Reactive Microservices with Eclipse Vert.x

Burr Sutter

RED HAT
DEVELOPERS

Reactive Microservices with Eclipse Vert.x

@burrsutter

vertx.io

http://developers.redhat.com

http://bit.ly/reactivemsa

Change History

0.1 - Forked from bit.ly/helloworldmsa

1.0 - JavaOne 2016

2.0 - http://javaday.org.ua/kyiv/

2.1 - Added “Be This Tall” and 2 Gene Kim books slides

2.2 - Changed color scheme/art on several slides

2.3 - reordered slides, added links to mono2micro data book and Reactive Tutorial, updated getting started section

RED HAT DEVELOPERS

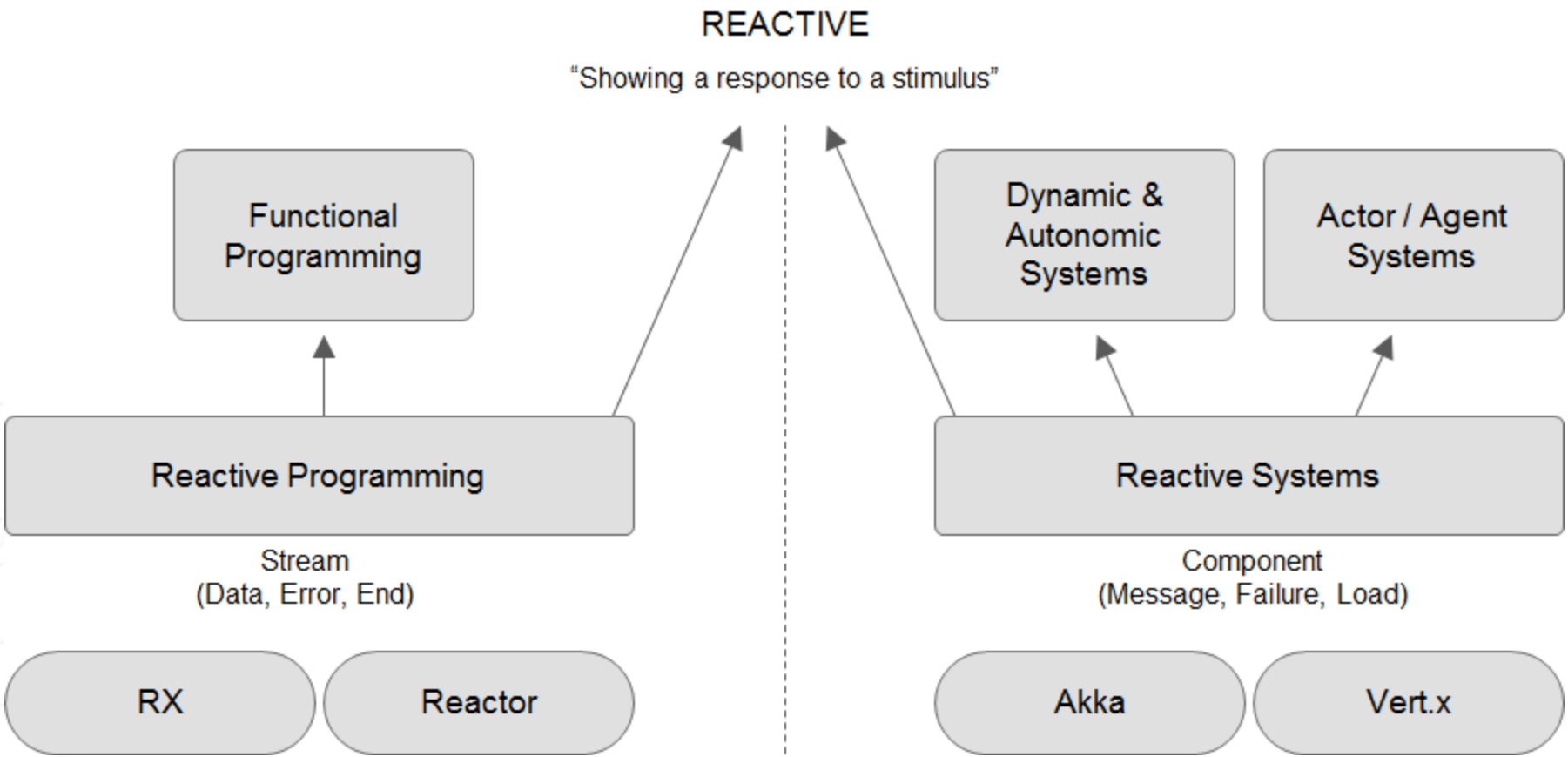
developers.redhat.com

Developer Evolution :-)

70s	80s	90s	00s	10s	
					
COBOL JCL WFL	C/C++ 4GLs CASE Netware RDBMS SQL Unix	HTTP/HTML CGI GET/POST Cookies Java Servlet EJB Windows NT Solaris/AIX OOP	MVC - Struts DI - Spring ORM - Hibernate XML WS-* JSF AJAX Agile Automated Testing CI SVN Linux	HTML5 Mobile iOS/Android Maven/Gradle Git Node.js MongoDB/Redis Hadoop/Spark *-aaS/Cloud **Reactive** Functional	Mythical Beast

Reactive

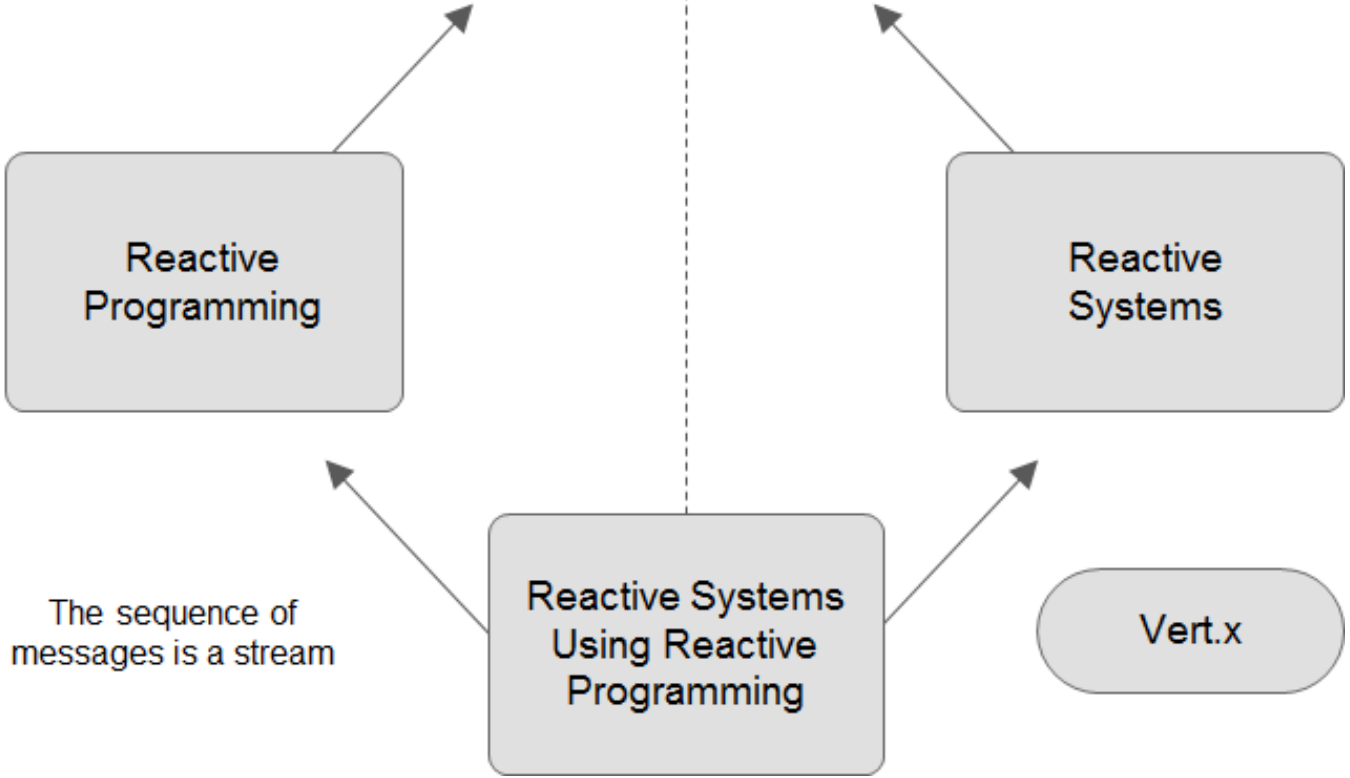
RED HAT DEVELOPERS



RED HAT DEVELOPERS

REACTIVE

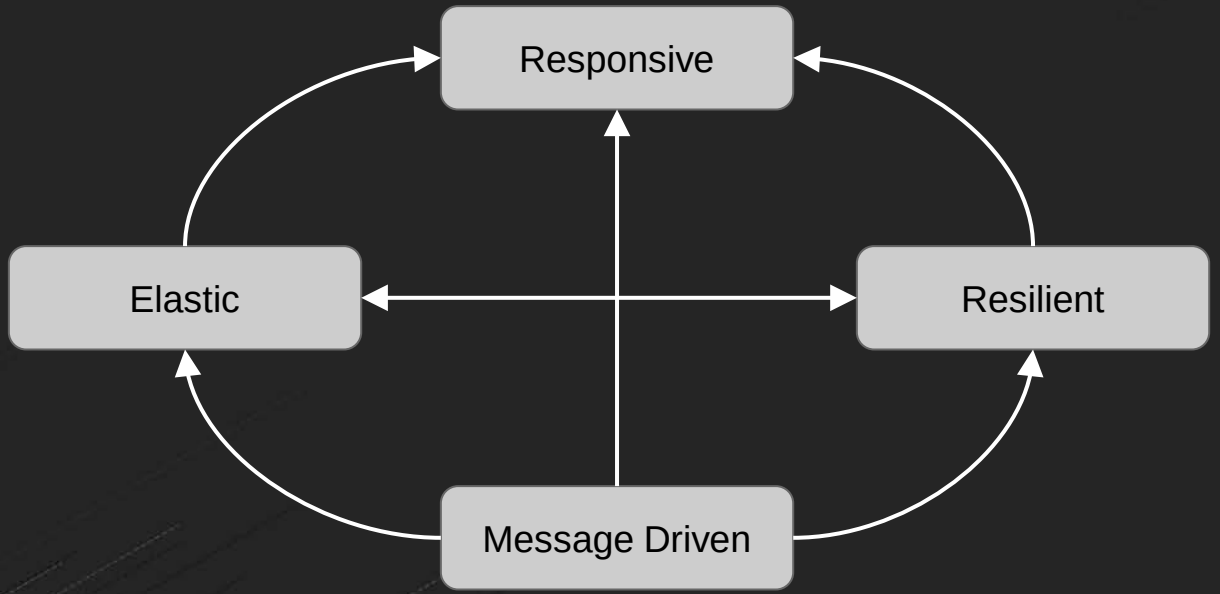
"Showing a response to a stimulus"



RED HAT DEVELOPERS

Reactive Manifesto

Reactive Systems (not Programming)



http://www.reactivemanifesto.org/

RED HAT DEVELOPERS

Responsive Demo

https://github.com/redhat-helloworld-msa/goodbye

RED HAT DEVELOPERS

Microservices

RED HAT DEVELOPERS

You Must Be This Tall

1. Self-Service, on-demand, elastic infrastructure as code

   (how many days/weeks to provision a new VM?)

2. Dev vs Ops

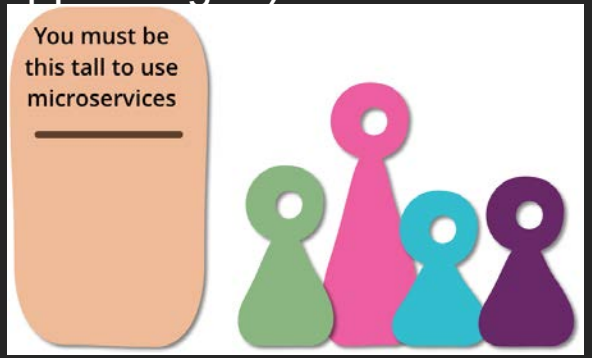
   (who is on the pager for production app outage?)

3. Automation

   (phoenix vs snowflake?)

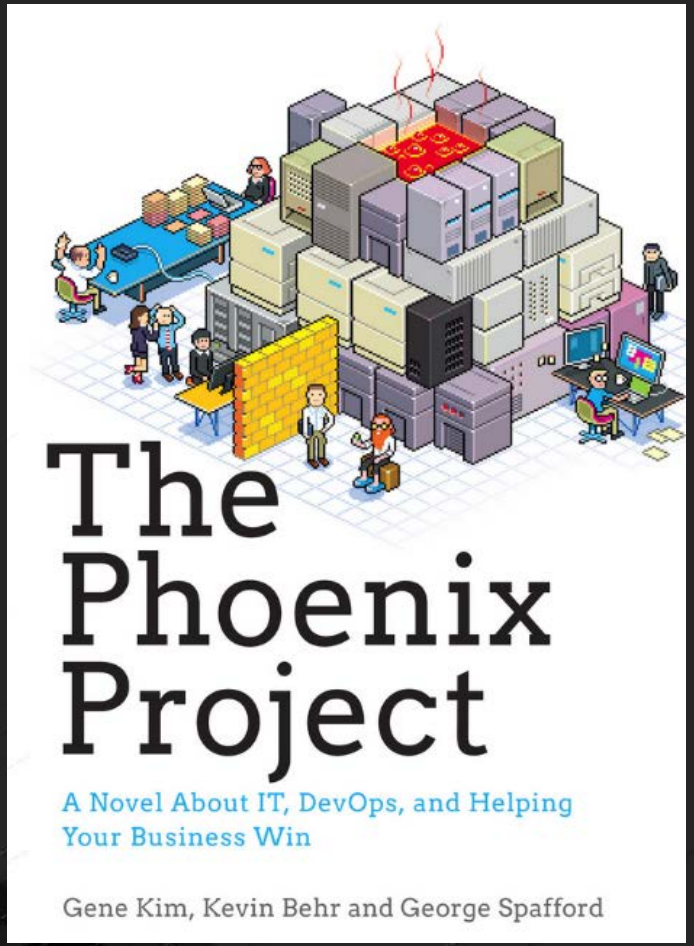
4. CI & CD

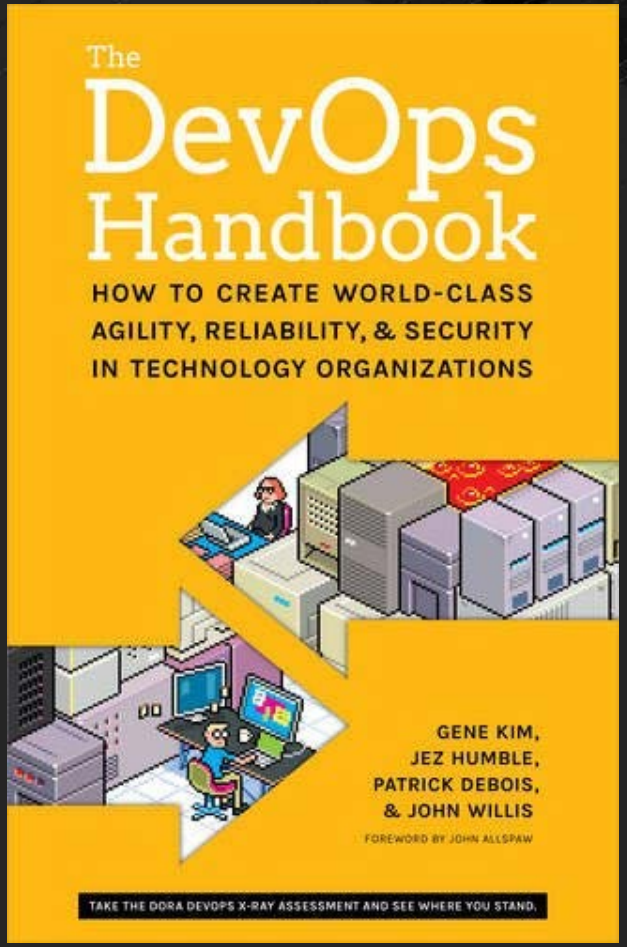
5. Deployment Pipeline



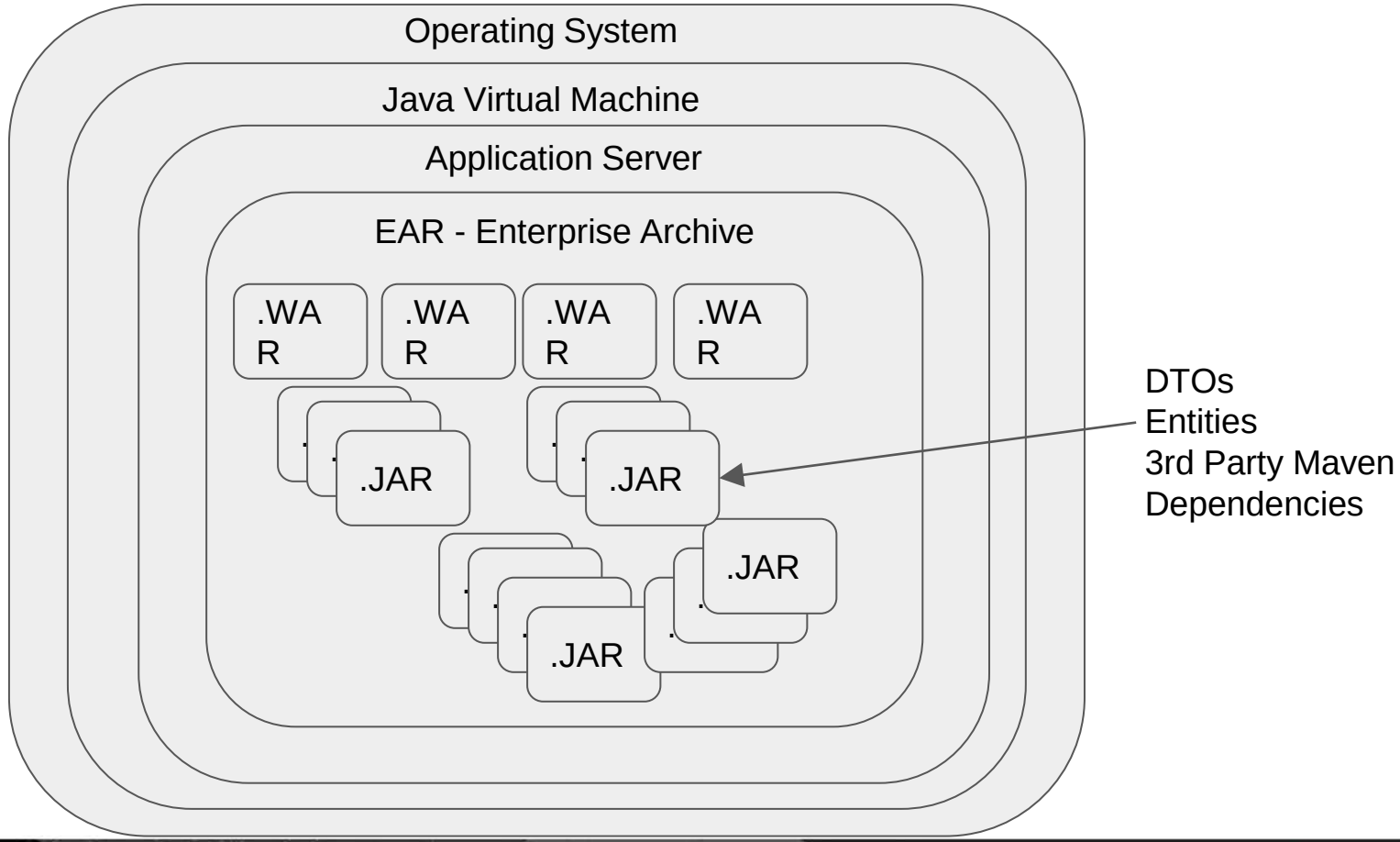
http://martinfowler.com/bliki/MicroservicePrerequisites.html

RED HAT DEVELOPERS

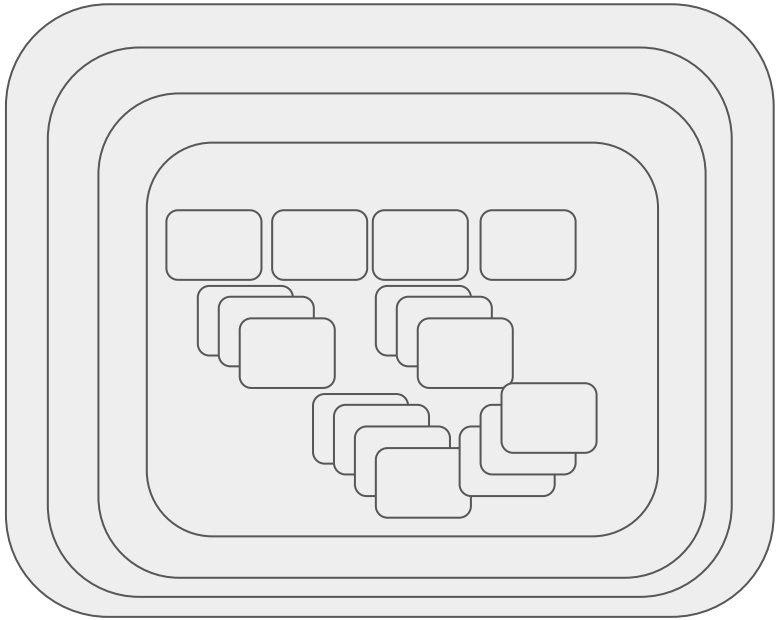




RED HAT DEVELOPERS



RED HAT DEVELOPERS



Programmers
(18)

Business
Analysts
(4)

Project
Managers
(2)

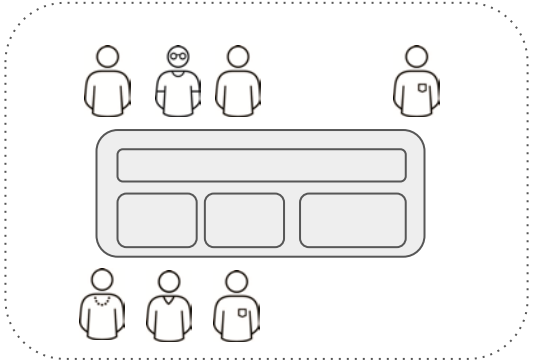
Quality
Assurance
(6)

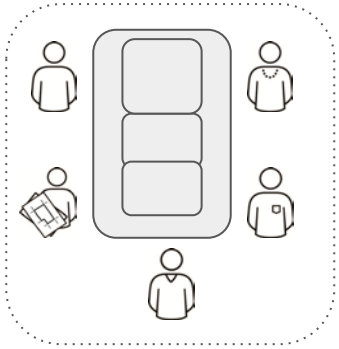
Compliance
(2)

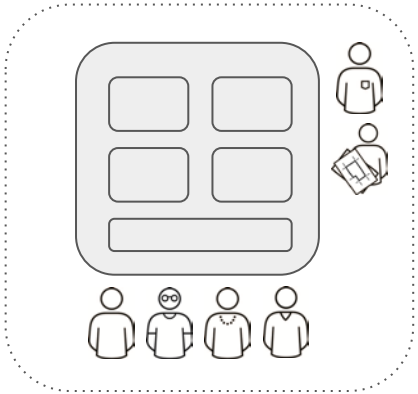
Operators
(6)

DBAs
(3)

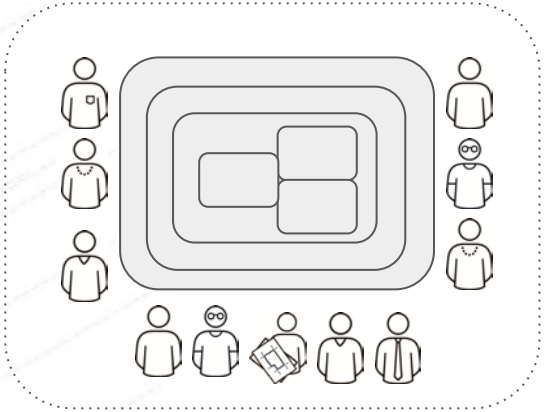
RED HAT DEVELOPERS

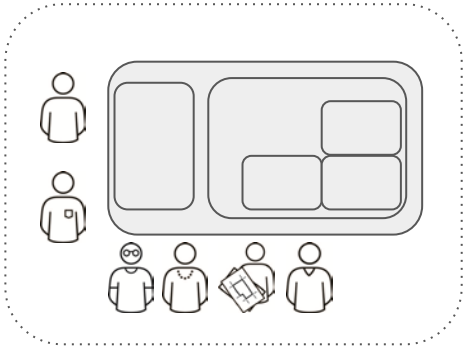


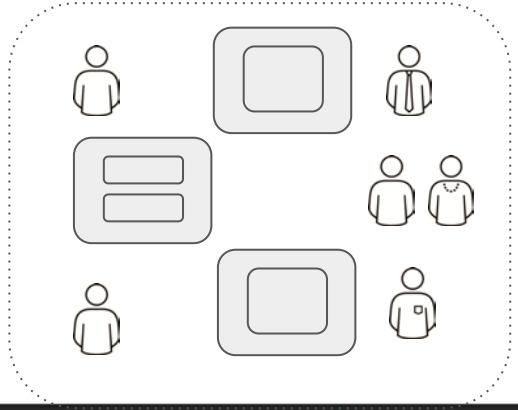




2-Pizza Teams
You Build It, You Own it







RED HAT DEVELOPERS

Microservice Principles/Characteristics

1. Deployment **Independence**: updates to an individual microservice have no negative impact to any other component of the system. Optimized for **Replacement**.
2. Organized around **business** capabilities
3. **Products** not Projects
4. **API**-Focused
5. **Smart** endpoints and dumb pipes
6. Decentralized Governance
7. Decentralized Data Management
8. Infrastructure Automation (infrastructure as code)

RED HAT DEVELOPERS

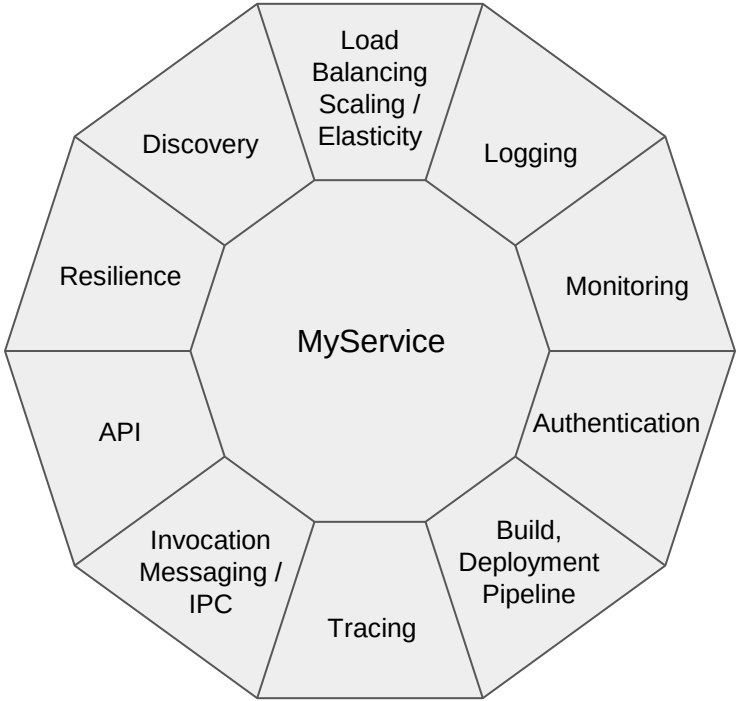
Decentralized Data Management

What?

https://developers.redhat.com/promotions/migrating-to-microservice-databases/

RED HAT DEVELOPERS

Microservice Concerns

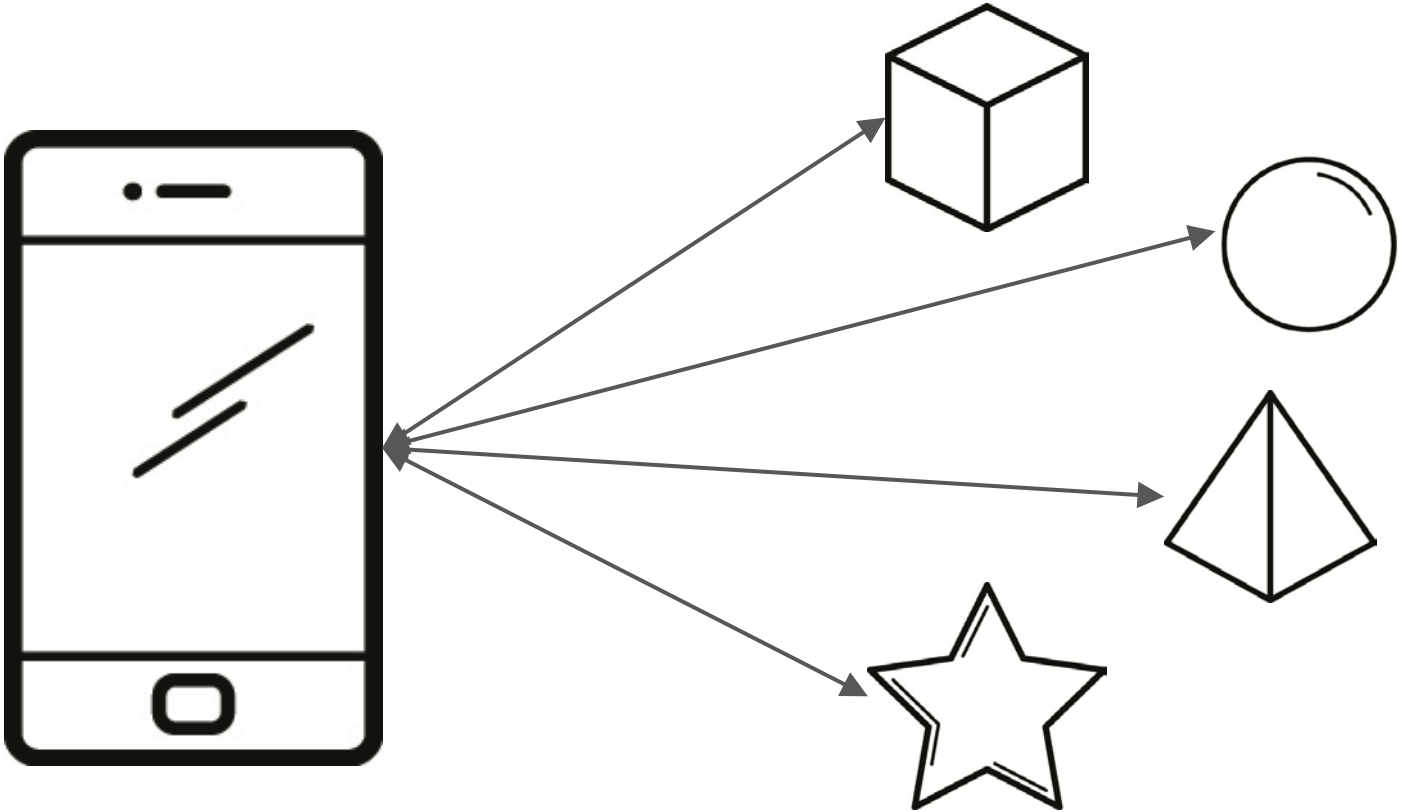


RED HAT DEVELOPERS

Microservice Patterns

RED HAT DEVELOPERS

Browser

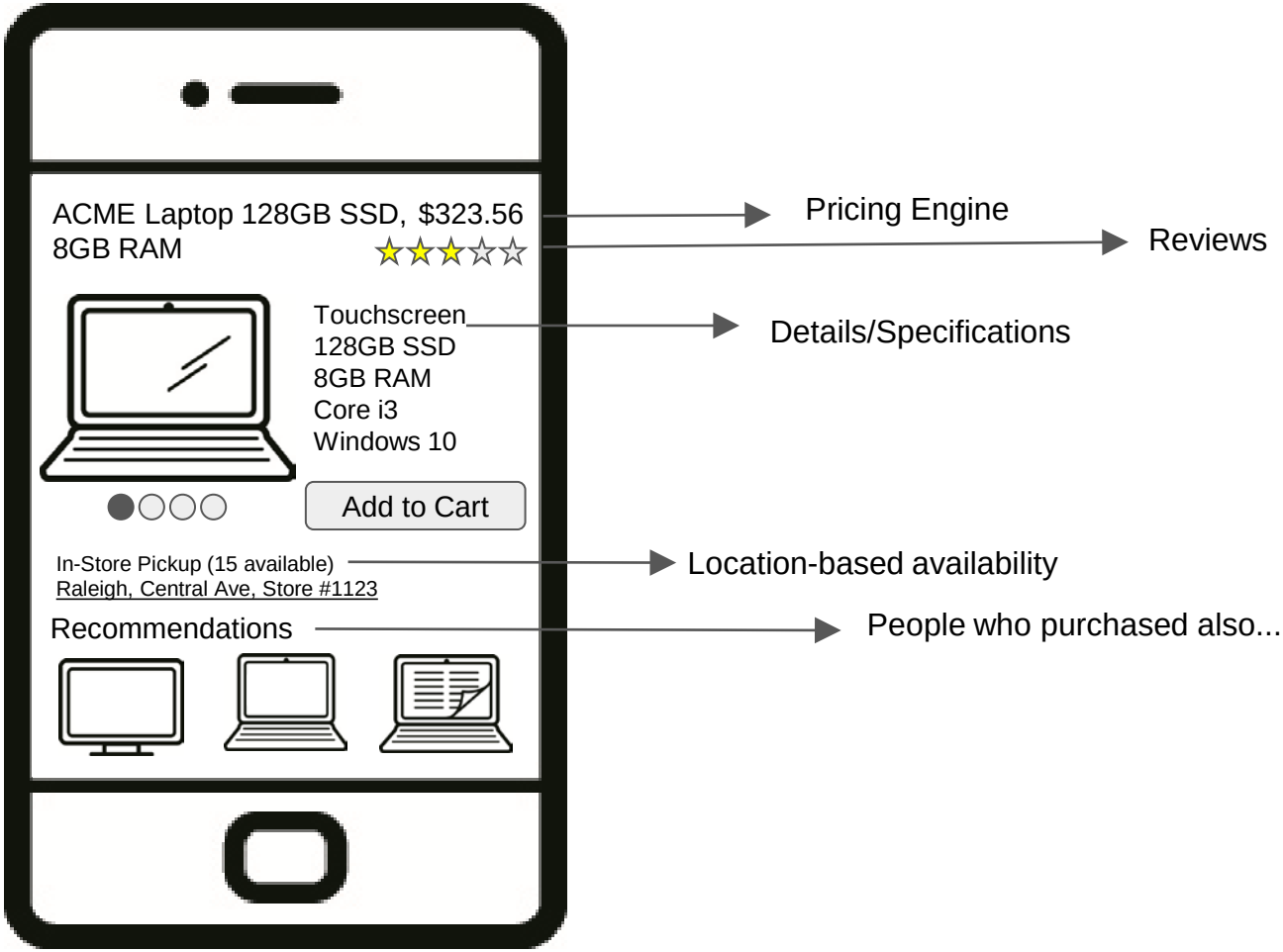


RED HAT DEVELOPERS

Example

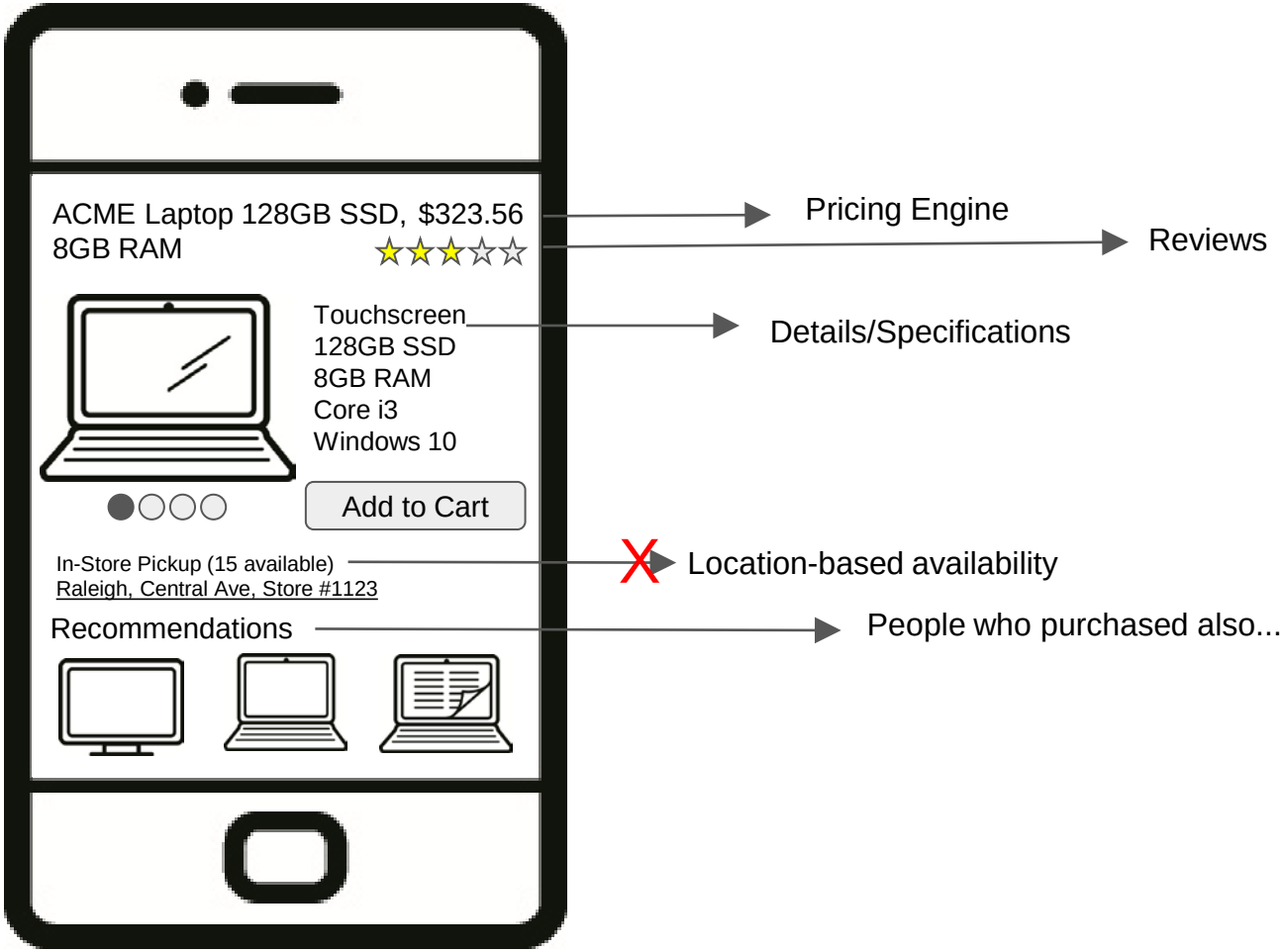
RED HAT DEVELOPERS

Example



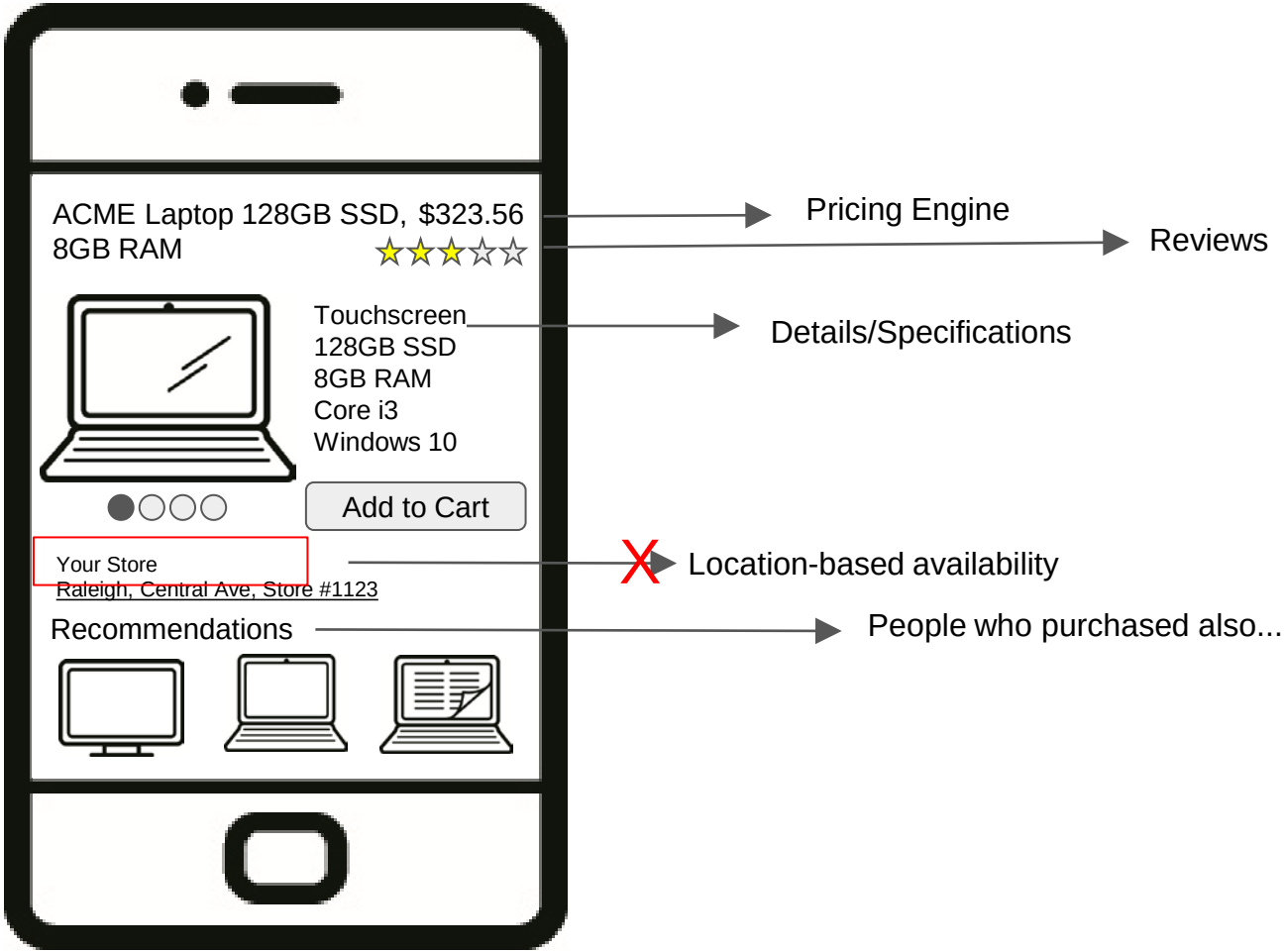
RED HAT DEVELOPERS

Example (Fail)



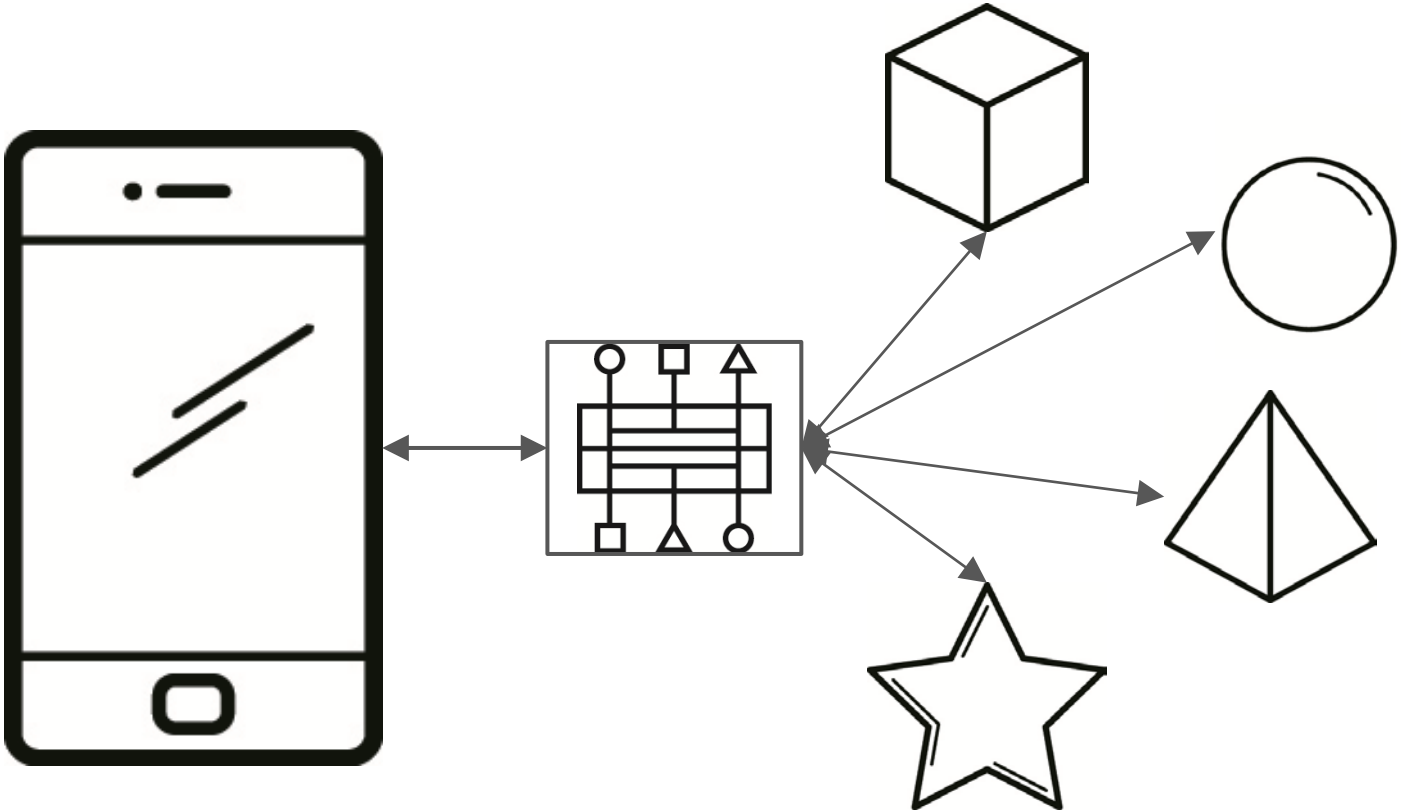
RED HAT DEVELOPERS

Example (Fallback)



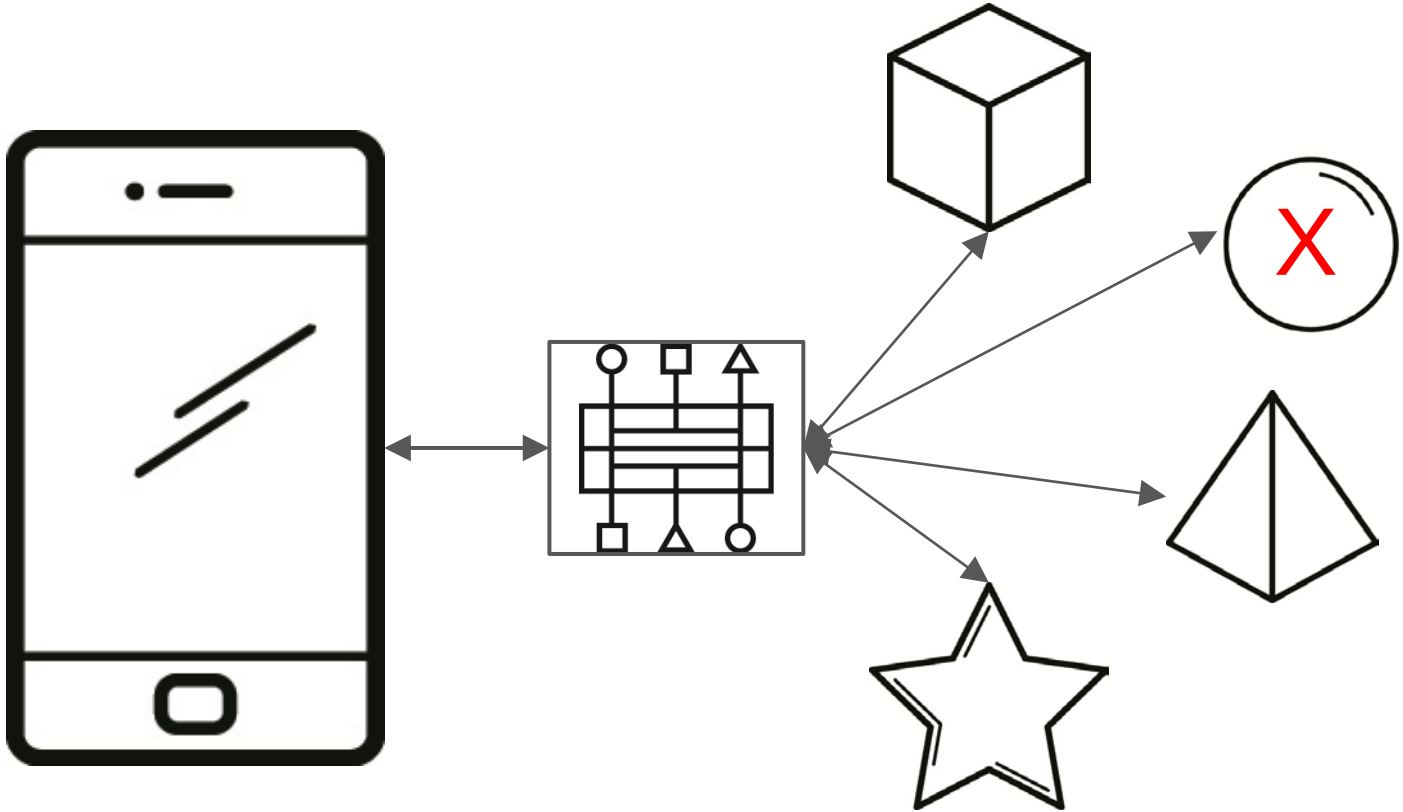
RED HAT DEVELOPERS

API “Gateway”

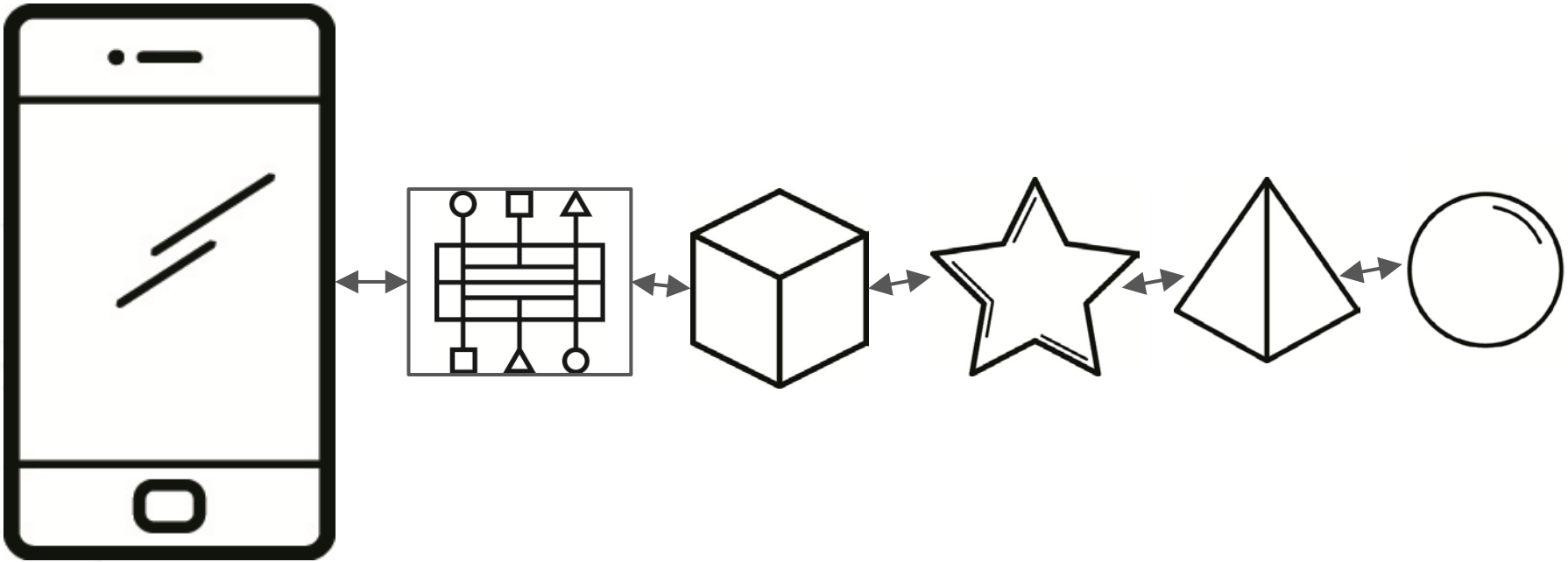


RED HAT DEVELOPERS

API “Gateway” (Fail)

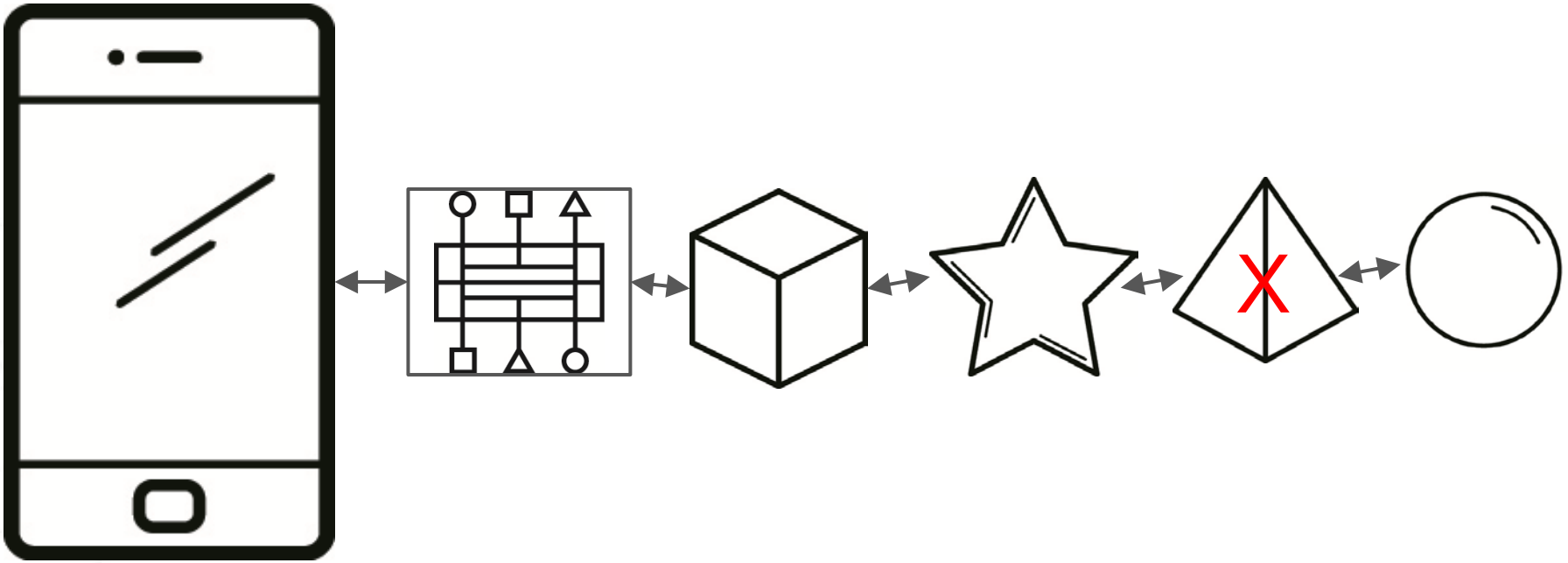


Chaining



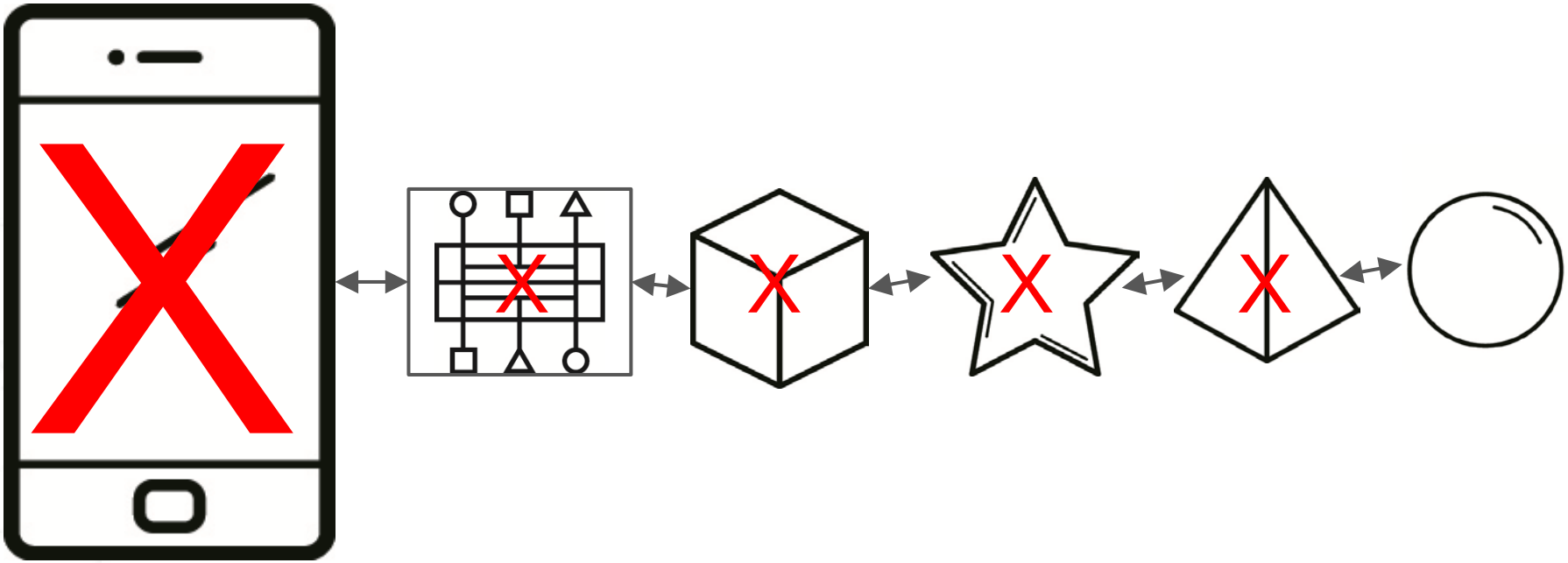
RED HAT DEVELOPERS

Chaining (Fail)



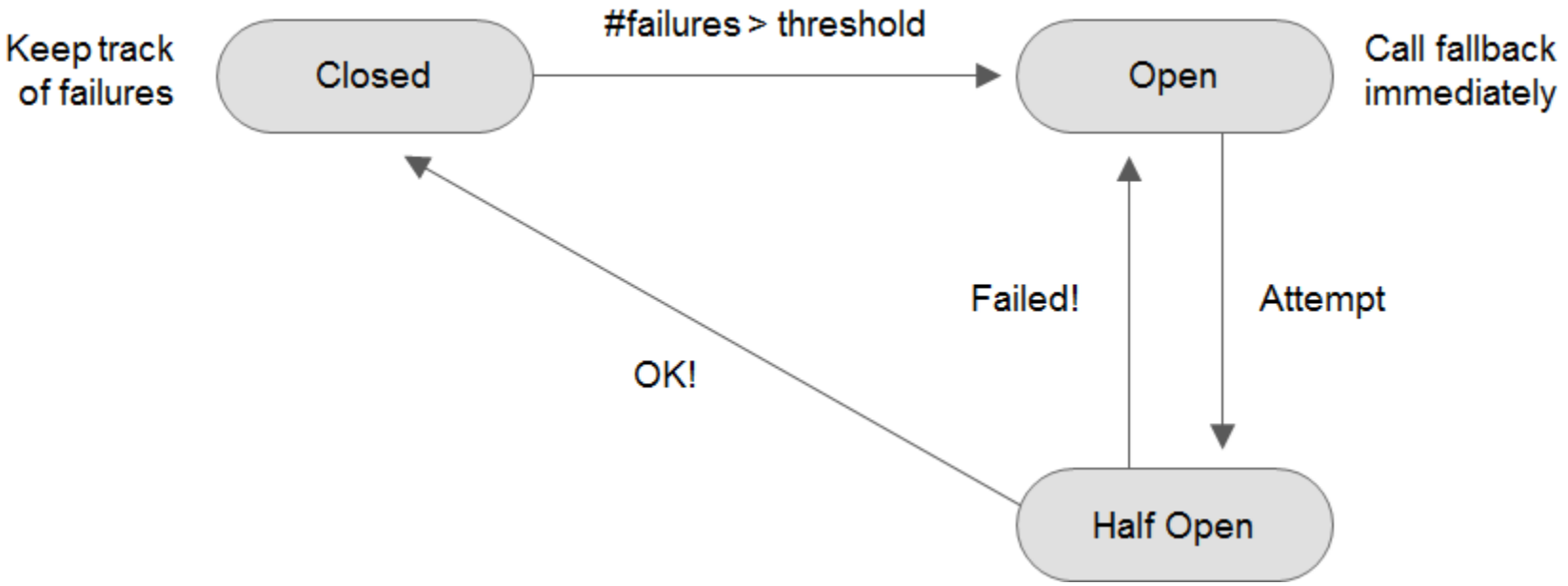
RED HAT DEVELOPERS

Chaining (Cascading Fail)



RED HAT DEVELOPERS

Circuit Breaker



RED HAT DEVELOPERS

Demo

bit.ly/msa-instructions

https://github.com/burrsutter/kube4docker

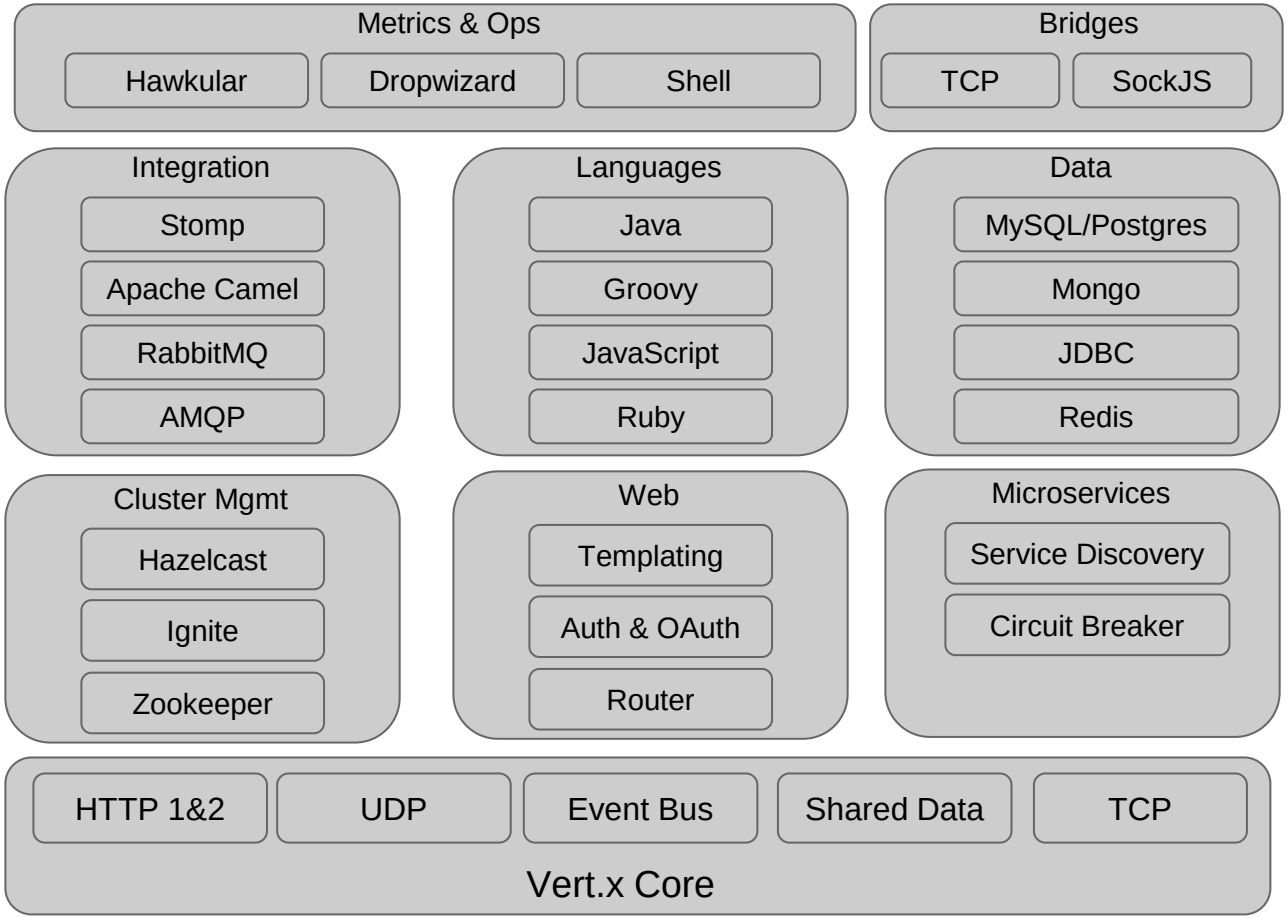
Videos: https://www.youtube.com/watch?v=SPATMHP-xw8&list=PLuWlr4oKSRUYjjwDOaZOX-54X4_OIUUeS

RED HAT DEVELOPERS

Vertx.io—an Eclipse project

Vert.x is a toolkit
for building reactive
applications on the JVM.

RED HAT DEVELOPERS



RED HAT DEVELOPERS

Vert.x

(**Un**-Opinionated) Toolkit for reactive, async applications

Every user has their own Vert.x way

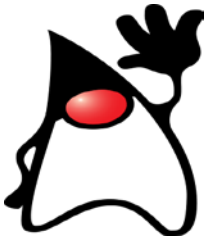
Low-latency microservices that interact using the EventBus

Built in elasticity and resiliency

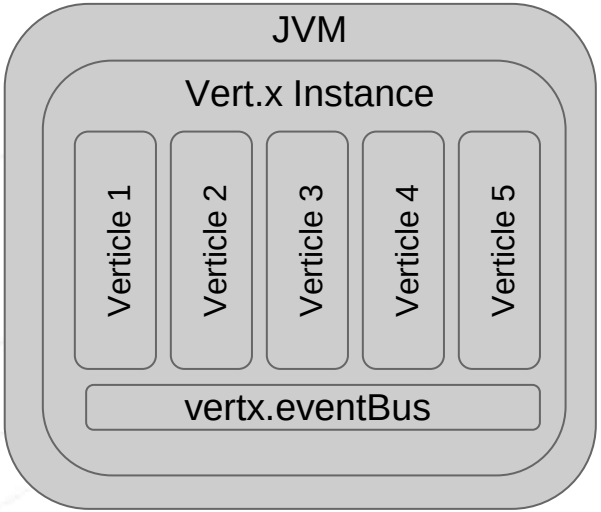
Polyglot: Java, JavaScript, Groovy, Ruby, Ceylon

vertx.io

http://vertx.io/blog/my-first-vert-x-3-application/



Vert.x Verticle



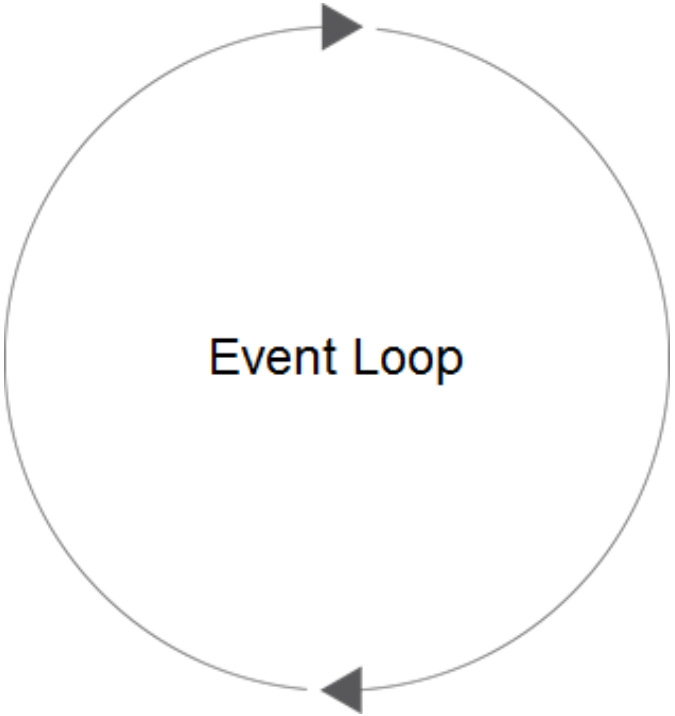
A Verticle is the programmable unit – a component

A Verticle is always executed on the same thread

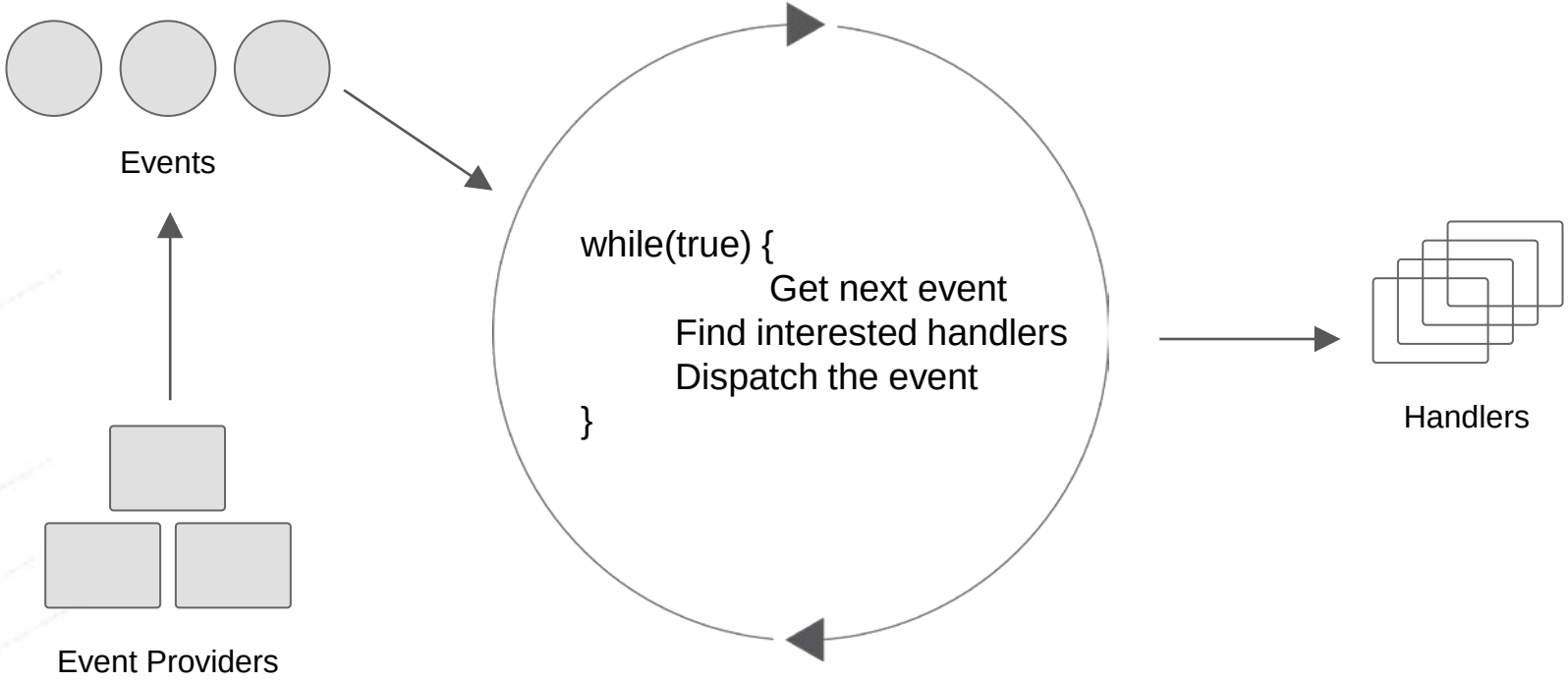
A single thread may execute several verticles – event loop

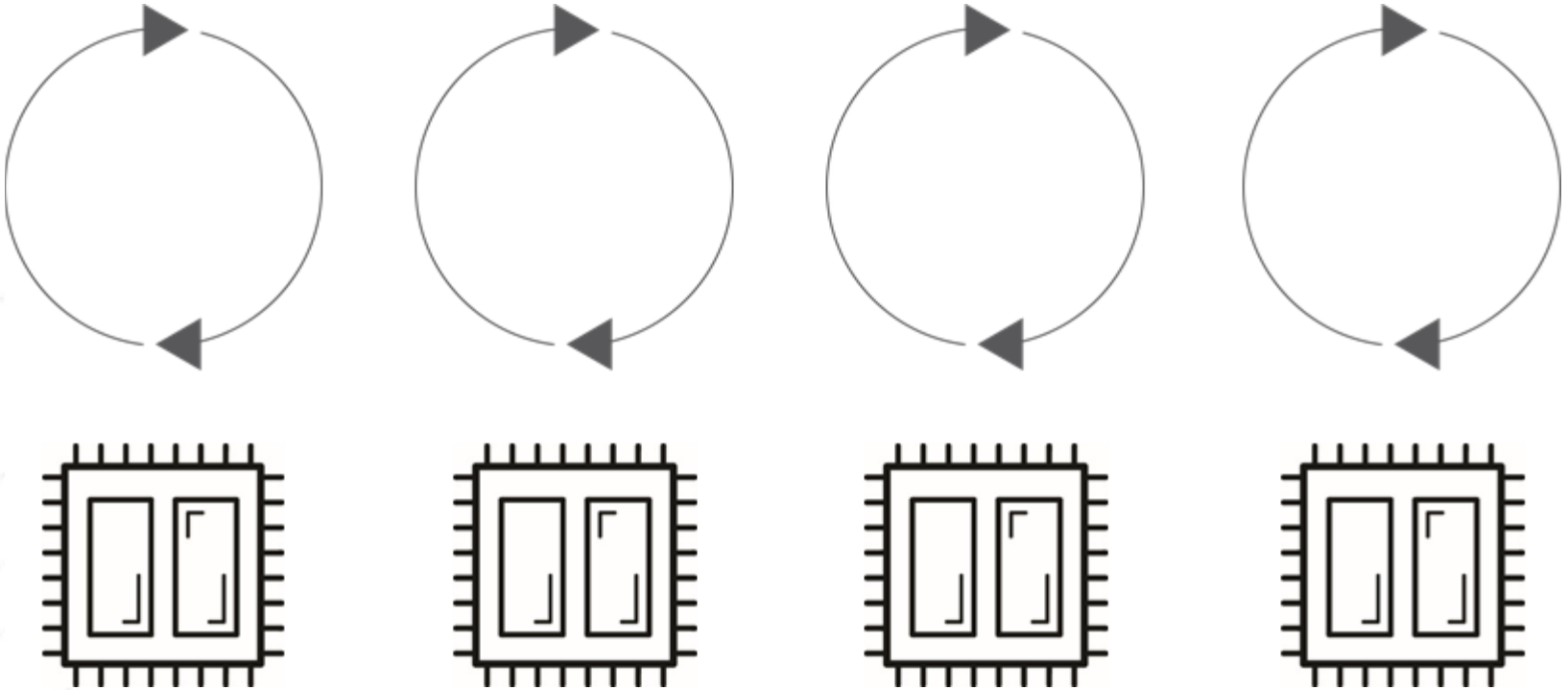
A Verticle Instance normally starts 1 thread/event-loop per core

RED HAT DEVELOPERS



RED HAT DEVELOPERS


Events
while(true) {
Get next event
Find interested handlers
Dispatch the event
}
Handlers
Event Providers



RED HAT DEVELOPERS

Vert.x HTTP Verticle

```
import io.vertx.core.AbstractVerticle;

public class MainVerticle extends AbstractVerticle {

  @Override
  public void start() {
    vertx.createHttpServer()
        .requestHandler(req ->
          req.response().end("Hello Vert.x v3!"))
        .listen(8080);
  }
}
```


RED HAT DEVELOPERS

Super-Fast Getting Started: Step 1

```
mkdir hello-microservice-http
cd hello-microservice-http

mvn io.fabric8:vertx-maven-plugin:1.0.5:setup -
DprojectGroupId=io.vertx.microservice -
DprojectArtifactId=hello-microservice-http -
Dverticle=io.vertx.book.http.HelloMicroservice -
Ddependencies=web

mvn compile vertx:run
```


RED HAT DEVELOPERS

Super-Fast Getting Started: Step 2

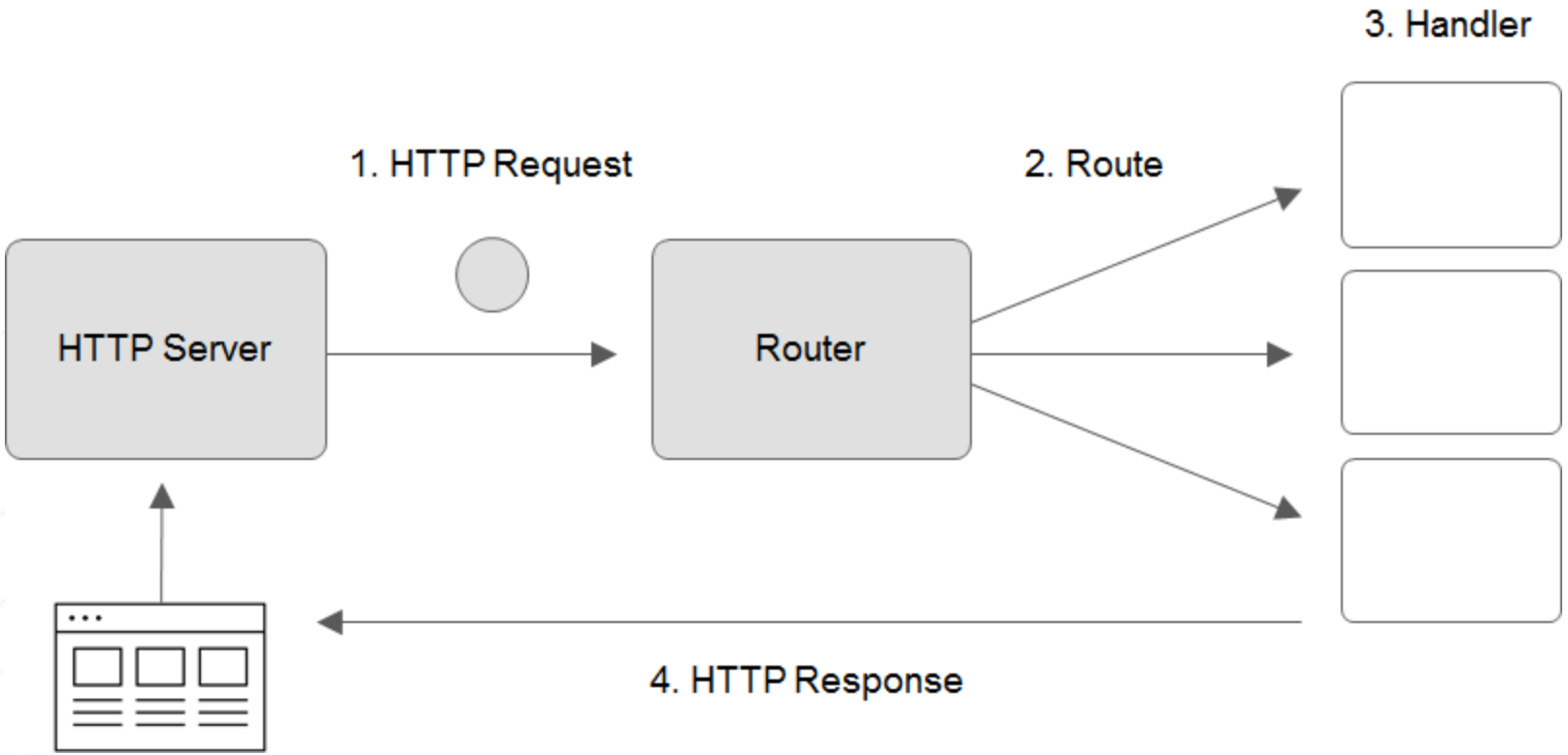
```
public void start() {
  Router router = Router.router(vertx);
  router.get("/").handler(this::hello);
  router.get("/:name").handler(this::hello);
  vertx.createHttpServer()
      .requestHandler(router::accept)
      .listen(8080);
}

private void hello(RoutingContext rc) {
  String message = "hello";
  if (rc.pathParam("name") != null) {
     message += " " + rc.pathParam("name");
  }
  JsonObject json = new JsonObject().put("message", message);
  rc.response()
       .putHeader(HttpHeaders.CONTENT_TYPE, "application/json")
       .end(json.encode());
}
```


RED HAT DEVELOPERS

Web Apps & REST

RED HAT DEVELOPERS


3. Handler
1. HTTP Request
2. Route
HTTP Server
Router
4. HTTP Response

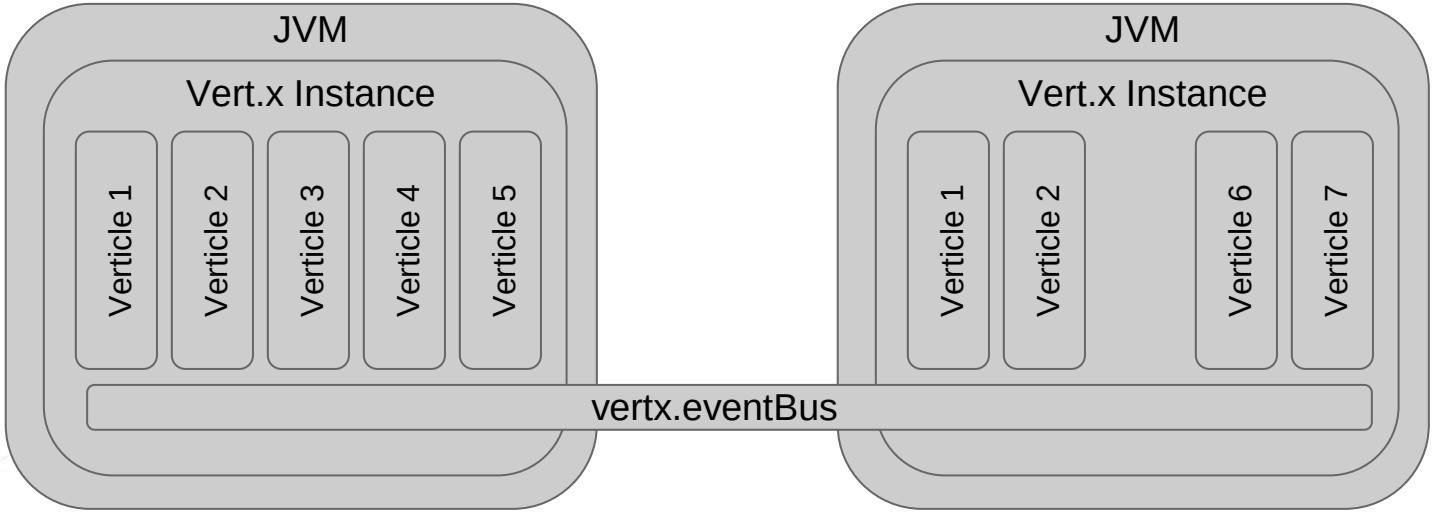
Vert.x HTTP Router

```
router.get("/api/whiskies").handler(this::getAll);
router.get("/api/whiskies/:id").handler(this::getOne);
router.post("/api/whiskies").handler(this::addOne);
router.delete("/api/whiskies/:id").handler(this::deleteOne);
router.put("/api/whiskies/:id").handler(this::updateOne);
// Serve static resources from the /assets directory
router.route("/assets/*").handler(StaticHandler.create("assets"));

vertx
  .createHttpServer()
  .requestHandler(router::accept)
  .listen(8080, result -> {
    if (result.succeeded()) {
      future.complete();
    } else {
      // could be something else is running on our port
      future.fail(result.cause());
    }
  });
```


RED HAT DEVELOPERS

Vert.x EventBus



RED HAT DEVELOPERS

Demo

https://github.com/burrsutter/vertx3_introduction

RED HAT DEVELOPERS

HTTP getNow() Circuit Breaker

```
CircuitBreakerOptions options = new
   CircuitBreakerOptions()
    .setMaxFailures(5)
    .setTimeout(5000)
    .setFallbackOnFailure(true);

CircuitBreaker breaker =
    CircuitBreaker.create("my-circuit-breaker", vertx,
        options)
    .openHandler(v -> {
      System.out.println("Circuit opened");
    }).closeHandler(v -> {
      System.out.println("Circuit closed");
    });
```

```
Future<String> result = breaker.executeWithFallback(future -> {
  vertx.createHttpClient().getNow(8080, "localhost", "/",
    response -> {
    if (response.statusCode() != 200) {
      future.fail("HTTP error");
    } else {
      response.exceptionHandler(future::fail).bodyHandler(buffer
       -> {
        future.complete(buffer.toString());
      });
    }
  });
}, v -> {
  // Executed when the circuit is opened
  return "Hello (fallback)";
});
```

https://github.com/burrsutter/vertx3_introduction/blob/master/7_httpinvoker/Client.java#L13

RED HAT DEVELOPERS

Demos

https://github.com/redhat-reactive-msa

https://github.com/burrsutter/reactive_tutorial_jfokus17

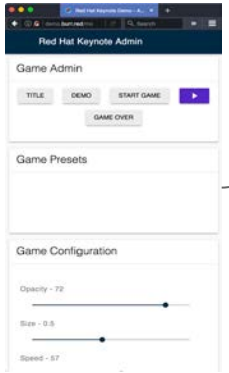
RED HAT DEVELOPERS

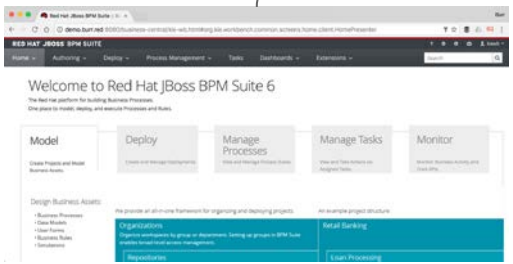
Recording: https://youtu.be/ooA6FmTL4Dk?t=1739

vertx-game-server

vertx-achievement-server

Java EE on JBoss with Drools+jBPM





RED HAT
DEVELOPERS

Reactive Microservices with Eclipse Vert.x

@burrsutter

http://developers.redhat.com
http://bit.ly/reactivemsa

October 5-6, 2017
IISc, Bangalore

www.modsummit.com

Register early and get the best discounts

April 23-28, 2018
IISc, Bangalore

www.developersummit.com

GREAT INDIAN DEVELOPER SUMMIT™ 2018