

**JET
BRAINS**



Real World Java 9



Trisha Gee (@trisha_gee)

Developer & Technical Advocate, JetBrains

Real World Java 9

- Jigsaw in Action
- Reactive Programming in Java 9
- The Other Good Stuff

Why Java 9?



Free Lunch!

- Better Use of Memory
- Better Performance of:
 - Locking
 - Secure Apps
 - Graphics
- Better Use of Hardware
- Better Documentation
- Prettier Graphics (Windows and Linux)
- Faster Compilation

Jigsaw

REPL

And... other things?

[274: Enhanced Method Handles](#)
[275: Modular Java Application Packaging](#)
[276: Dynamic Linking of Language-Defined Object Models](#)
[277: Enhanced Deprecation](#)
[278: Additional Tests for Humongous Objects in G1](#)
[279: Improve Test-Failure Troubleshooting](#)
[280: Indify String Concatenation](#)
[281: HotSpot C++ Unit-Test Framework](#)
[282: jlink: The Java Linker](#)
[283: Enable GTK 3 on Linux](#)
[284: New HotSpot Build System](#)
[285: Spin-Wait Hints](#)
[287: SHA-3 Hash Algorithms](#)
[288: Disable SHA-1 Certificates](#)
[289: Deprecate the Applet API](#)
[290: Filter Incoming Serialization Data](#)
[292: Implement Selected ECMAScript 6 Features in Nashorn](#)
[294: Linux/s390x Port](#)
[295: Ahead-of-Time Compilation](#)
[297: Unified arm32/arm64 Port](#)
[298: Remove Demos and Samples](#)

Caveats & Disclaimers



Java 9 in Production?

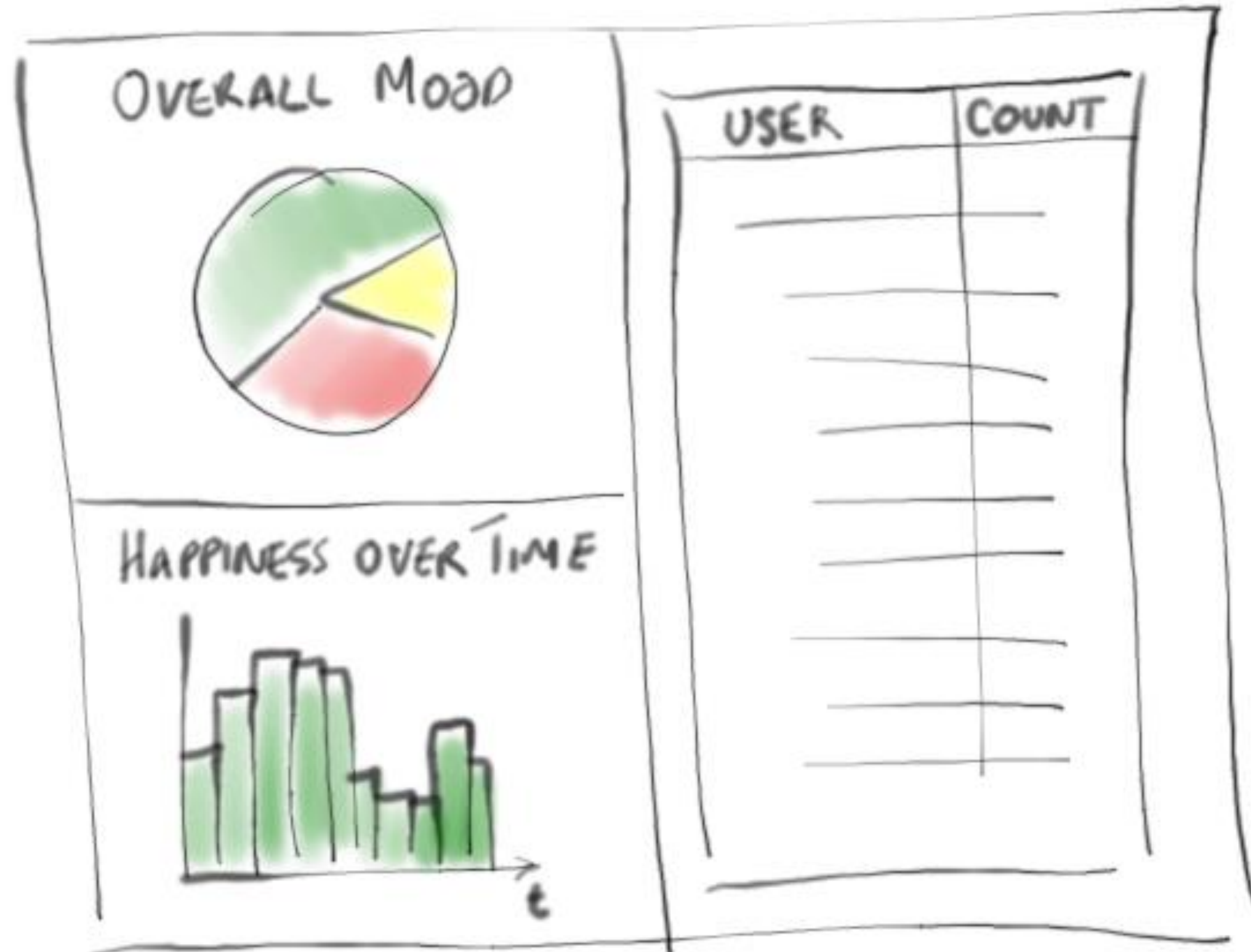
Questionable

Everything is EA

The Case Study



Twitter Dashboard



TWITTER
FIREHOSE

CONNECT
&
EXTRACT

MOOD
SERVICE

OVERALL
MOOD
MVC

HAPPINESS
MVC

USER
SERVICE

LEADERBOARD
MVC



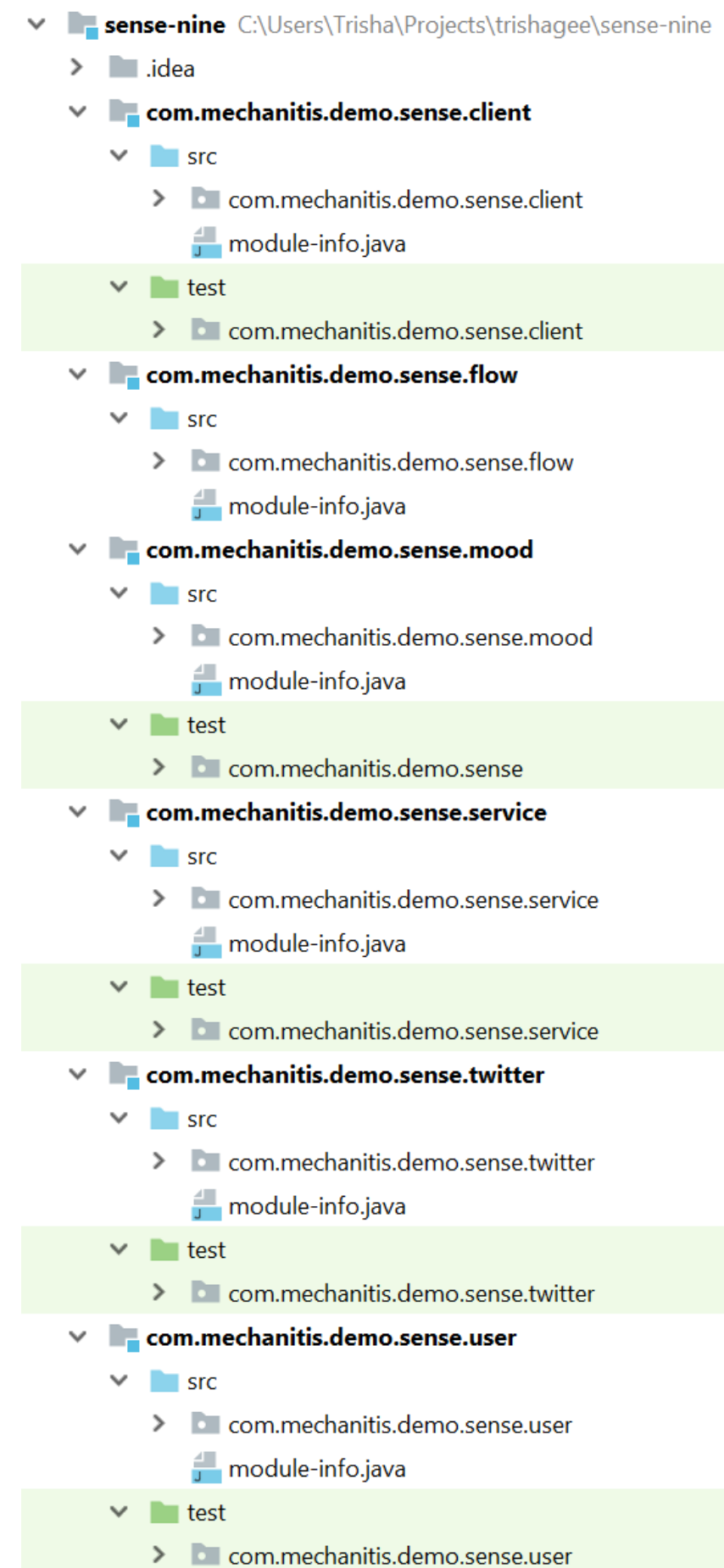
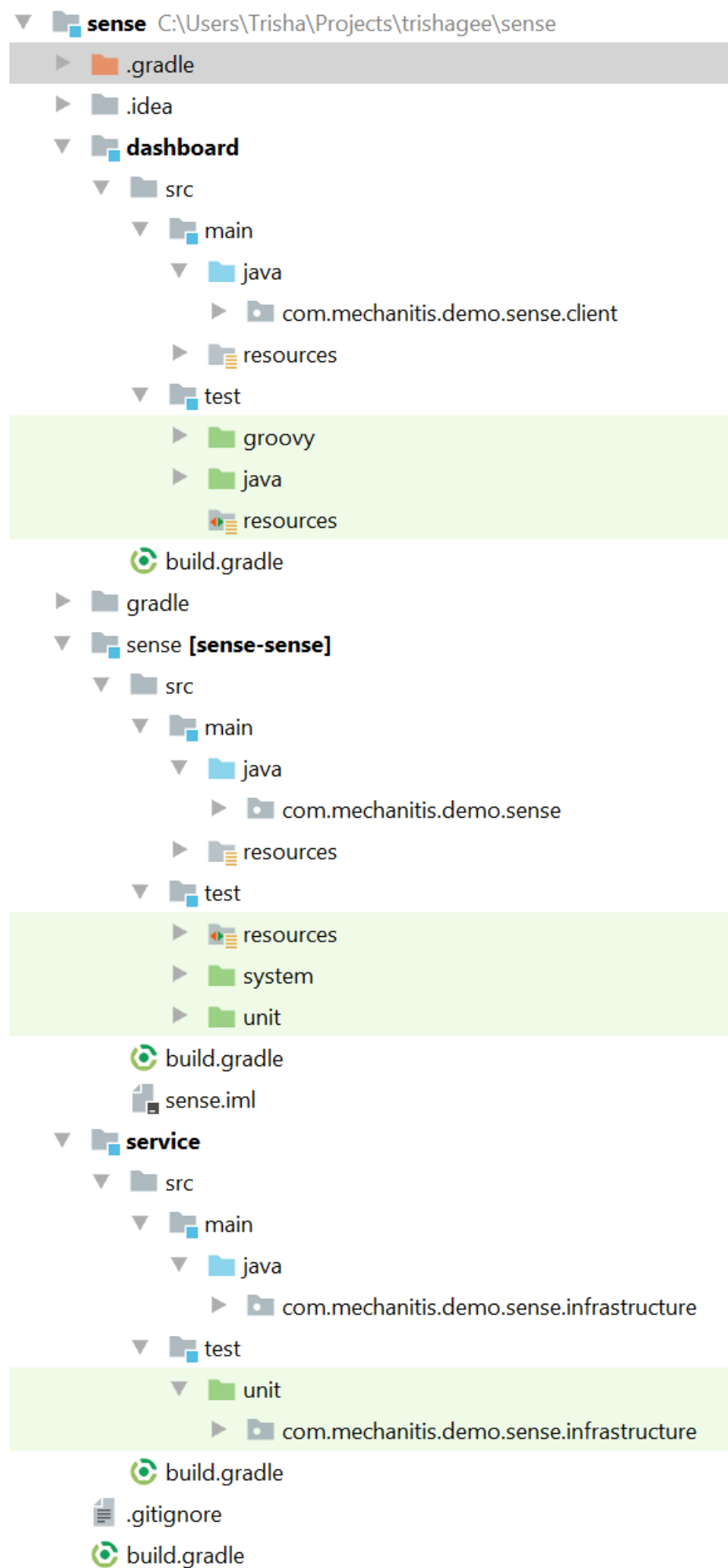
<http://bit.ly/Java8InAnger>

Compiling with Java 9



Jigsaw





▼  **jdk9** sources root, C:\Users\Trisha\Projects\sa

- ▶  .idea
- ▶  jcheck
- ▶  make
- ▼  src
 - ▶  bsd.doc.man
 - ▶  demo
 - ▶  **java.base**
 - ▶  **java.datatransfer**
 - ▶  **java.desktop**
 - ▶  **java.instrument**
 - ▶  **java.logging**
 - ▶  **java.management**
 - ▶  **java.management.rmi**
 - ▶  **java.naming**
 - ▶  **java.prefs**
 - ▶  **java.rmi**
 - ▶  **java.scripting**
 - ▶  **java.se**
 - ▶  **java.se.ee**
 - ▶  linux.doc.man
 - ▶  sample
 - ▶  solaris.doc.sun.man.man1

- ▼  test
 - ▶  com
 - ▶  **demo**
 - ▼  **java [test1]**
 - ▶  **awt**
 - ▶  **beans**
 - ▶  io
 - ▶  lang
 - ▶  **math**
 - ▶  net
 - ▶  **nio [nio3]**
 - ▶  **rmi**
 - ▶  **security [security3]**
 - ▶  sql
 - ▶  **text**

▼  **sense-nine** C:\Users\Trisha\Projects\trishagee\sense-nine

- >  .idea
- ▼  **com.mechanitis.demo.sense.client**
 - ▼  src
 - >  com.mechanitis.demo.sense.client
 -  module-info.java

- ▼  test
 - >  com.mechanitis.demo.sense.client

- ▼  **com.mechanitis.demo.sense.flow**
 - ▼  src
 - >  com.mechanitis.demo.sense.flow
 -  module-info.java

- ▼  **com.mechanitis.demo.sense.mood**
 - ▼  src
 - >  com.mechanitis.demo.sense.mood
 -  module-info.java

- ▼  test
 - >  com.mechanitis.demo.sense

- ▼  **com.mechanitis.demo.sense.service**
 - ▼  src
 - >  com.mechanitis.demo.sense.service
 -  module-info.java

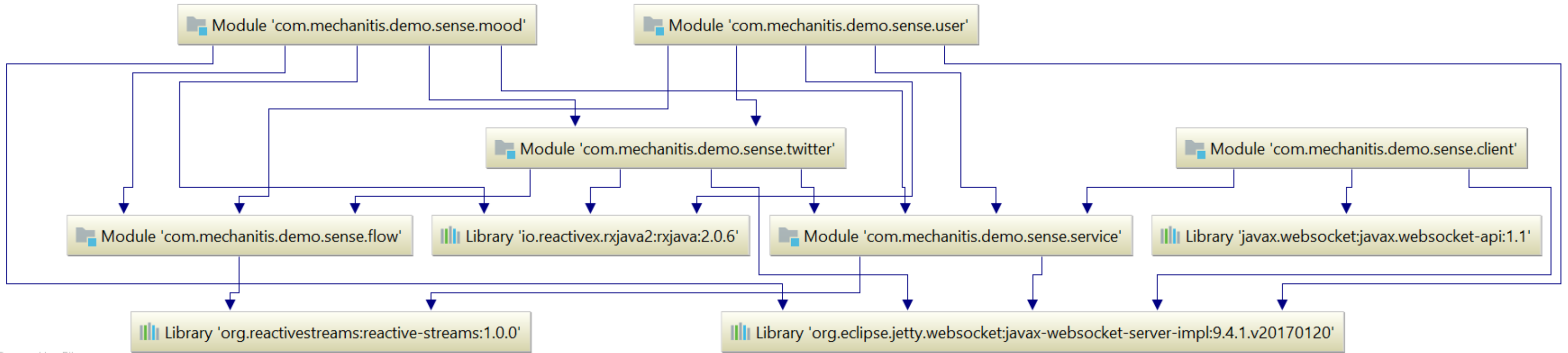
- ▼  test
 - >  com.mechanitis.demo.sense.service

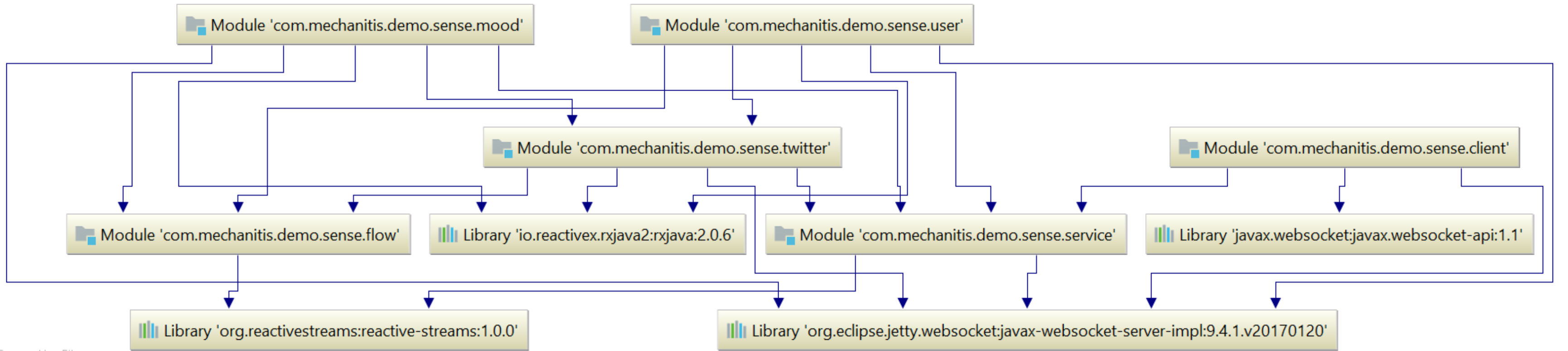
- ▼  **com.mechanitis.demo.sense.twitter**
 - ▼  src
 - >  com.mechanitis.demo.sense.twitter
 -  module-info.java

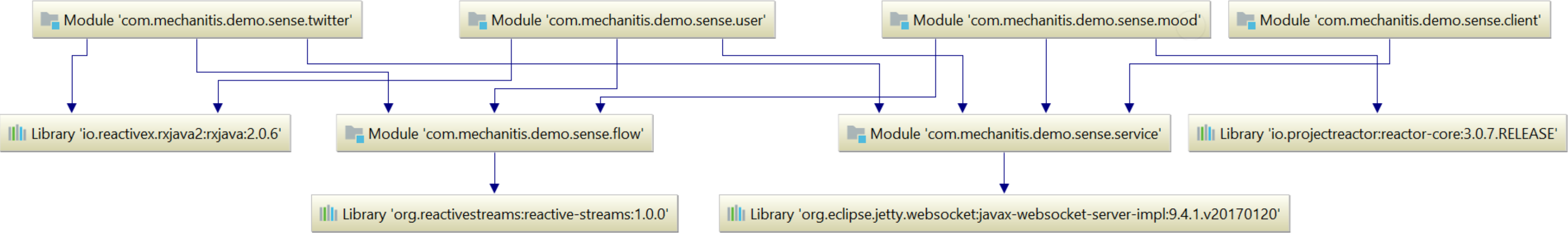
- ▼  test
 - >  com.mechanitis.demo.sense.twitter

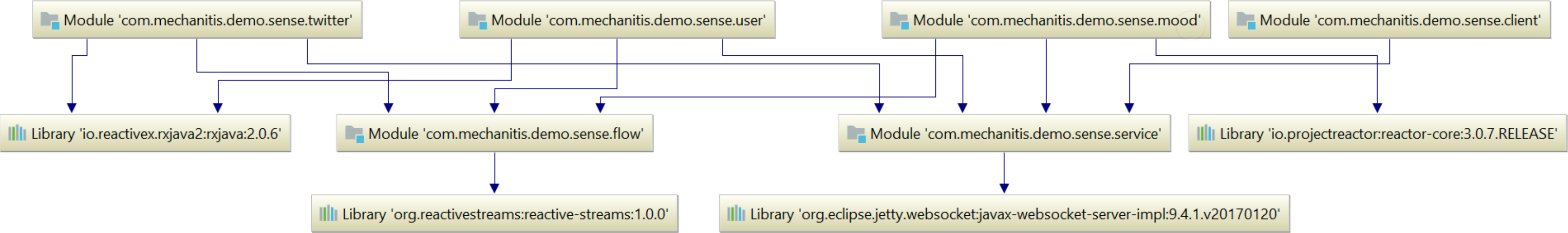
- ▼  **com.mechanitis.demo.sense.user**
 - ▼  src
 - >  com.mechanitis.demo.sense.user
 -  module-info.java

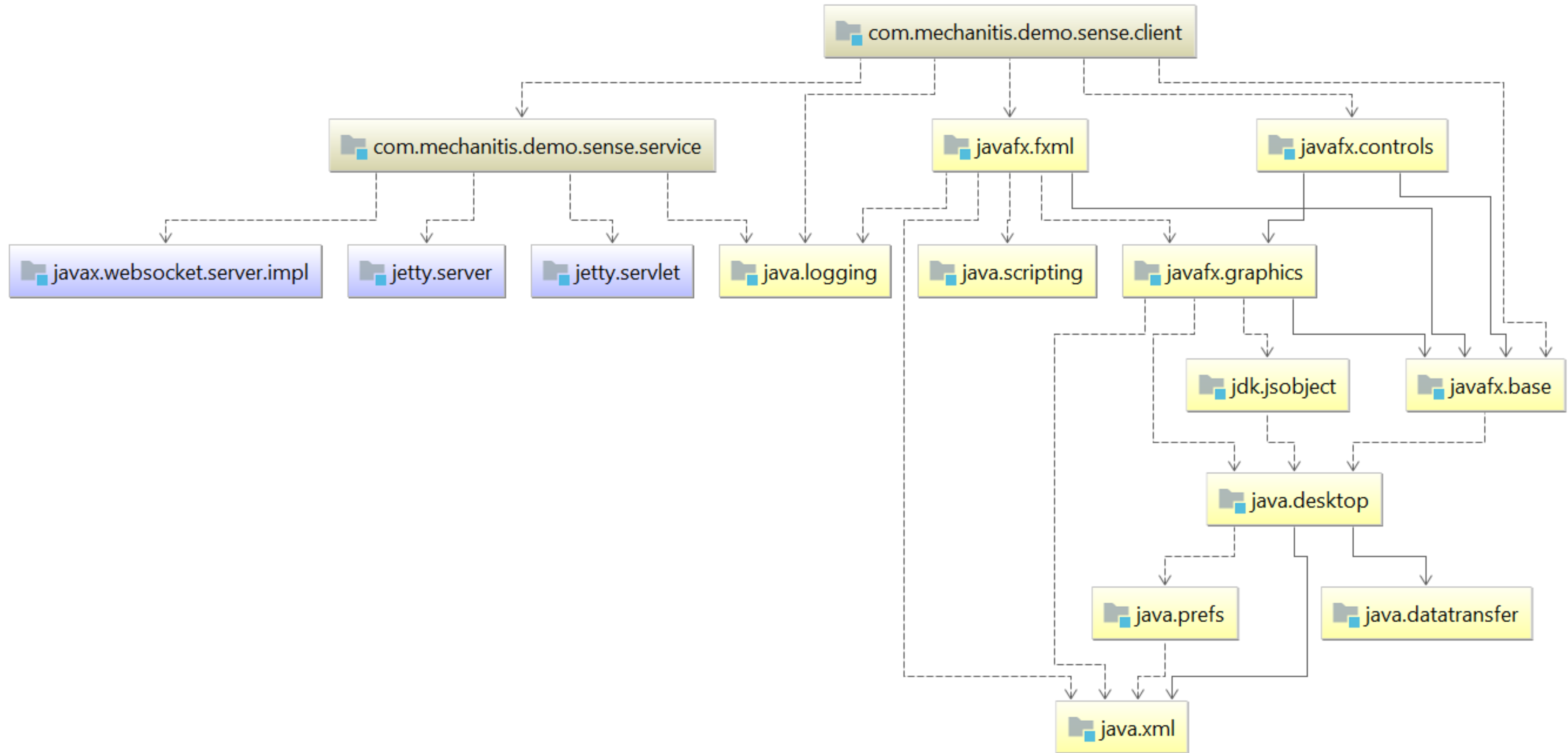
- ▼  test
 - >  com.mechanitis.demo.sense.user











module-info.java

Pain

- Some structural reorganisation may be needed
- This project has no command-line build
- IDE support is not what we're used to
- RTFM. No, really do.

Gain

- Cleaner design
- Smaller deployables

Reactive API



Reactive Streams API

▼  org.reactivestreams:reactive-streams:1.0.0

▼  reactive-streams-1.0.0.jar library root

▶  META-INF

▼  org.reactivestreams

 Processor

 Publisher

 Subscriber

 Subscription

Implementations

- [Akka Streams](#)
- [MongoDB Reactive Java Driver](#)
- [Ratpack](#)
- [Reactive Rabbit](#)
- [Reactor](#)
- [RxJava](#)
- [Vert.x](#)

Java 9 Flow API

▼  < 9-ea > C:\Program Files\Java\jdk-9

▼  java.base library root

▼  java.util.concurrent

▼  Flow

 Processor

 ▶  Publisher

 ▶  Subscriber

 ▶  Subscription

▼  < 9-ea > C:\Program Files\Java\jdk-9

▼  java.base library root

▼  java.util.concurrent

▼  Flow

▶  Processor

▶  Publisher

▶  Subscriber

▶  Subscription

▼  org.reactivestreams:reactive-streams:1.0.0

▼  reactive-streams-1.0.0.jar library root

▶  META-INF

▼  org.reactivestreams

 Processor

 Publisher

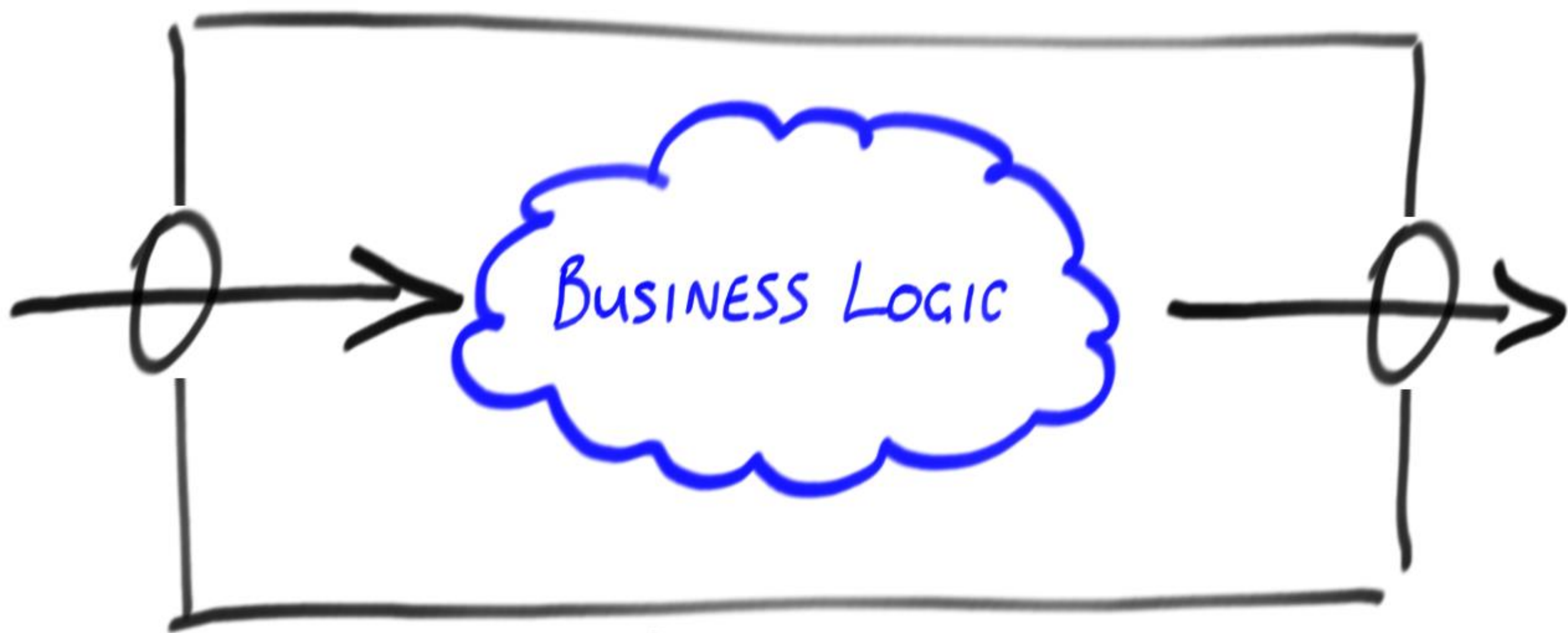
 Subscriber

 Subscription

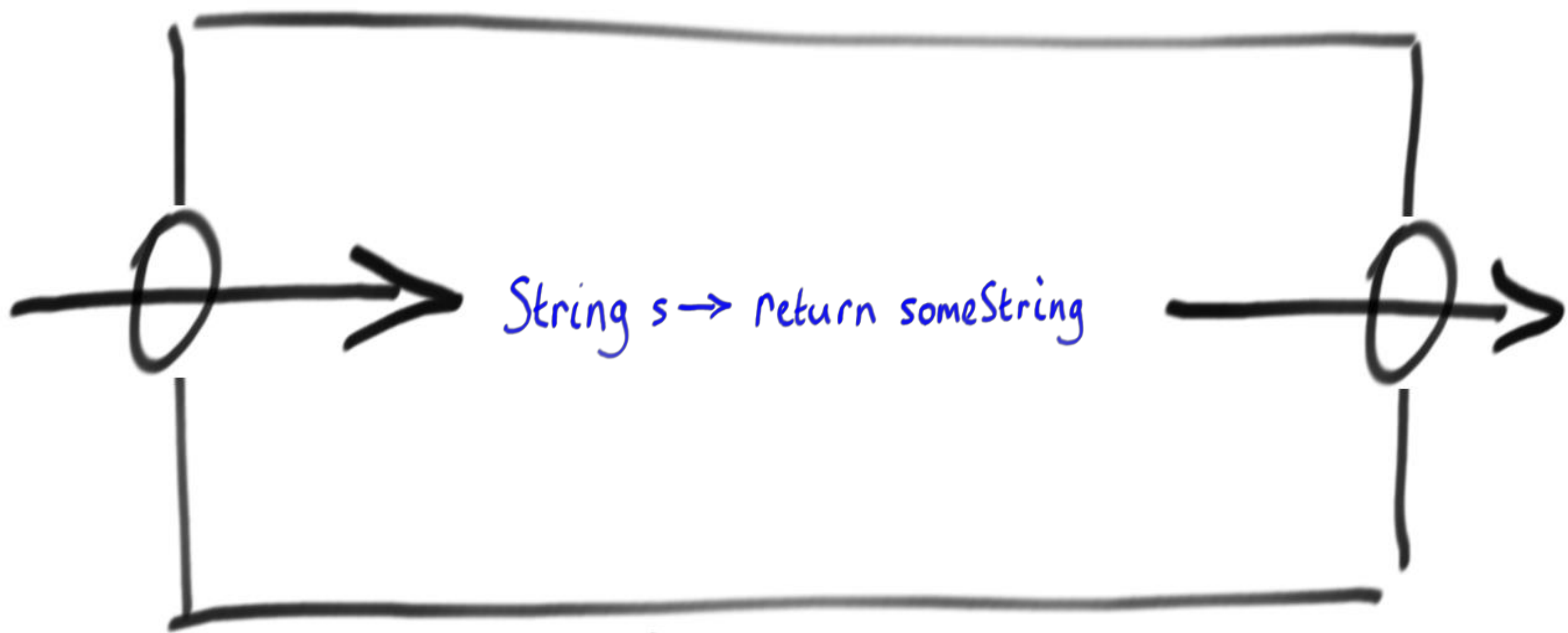
Where to use it?



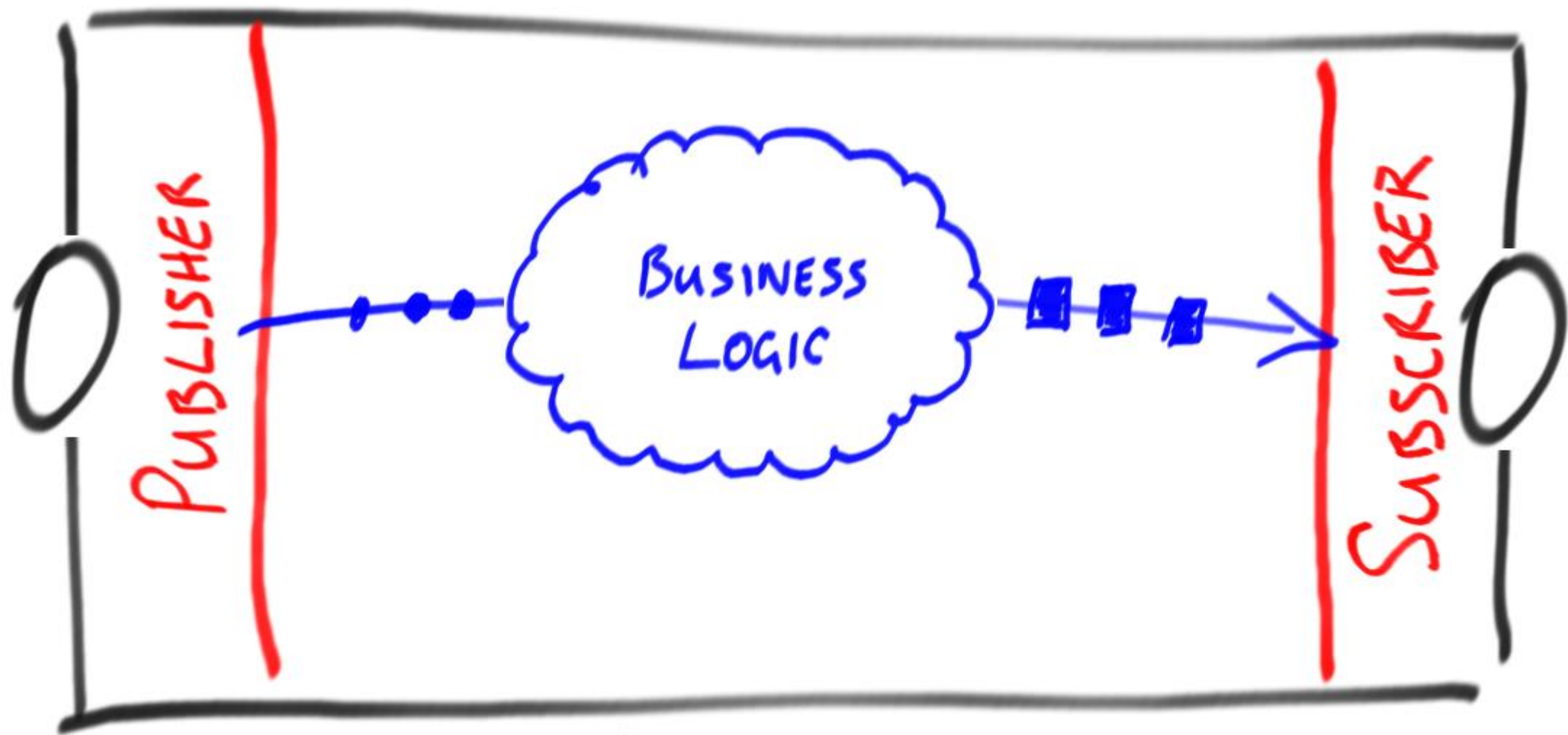
SERVICE



SERVICE



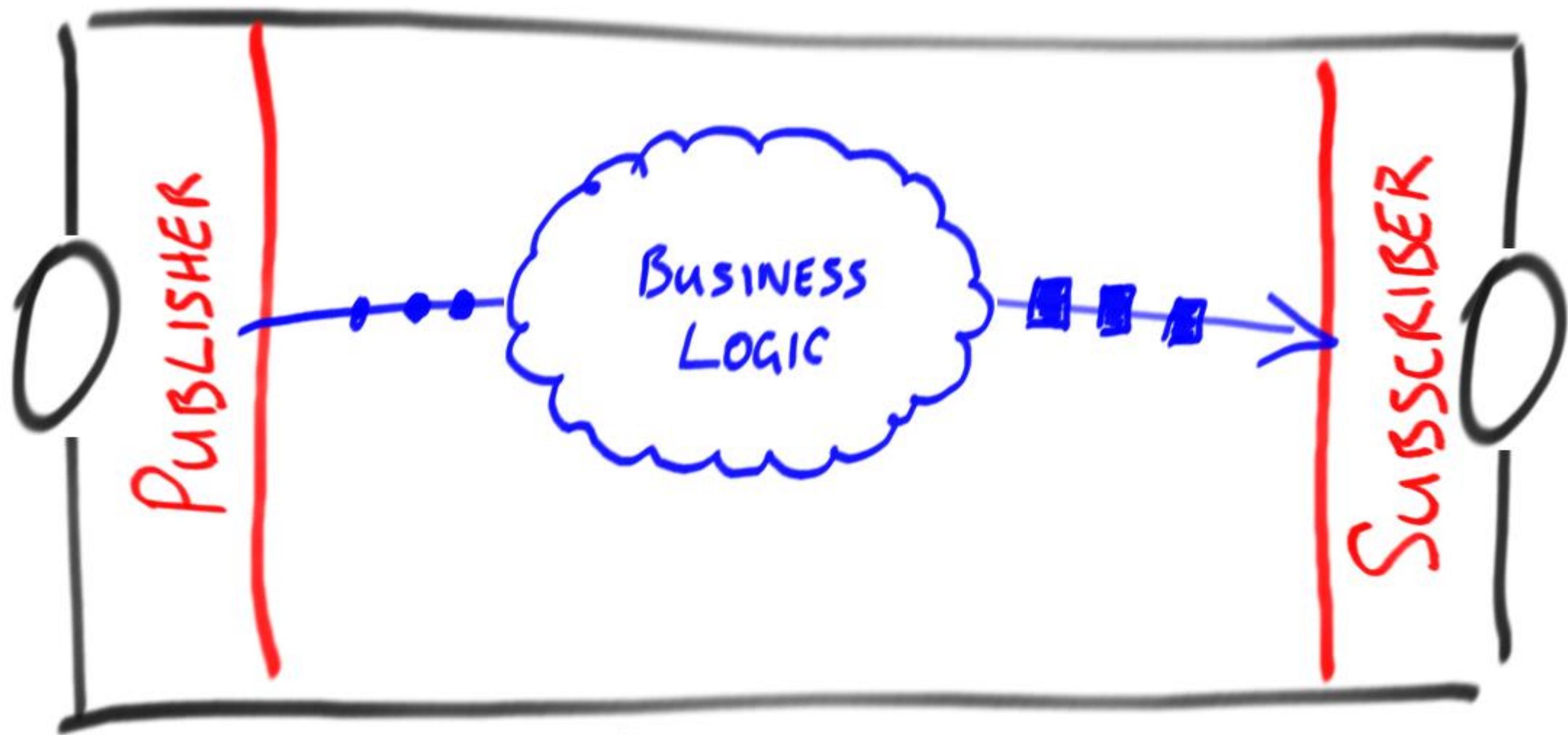
SERVICE



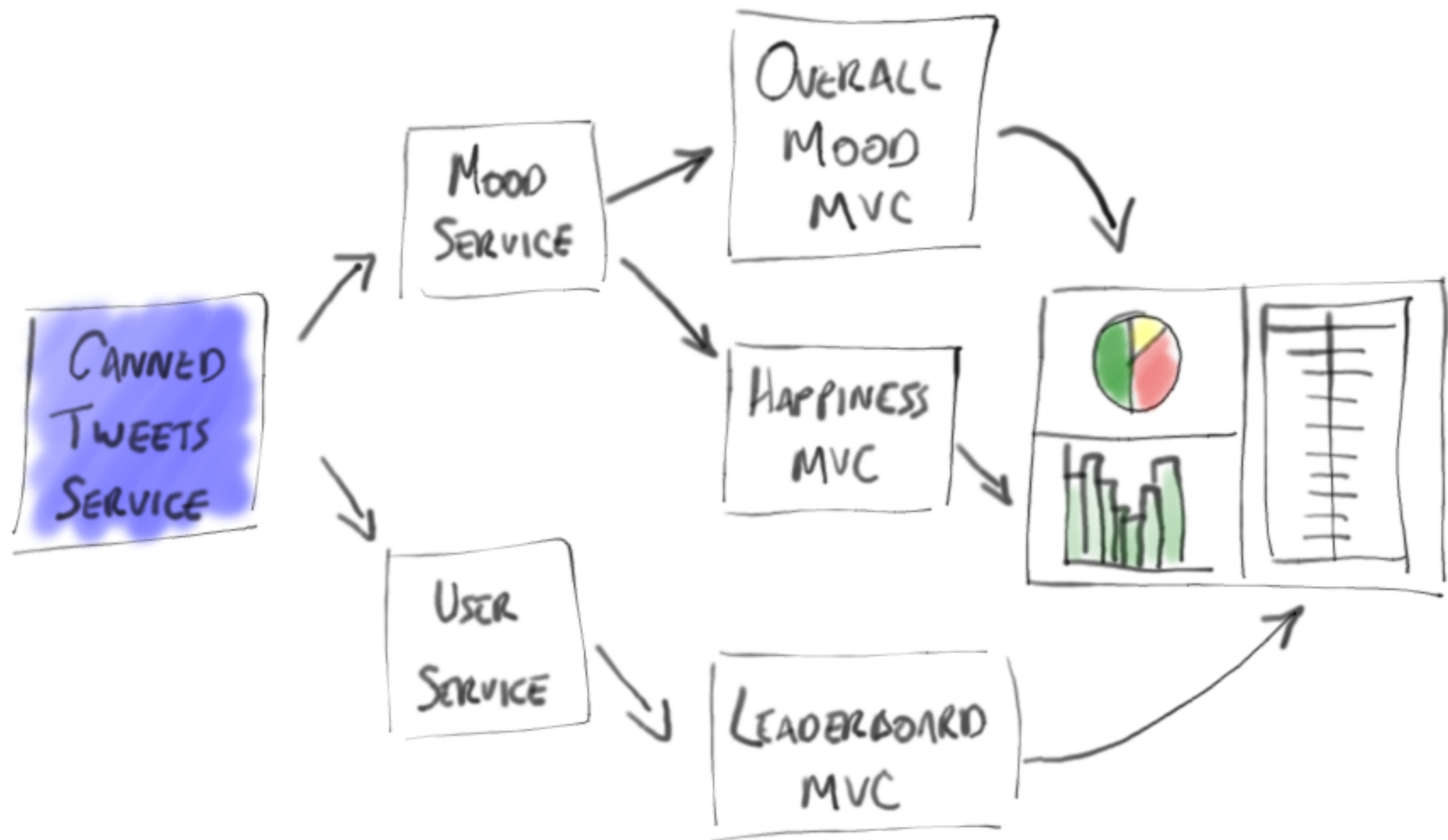
SERVICE

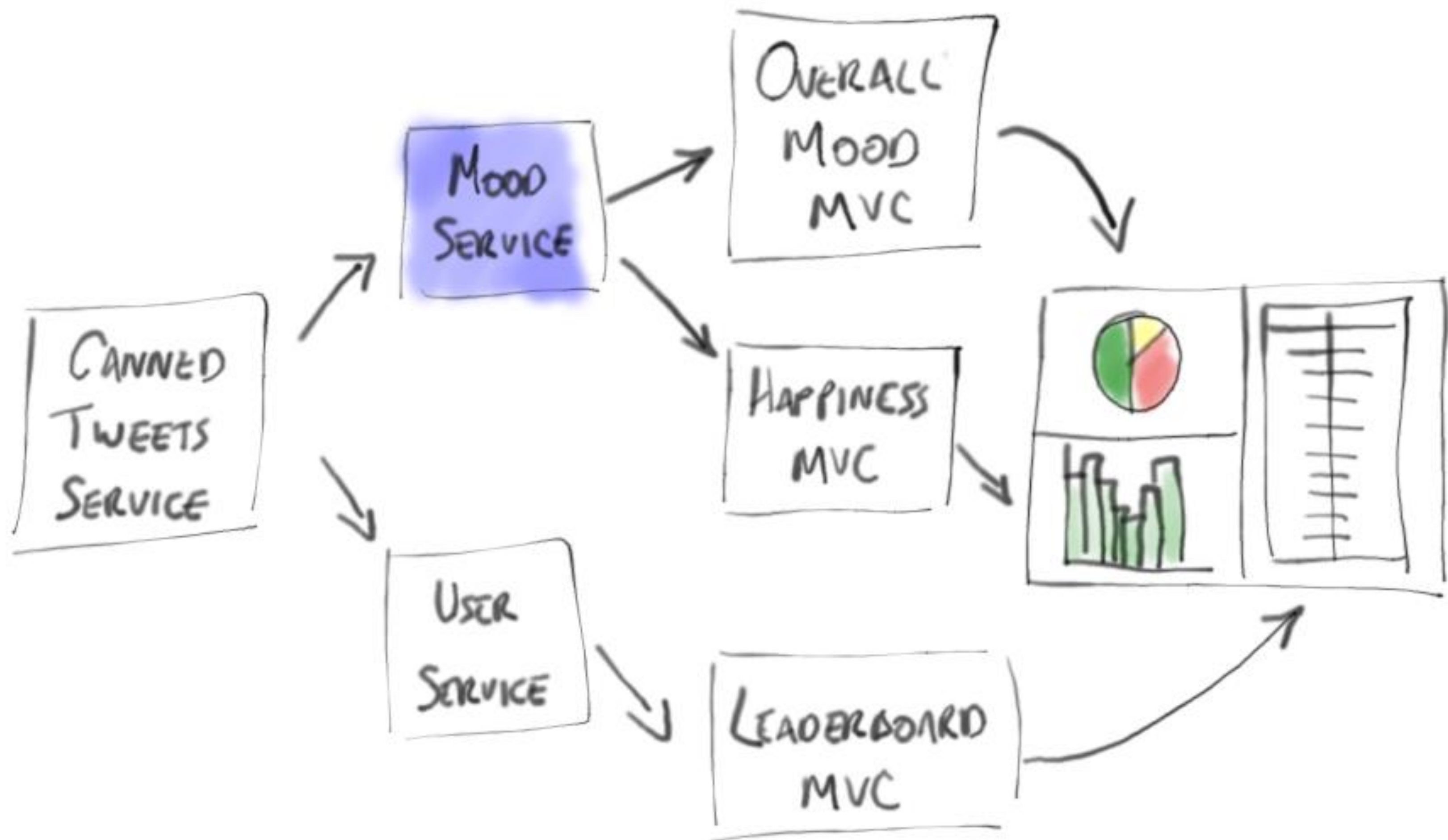
Reactive In Practice





SERVICE





Java 8 vs Reactive



```
try (Stream<String> lines = lines(filePath)) {  
    lines.filter(s -> !s.equals("OK"))  
        .peek(s -> this.addArtificialDelay())  
        .forEach(tweetsEndpoint::onNext);  
}
```

```
try {  
    Flowable.fromIterable(Files.readAllLines(filePath)  
        .filter(s -> !s.equals("OK"))  
        .doOnNext(s -> this.addArtificialDelay()))  
        .forEach(tweetsEndpoint::onNext);  
}
```

Reactive

- Work with multiple streams
- Have multiple consumers
- Backpressure
- Very different concurrency model
- Reactive API lets you work with different implementations

Other Java 9 Features



269: Convenience Factory Methods for Collections

213 : Private Methods on Interfaces

New Methods on the Streams API

259 : Stack-Walking API

238 : Multi-Release JAR Files

102 : Process API Updates

277 : Updated Deprecation

224 : HTML5 Javadoc

225 : Javadoc search

Java™ Platform, Standard Edition Development Kit (JDK™) 9

API Specification

This document is the API specification for the Java™ Platform, Standard Edition.

Modules	
Module	Description
java.activation	Defines the JavaBeans Activation Framework (JAF) API.
java.base	Defines the foundational APIs of the Java SE Platform.
java.compiler	Defines the Language Model, Annotation Processing, and Java Compiler APIs.
java.corba	Defines the Java binding of the OMG CORBA APIs, and the RMI-IIOP API.
java.datatransfer	Defines the API for transferring data between and within applications.
java.desktop	Defines the AWT and Swing user interface toolkits, plus APIs for accessibility, audio, imaging, printing, and JavaBeans.
java.instrument	Defines services that allow agents to instrument programs running on the JVM.
java.jnlp	Defines the API for Java Network Launch Protocol (JNLP).
java.logging	Defines the Java Logging API.
java.management	Defines the Java Management Extensions (JMX) API.
java.management.rmi	Defines the RMI Connector for the Java Management Extensions (JMX) Remote API.
java.naming	Defines the Java Naming and Directory Interface (JNDI) API.
java.prefs	Defines the Preferences API.
java.rmi	Defines the Remote Method Invocation (RMI) API.
java.scripting	Defines the Scripting API.
java.se	Defines the core Java SE API.
java.se.ee	Defines the full API of the Java SE Platform.
java.security.jgss	Defines the Java binding of the IETF Generic Security Services API (GSS-API).

JShell (REPL)



https://java.net/downloads/adoptopenjdk/REPL_Tutorial.pdf

We'd Love Your Help

<https://youtrack.jetbrains.com/issue/IDEA-161611>



The Pain



Your code might break!

Try it out now to find out sooner, not later

Going Away

- 214: Remove Deprecated GC Combinations
- 231: Remove Launch-Time JRE Version Selection
- 240: Remove the JVM TI hprof Agent
- 241: Remove the jhat Tool
- 260: Encapsulate Most Internal APIs
- 289: Deprecate the Applet API
- 298: Remove Demos and Samples

Possible Unexpected Behaviour

- 158 : Unified JVM Logging
- 223 : New Version-String Scheme
- 245 : Validate JVM Command-Line Flag Arguments
- 248 : Make G1 the Default Garbage Collector
- 271 : Unified GC Logging

Java 9 is still evolving

So are the tools we use

Don't rush in to modularisation

It might not be for you

Reactive Streams are not the same as Java 8 Streams

Different behaviour, different use cases

The Gain



Encourages good design

Encourages new ways of working

Just a bit nicer to write code



<http://bit.ly/RealJava9>