Driven DESIGN
Tackling Complexity in the Heart of Software
Eric Evans
Foreword by Martin Fowler


The Addison-Wesley Signature Series
CONTINUOUS DELIVERY
RELIABLE SOFTWARE RELEASES THROUGH BUILD, TEST, AND DEPLOYMENT AUTOMATION
JEZ HUMBLE
DAVID FARLEY
The Pragmatic Programmers
Release It!
Design and Deploy Production-Ready Software
Michael T. Nygard

In summary…

Organisation, Architecture, Operations

products over projects

small inter-discipline teams over large homogeneous teams

multiple cohesive services over highly coupled monoliths

goal-driven technical leadership over command and control

automated continuous deployment over manual big bangs

individuals and interactions over processes and tools

A Big Thanks To...

OC OpenCredo

Jonas Partner, CEO
Tareq Abedrabbo, CTO
Don Cleeve, Sales

Consultants

Daniel Bryant
Rafal Gancarz
Stephen Judd
Nicki Watt
Erich Eichinger
Noah Cantor
Guy Richardson
Matt Calthrop
Mateus Pimenta
Dominic Fox
Bart Spaans
Jean-Marie Ferdegue
Tristan McCarthy
David Borsos
Marco Cullen
Norbert Annus
Carlos Lindo
Adrian Ursu
Maartens Lourens
Andrew Morgan
Karolis Rusenas
William May
Barnaby Dawson
Lorenzo Nicora
Antonio Cobo Cuenca
Brett Mack
Matthew Long

NOT ON THE HIGH STREET .com

for a life less ordinary

London µServices (Microservices) User Group

London, United Kingdom

Founded Apr 11, 2014

Welcome!

✚ SUGGEST A NEW MEETUP

µCon 2015: The Microservices Conference

The Skills Matter eXchange
116-120 Goswell Road, London (map)

Mon Nov 9
9:00 AM

RSVP

61 going

How can moving from a monolithic

What's new

NEW RSVP
Dimitri Koutsos RSVPed Yes for Microservices Taking on Legacy
1h ago

NEW RSVP
Dimitri Koutsos RSVPed Yes for Continuous Delivery and Microservices
1h ago

THANKS… QUESTIONS?

We are hiring in UK and DE!

daniel.bryant@opencredo.com

'Microservices Weekly' (muserviceswweekly.com)

www.opencredo.com/blog

Additional References

- http://www.infoq.com/presentations/7-sins-microservices
- https://www.opencredo.com/2016/01/08/the-seven-deadly-sins-of-microservices-redux/
- https://www.opencredo.com/2015/11/11/testing-with-microservices/
- https://www.opencredo.com/2015/09/20/working-locally-with-microservices/
- https://www.opencredo.com/2015/06/22/microservices/
- https://dzone.com/guides/cloud-development
- https://www.youtube.com/watch?v=wyZNxB172VI