WTF in Clichés

Balajee Mohan
@balajeetm
What’s The Fun in Clichés

Balajee Mohan
@balajeetm

TESCO Technology
K.I.S.S

Keep It Short & Simple

JARGON
What’s The Fun in Clichés

Balajee Mohan
@balajeetm

TESCO Technology
What’s The Fun in Clichés

Balajee Mohan
@balajeetm
What’s The Fun in Clichés
Agile, Microservices, DevOps

Balajee Mohan
@balajeetm
Balajee Mohan

@balajeetm

http://blog.balajeetm.com/
Balajee Mohan

TESCO Technology

Transform the way we build software

@balajeetm

http://blog.balajeetm.com/
I’m NO EXPERT
Balajee Mohan

Transform the way we build software

@balajeetm

http://blog.balajeetm.com/
SOFTWARE ENGINEERS

!=

ENGINEERS
Plan

Build

Share

RISK!

RISK MITIGATION

Cost

Value

Time
RISK MITIGATION

Time
SOFTWARE

!=

BRIDGE BUILDING
Low Cost to Change
+ Low Cost to Distribute

Express Iteration
BAD MITIGATION = GREATER RISK
RISK MITIGATION \(\bowtie\) FEEDBACK
INTERNET ERA

FEEDBACK  INNOVATE  DELIVER

FASTER
VELOCITY

\[ \vec{V} = |V| \hat{V} \]

Culture Velocity

Build Expertise

Learn & Unlearn
Low Cost to Distribute
Low Cost to Change

Low Cost to Distribute
Course Correction

Risk? Risker Risk

Time

Cost

Value
AGILE & EXPRESS ITERATION

Course Correction

Lower Risk
AGILE & EXPRESS ITERATION are the way to succeed in a competitive & unpredictable software economy
What is AGILE?

Is a philosophy
What is AGILE?

Is a philosophy

Deutsche Bank


To achieve velocity by being flexible to change
Low Cost to Change
Low Cost to Distribute
Low Cost to Change?

Low Cost to Distribute
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
Low Cost to Change
AGILE & EXPRESS ITERATION

Course Correction

Lower Risk

Cost

Value

Time
PAINFULLY SLOW ITERATION

Hard to Course Correction

RISK!
PAINFULLY SLOW ITERATION

Hard to Course Correction

RISK!
SOFTWARE

!=

BRIDGE BUILDING
VELOCITY

\[ \vec{V} = |V| \cdot \hat{V} \]

Development Velocity

How many features are delivered

How soon are they delivered
ISN’T IT SIMPLE MATH?

More the people = More the features
Thinks nine women can deliver a baby in one month
Why art thou a MONOLITH?
YOU ONLY HAVE ONE BRAIN
ORG STRUCTURE

CONWAY’S LAW

SOFTWARE ARCHITECTURE
CONWAY’S LAW
I THINK, THEREFORE I AM
Design Patterns to the rescue!
Design Patterns

YAGNI

DRY

S.O.L.I.D
BAD LUCK
DEV

CAME UP WITH THE
BEST DESIGN EVER

REQUIREMENTS CHANGED
BAD LUCK DEV

CAME UP WITH THE BEST DESIGN EVER

REQUIREMENTS CHANGED
BAD LUCK DEV

CAME UP WITH THE
BEST DESIGN EVER

REQUIREMENTS CHANGED
BAD LUCK

DEV

THERE GOES MY DESIGN!

CAME UP WITH THE
BEST DESIGN EVER

REQUIREMENTS CHANGED
CONSEQUENCES
CONSEQUENCES

Invasive Code Changes

Functionality

NFRs

Design
ISOLATION

DOMAIN

THOUGHT  DESIGN
IMPLEMENTATION

THOUGHT  DESIGN
IMPLEMENTATION

THOUGHT  DESIGN
IMPLEMENTATION
ISOLATION

DOMAIN

THOUGHT  DESIGN
IMPLEMENTATION

THOUGHT  DESIGN
IMPLEMENTATION

THOUGHT  DESIGN
IMPLEMENTATION
ISOLATION

THOUGHT DESIGN IMPLEMENTATION

THOUGHT DESIGN IMPLEMENTATION

THOUGHT DESIGN IMPLEMENTATION

Duplication
DRY?

DO NOT REPEAT YOURSELF
DRY?

DO REPEAT YOURSELF
JAVA IS DEAD, LONG LIVE JAVA
We can build enterprise apps without OOPs
We SHOULD build enterprise apps without OOPs
We can build enterprise apps without OOPs
SOFTWARE ENGINEERS

==

ENGINEERS
EOL OF RIGID STRUCTURES

SQL -> NOSQL
NORMALIZATION -> DENORMALIZATION
OOPS -> FUNCTIONAL
DRY -> DRY
MONOLITH -> MICROSERVICES
WATERFALL -> AGILE
Low Cost to Change
Low Cost to Distribute
Low Cost to Change
Low Cost to Distribute?
VELOCITY

\[ \vec{V} = |V| \bowtie \mathbf{V} \]
I DON'T KNOW WHAT YOUR SOFTWARE IS AND HOW IT'S BUILT

BUT I'LL LOOK FOR IT, FIND IT AND "KILL -9" IT
ISOLATION

DOMAIN

THOUGHT  DESIGN
IMPLEMENTATION

THOUGHT  DESIGN
IMPLEMENTATION

THOUGHT  DESIGN
IMPLEMENTATION
ISOLATION

DOMAIN

THOUGHT DESIGN IMPLEMENTATION
THOUGHT DESIGN IMPLEMENTATION
THOUGHT DESIGN IMPLEMENTATION

Give up heterogenous tech stacks
Use similar data stores

Forsake freedom of choice

Give up heterogenous tech stacks
ISOLATION

Forsake freedom of choice

Structures EMERGE

THOUGHT DESIGN IMPLEMENTATION

Use similar data stores
Give up heterogenous tech stacks
ISOLATION

Use similar data stores

Forsake freedom of choice

Give up heterogenous tech stacks

RESOURCE CONTENTION

THOUGHT  DESIGN  IMPLEMENTATION

THOUGHT  DESIGN  IMPLEMENTATION

THOUGHT  DESIGN  IMPLEMENTATION

THOUGHT  DESIGN  IMPLEMENTATION

Structures EMERGE
ISOLATION FAILED @ DEPLOYMENT

WHAT IF I TOLD YOU

MICROSERVICES DO NOT WORK

ITS WAY TOO COMPLICATED

ITS JUST LIKE BEFORE

ITS IMPOSSIBLE TO SCALE THIS APPLICATION
IT WAS YOU ALL ALONG

DISTRIBUTED MONOLITH

MICROSERVICE
<table>
<thead>
<tr>
<th>APP SOFTWARE</th>
<th>INFRASTRUCTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spawn app resources</td>
<td>Spawn infra resources</td>
</tr>
<tr>
<td>Scale on need</td>
<td>Scale on need</td>
</tr>
<tr>
<td>Continuously monitored</td>
<td>Continuously monitored</td>
</tr>
<tr>
<td>Improved on constant feedback</td>
<td>Improved on constant feedback</td>
</tr>
</tbody>
</table>

**DEVELOPMENT**

**OPERATIONS**
DEVELOPMENT  SHAKING HANDS  OPERATIONS
CONTINUOUS INTEGRATION

CONTINUOUS DELIVERY

CONTINUOUS DEPLOYMENT
Being flexible to change is the way to succeed in a competitive & unpredictable software economy
THE MARRIAGE OF AGILE, MICROSERVICES & DEVOPS

Culture Velocity

Build Expertise

Learn & Unlearn

Development Velocity

How many features are delivered

How soon are they delivered

Deployment Velocity

Deploy to live

Again and again
All of us are capable of being a walking software factory
BDD
BRUCE-LEE DRIVEN DEVELOPMENT
The stiffest tree is most easily cracked, while the bamboo, survives by bending with the wind.

~ Bruce Lee

Moving away from rigidity
Use only that which works, and take it from any place you can find it.

~ Bruce Lee

Picking heterogenous tools
I fear not the man who has practiced 10,000 kicks once, but I fear the man who has practiced one kick 10,000 times.

~ Bruce Lee
Empty your mind.

Be formless. Shapeless. **LIKE WATER.**

You put water into a cup, it becomes the cup.
You put water into a bottle, it becomes the bottle.
You put water into a teapot, it becomes the teapot.

**WATER can FLOW or it can CRASH.**

**BE WATER, MY FRIEND.**
As you think, so you shall be

References - https://www.youtube.com/watch?v=xdw_9dADM-4