Becoming Fully Buzzword Compliant

JET BRAINS
—

Trisha Gee (@trisha_gee)
Developer & Technical Advocate, JetBrains

Trisha Gee

Developer Advocate at JetBrains

Digital Aguacates • University of Sussex

Sevilla Area, Spain • 500+ 👥

Note for recruiters: I am happily based in Spain and NOT looking to relocate back to London, or to anywhere else.I'm a developer / technical advocate / educator, based in Spain and working remotely for JetBrains. I love the combina…

See more ⌄

⭐ **All-Star profile** | 447 Who's viewed your profile | 533 Views of your post in the feed

https://www.linkedin.com/in/trishagee/

Skills & Endorsements

Java · 99+

Endorsed by **Peter Lawrey and 20 others who are highly skilled at this**

Endorsed by **5 of Trisha's colleagues at MongoDB**

Software Development · 32

Endorsed by **Frank Greco and 2 others who are highly skilled at this**

Endorsed by **2 of Trisha's colleagues at MongoDB**

Industry Knowledge

Continuous Integration · 35

Agile Methodologies · 26

Agile Project Management · 13

Scrum · 11

Distributed Systems · 10

Design Patterns · 10

Object Oriented Design · 9

Web Applications · 9

Tools & Technologies

MongoDB · 44

MySQL · 23

REST · 19

Git · 13

or: Tips on Surviving the Technology Industry

Recent Past (~5 years ago)

- Asynchronous Programming
- NoSQL
- Distributed Version Control
- JavaScript & HTML5
- Continuous Delivery
- DevOps
- Cloud

More Recent Past (~2 years ago)

- Reactive
- Big Data
- Git
- JavaScript & HTML5
- Continuous Delivery
- DevOps
- Cloud

Today

- Blockchain
- Containers
- Serverless
- Security
- Machine Learning / AI
- Ethics

Looking back further...

- Prince2
- Scrum
- SVN
- Flash
- AWT
- Test Driven Development
- Static Typing

Looking back further...

- Prince2
- Scrum
- SVN
- Flash
- Swing
- Test Driven Development
- Static Typing

How Can We Tell What's Important?

How can we avoid Extinction?

Step 1: Denial

Step 1: Awareness

NEWS

San Francisco: QCon.ai Schedule Nearing Completion

WESLEY REISZ   ON MAR 01

Hyperscript Tagged Markup: A JSX Alternative Based on Standard JavaScript

BRUNO COURIOL   ON FEB 28   2

On the road to Python 3.8: Assignment Expressions, Merging typed_ast and Multiprocessing Improvement

ALEX GIAMAS   ON FEB 28

The Risk of Climate Change and What Tech Can Do: QCon London Q&A

BEN LINDERS   ON FEB 28

Microsoft Announces Two New Azure Cloud Services Allowing Developers to Build Cross-Platform AR Apps

STEEF-JAN WIGGERS   ON MAR 01

Red Hat Releases Eclipse Che-Based IDE CodeReady Workspaces

MATT CAMPBELL   ON FEB 28   1

Portia Tung on Playful Leadership

PORTIA TUNG , **SHANE HASTIE**   ON FEB 28

The Importance of Event-First Thinking

JAN STENBERG   ON FEB 28

🔥 TRENDING

10 days   40 days   6 months

1 ▶ Goodbye Client Side JavaScript, Hello C#'s Blazor
   1

2 DevOps and Cloud InfoQ Trends Report - February 2019

3 ▶ Getting from Monolith to Microservices

4 2019 Scrum Master Trends Report Published

5 Google's Software-Driven Hybrid Cloud Platform Moves from Alpha to Beta Stage

6 ▶ Using CredHub for Kubernetes Deployments

7 Are Frameworks Good or Bad, or Both?   6

8 🔊 Grady Booch on Today's Artificial Intelligence Reality and What it Means for Developers

DZone

Thanks for visiting DZone today, **Trisha**   **Post** +

REFCARDZ   GUIDES   ZONES   JOBS | Agile   AI   Big Data   Cloud   Database   DevOps   Integration   IoT   Java   Microservices   Open Source   Performance   Security   Web Dev

DZone Spotlight   Friday, March 1

Describe, Then Interpret: HTTP Endpoints Using Tapir

by Adam Warski   ♔ MVB

There's no shortage of great HTTP server libraries in Scala: akka-http, http4s, play, finch, just to name some of the more popular ones. However, a common pain point... More
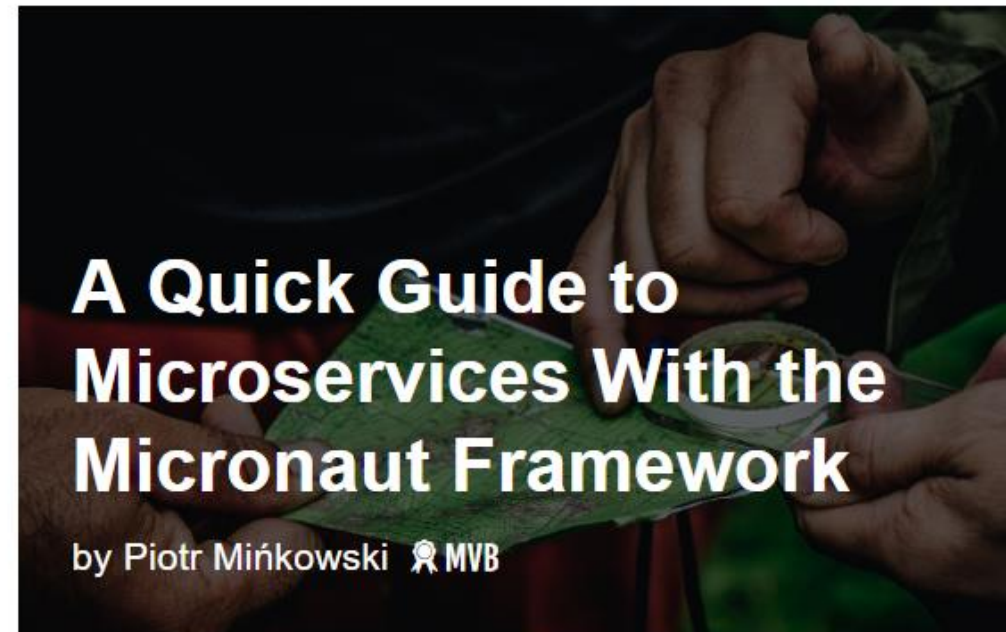
A Quick Guide to Microservices With the Micronaut Framework

by Piotr Mińkowski   ♔ MVB

The Micronaut framework was introduced as an alternative to Spring Boot for building microservice applications. At first glance, it is very similar to Spring. It... More

DZone's Guide to

Big Data: Volume, Variety, and Velocity

⬇ DOWNLOAD   ☆ SAVE

Managing Bitemporal Data With BarbelHisto
by Niklas Schlimm

Will Our Software Bankrupt Us? [Interview]
by Pete Pizzutillo

AWS DevOps: Introduction to DevOps on AWS
by Vishal Padghan   ♔ MVB

Why We Need to Rip Off the Cybersecurity Band-Aids
by Anne Baker

Refcard #289
Working With Time Series Data
by Daniella Pontes

⬇ DOWNLOAD   ☆ SAVE

Refcard #288
Low-Code Application Development
by Mike Hughes

⬇ DOWNLOAD   ☆ SAVE

Duplicate Objects in Java: Not Just Strings
by Misha Dmitriev

My 2019 Predictions for Big Data in the Enterprise
by Alex Gorelik

AWS S3 Bucket Naming Challenge [Comic]
by Jayashree Hegde Adkoli   ♔ MVB

Proposed SQL Server Defaults: Disable Lightweight Pooling
by Randolph West   ♔ MVB

WARNING

CLOUD DISASTER AHEAD

2 reasons to avoid industry-specific cloud technology

Too often, industry-specific platforms use inferior technology and take too much work to reconfigure

BY DAVID LINTHICUM

BY INFOWORLD STAFF

Deep dive: AI, machine learning, and deep learning
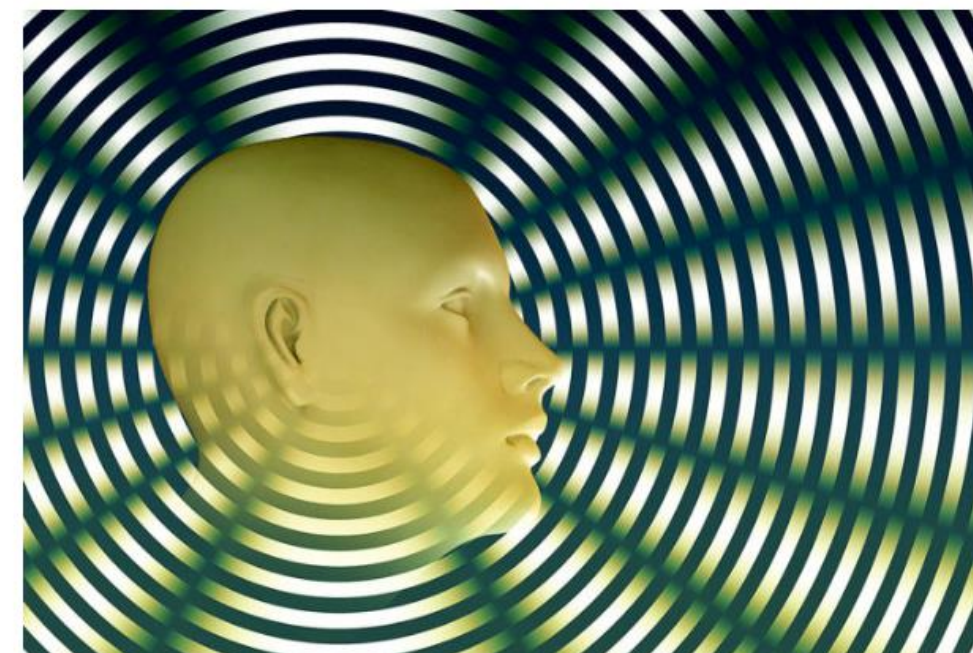
JavaScript dapps: Build decentralized apps with Blockstack

How to validate data, analytics, and data visualizations

Why 35 percent of cloud spending is wasted

What's new in Google's Dart 2.2 language

BY ISAAC SACOLICK

INFOWORLD REVIEWS

Comparison
Cloud IDE review: AWS Cloud9 vs. Eclipse Che vs. Eclipse Theia

Here are the 10 top tech trends you need to understand right now

From autonomous things and blockchain to quantum computing; how many of these technologies are you ready for?

By Steve Ranger | October 16, 2018 -- 12:13 GMT (13:13 BST) | Topic: Innovation

Tl;DR:

Autonomous things, Augmented analytics, AI-driven development, Digital twins, Edge computing, Immersive experience, Blockchain, Smart spaces, Digital ethics and privacy, Quantum computing

Top Tech Trends In 2019: 11 Experts Detail What You Need To Watch

Forbes Technology Council CommunityVoice ⓘ

Tl;DR:

Increased Automation, Blockchain Comeback, Human/AI Collaboration, Connected Devices, Augmented Reality, Cybersecurity Using ML And AI, Solutions To The Tech Backlash, Technology Convergence, Augmented Analytics Using Natural Language, Data Security

TechRepublic ✔ @TechRepublic · 59s
AWS Firecracker is tiny, efficient, fast, and might redefine the virtual machine. Here's what you need to know about this AWS product.
tek.io/2Eti7bM

🖼 Image

💬 ⟲ ♡ •••

⟲ Ali Spittel 👩 Retweeted

DCFemTech @DCFemTech · 1m
🎉 It's that time of year! Nominations for the 2019 DCFemTech Awards are now open! Know an amazing woman who codes, designs, or works with data? Are you one? #dcft19 #dctech

Nominate yourself or others ➡
dcfemtech.typeform.com/to/aodjHg

🖼 Image

💬 ⟲ 3 ♡ 2 •••

𝒢 ··· 1m

⟲ Ali Spittel 👩 Retweeted

Prince @maxcell · 6m
Being a teacher has really taught me to value expressive code for others and replicable code by others. It isn't cool to write smaller lines simply because it works. You aren't getting points for being "clever". It just makes it hard to maintain for others who come after you.

💬 ⟲ 1 ♡ 4 •••

Ron Jeffries @RonJeffries · 1m

Simon Brown @simonbrown · 14m
👇 A first version is live ... the "Create animated GIF" button will become enabled once you've exported one or more diagrams in your workspace. #WorkInProgress

> **Simon Brown** @simonbrown
> I'm playing with generating animated GIFs ... is this a useful feature to have in @structurizr?

💬 1 ⟲ ♡ 3 •••

⟲ Martijn Verburg Retweeted

Matthew Cornford @bofalot · 48m
Also, @karianna well done for putting together what looks like a great #java track @qconlondon I'm really looking forward to hearing all the other great speakers @dianecraigdavis @trisha_gee @aardvark179 @speakjava

💬 ⟲ 1 ❤ 3 •••

⟲ Simon Brown Retweeted

Scott W. Ambler @scottwambler · 35m
The other day in a workshop I had someone ask me if I had ever heard of a technique called "whiteboarding" and whether I thought it was a good idea for software architecture. Sigh.

💬 2 ⟲ 2 ♡ 8 •••

Justin Lee @evanchooly · 36m

Filtering by 🙌

JAX London @jaxlondon · 38m
#IntelliJIDEA 2019 progress report: 2019.1 Beta is here! bit.ly/2TuJCeV

💬 ⟲ ♡ •••

Dylan Beattie 🇪🇺 @ NDC Porto 🇵🇹 · 1h
"We call it the dilophosaurus factor: how many developers would need to be eaten by a dilophosaurus before you lost the ability to run your critical systems? It turned out our dilophosaurus factor was... one." - @toddhgardner sharing tech lessons from Jurassic Park #NDCPorto

💬 ⟲ 3 ♡ 7 •••

Simon Brown @simonbrown · 1h
I'm playing with generating animated GIFs ... is this a useful feature to have in @structurizr?

Newsletters

http://androidweekly.net/

http://www.kotlinweekly.net/

http://scalatimes.com/

http://groovycalamari.com/

http://www.baeldung.com/java-web-weekly/

https://info.jetbrains.com/Java-Annotated-Subscription.html

pocket

Trisha ⌄

My List

Favorites

Archive

Articles

Videos

Images

Tags >

Performance After Eight
youtube.com   beyond java 8

Enhanced-For Statement Should Allow Streams
cr.openjdk.java.net   annotated

Beware of computation in static initializer
pangin.pro   annotated

Why You NEED To Be On Twitter If You Identify As Part of the #womenintech ...
womenintechpgh.com   annotated   gender   ifttt   twitter

FYI: Changes to Garbage Collection in JDK11
mail.openjdk.java.net   beyond java 8

OpenJDK Startup From 8 Through 11
cl4es.github.io   ifttt   twitter

A Bird's-Eye View on Java Concurrency Frameworks
dzone.com   annotated

Spring Boot With Spring Data JPA
dzone.com   annotated

How To Build A Daily Habit Tracker In Trello (And Reach Those Goals!)
blog.trello.com   Best Of   annotated

Migrating Spring Boot Applications to the Latest Java Version (Java 11)
dzone.com   beyond java 8

Step 2: Speaking the Lingo

Containerised reactive serverless microservice blockchain big data machine learning applications

Containerised reactive serverless microservice blockchain big data machine learning applications

Containers for Java: Optimizing Your Applications

Mofe Salami

Location: Henry Moore, 4th flr.

Kubernetes AMA w/ Cebula, Petazzoni & Hoban

Jérôme Petazzoni, Luke Hoban, Melanie Cebula

Location: Guild, 3rd flr.

Develop Hundreds of Kubernetes Services at Scale With Airbnb

Melanie Cebula

Location: Fleming, 3rd flr.

Choosing Kubernetes: Managing Risk in Cloud Infrastructure

Ben Butler-Cole

Location: Whittle, 3rd flr.

Introduction to Containers

Adrian Mouat & Aliou Ba

[9:00am - 12:00pm]

Getting Started With Kubernetes and Container Orchestration

Jérôme Petazzoni & AJ Bowen

[9:00am - 4:00pm]

Programming the Cloud with TypeScript: Serverless, Containers and Kubernetes

Luke Hoban & Sean Gillespie

[9:00am - 4:00pm]

Presentation: Why Developers Shouldn't Care About Containers

Track: **Solutions Track IV**

Location: **Mountbatten, 6th flr.**

Duration: **11:50am - 12:40pm**
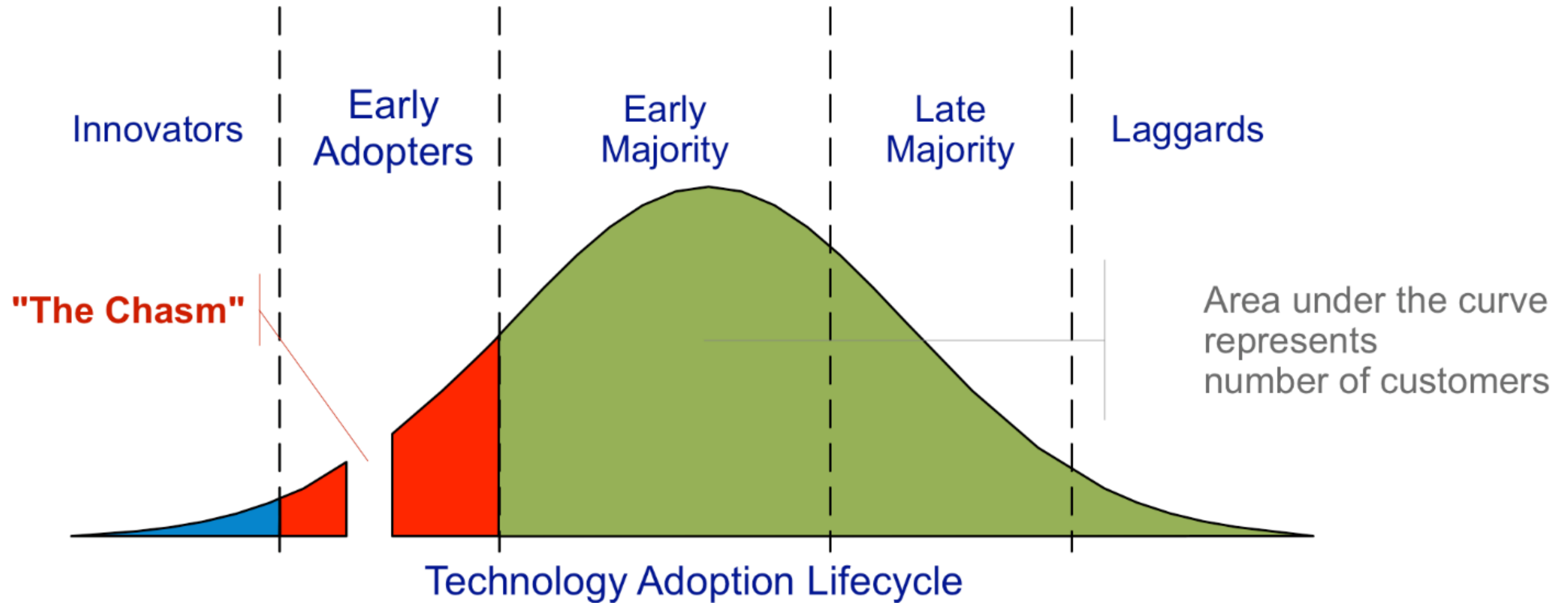
Day of week: **Wednesday**

Share this on:

Abstract

Imagine a world where developers don't even need to know what a container is? A world where writing code is literally all a developer needs to worry about. Turns out that world already exists - and in multiple forms! Ed will cover how to increase development efficiency and will use the open
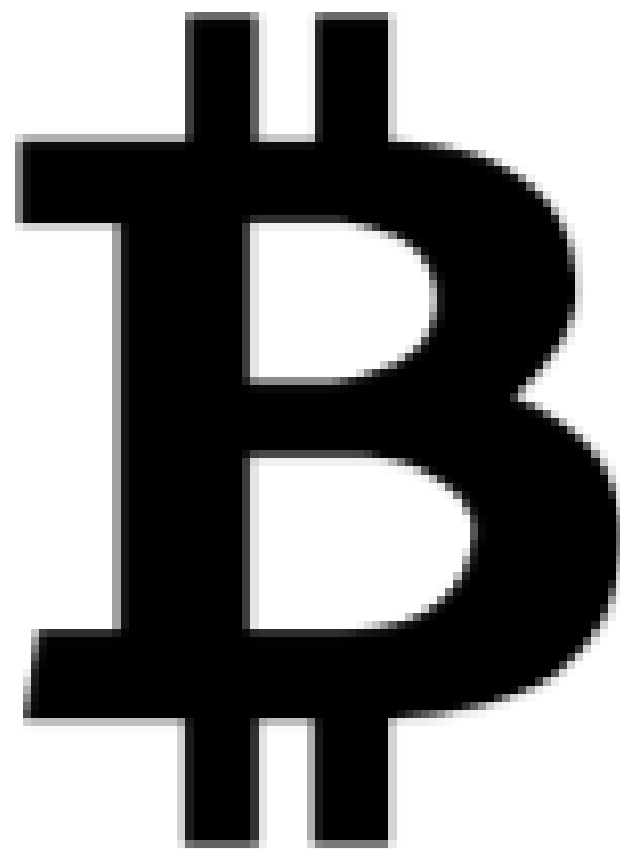
Containerised reactive serverless **microservice** blockchain big data machine learning applications

Innovators

Early Adopters

Early Majority

Late Majority

Laggards

"The Chasm"

Area under the curve represents number of customers

Technology Adoption Lifecycle

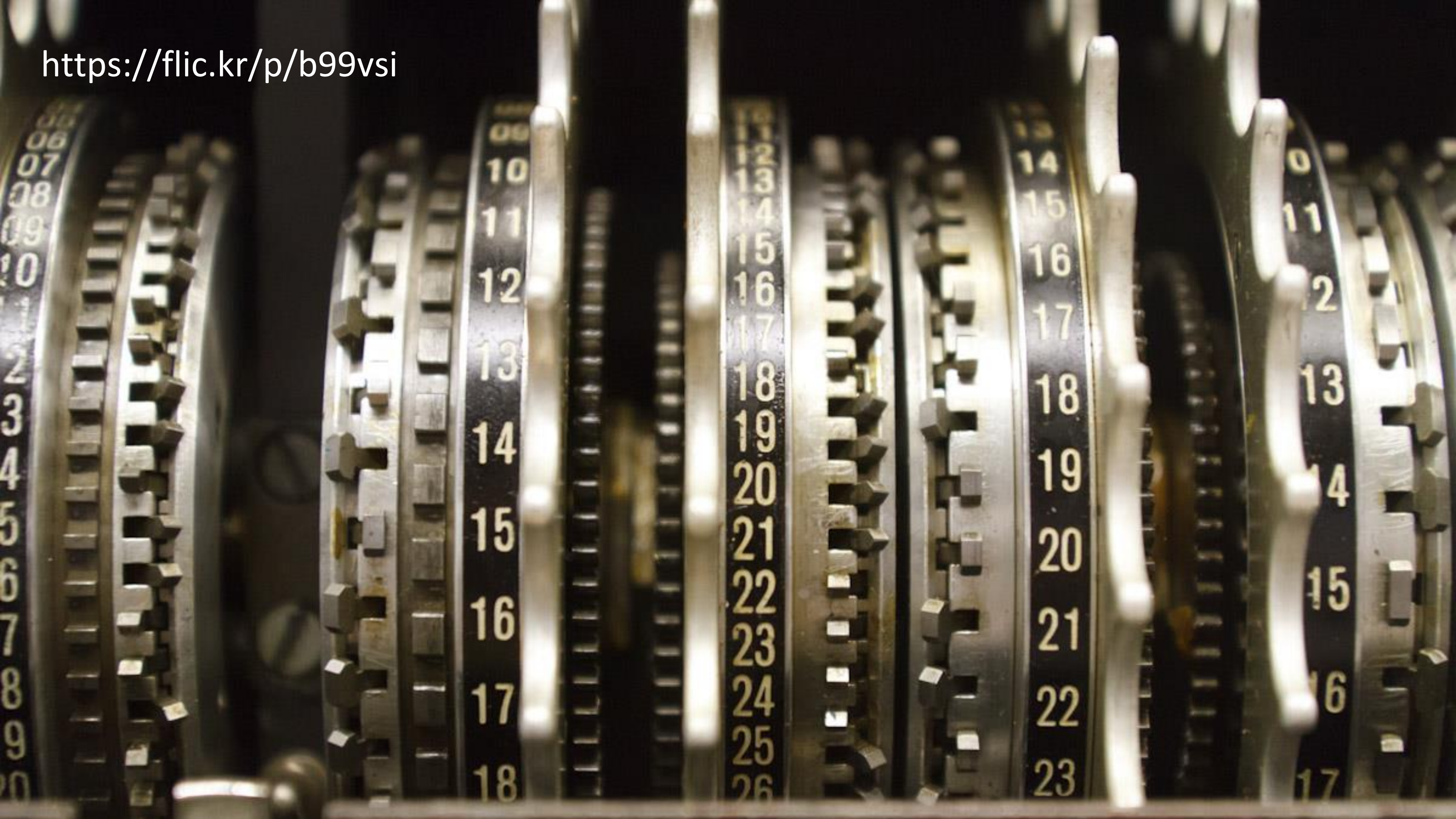https://en.wikipedia.org/wiki/File:Technology-Adoption-Lifecycle.png

Containerised reactive serverless microservice blockchain big data machine learning applications

Containerised reactive serverless microservice **blockchain** big data machine learning applications

£$€

Containerised reactive serverless microservice blockchain big data machine learning applications

Containerised reactive serverless microservice blockchain big data machine learning applications

Containerised reactive serverless microservice blockchain big data **machine learning** applications

Containerised **reactive** serverless microservice blockchain big data machine learning applications

Step 3: Enough Knowledge to be Dangerous

Containerised reactive serverless microservice blockchain big data machine learning applications

Reactive Systems?

Reactive Programming?

Functional Reactive Programming?

Functional reactive programming, commonly called *FRP*, is most frequently misunderstood. FRP was very [precisely defined](#) 20 years ago by Conal Elliott. The term has most recently been used incorrectly[1] to describe technologies like Elm, Bacon.js, and Reactive Extensions (RxJava, Rx.NET, RxJS) amongst others. Most libraries claiming to support FRP are almost exclusively talking about *reactive programming* and it will therefore **not be discussed further.**

https://www.oreilly.com/ideas/reactive-programming-vs-reactive-systems

Reactive programming vs. Reactive systems

- Reactive is a set of design principles
- Event-driven vs. message-driven
- From programs to systems
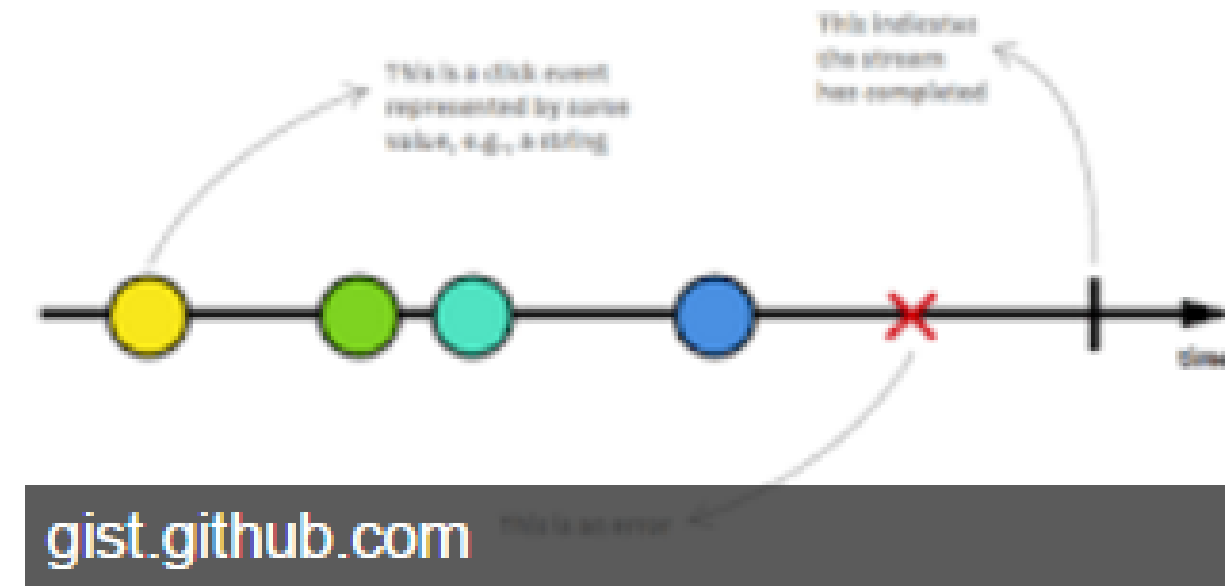- The resilience of reactive systems
- The elasticity of reactive systems

https://www.oreilly.com/ideas/reactive-programming-vs-reactive-systems
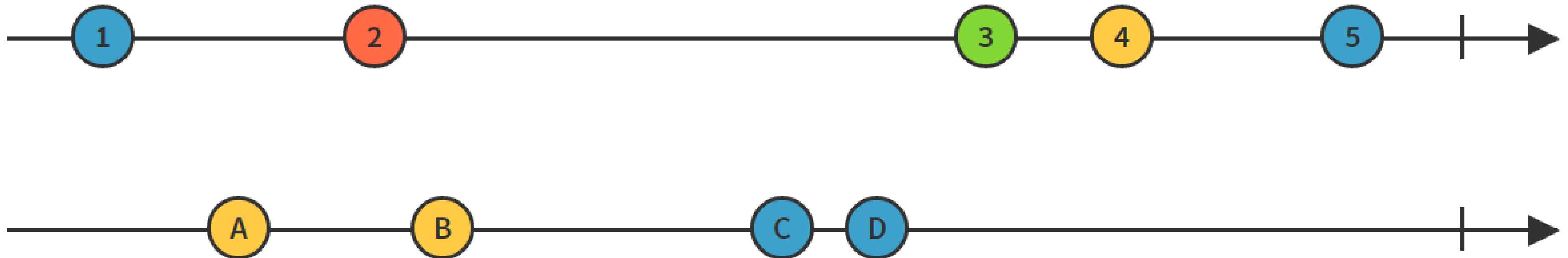
Reactive Systems sound hard

Reactive Programming is available in All Good Languages

In computing, **reactive programming** is an asynchronous **programming** paradigm oriented around data streams and the propagation of change.

Reactive programming - Wikipedia

https://en.wikipedia.org/wiki/Reactive_programming

combineLatest((x, y) => "" + x + y)

ENGINEERING

Understanding Reactive types

ENGINEERING | SÉBASTIEN DELEUZE | APRIL 19, 2016 | 2 COMMENT

Following previous Reactive Spring and Reactor Core 3.0 blog posts, I would like to
Reactive types are useful and how they compare to other asynchronous types, base
we have learned while working on the Spring Framework 5 upcoming Reactive sup

Why using Reactive types?

Reactive types are not intended to allow you to process your requests or data faster

| All Posts | RebelLabs | ZeroTurnaround | Android | Virtual JUG |

RxJava for easy concurrency and backpressure

December 19, 2016 | Anton Arhipov | 2 Comments

Tweet | Like 19 | Share 41 | G+1 4

RxJava is growing in popularity in Android application development
also very capable for server-side apps. RxJava makes concurrency
though a seasoned Java developer would have to re-learn the con

---

Reactive Programming with JDK 9 Flow API

Created by Rahul_Srivastava-Oracle on 26-Sep-2016 06:24. Last modified by Bob Rhubart-Oracle on 26-Sep-2016 14:03.

Reactive programming is about processing an asynchronous stream of data items, where applicatio
example using the JDK 9 Flow API.

By Rahul Srivastava

What is Reactive Programming ?

Reactive programming is about processing an asynchronous stream of data items, where applicatio
data items occurring over time. This model is more memory efficient because the data is processed

RxJava 2.0 - Tutorial

Lars Vogel, Simon Scholz (c) 2017 vogella GmbH – Ve

Table of Contents

RXJava2 by Example

| Posted by Victor Grazi on Feb 13, 2017. Estimated reading time: 16 minutes | Discuss

Share

Key Takeaways

- Reactive programming is a specification for dealing with asynchronous streams of data
- Reactive provides tools for transforming and combining streams and for managing flow-control
- Marble diagrams provide an interactive canvas for visualizing reactive constructs
- Resembles Java Streams API but the resemblance is purely superficial

---

ReactiveX / RxJava   Watch

<> Code | Issues 43 | Pull requests 0 | Projects 0 | Wiki | Pulse | Gra

Home

David Gross edited this page on Jan 12, 2015 · 95 revisions

RxJava is a Java VM implementation of ReactiveX (Reactive Extensions): a library for composing
asynchronous and event-based programs by using observable sequences.

For more information about ReactiveX, see the Introduction to ReactiveX page.

Step 4: Code

```
Observable.fromIterable(words)
        .flatMap(word -> fromArray(word.split("")))
        .zipWith(range(1, Integer.MAX_VALUE),
                (str, count) -> format("%2d. %s", count, str))
        .subscribe(System.out::println);
```

```java
static String filterMessagesForMoods(String s) {
    return Stream.of(s)
                .map(MoodService::getTweetMessageFrom)
                .flatMap(s1 -> Stream.of(splitMessageIntoWords(s1)))
                .map(String::toLowerCase)
                .map(MoodAnalyser::getMood)
                .filter(Optional::isPresent)
                .distinct()
                .map(mood -> mood.get().name())
                .collect(Collectors.joining(","));
}
```

```java
static void filterMessagesForMoods(Publisher<String> stringPublisher,
                                   Subscriber<String> stringSubscriber) {
    fromPublisher(stringPublisher)
            .map(MoodService::getTweetMessageFrom)
            .flatMap(s1 -> fromArray(splitMessageIntoWords(s1)), maxConcurrency: 1)
            .map(String::toLowerCase)
            .map(MoodAnalyser::getMood)
            .filter(Optional::isPresent)
            .map(mood -> mood.get().name())
            .subscribe(stringSubscriber);
}
```

```java
static String filterMessagesForMoods(String s) {            static void filterMessagesForMoods(Publisher<String> stringPublisher,
    return Stream.of(s)                                                                         Subscriber<String> stringSubscriber) {
        .map(MoodService::getTweetMessageFrom)                  fromPublisher(stringPublisher)
        .flatMap(s1 -> Stream.of(splitMessageI                       .map(MoodService::getTweetMessageFrom)
        .map(String::toLowerCase)                                    .flatMap(s1 -> fromArray(splitMessageIntoWords(s1)), maxConcurrency: 1)
        .map(MoodAnalyser::getMood)                                  .map(String::toLowerCase)
        .filter(Optional::isPresent)                                 .map(MoodAnalyser::getMood)
        .distinct()                                                  .filter(Optional::isPresent)
        .map(mood -> mood.get().name())                              .map(mood -> mood.get().name())
        .collect(Collectors.joining(","));                           .subscribe(stringSubscriber);
}                                                           }
```

```java
static String filterMessagesForMoods(String s) {
    return Stream.of(s)
        .map(MoodService::getTweetMessageFrom)
        .flatMap(s1 -> Stream.of(splitMessageI
        .map(String::toLowerCase)
        .map(MoodAnalyser::getMood)
        .filter(Optional::isPresent)
        .distinct()
        .map(mood -> mood.get().name())
        .collect(Collectors.joining(","));
}
```

```
19    25
20    26
21    27
22    28
23    29
24    30
25    31
26    32
27    33
28    34
29    35
30    36
```

```java
static void filterMessagesForMoods(Publisher<String> stringPublisher,
                                   Subscriber<String> stringSubscriber) {
    fromPublisher(stringPublisher)
        .map(MoodService::getTweetMessageFrom)
        .flatMap(s1 -> fromArray(splitMessageIntoWords(s1)), maxConcurrency: 1)
        .map(String::toLowerCase)
        .map(MoodAnalyser::getMood)
        .filter(Optional::isPresent)
        .map(mood -> mood.get().name())
        .subscribe(stringSubscriber);
}
```

RxJava for easy concurrency and backpressure

December 19, 2016   Anton Arhipov   2 Comments

Tweet   Like 19   Share 41   G+1 4

RxJava is growing in popularity in Android application development. Howev
also very capable for server-side apps. RxJava makes concurrency easier ev
though a seasoned Java developer would have to re-learn the concepts to

How to limit the concurrency of flatMap?

▲

4

▼

★

2

I'm trying to use RxJS to write a script to process several hundreds of log files, each of which is about 1GB. The skeleton of the script looks like

```
Rx.Observable.from(arrayOfLogFilePath)
.flatMap(function(logFilePath){
    return Rx.Node.fromReadStream(logFilePath)
        .filter(filterLogLine)
})
.groupBy(someGroupingFunc)
.map(someFurtherProcessing)
.subscribe(...)
```

The code works, but notice that the filtering step of all log files will start concurrently. However, from file system IO performance perspective, it is preferable to process one file after another (or at least

▼ OK Test Results 322ms
  ▼ OK MoodServiceTest 322ms
    OK should correctly identify happy messages that are not lower case 264ms
    OK should correctly identify mixed messages with multiple moods 8ms
    OK should not have any mood for messages that are neither happy or sad 5ms
    OK should correctly identify happy messages 12ms
    OK should correctly identify mixed messages 17ms
    OK should correctly identify sad messages 16ms

Step 5: Update CV

Featured Skills & Endorsements   Add a new skill

Java   99+

Endorsed by Peter Lawrey and 24 others who are highly skilled at this

Endorsed by 5 of Trisha's colleagues at MongoDB

MongoDB   47

Add new profile section

Continuous Integr...

Trisha is also good at...

Software Develop...

MySQL   22

Agile Project Man...

Scrum   29

Design Patterns   8

JavaScript   9

Continuous Delive...

Ant   2

API Development

Community   4

Add Skill   ✕

React

React

Polymerase Chain Reaction (PCR)

Real-Time Polymerase Chain Reaction (qPCR)

Reverse Transcription Polymerase Chain Reaction (RT-PCR)

React.js

Reactor

Reaction Engineering

Yes   ⬤   **Share profile changes**
If enabled, your network may see this change.

Add

In Summary

Step 1: Awareness

Find Zen in Surfing the Tsunami

Where?

- Twitter
- Newsletters
- User Groups
- Blogs & Tutorials

Step 2: Speaking the Lingo

"Wouldn't a Reactive approach solve that problem?"

Step 3: Enough Knowledge to be Dangerous

Guideline: Enough to blag your way through a conversation in the pub

Where?

- Twitter
- Newsletters
- User Groups
- Blogs & Tutorials
- Online courses
- Conferences

Step 4: Code!

Remember: You must complete Steps 1-3 first

Where?

- Twitter
- Newsletters
- User Groups
- Blogs & Tutorials
- Online courses
- Conferences
- Books
- StackOverflow

How?

- Pet project
- Join an open source project
- Find a project at work

Step 5: Update CV

Or LinkedIn

Machine Learning	Security	Scalability	**Design Patterns**	TensorFlow
Go	**HTML5**	**AngularJS**	**Git**	**TDD**
Big Data	Cloud	**Microservices**	Performance	**NoSQL**
Reactive	Serverless	Blockchain	Resiliance	Distributed Systems
RxJava	Reactor	React	Kotlin	**Agile**

http://bit.ly/buzz-pc