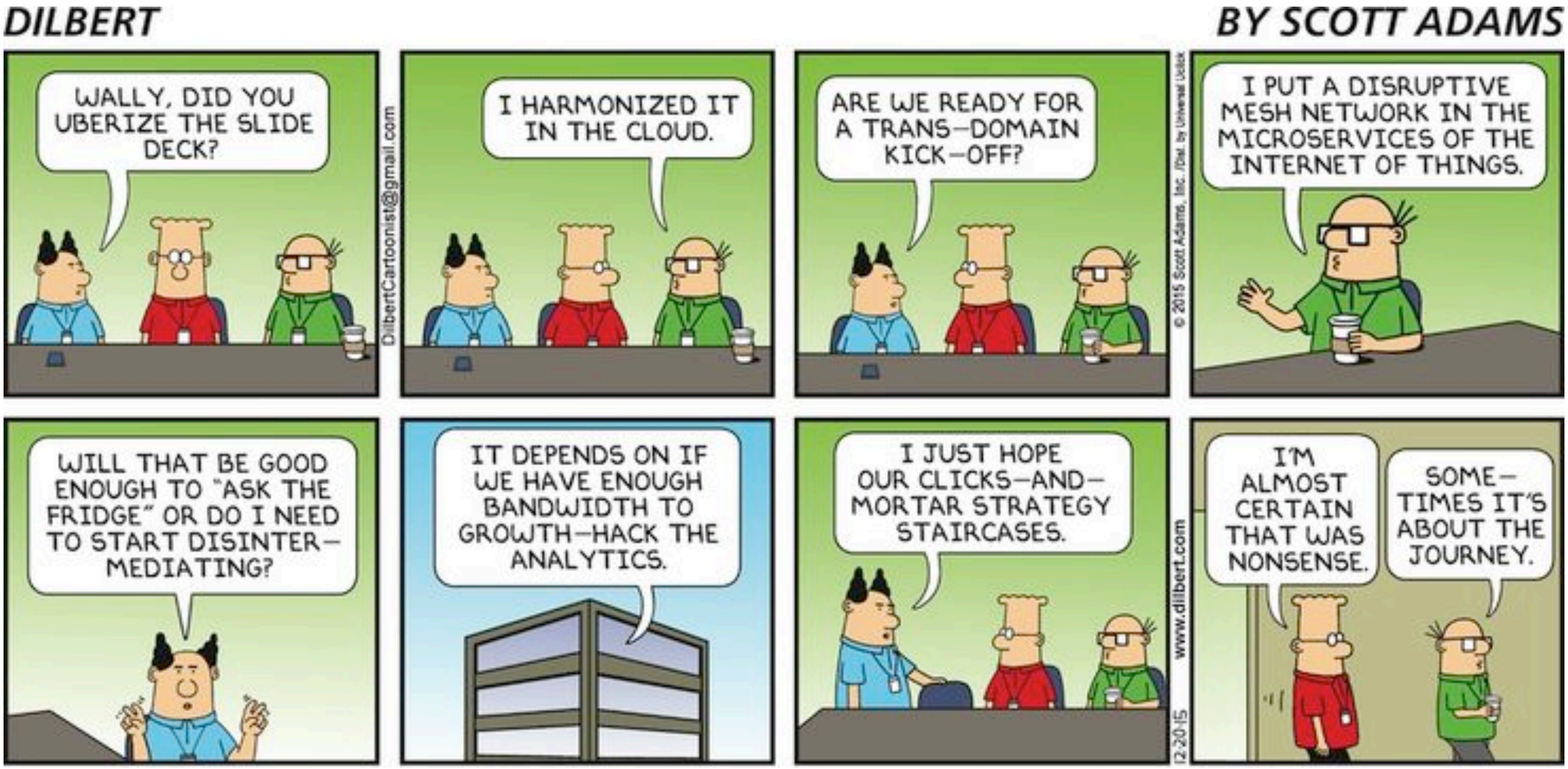
Introduction
Service Oriented Architecture

Oxford University
Software Engineering Programme
December 2019

Introduction

- Aims
- Pre-requisites
- Contents
- Connections
- Resources
- Rules of Engagement
- Introductions


DILBERT
BY SCOTT ADAMS
WALLY, DID YOU UBERIZE THE SLIDE DECK?
I HARMONIZED IT IN THE CLOUD.
ARE WE READY FOR A TRANS-DOMAIN KICK-OFF?
I PUT A DISRUPTIVE MESH NETWORK IN THE MICROSERVICES OF THE INTERNET OF THINGS.
WILL THAT BE GOOD ENOUGH TO "ASK THE FRIDGE" OR DO I NEED TO START DISINTER-MEDIATING?
IT DEPENDS ON IF WE HAVE ENOUGH BANDWIDTH TO GROWTH-HACK THE ANALYTICS.
I JUST HOPE OUR CLICKS-AND-MORTAR STRATEGY STAIRCASES.
I'M ALMOST CERTAIN THAT WAS NONSENSE.
SOME-TIMES IT'S ABOUT THE JOURNEY.
DilbertCartoonist@gmail.com
© 2015 Scott Adams, Inc. /Dist. by Universal Uclick
www.dilbert.com
12-20-15

Apologies for the Jargon

- There is a lot!
  - Microservices, SOA, DevOps, REST, SOAP, WSDL, Swagger, JSON, XML, OAuth2, TLS, Service Mesh, etc
  - Please ask if I fail to explain an acronym

Aims

- To understand:
  - Benefits and challenges of SOA
  - Services, Microservices and APIs
  - Security models
  - Mediation, Composition, Governance
- Implementation of
  - REST based services
  - Event based architectures
  - gRPC and binary protocol based services
  - Microservices
  - Mediation and BPMN flows
  - OAuth2 and SSL secured services
  - API Gateways and clients

Pre-requisites

(Some familiarity required)

- Languages: Java, Node, Python
- Data formats: JSON and XML
- Tools: Unix shell, Eclipse, Text editors

Contents

- Overview and course outline
- Case studies and motivations
- REST introduction
- REST example flows
- Advanced REST
- SOAP and WSDL
- Microservices architecture
- Event Architectures
- gRPC and binary protocols
- Deployment, DevOps, containers and cloud-native applications
- Integration and ESBs
- Security
- API and API Management
- Orchestration and Choreography
- Governance
- Overview, futures, recap

Practicals

A. My aim is to have **more** practicals than is reasonable:

- Some people finish early, so there are extensions and bonus practicals for them.
- You might even wish to do more at home?!?

B. The practicals are quite directive to start with:

- This is a complex area with a lot to cover.
- Extensions are more freeform.
- *You need to **think** and not just do as I say to get the most out of them.*

"This really is an innovative approach, but I'm afraid we can't consider it. It's never been done before."

Practicals

- Basic HTTP server and client
- HTTP service in Java
- Evolving the Richardson Maturity Model towards a RESTful service
- Microservice and Docker deployment
- gRPC and Async
- SSL and OAuth2 security
- API Management and Analytics
- BPMN workflows
  - Plus some bonus exercises

Resources

- Weerawarana et al, *Web Services Platform Architecture,* (Pearson, 2005)
- Erl, *SOA* (Prentice-Hall, 2005)
- Richardson and Ruby, *RESTful Web Services* (O'Reilly, 2007)
- Webber et al, *REST in Practice* (O'Reilly, 2010)
- Fielding, *Architectural Styles and the Design of Network-based Software Architectures,* (University of California, 2000)
- Various W3C, OASIS, IETF, OMG standards

Rules of Engagement

- Ask questions as we go along
  - We will “park” any that are better answered later
  - **Don’t wait till the end to ask or raise concerns**
- Timings are flexible
- Please keep mobile phones silent or better still turned off
- If you have improvements or bug reports, please submit issues or pull requests:
  - https://github.com/pzfreo/ox-soa2/issues/new

Paul Fremantle

- CTO and Co-Founder of WSO2
  - An Open Source SOA and API focused company
- Previously Senior Technical Staff Member, IBM WebSphere architecture
- Co-Chair Web Services Reliable eXchange at OASIS (WSRM)
- VP, Apache Synapse and Member of ASF
- MA in Maths and Philosophy
- MSc in Computation (here)
- PhD in Computing – IoT privacy and security

You?

Approximate Schedule

Monday	Tuesday	Wednesday	Thursday	Friday
Introduction	Evolving REST practicals	Docker and DevOps	API Manage-ment	Design Exercise
Case Studies and motivation	REST Description Advanced REST	Async and Events	Integration	Conclusions
REST Introduction	SOAP and XML	Binary / gRPC	Composition	
REST example flow	Microservices	Security	Governance and Org	

Let's get started