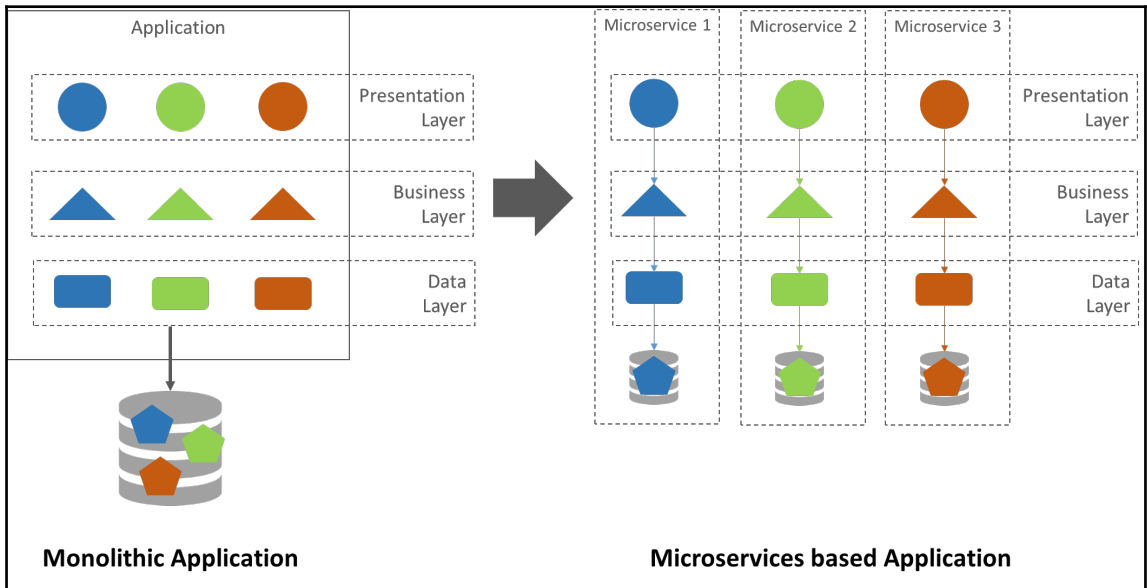
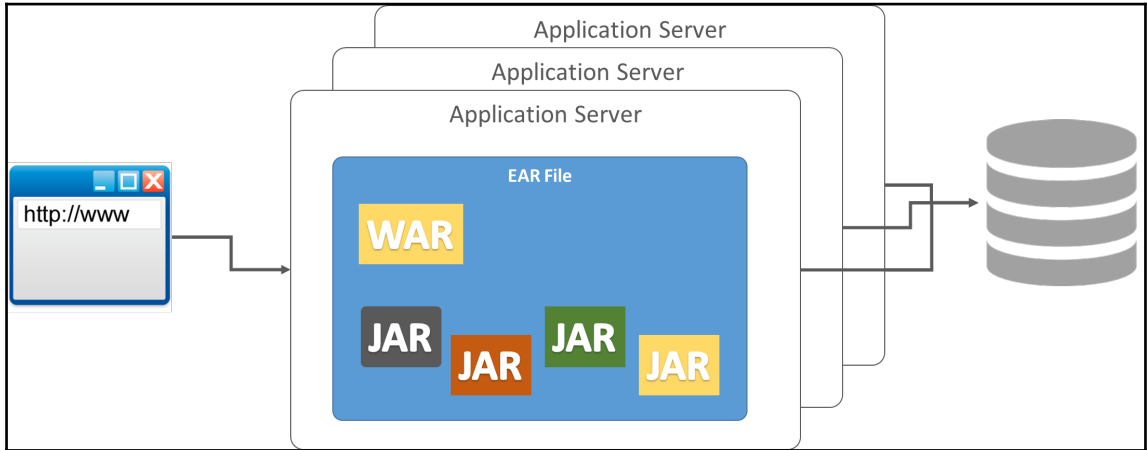
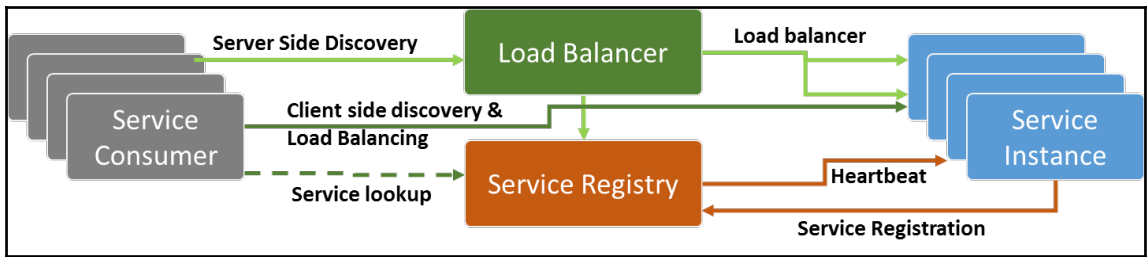
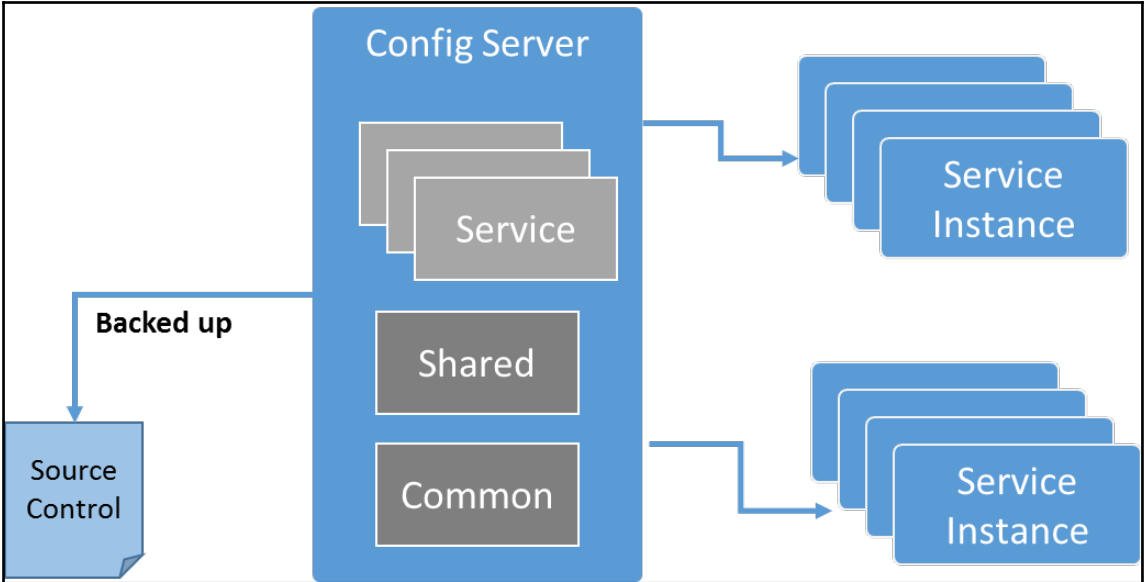


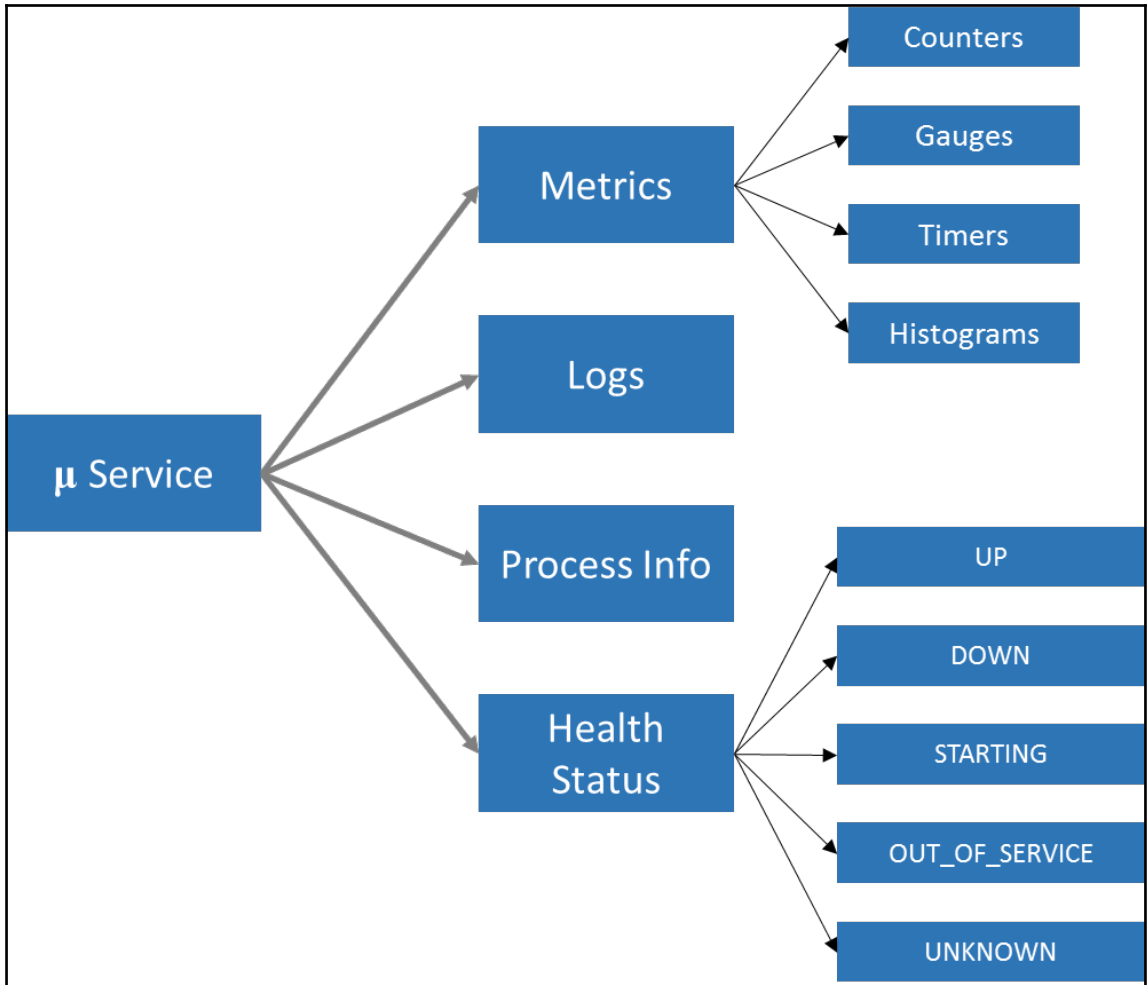
Chapter 1: Introduction to Cloud-Native

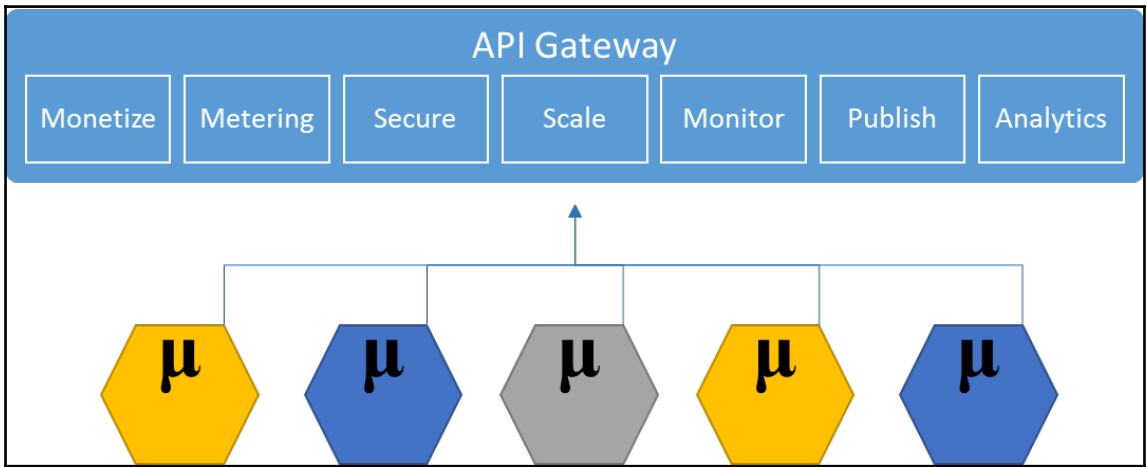
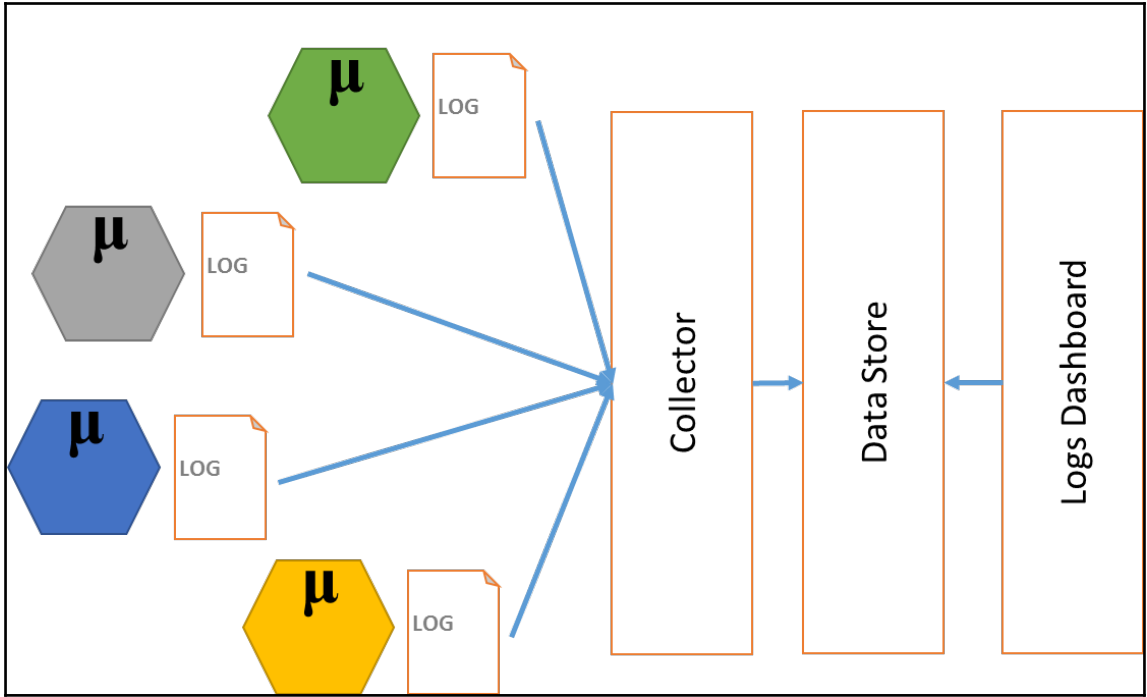


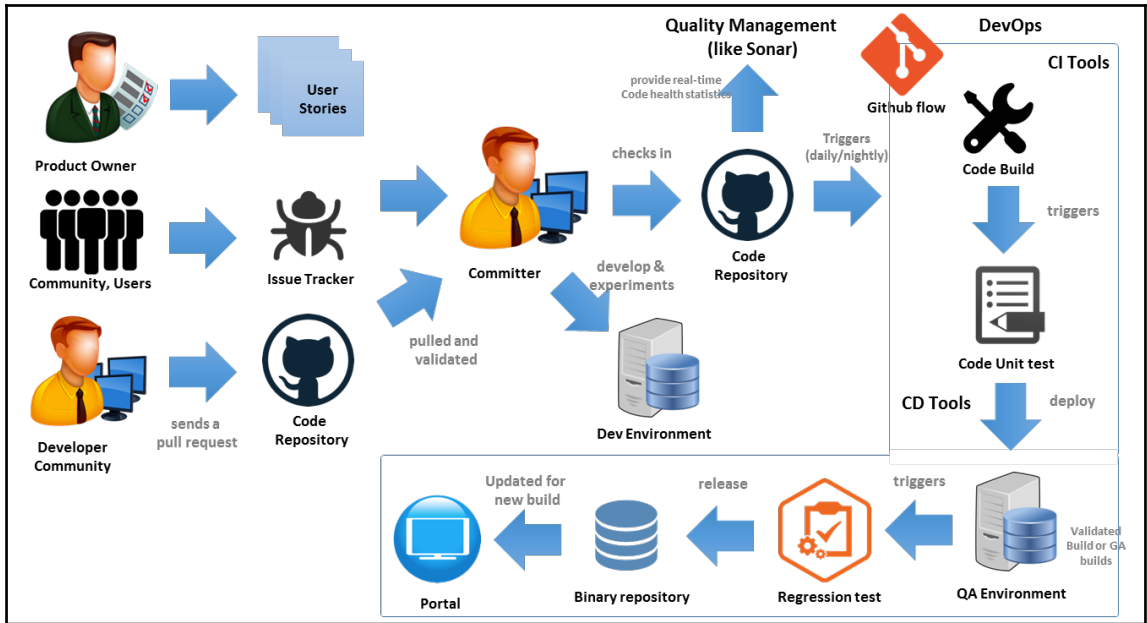
Single Codebase	Explicit Dependencies	Externally managed Config	Backing services
Build, release, run	Processes	Port binding	Concurrency
Disposability	Dev/prod parity	Logs as event streams	Admin processes



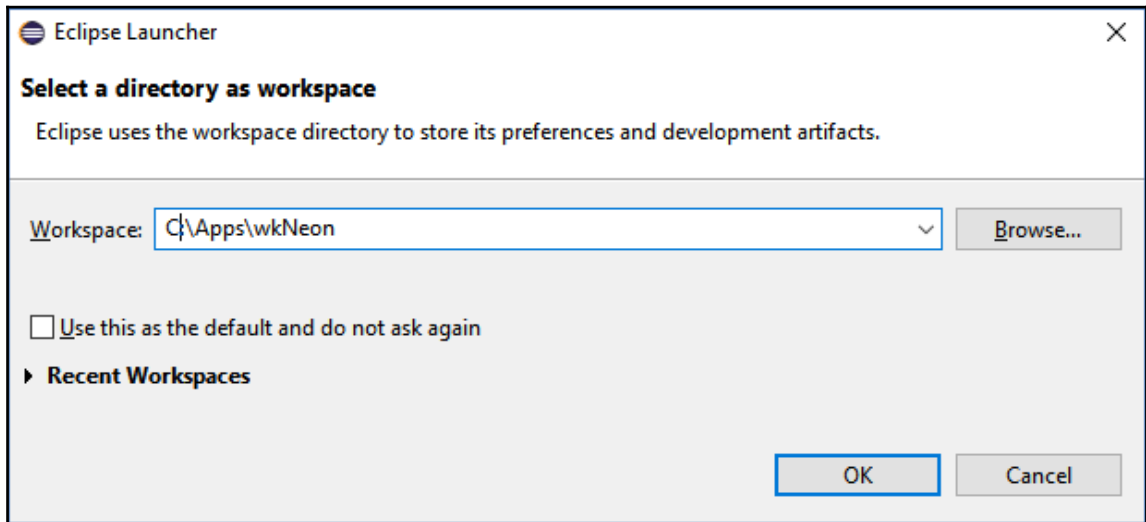


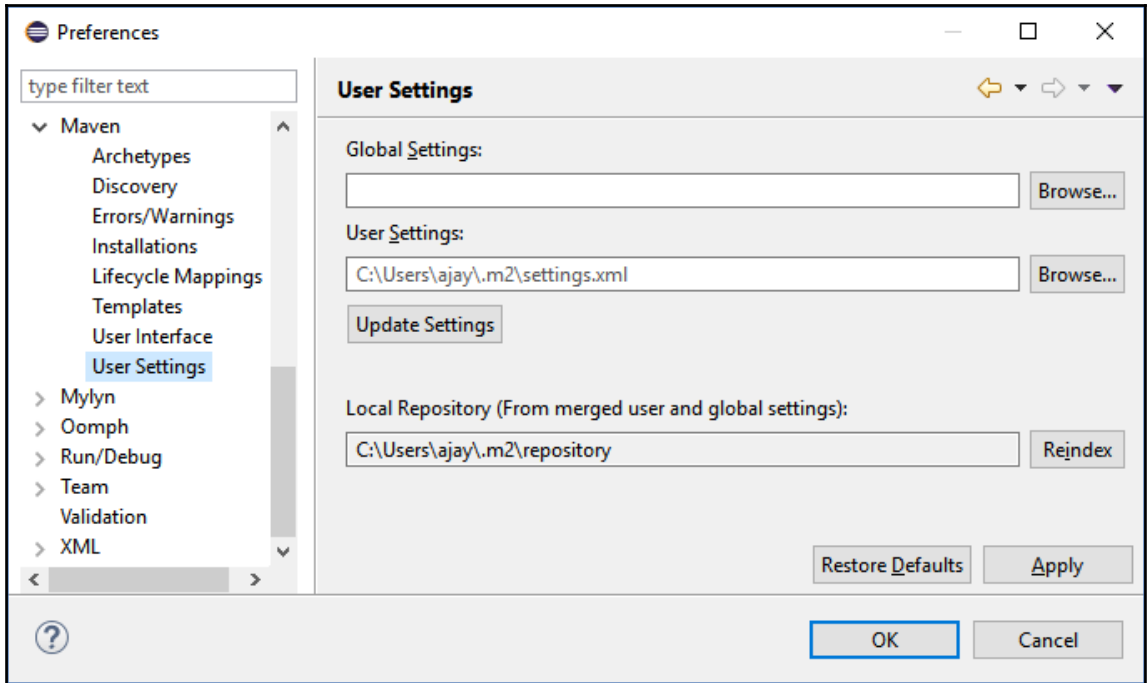


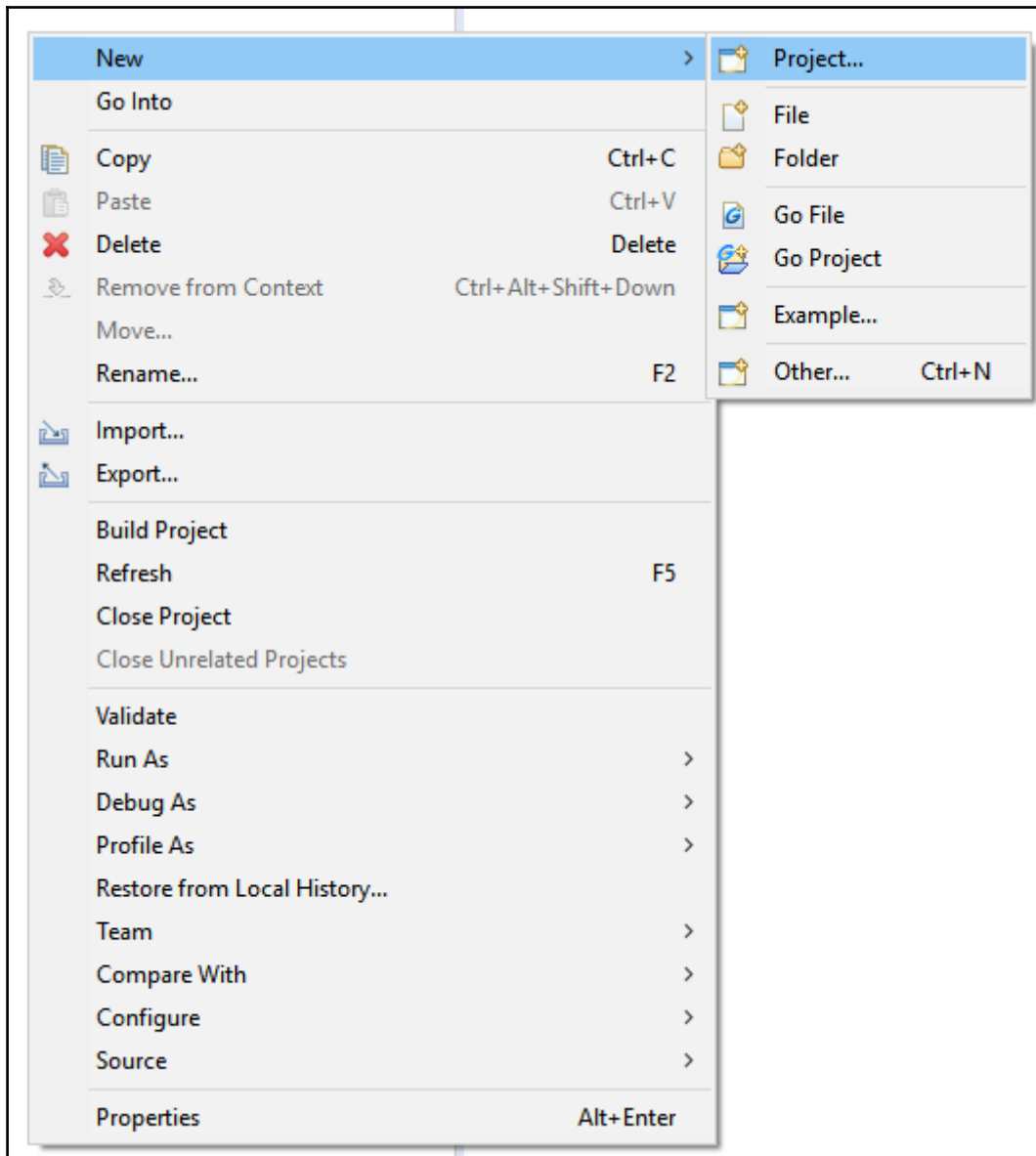


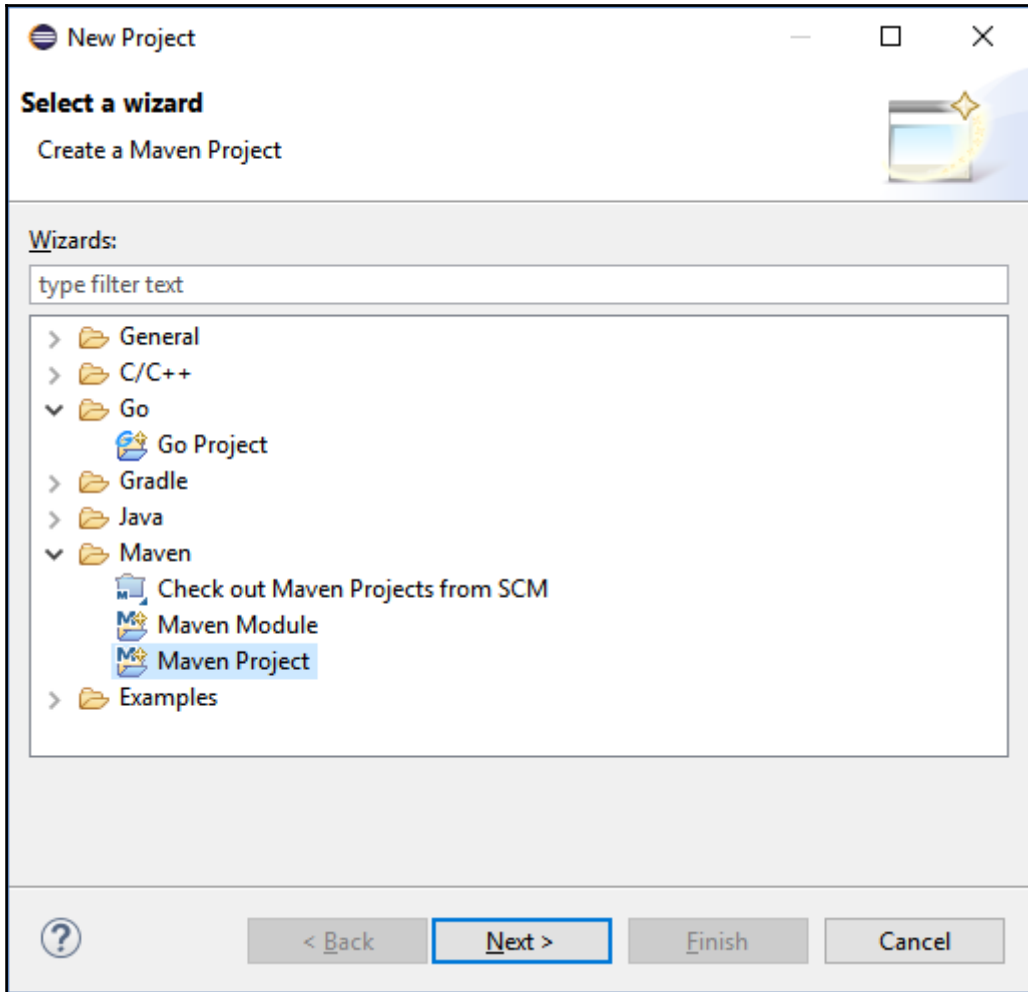


Chapter 2: Writing Your First Cloud-Native Application









New Maven project

Configure project



Artifact

Group Id:

Artifact Id:

Version:

Packaging:

Name:

Description:

Parent Project

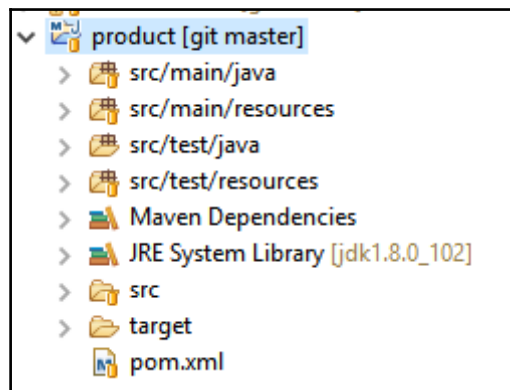
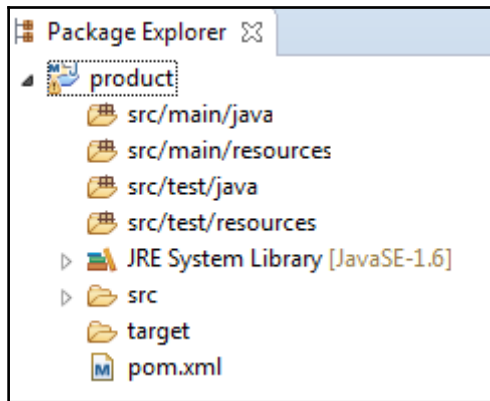
Group Id:

Artifact Id:

Version:

▶ Advanced





```
product/pom.xml ProductSpringApp.java ProductService.java
1 package com.mycompany.product;
2
3 import java.util.Arrays;
4 import java.util.List;
5
6 import org.springframework.web.bind.annotation.PathVariable;
7 import org.springframework.web.bind.annotation.RequestMapping;
8 import org.springframework.web.bind.annotation.RequestParam;
9 import org.springframework.web.bind.annotation.RestController;
10
11 @RestController
12 public class ProductService {
13
14     @RequestMapping("/product/{id}")
15     Product getProduct(@PathVariable("id") int id) {
16         return new Product(id);
17     }
18
19     @RequestMapping("/productIds")
20     List<Integer> getProductIds(@RequestParam("id") int id) {
21         return Arrays.asList(id + 1, id + 2, id + 3);
22     }
23 }
```

Run Configurations

Create, manage, and run configurations

Name: product

Main JRE Refresh Source Environment Common

Base directory: \${project_loc:product}

Workspace... File System... Variables...

Goals: spring-boot:run

Profiles:

User settings: C:\Users\ajay\.m2\settings.xml

Workspace... File System... Variables...

Offline Update Snapshots

Debug Output Skip Tests Non-recursive

Resolve Workspace artifacts

1 Threads

Parameter Name	Value	
		Add...
		Edit...
		Remove

Maven Runtime: EMBEDDED (3.3.9/1.7.0.20160603-1931) Configure...

Revert Apply

Run Close

type filter text

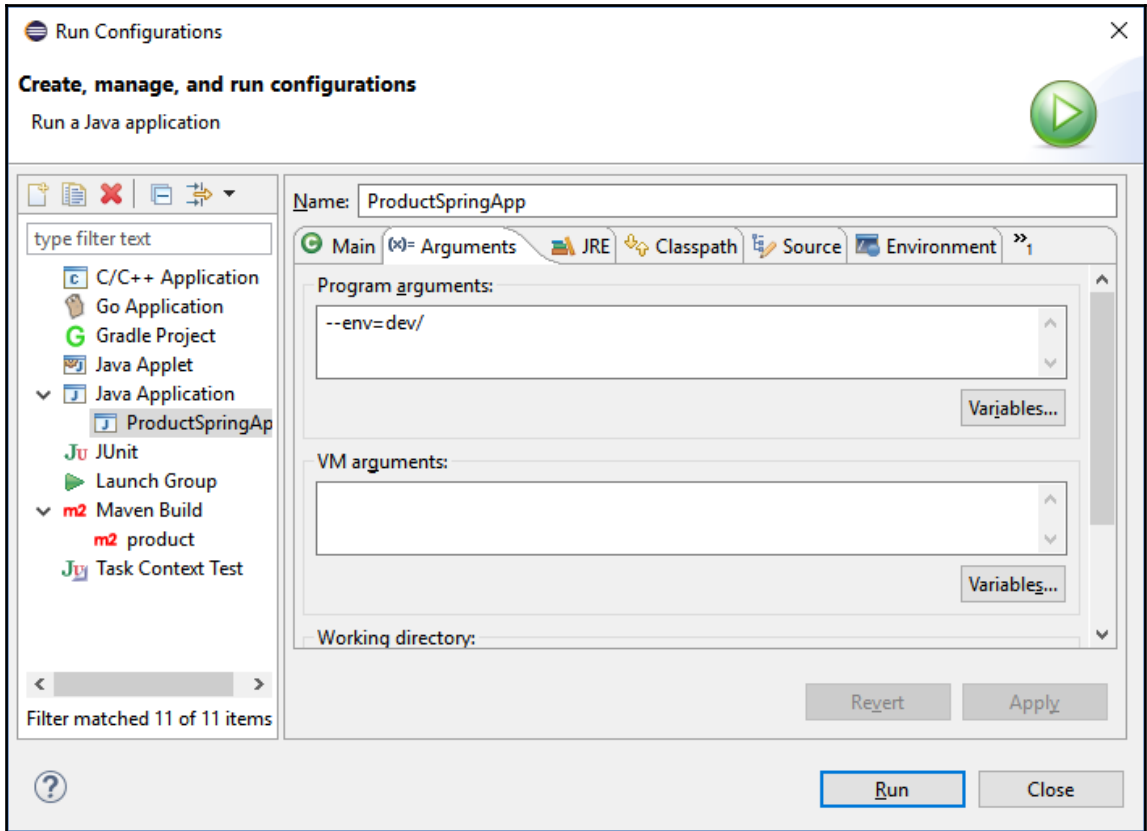
- C/C++ Application
- Go Application
- Gradle Project
- Java Applet
- Java Application
 - ProductSpringApj
- JUnit
- Launch Group
- Maven Build
 - product
- Task Context Test

Filter matched 11 of 11 items

```
scala -java -jar target\product-0.0.1-SNAPSHOT.jar
D:\Apps\wkNeon\product>java -jar target\product-0.0.1-SNAPSHOT.jar

=====
:: Spring Boot ::                (v1.4.1.RELEASE)

2016-10-31 12:44:11.308 INFO 1332 --- [main] com.mycompany.product.ProductSpringApp : Starting ProductSpringApp
jay in D:\Apps\wkNeon\product>
2016-10-31 12:44:11.322 INFO 1332 --- [main] com.mycompany.product.ProductSpringApp : No active profile set, using default
2016-10-31 12:44:11.431 INFO 1332 --- [main] org.springframework.context.support.AnnotationConfigEmbeddedWebApplicationContext : Refreshing org.springframework.context.support.AnnotationConfigEmbeddedWebApplicationContext
44:11 IST 2016]; root of context hierarchy
2016-10-31 12:44:13.580 INFO 1332 --- [main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat initialized with port(s) 8080->
2016-10-31 12:44:13.661 INFO 1332 --- [main] o.apache.catalina.core.StandardService : Starting service Tomcat
2016-10-31 12:44:13.694 INFO 1332 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet Engine: org.apache.catalina.
2016-10-31 12:44:13.723 INFO 1332 --- [lost-startStop-1] o.a.e.c.c.[Tomcat].[localhost].[/] : Initializing Spring WebAppConfiguration
2016-10-31 12:44:13.724 INFO 1332 --- [lost-startStop-1] o.s.web.context.ContextLoader : Root WebApplicationContext
2016-10-31 12:44:13.933 INFO 1332 --- [lost-startStop-1] o.s.b.w.servlet.ServletRegistrationBean : Mapping servlet: 'dispatcherServlet'
2016-10-31 12:44:13.940 INFO 1332 --- [lost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'characterEncodingFilter'
2016-10-31 12:44:13.941 INFO 1332 --- [lost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'hiddenHttpMethodFilter'
2016-10-31 12:44:13.942 INFO 1332 --- [lost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'httpMethodNonIdempotentFilter'
2016-10-31 12:44:13.944 INFO 1332 --- [lost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'requestContextFilter'
2016-10-31 12:44:14.320 INFO 1332 --- [main] s.w.s.m.n.a.RequestMappingHandlerAdapter : Looking for @Component
date [Mon Oct 31 12:44:11 IST 2016]; root of context hierarchy
2016-10-31 12:44:14.432 INFO 1332 --- [main] s.w.s.m.n.a.RequestMappingHandlerMapping : Mapped "/product/<
2016-10-31 12:44:14.436 INFO 1332 --- [main] s.w.s.m.n.a.RequestMappingHandlerMapping : Mapped "/productId
2016-10-31 12:44:14.440 INFO 1332 --- [main] s.w.s.m.n.a.RequestMappingHandlerMapping : Mapped "/error/"
ework.boot.autoconfigure.web.BasicErrorController.error(javax.servlet.http.HttpServletRequest) : Mapped "/error/{pr
web.BasicErrorController.errorHtml(javax.servlet.http.HttpServletRequest,javax.servlet.http.HttpServletResponse) : Mapped "/error/{pr
2016-10-31 12:44:14.441 INFO 1332 --- [main] s.w.s.m.n.a.RequestMappingHandlerMapping : Mapped "/error/{pr
2016-10-31 12:44:14.482 INFO 1332 --- [main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/we
2016-10-31 12:44:14.484 INFO 1332 --- [main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/we
2016-10-31 12:44:14.533 INFO 1332 --- [main] o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/we
2016-10-31 12:44:14.703 INFO 1332 --- [main] o.s.j.e.a.AnnotationMBeanExporter : Registering beans fo
2016-10-31 12:44:14.793 INFO 1332 --- [main] s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on po
2016-10-31 12:44:14.807 INFO 1332 --- [main] com.mycompany.product.ProductSpringApp : Started ProductSpringApp
```




```
eureka-server/pom.xml
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
3   <modelVersion>4.0.0</modelVersion>
4
5   <groupId>com.mycompany.infra</groupId>
6   <artifactId>eureka-server</artifactId>
7   <version>0.0.1-SNAPSHOT</version>
8
9   <parent>
10    <groupId>org.springframework.boot</groupId>
11    <artifactId>spring-boot-starter-parent</artifactId>
12    <version>1.4.1.RELEASE</version>
13  </parent>
14
15  <dependencyManagement>
16    <dependencies>
17      <dependency>
18        <groupId>org.springframework.cloud</groupId>
19        <artifactId>spring-cloud-netflix</artifactId>
20        <version>1.2.1.RELEASE</version>
21        <type>pom</type>
22        <scope>import</scope>
23      </dependency>
24    </dependencies>
25  </dependencyManagement>
26
27  <dependencies>
28    <dependency>
29      <groupId>org.springframework.cloud</groupId>
30      <artifactId>spring-cloud-starter-eureka-server</artifactId>
31    </dependency>
32  </dependencies>
33
34 </project>
```

```
eureka-server/pom.xml  EurekaApplication.java
1 package com.mycompany.infra.eureka;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import org.springframework.cloud.netflix.eureka.server.EnableEurekaServer;
6
7 @SpringBootApplication
8 @EnableEurekaServer
9 public class EurekaApplication {
10
11     public static void main(String[] args) throws Exception {
12         SpringApplication.run(EurekaApplication.class, args);
13     }
14 }
```

```
: Initializing Spring FrameworkServlet 'dispatcherServlet'
: FrameworkServlet 'dispatcherServlet': initialization started
: FrameworkServlet 'dispatcherServlet': initialization completed in 19 ms
: Registered instance EUREKA/localhost:eureka:8761 with status UP (replication=false)
: DiscoveryClient_EUREKA/localhost:eureka:8761 - registration status: 204
: Disable delta property : false
: Single vip registry refresh property : null
: Force full registry fetch : false
: Application is null : false
: Registered Applications size is zero : true
: Application version is -1: true
: Getting all instance registry info from the eureka server
: The response status is 200
: Registered instance EUREKA/localhost:eureka:8761 with status UP (replication=true)
: Got 1 instances from neighboring DS node
: Renew threshold is: 1
: Changing status to UP
: Started Eureka Server
```

Eureka

localhost:8761

Spring Boot Reference

spring Eureka

HOME LAST 1000 SINCE STARTUP

System Status

Environment	test	Current time	2016-11-01T23:05:27 +0530
Data center	default	Uptime	00:05
		Lease expiration enabled	true
		Renews threshold	1
		Renews (last min)	2

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
EUREKA	n/a (1)	(1)	UP (1) - localhost:eureka:8761

General Info

Name	Value
total-avail-memory	399mb
environment	test
num-of-cpus	4
current-memory-usage	163mb (40%)
server-uptime	00:05
registered-replicas	
unavailable-replicas	

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-netflix</artifactId>
      <version>1.2.1.RELEASE</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>

<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-eureka</artifactId>
  </dependency>
</dependencies>
```

```
bootstrap.yml  product/pom.xml  ProductServi...  ProductSprin...  ⌵
1 package com.mycompany.product;
2
3+ import org.springframework.boot.SpringApplication;
4
5
6
7 @SpringBootApplication
8 @EnableDiscoveryClient
9
10 public class ProductSpringApp {
11
12- public static void main(String[] args) throws Exception {
13     SpringApplication.run(ProductSpringApp.class, args);
14 }
15 }
```

```
com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server
com.netflix.discovery.DiscoveryClient : The response status is 200
com.netflix.discovery.DiscoveryClient : Starting heartbeat executor: renew interval is: 10
c.n.discovery.InstanceInfoReplicator : InstanceInfoReplicator onDemand update allowed rate per min is 4
com.netflix.discovery.DiscoveryClient : Discovery Client initialized at timestamp 1478022547463 with initial
c.n.e.EurekaDiscoveryClientConfiguration : Registering application product with eureka with status UP
com.netflix.discovery.DiscoveryClient : Saw local status change event StatusChangeEvent [timestamp=147802254
com.netflix.discovery.DiscoveryClient : DiscoveryClient_PRODUCT/localhost:product:8081: registering service.
com.netflix.discovery.DiscoveryClient : DiscoveryClient_PRODUCT/localhost:product:8081 - registration status
s.b.c.e.t.TomcatEmbeddedServletContainer : Tomcat started on port(s): 8081 (http)
c.n.e.EurekaDiscoveryClientConfiguration : Updating port to 8081
com.mycompany.product.ProductSpringApp : Started ProductSpringApp in 10.099 seconds (JVM running for 10.567)
```

Eureka

localhost:8761

Spring Boot Reference

spring Eureka

HOME LAST 1000 SINCE STARTUP

System Status

Environment	test	Current time	2016-11-01T23:33:47 +0530
Data center	default	Uptime	00:33
		Lease expiration enabled	true
		Renews threshold	3
		Renews (last min)	8

DS Replicas

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
EUREKA	n/a (1)	(1)	UP (1) - localhost:eureka:8761
PRODUCT	n/a (1)	(1)	UP (1) - localhost:product:8081

General Info

Name	Value
total-avail-memory	399mb
environment	test
num-of-cpus	4
current-memory-usage	184mb (46%)
server-uptime	00:33
registered-replicas	

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-eureka</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-ribbon</artifactId>
  </dependency>
</dependencies>
```

```
@RestController
public class ProductClient {

    @Autowired
    private RestTemplate rTemplate ;

    @Value("${env}")
    private String env ;

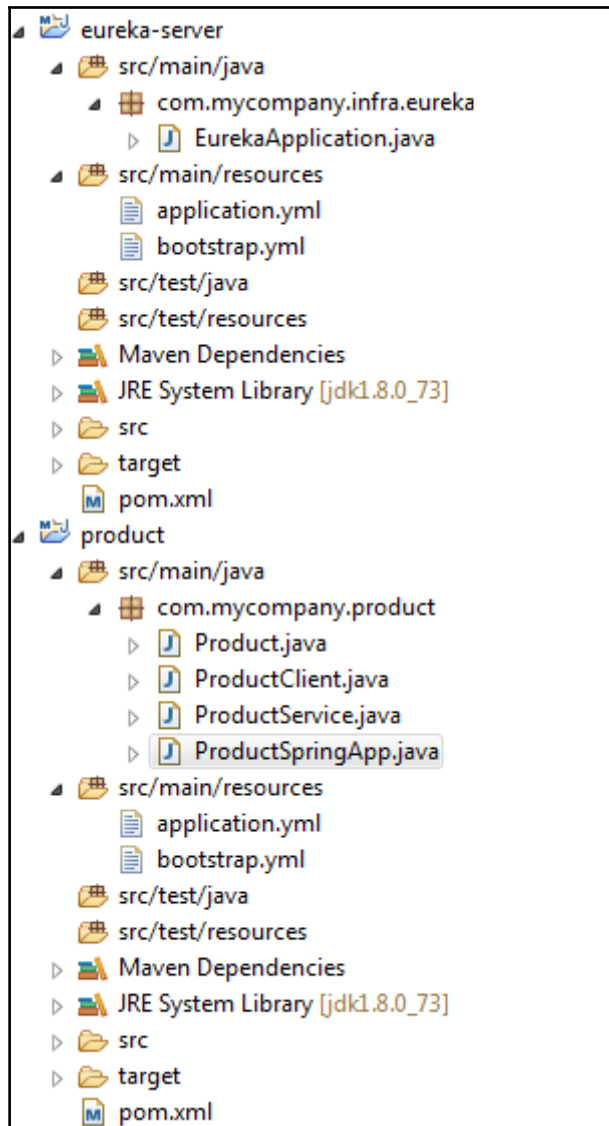
    @RequestMapping("/client/{id}")
    Product getProduct(@PathVariable("id") int id) {

        Product product = rTemplate.getForObject(
            "http://PRODUCT/" + env + "/product/" + id,
            Product.class);
        return product ;
    }
}
```

```
@SpringBootApplication
@EnableDiscoveryClient
public class ProductSpringApp {

    public static void main(String[] args) throws Exception {
        SpringApplication.run(ProductSpringApp.class, args);
    }

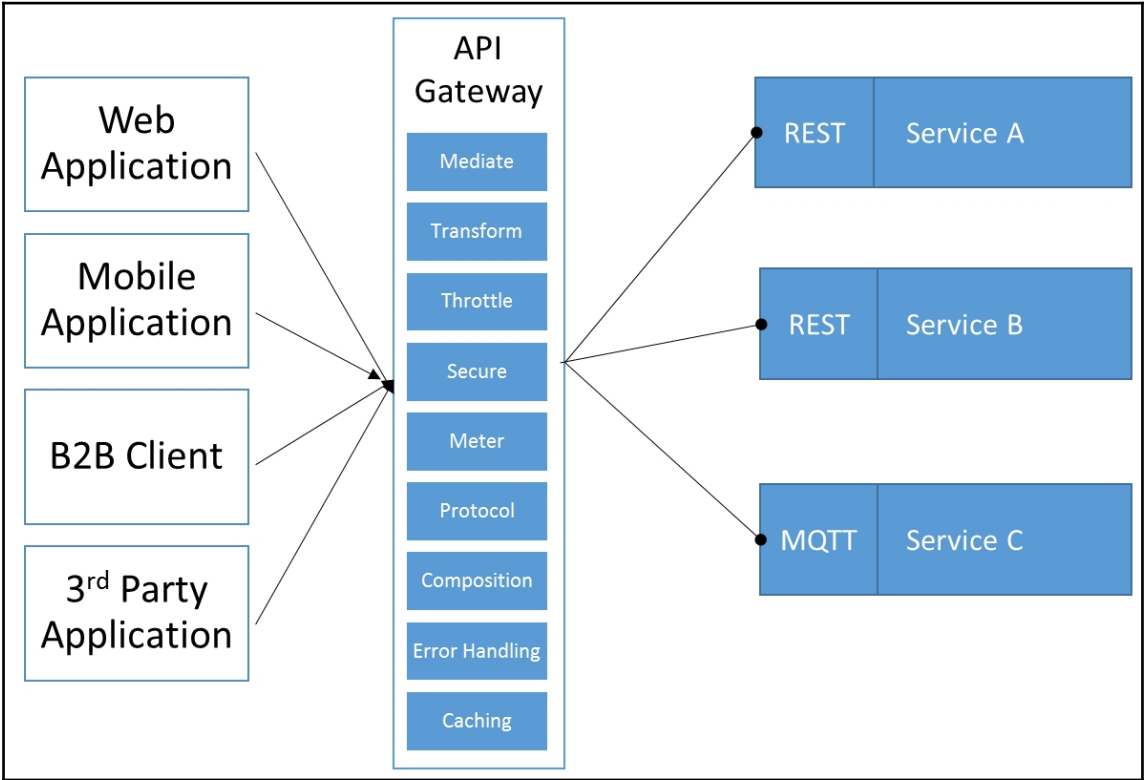
    @Bean
    @LoadBalanced
    public RestTemplate restTemplate() {
        return new RestTemplate();
    }
}
```

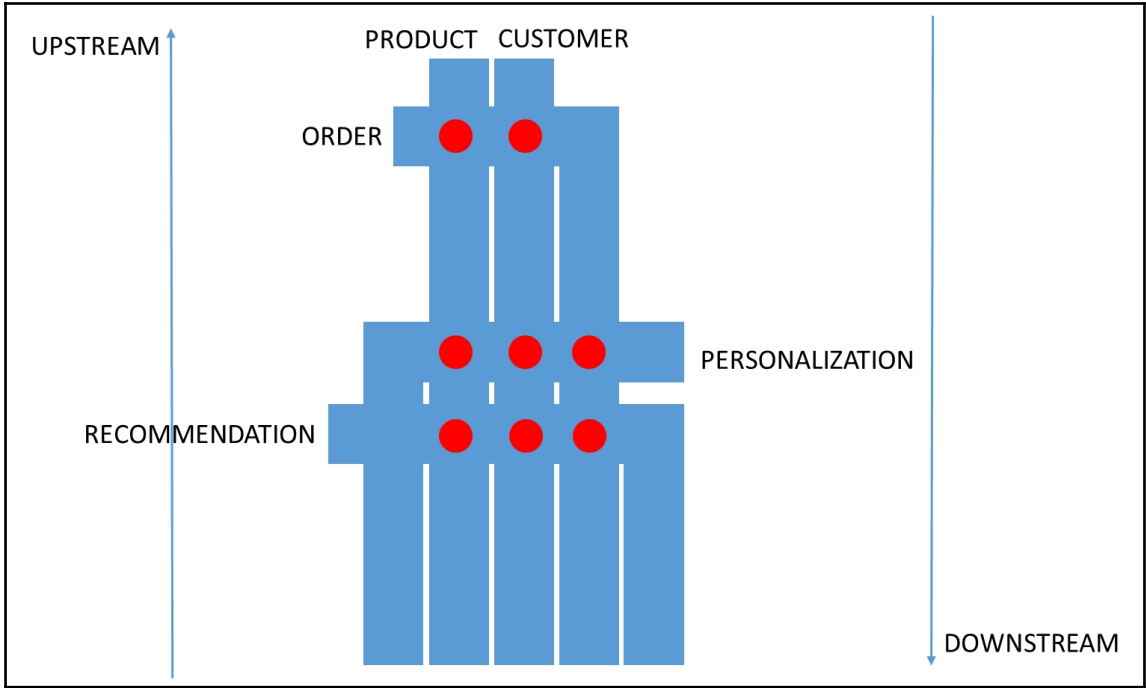



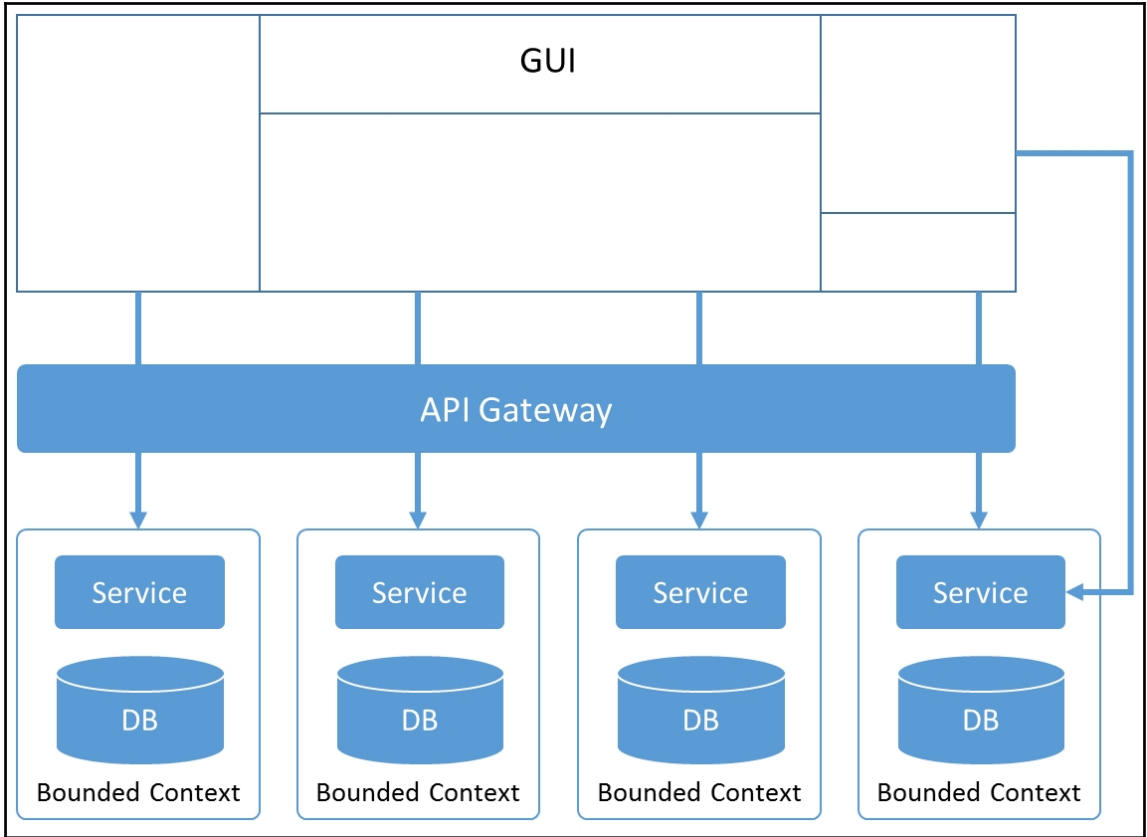
```
Initializing Spring FrameworkServlet 'dispatcherServlet'  
FrameworkServlet 'dispatcherServlet': initialization started  
FrameworkServlet 'dispatcherServlet': initialization completed in 34 ms  
Registered instance EUREKA/L-156025577.wipro.com:eureka:8761 with status UP (replication=false)  
DiscoveryClient_EUREKA/L-156025577.wipro.com:eureka:8761 - registration status: 204  
Registered instance EUREKA/L-156025577.wipro.com:eureka:8761 with status UP (replication=true)  
Disable delta property : false  
Single vip registry refresh property : null  
Force full registry fetch : false  
Application is null : false  
Registered Applications size is zero : true  
Application version is -1: true  
Getting all instance registry info from the eureka server  
The response status is 200  
Registered instance EUREKA/L-156025577.wipro.com:eureka:8761 with status UP (replication=true)  
Got 1 instances from neighboring DS node  
Renew threshold is: 1  
Changing status to UP  
Started Eureka Server
```

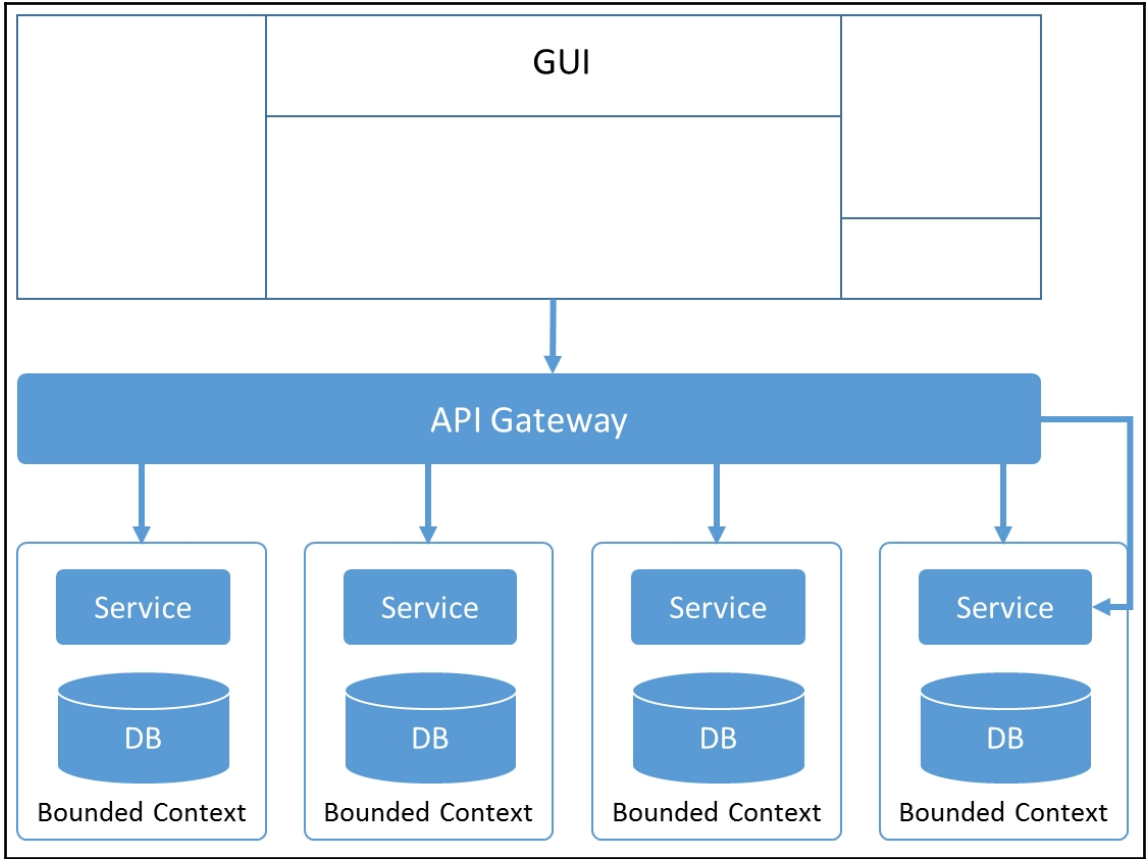
```
InstanceInfoReplicator onDemand update allowed rate per min is 4  
Discovery Client initialized at timestamp 1478077808353 with initial instances count: 1  
Registering application product with eureka with status UP  
Saw local status change event StatusChangeEvent [timestamp=1478077808403, current=UP, previous=STARTING]  
DiscoveryClient_PRODUCT/L-156025577.wipro.com:product:8082: registering service...  
DiscoveryClient_PRODUCT/L-156025577.wipro.com:product:8082 - registration status: 204  
Tomcat started on port(s): 8082 (http)  
Updating port to 8082  
Started ProductSpringApp in 11.65 seconds (JVM running for 12.316)
```

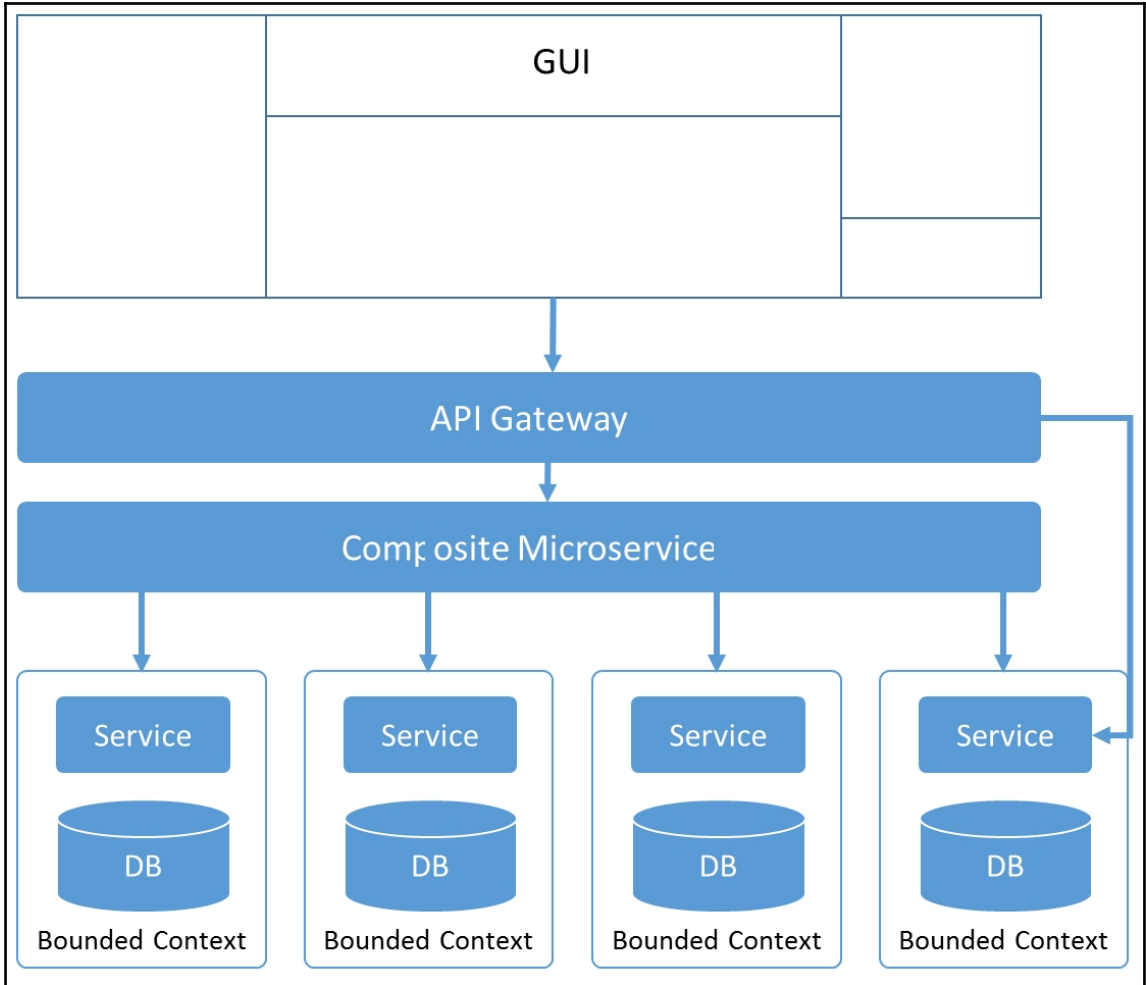
Chapter 3: Designing Your Cloud-Native Application

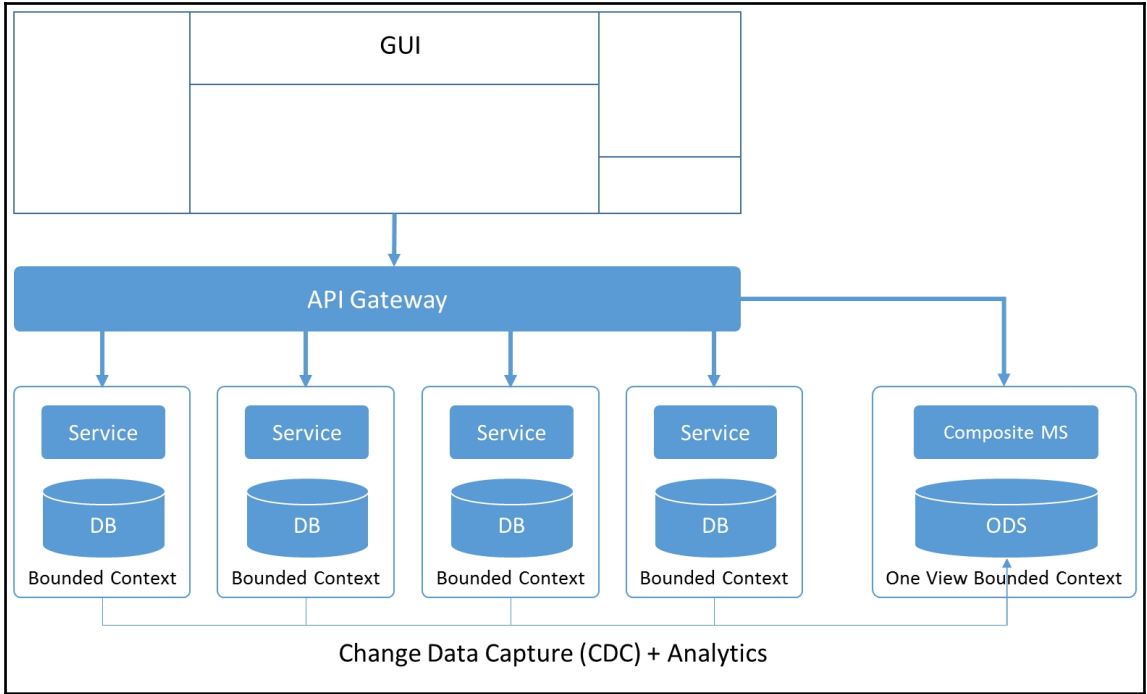


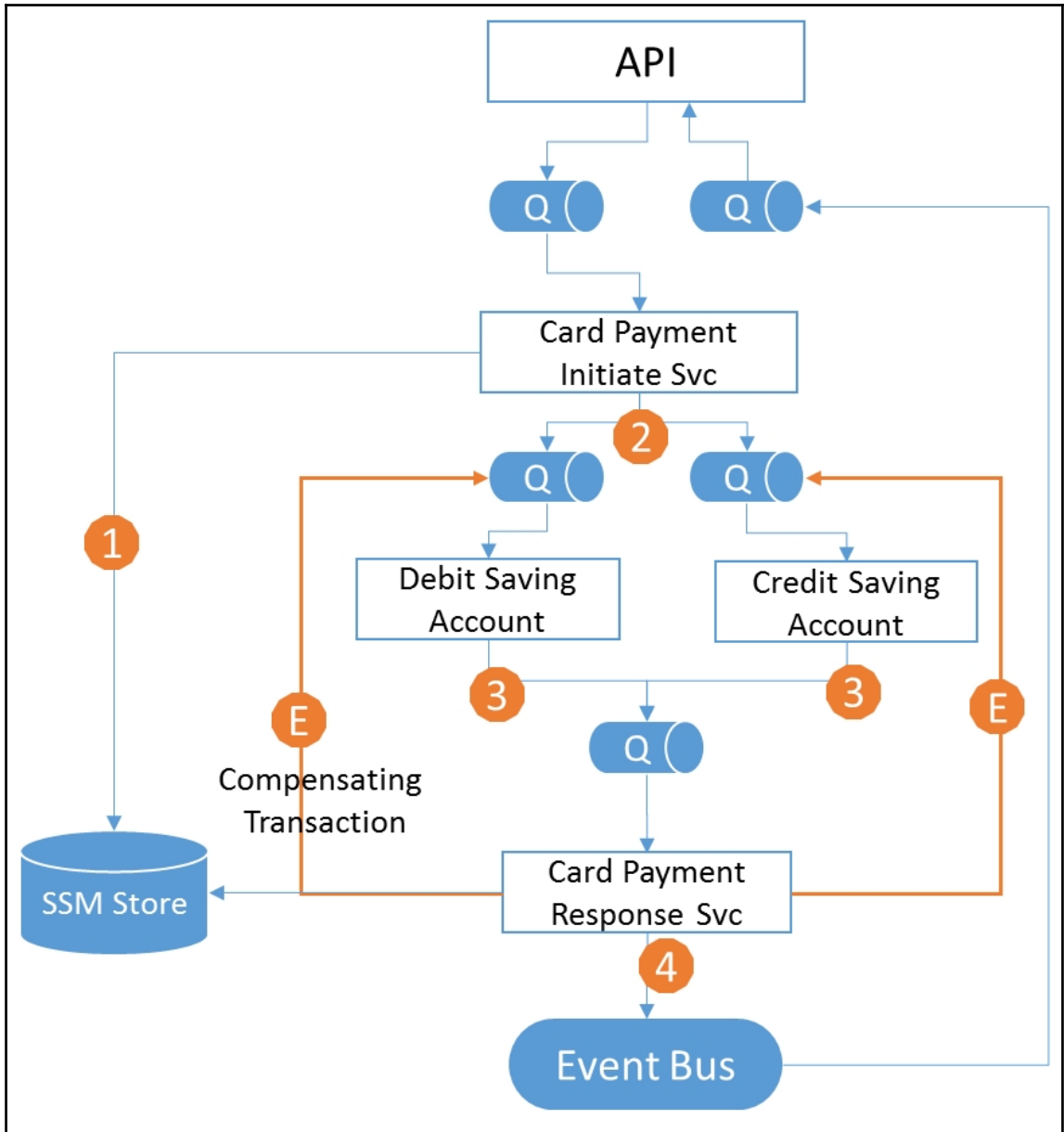


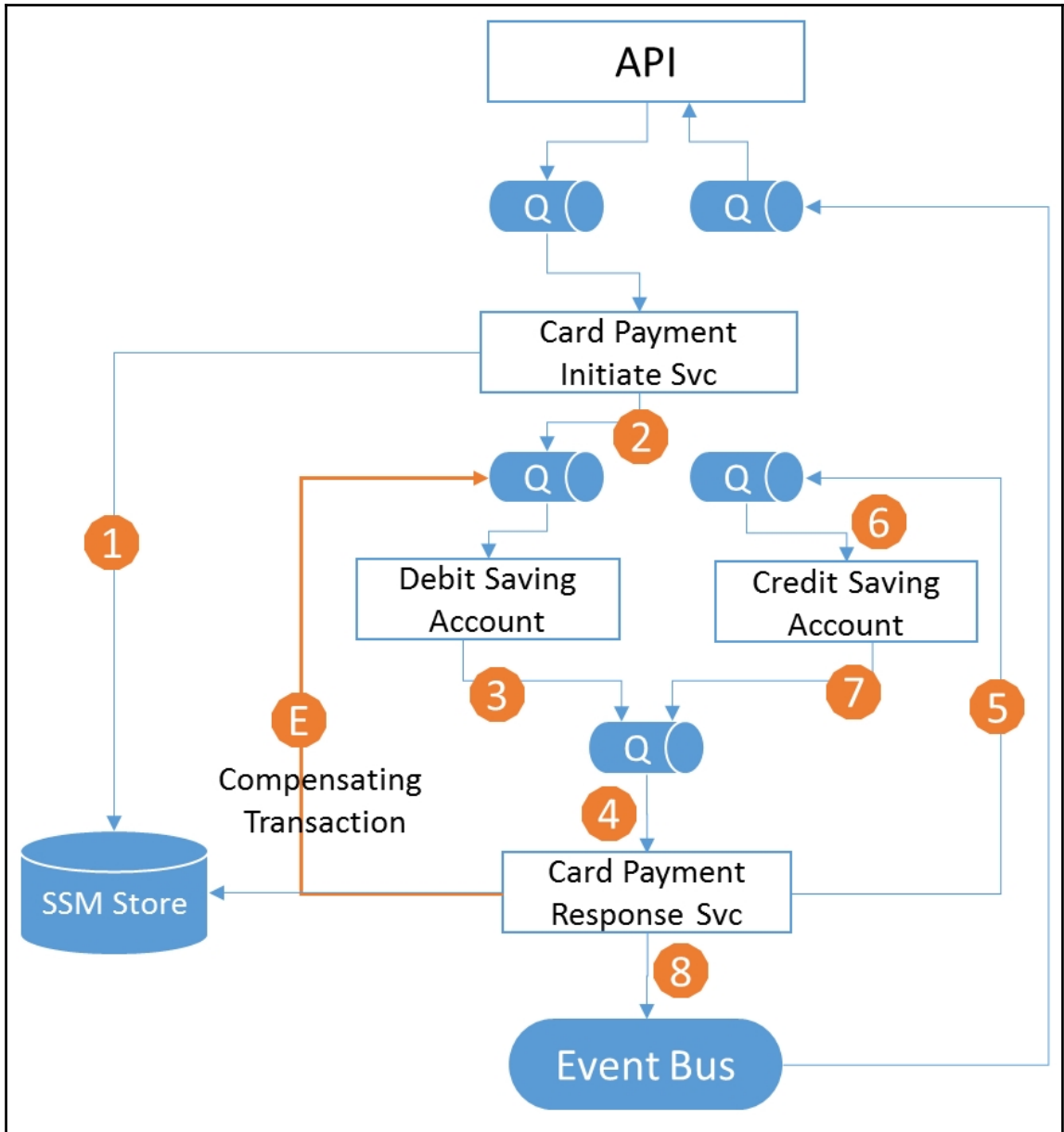


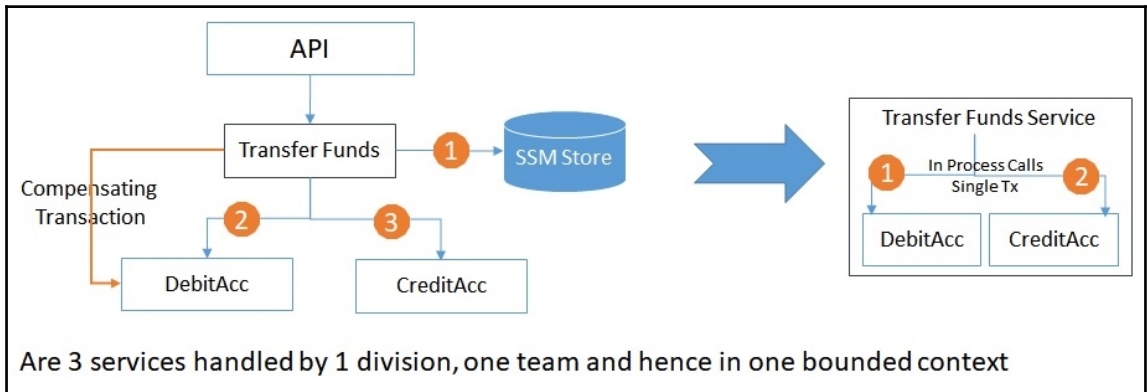
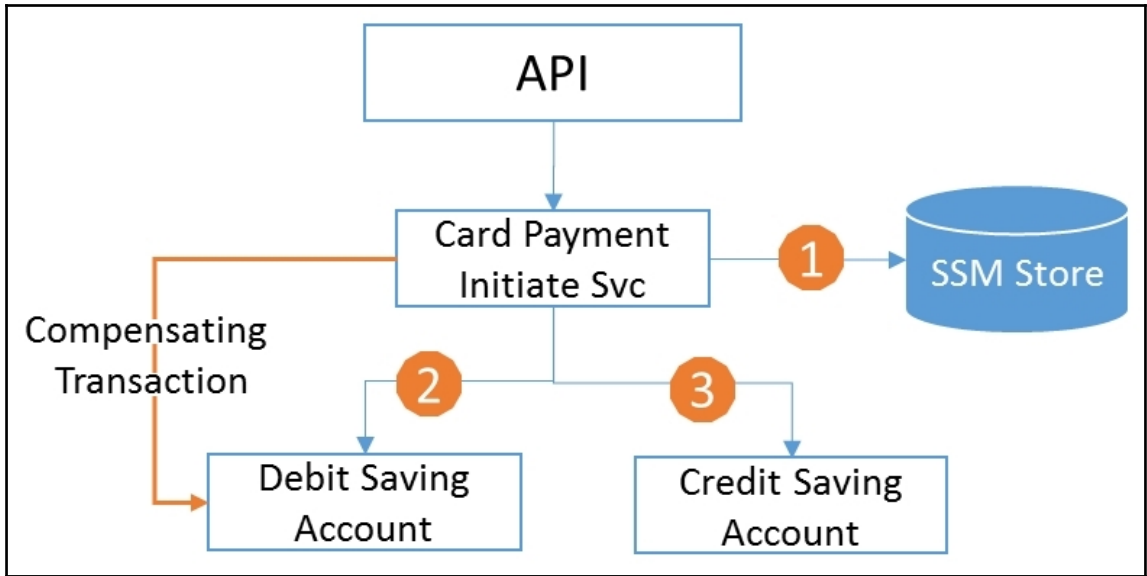




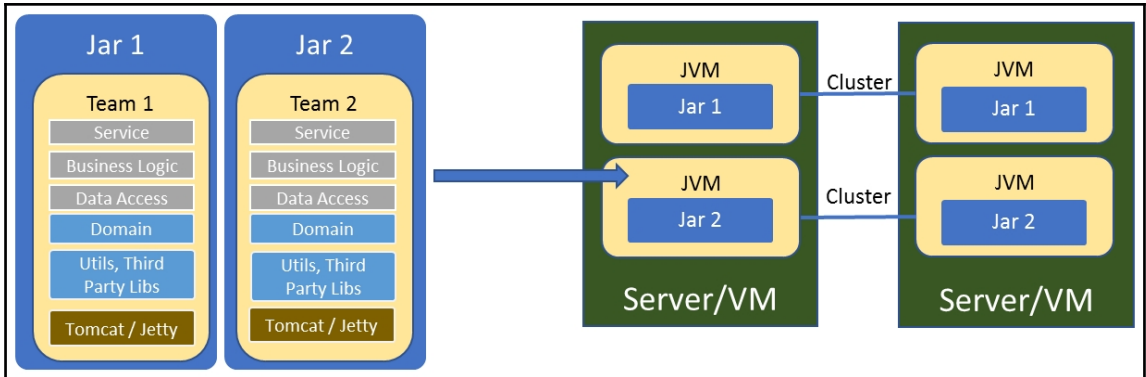
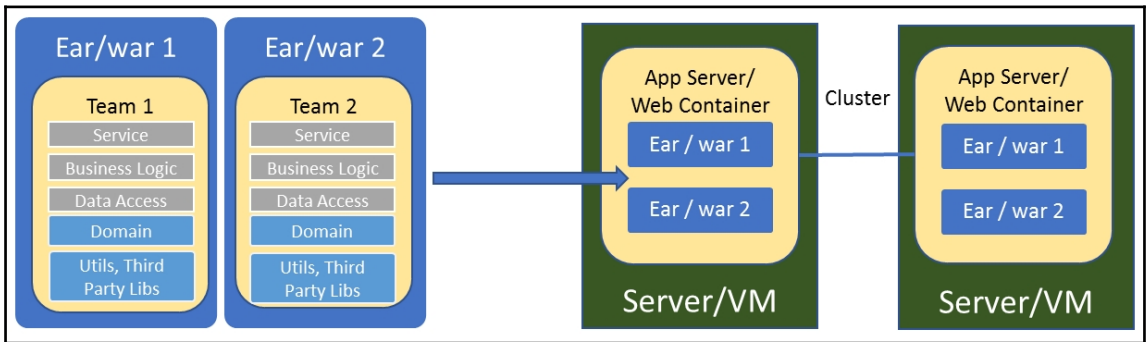
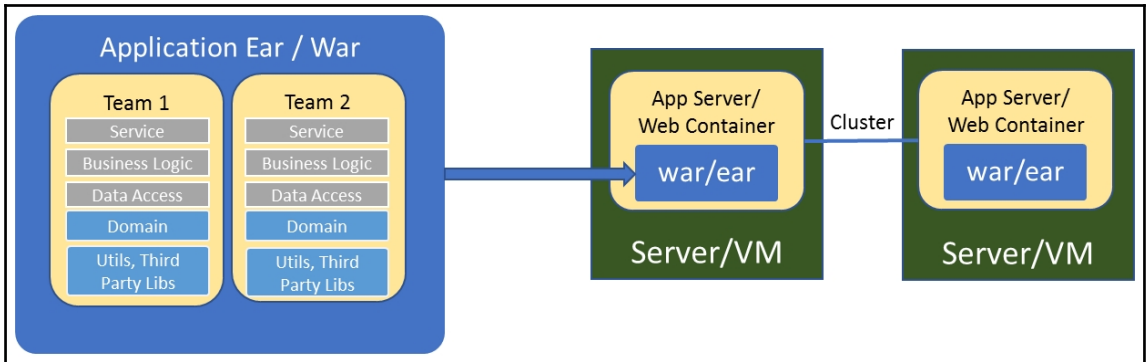


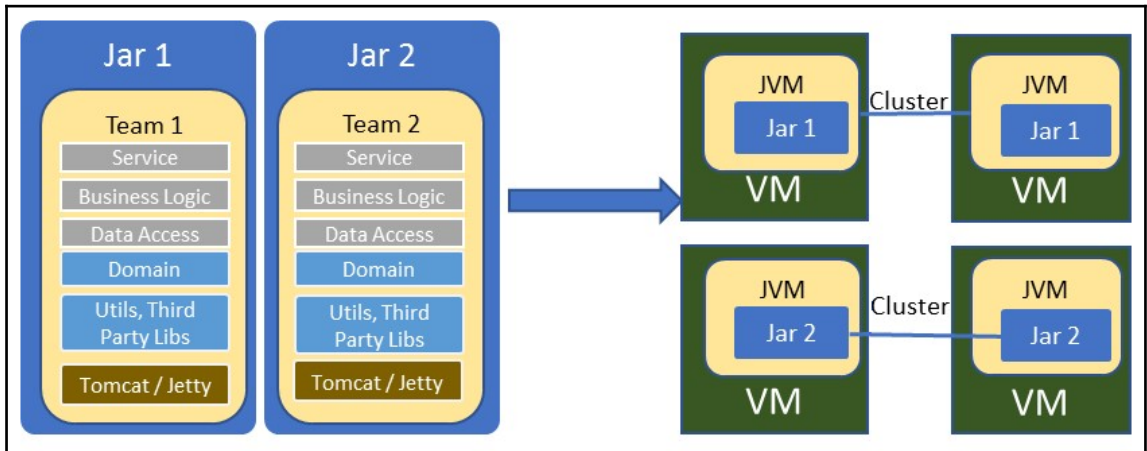
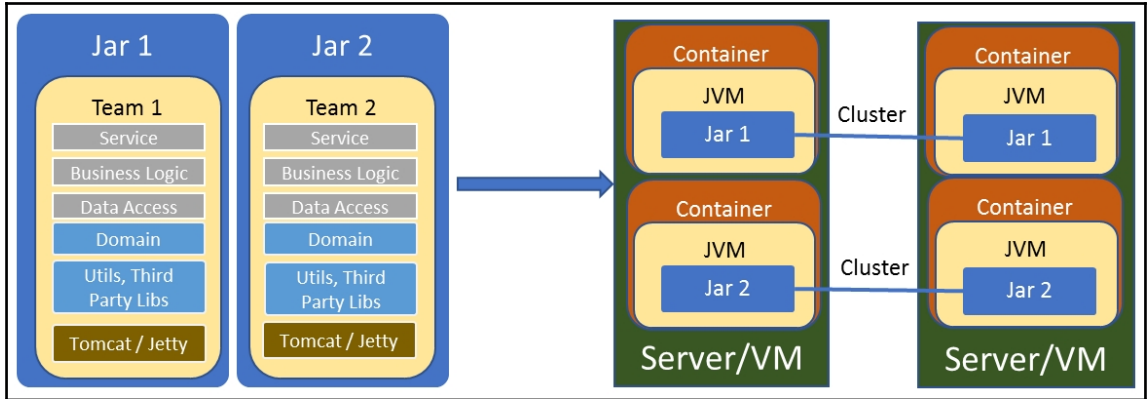


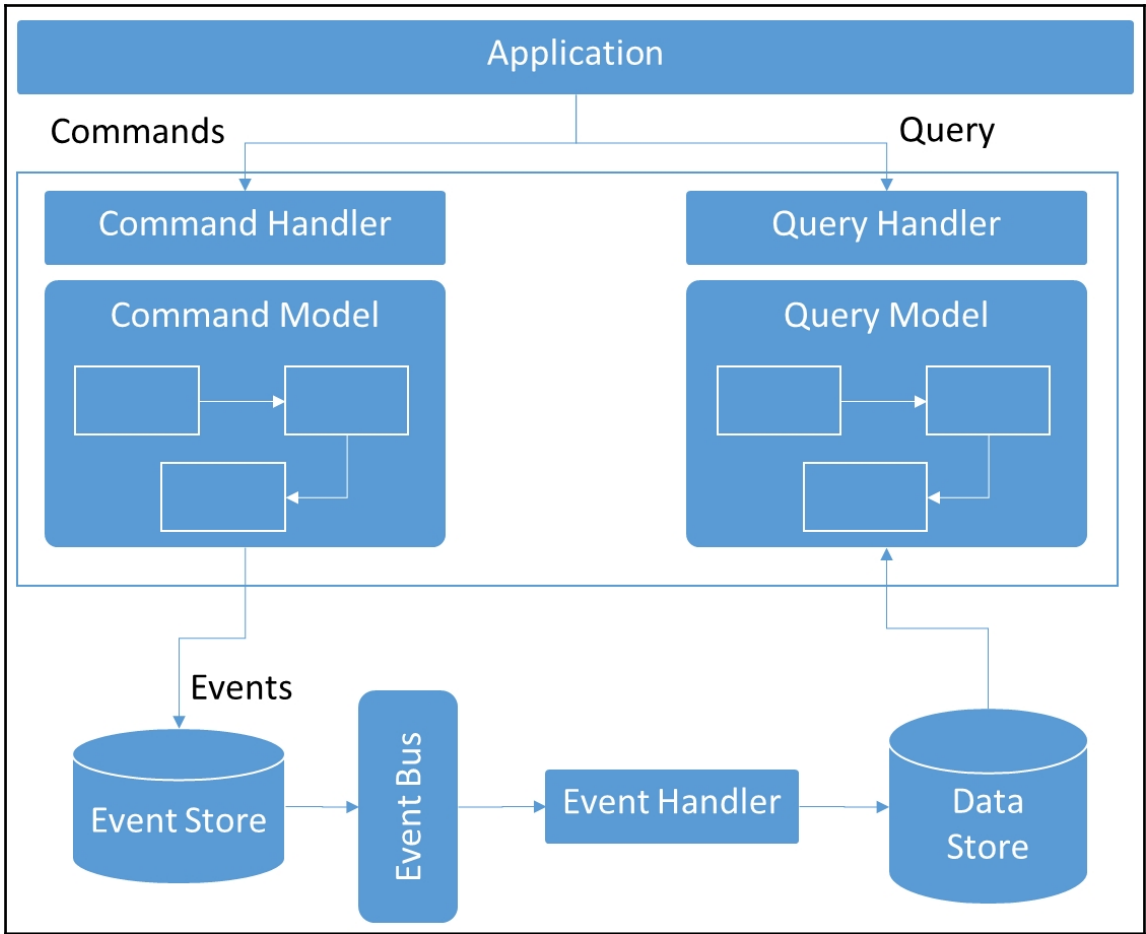


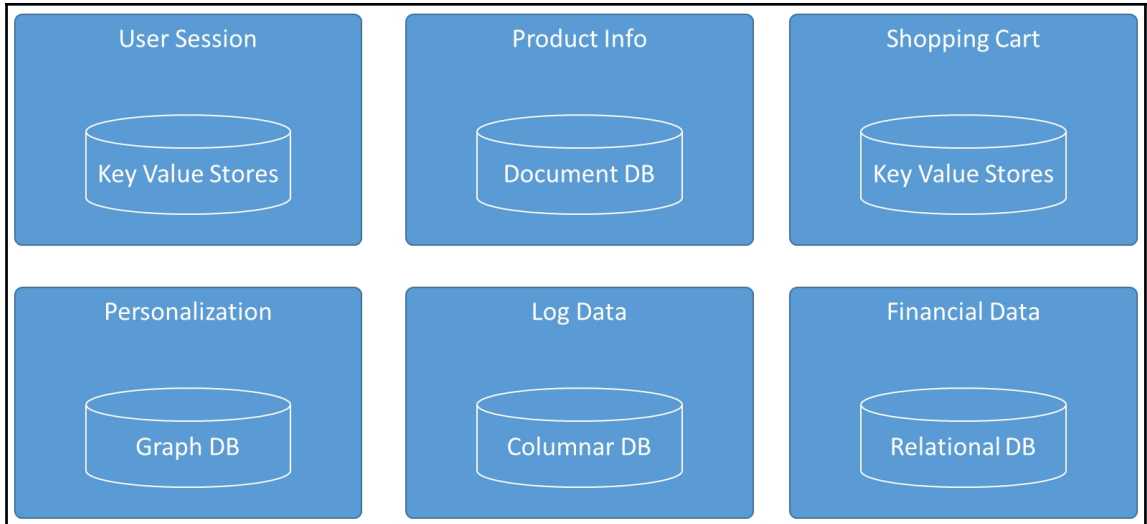
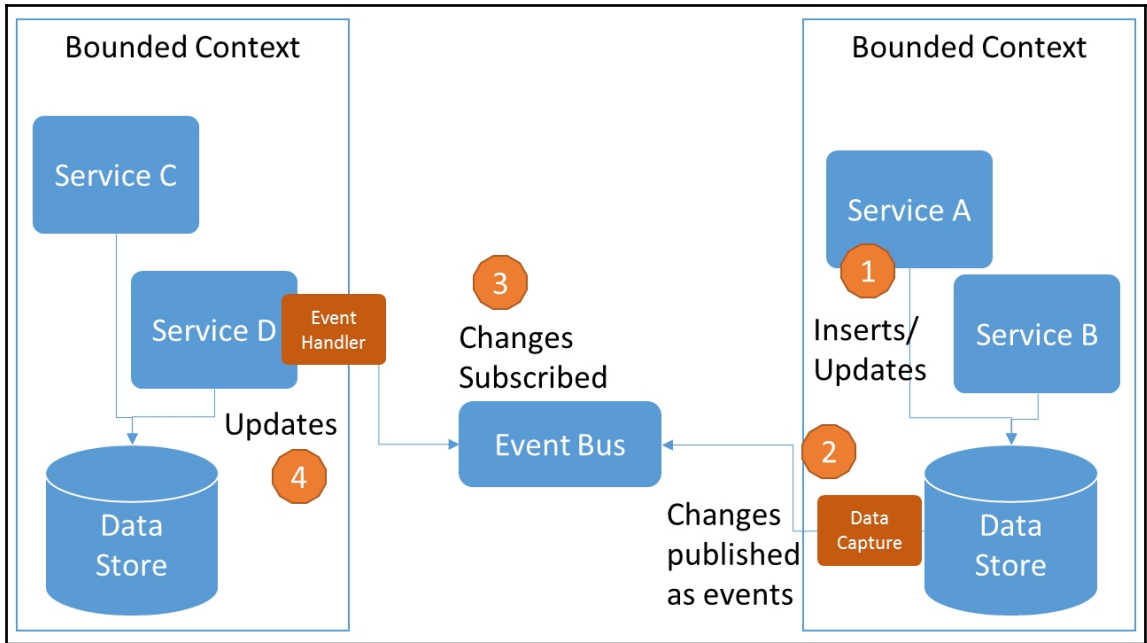


Are 3 services handled by 1 division, one team and hence in one bounded context







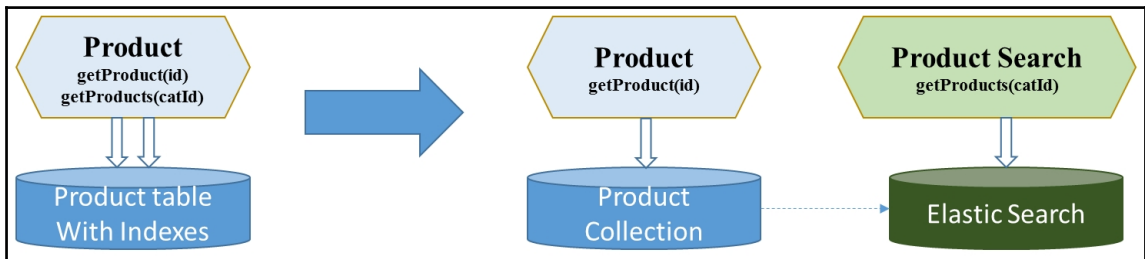


Chapter 4: Extending Your Cloud-Native Application

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-eureka</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.cloud</groupId>
  <artifactId>spring-cloud-starter-ribbon</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>
<dependency>
  <groupId>org.hsqldb</groupId>
  <artifactId>hsqldb</artifactId>
  <scope>runtime</scope>
</dependency>
</dependencies>
```



```
D:\Program Files\MongoDB\Server\3.4\bin>mongo
MongoDB shell version v3.4.1
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.4.1
Server has startup warnings:
2016-12-30T11:03:23.012+0530 I CONTROL [initandlisten]
2016-12-30T11:03:23.012+0530 I CONTROL [initandlisten]
2016-12-30T11:03:23.013+0530 I CONTROL [initandlisten]
2016-12-30T11:03:23.014+0530 I CONTROL [initandlisten]
> use masterdb
switched to db masterdb
> db.product.find()
{ "_id" : 1, "name" : "Apples", "catId" : 1 }
{ "_id" : 2, "name" : "Oranges", "catId" : 1 }
{ "_id" : 3, "name" : "Bananas", "catId" : 1 }
{ "_id" : 4, "name" : "Carrot", "catId" : 2 }
>
```



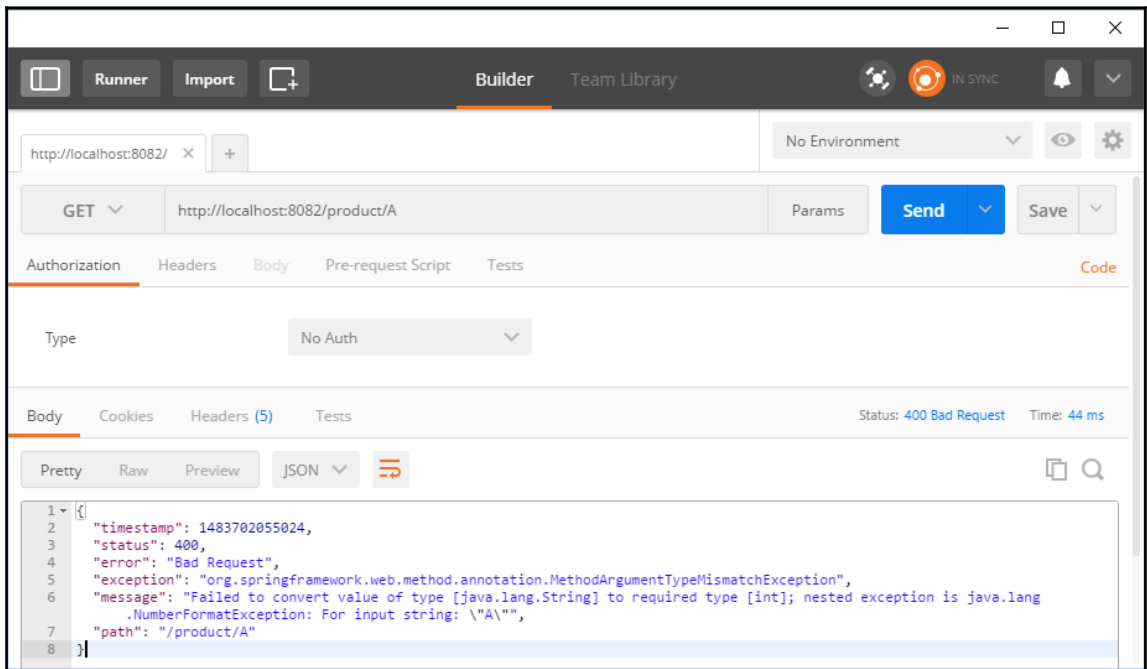
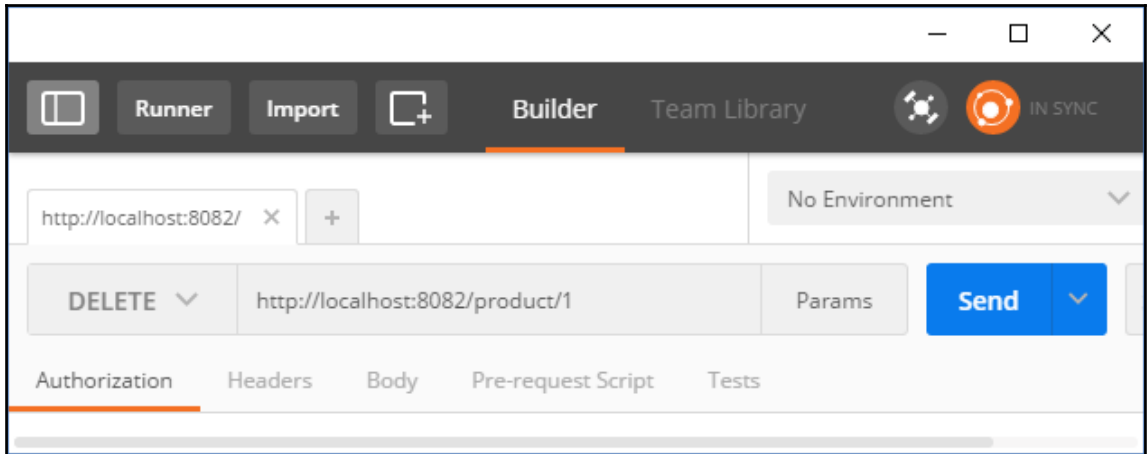
The image shows a REST client interface with the following components:

- Runner** and **Import** buttons.
- Builder** tab selected.
- Environment: **No Environment**.
- URLs: `http://localhost:8085/produ` and `http://localhost:9200/`.
- Method: **POST**.
- URL: `http://localhost:9200/product/external/_bulk.`
- Buttons: **Params** and **Send**.
- Tab: **Body** (selected).
- Content Type: **raw** (selected).
- Body content (lines 1-12):

```
1 {"index":{"_id":"1"}}
2 {"id":"1","name":"Apples","catId":1}
3
4 {"index":{"_id":"2"}}
5 {"id":"2","name":"Oranges","catId":1}
6
7 {"index":{"_id":"3"}}
8 {"id":"3","name":"Bananas","catId":1}
9
10 {"index":{"_id":"4"}}
11 {"id":"4","name":"Carrot","catId":2}
12
```

The screenshot shows the Postman Builder interface. The top navigation bar includes 'Runner', 'Import', 'Builder', and 'Team Library'. The address bar shows 'http://localhost:8082/'. The request method is 'POST' and the URL is 'http://localhost:8082/product'. The 'Body' tab is selected, and the 'raw' radio button is chosen. The JSON body is: `{ "name": "Grape", "catId": 1 }`. The 'Send' button is highlighted in blue.

The screenshot shows the Postman Builder interface. The top navigation bar includes 'Runner', 'Import', 'Builder', and 'Team Library'. The address bar shows 'http://localhost:8082/'. The request method is 'PUT' and the URL is 'http://localhost:8082/product/1'. The 'Body' tab is selected, and the 'raw' radio button is chosen. The JSON body is: `{ "id": 1, "name": "Fuji Apples", "catId": 1 }`. The 'Send' button is highlighted in blue.



The screenshot shows an API client interface with the following elements:

- Top navigation: Runner, Import, Builder (active), Team Library, and IN SYNC status.
- Address bar: `http://localhost:8082/` with a dropdown menu set to "No Environment".
- Request bar: `DELETE` method, `http://localhost:8082/product/534` URL, and a "Send" button.
- Request configuration tabs: Authorization, Headers (1), Body, Pre-request Script, Tests.
- Authorization dropdown: "No Auth".
- Response status: "Status: 500 Internal Server Error".
- Response format: Pretty, Raw, Preview (selected), JSON.
- Response body (JSON):



```
1 {  
2   "timestamp": 1483780254653,  
3   "status": 500,  
4   "error": "Internal Server Error",  
5   "exception": "com.mycompany.product.exception.BadRequestException",  
6   "message": "Product Not found with code 534",  
7   "path": "/product/534"  
8 }
```

```
{  
  "timestamp": 1483780958138,  
  "status": 404,  
  "error": "Not Found",  
  "exception": "com.mycompany.product.exception.BadRequestException",  
  "message": "1:Product Not found with code 156",  
  "path": "/product/156"  
}
```

localhost: ActiveMQ Con x

localhost:8161/admin/index.jsp

New Tab tech D-Link DSL-2750U wipro Welcome to BROADB: BSNL data usage general Apple Other bookmarks



[Home](#) | [Queues](#) | [Topics](#) | [Subscribers](#) | [Connections](#) | [Network](#) | [Scheduled](#) | [Send](#) [Support](#)

Welcome!

Welcome to the Apache ActiveMQ Console of **localhost** (ID:ajay-pc-60274-1483852875739-0:1)

You can find more information about Apache ActiveMQ on the [Apache ActiveMQ Site](#)

Broker

Name	localhost
Version	5.14.2
ID	ID:ajay-pc-60274-1483852875739-0:1
Uptime	3 minutes
Store percent used	0
Memory percent used	0
Temp percent used	0

Queue Views

- Graph
- XML

Topic Views

- XML

Subscribers Views

- XML

Useful Links

- Documentation
- FAQ
- Downloads
- Forums

Copyright 2005-2015 The Apache Software Foundation.

Home | Queues | Topics | Subscribers | Connections | Network | Scheduled | Send

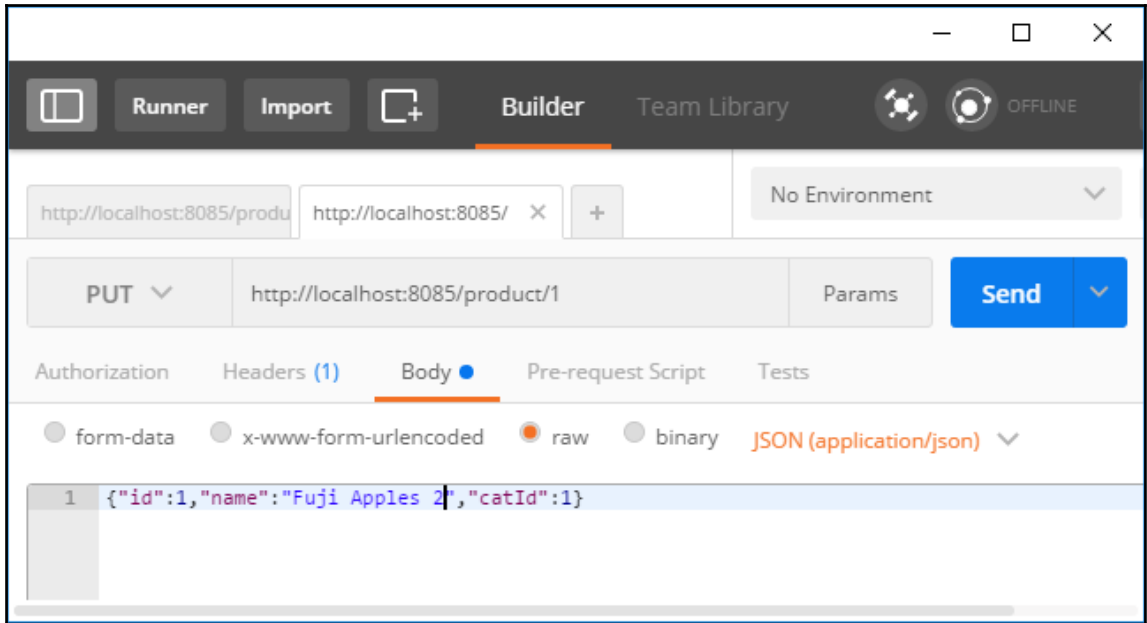
Topic Name

Topics

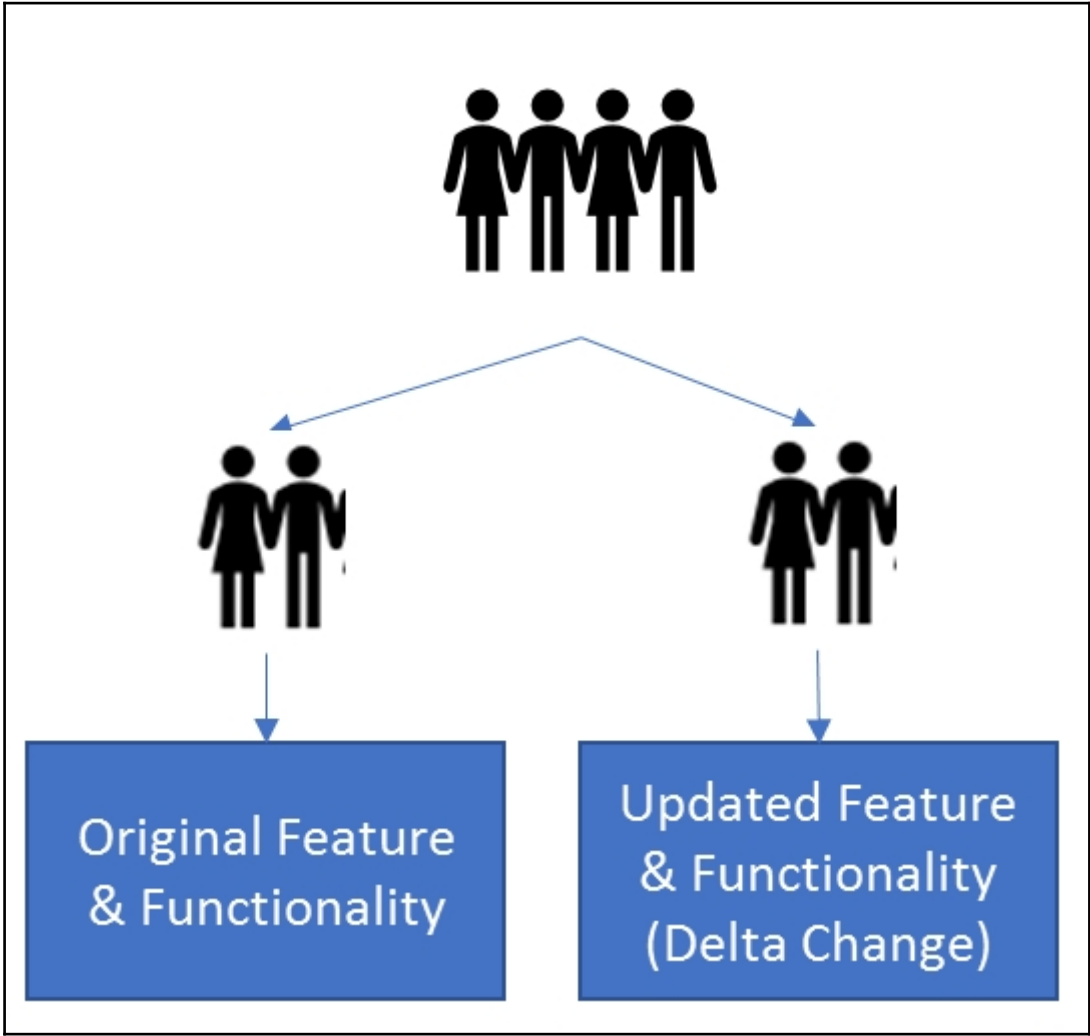
Name ↑	Number Of Consumers	Messages Enqueued	Messages Dequeued	Operations
ActiveMQ.Advisory.MasterBroker	0	1	0	Send To Active Subscribers Active Producers Delete
ActiveMQ.Advisory.Topic	0	1	0	Send To Active Subscribers Active Producers Delete
ProductT	0	0	0	Send To Active Subscribers Active Producers Delete

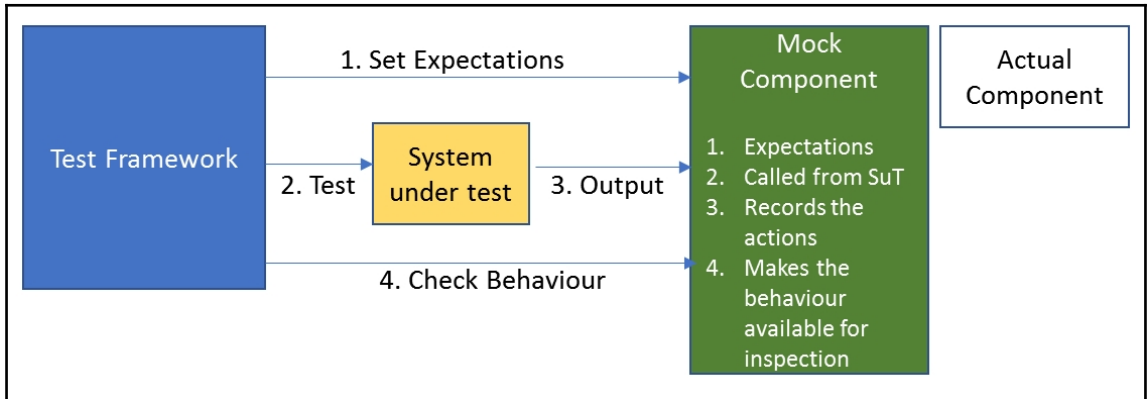
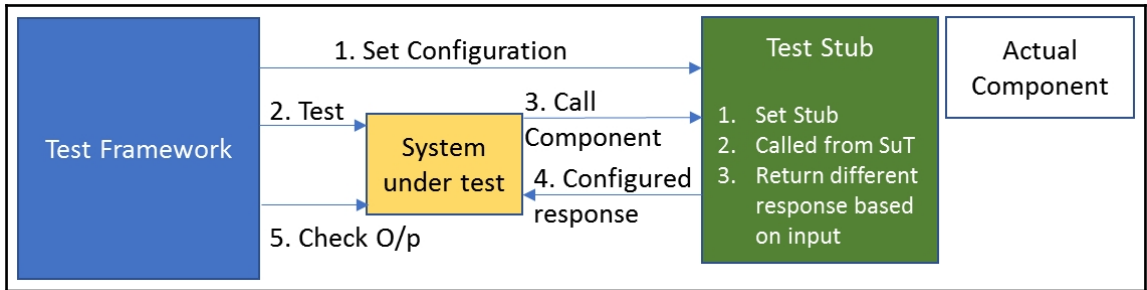
```

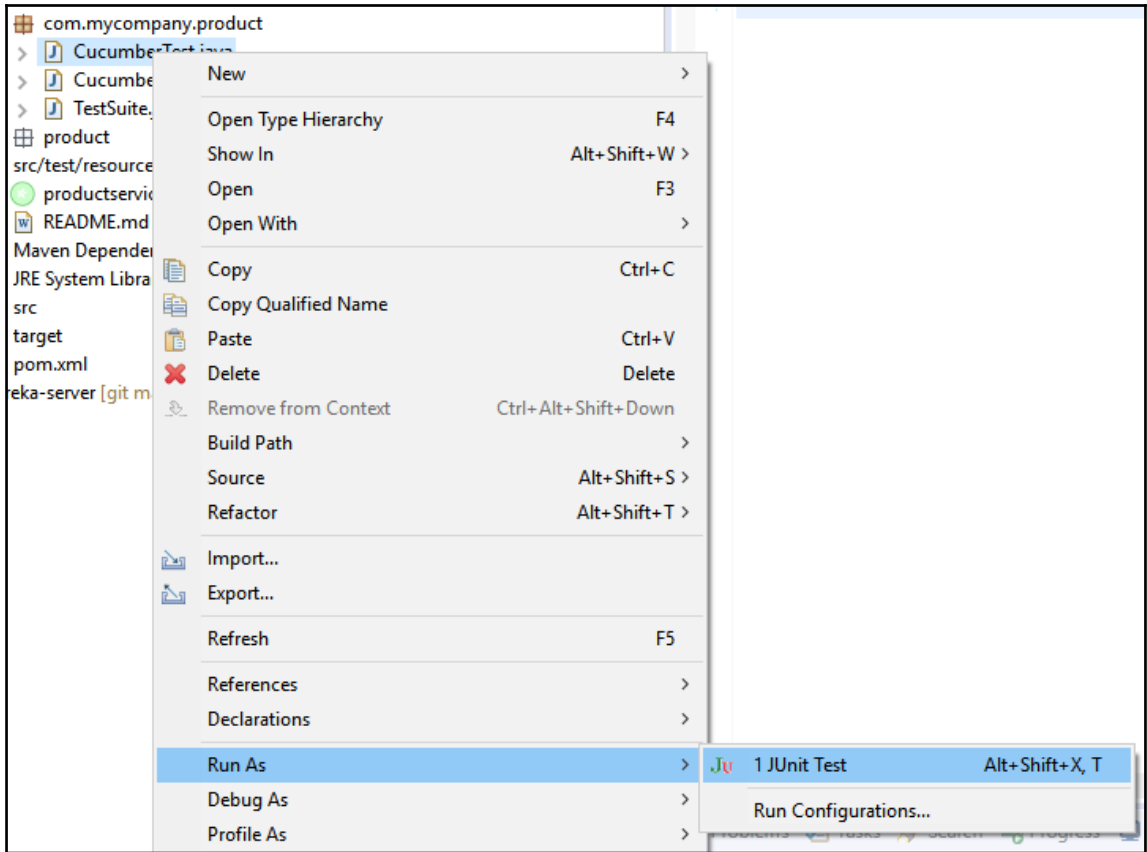
v ch3-product-nosql [git master]
  v src/main/java
    v com.mycompany.product
      > ProductService.java
      > ProductSpringApp.java
    v com.mycompany.product.dao
      > ProductRepository.java
    v com.mycompany.product.entity
      > Product.java
    v com.mycompany.product.exception
      > BadRequestException.java
      > GlobalControllerExceptionHandler.java
  v src/main/resources
    application.properties
    bootstrap.yml
  
```



Chapter 5: Testing Cloud-Native Applications



















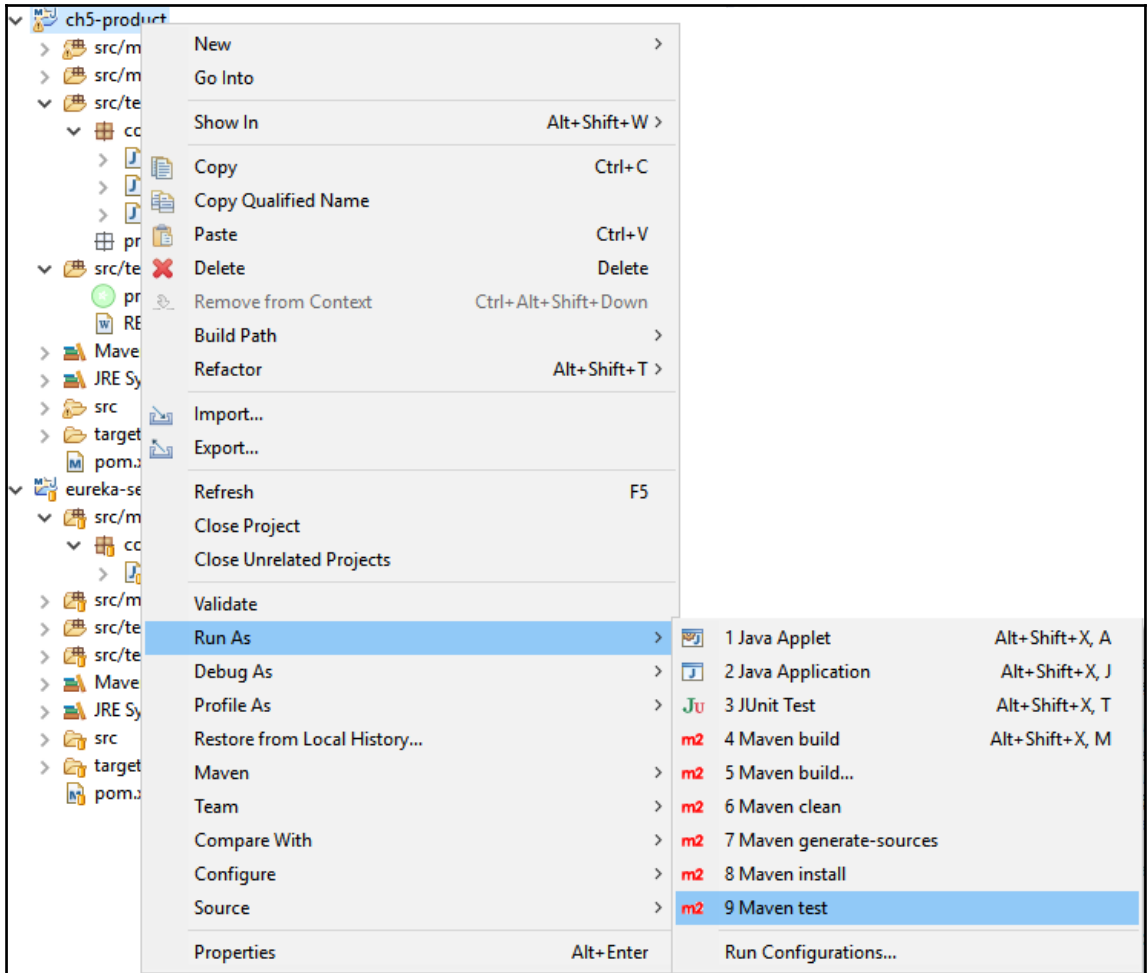
Finished after 26.23 seconds

Runs: 8/8

Errors: 0

Failures: 0

- ▼  com.mycompany.product.CucumberTest [Runner: JUnit 4] (0.400 s)
 - ▼  Feature: get Product (0.400 s)
 - ▼  Scenario: When product id is valid and exists (0.269 s)
 -  Given product Service is running (0.269 s)
 -  When get product service is called with existing product id 1 (0.000 s)
 -  Then I should get a response with HTTP status code 200 (0.000 s)
 -  And return Product details with name Apples and category 1 (0.000 s)
 - ▼  Scenario: When product id is invalid or does not exist (0.013 s)
 -  Given product Service is running (0.009 s)
 -  When get product Service is called with a non existing product id 456 (0.003 s)
 -  Then return a 404 not found status (0.001 s)
 -  And return error message "No product for id 456" (0.000 s)



PC > Data (D:) > Apps > wkNeon > ch5-product > target >

Name	Date modified	Type
classes	28-05-2017 23:15	File folder
coverage-reports	28-05-2017 23:08	File folder
generated-sources	28-05-2017 23:08	File folder
generated-test-sources	28-05-2017 23:08	File folder
maven-status	28-05-2017 23:08	File folder
site	28-05-2017 23:16	File folder
surefire-reports	28-05-2017 23:08	File folder
test-classes	28-05-2017 23:15	File folder

product

file:///D:/Apps/wkNeon/ch5-product/target/site/jacoco-ut/index.html

product

product

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed Cxty	Missed Lines	Missed Methods	Missed Classes
com.mycompany.product		26%		33%	8 14	23 31	6 11	0 3
com.mycompany.product.entity		76%	n/a	n/a	1 7	3 12	1 7	0 1
com.mycompany.product.exception		94%		50%	1 4	0 8	0 3	0 2
Total	109 of 199	45%	5 of 8	37%	10 25	26 51	7 21	0 6

Created with JaCoCo 0.7.9.201702052155

com.mycompany.product

file:///D:/Apps/wkNeon/ch5-product/target/site/jacoco-ut/com.mycompany.product/index.html

product > com.mycompany.product

com.mycompany.product

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
ProductService		24%		33%	6	9	17	22	4	6	0	1
ProductClient		15%		n/a	1	2	4	5	1	2	0	1
ProductSpringApp		58%		n/a	1	3	2	4	1	3	0	1
Total	101 of 137	26%	4 of 6	33%	8	14	23	31	6	11	0	3

Created with JaCoCo 0.7.9.201702052155

ProductService

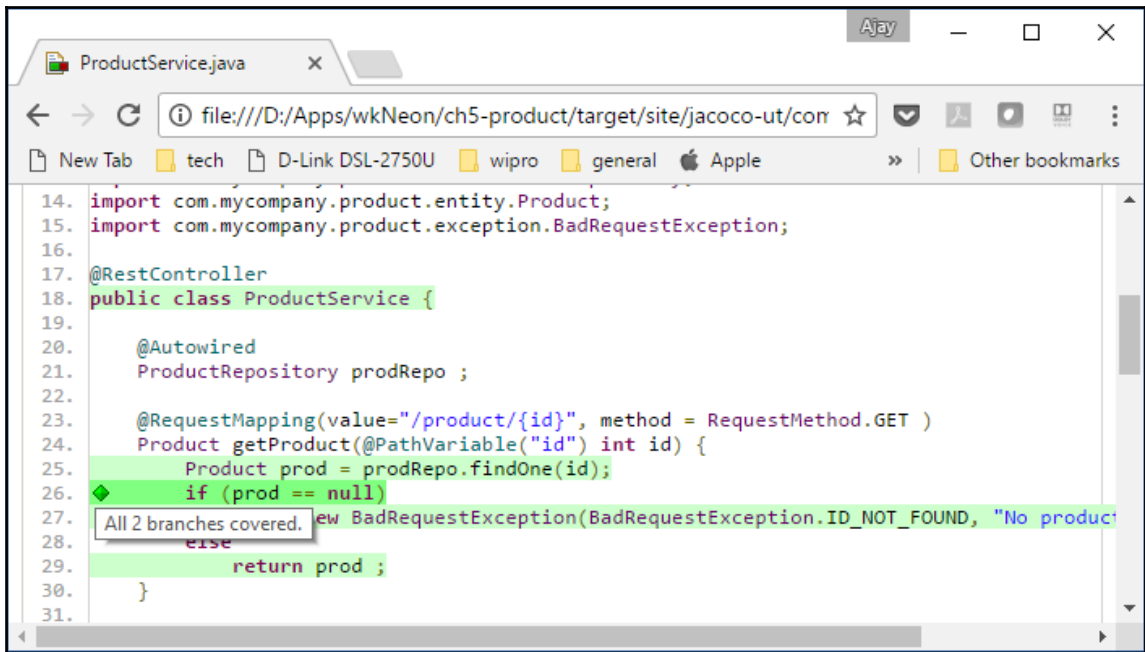
file:///D:/Apps/wkNeon/ch5-product/target/site/jacoco-ut/com.mycompany.product/ProductService

product > com.mycompany.product > ProductService

ProductService

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods
updateProduct(int, Product)		0%		0%	2	2	8	8	1	1
deleteProduct(int)		0%		0%	2	2	6	6	1	1
insertProduct(Product)		0%		n/a	1	1	2	2	1	1
getProductsForCategory(int)		0%		n/a	1	1	1	1	1	1
getProduct(int)		100%		100%	0	2	0	4	0	1
ProductService()		100%		n/a	0	1	0	1	0	1
Total	79 of 105	24%	4 of 6	33%	6	9	17	22	4	6

Created with JaCoCo 0.7.9.201702052155



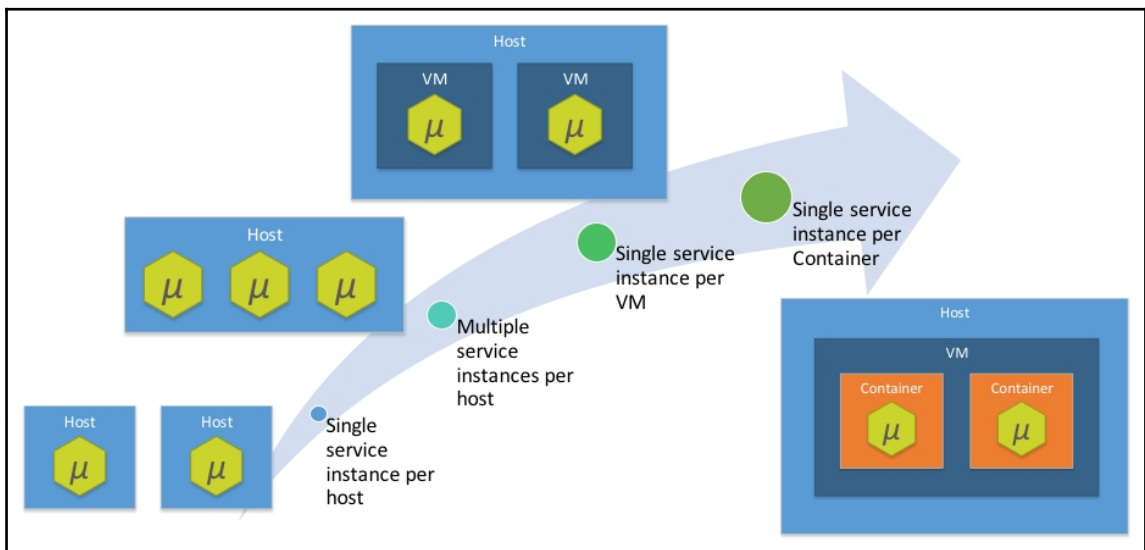
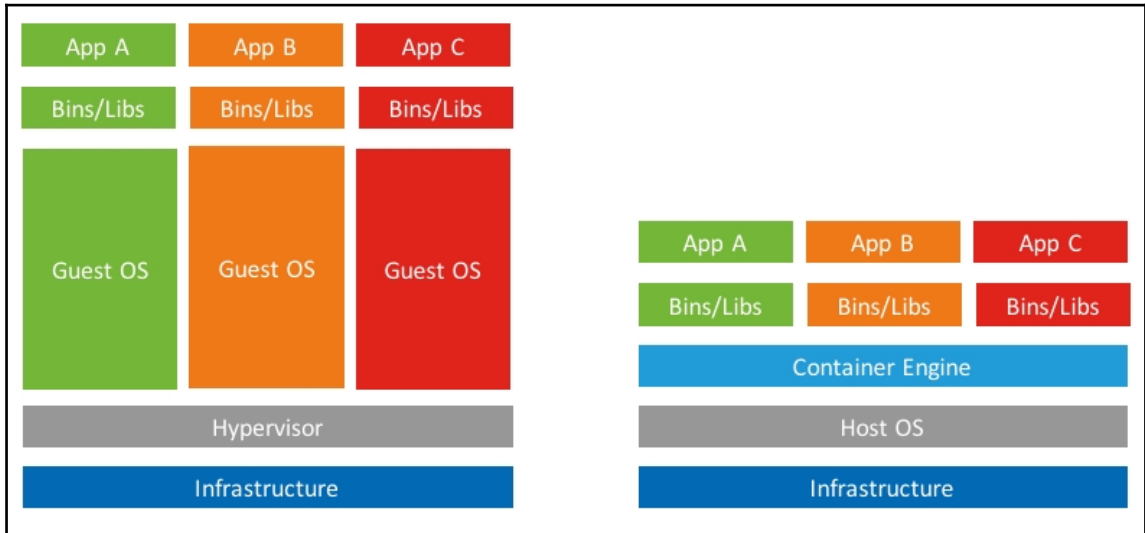
The image shows a screenshot of an IDE window titled "ProductService.java". The window contains the following Java code:

```
14. import com.mycompany.product.entity.Product;
15. import com.mycompany.product.exception.BadRequestException;
16.
17. @RestController
18. public class ProductService {
19.
20.     @Autowired
21.     ProductRepository prodRepo ;
22.
23.     @RequestMapping(value="/product/{id}", method = RequestMethod.GET )
24.     Product getProduct(@PathVariable("id") int id) {
25.         Product prod = prodRepo.findOne(id);
26.         if (prod == null)
27.             new BadRequestException(BadRequestException.ID_NOT_FOUND, "No product
28.         else
29.             return prod ;
30.     }
31.
```

A tooltip is visible over the code, stating "All 2 branches covered." This indicates that the code has been tested and both the 'if' and 'else' branches of the conditional statement have been executed during the test run.


```
1. package com.mycompany.product.exception;
2.
3. import java.io.IOException;
4.
5. import javax.servlet.http.HttpServletResponse;
6.
7. import org.springframework.http.HttpStatus;
8. import org.springframework.web.bind.annotation.ControllerAdvice;
9. import org.springframework.web.bind.annotation.ExceptionHandler;
10.
11. @ControllerAdvice
12. public class GlobalControllerExceptionHandler {
13.
14.     @ExceptionHandler(BadRequestException.class)
15.     void handleBadRequests(BadRequestException bre, HttpServletResponse response) throws IOException {
16.
17.         int respCode = (bre.errCode == BadRequestException.ID_NOT_FOUND) ?
18.             HttpStatus.NOT_FOUND.value() : HttpStatus.BAD_REQUEST.value();
19.
20.         response.sendError(respCode, bre.errCode + ":" + bre.getMessage());
21.     }
22. }
```

Chapter 6: Cloud-Native Application Deployment



```
$ docker build -t cloudnativejava/eureka-server .  
Sending build context to Docker daemon 39.92MB  
Step 1/5 : FROM openjdk:8-jdk-alpine  
----> 478bf389b75b  
Step 2/5 : RUN mkdir -p /app  
----> Using cache  
----> 9a09e0b0d6f2  
Step 3/5 : ADD target/eureka-server-0.0.1-SNAPSHOT.jar /app/app.jar  
----> 2964965b585e  
Removing intermediate container f757ed7f6818  
Step 4/5 : EXPOSE 8761  
----> Running in 848041f9fdee  
----> d9484343075e  
Removing intermediate container 848041f9fdee  
Step 5/5 : ENTRYPOINT /usr/bin/java -jar /app/app.jar  
----> Running in a6ea9f525f56  
----> 76e53a88d2ce  
Removing intermediate container a6ea9f525f56  
Successfully built 76e53a88d2ce  
Successfully tagged cloudnativejava/eureka-server:latest
```

```
[$ docker network create app_nw  
5828658b7d30f10391c6ce4dcfd7bef0c966873eb6df81c7b6f678fbf62c58f7
```

```
$ docker run -d --network app_nw --name eureka cloudnativejava/eureka-server  
c26a460a4cb384204dde3ff27a07773f0070fc809a9238145b88b34641a6c32c
```

```
$ docker build -t cloudnativejava/product-api .
Sending build context to Docker daemon 61.06MB
Step 1/5 : FROM openjdk:8-jdk-alpine
----> 478bf389b75b
Step 2/5 : RUN mkdir -p /app
----> Using cache
----> 9a09e0b0d6f2
Step 3/5 : ADD target/product-0.0.1-SNAPSHOT.jar /app/app.jar
----> Using cache
----> 4facc6546ab1
Step 4/5 : EXPOSE 8080
----> Using cache
----> 4982d78f6443
Step 5/5 : ENTRYPOINT /usr/bin/java -jar /app/app.jar --spring.profiles.active=docker
----> Using cache
----> f3fe921a8988
Successfully built f3fe921a8988
Successfully tagged cloudnativejava/product-api:latest
```

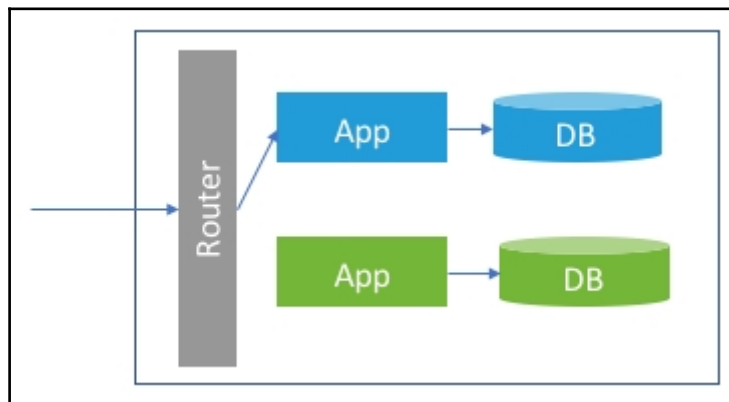
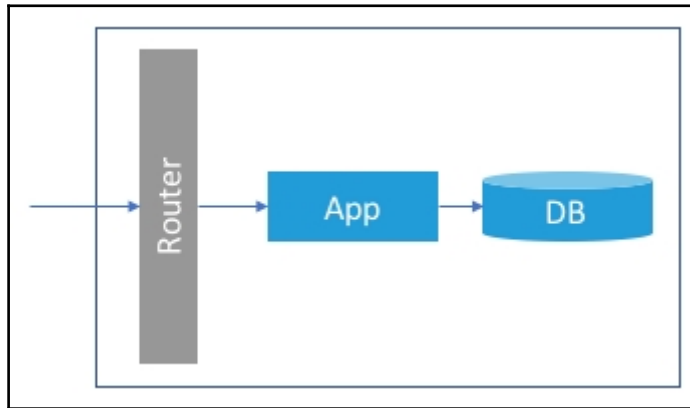
```
$ docker run -d -p 8011:8080 \
> --network app_nw \
> cloudnativejava/product-api
74cc71e3baf4c10df238235c642d3a264d22c03b669b4552821d901b64f33d20
$ docker run -d -p 8012:8080 \
> --network app_nw \
> cloudnativejava/product-api
f57adbd3bb3bb0628ef11d90b6ed98457e3290064cc588313e29cecc4e7c3708
```

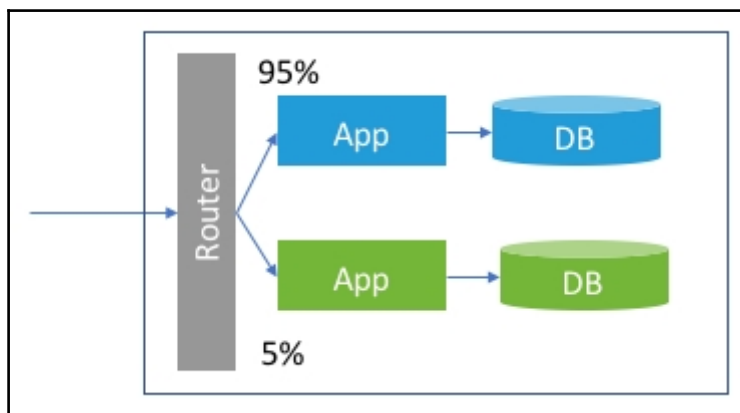
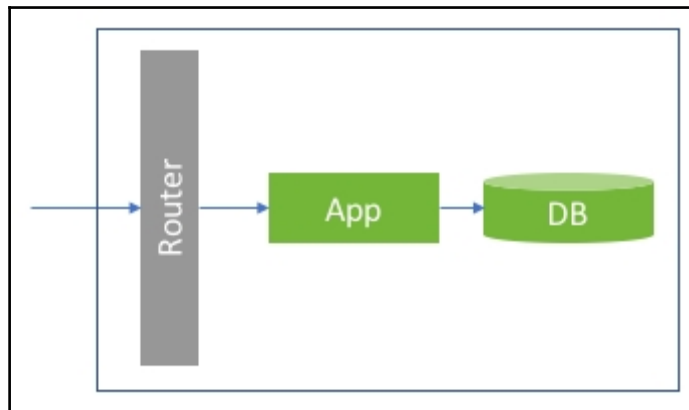
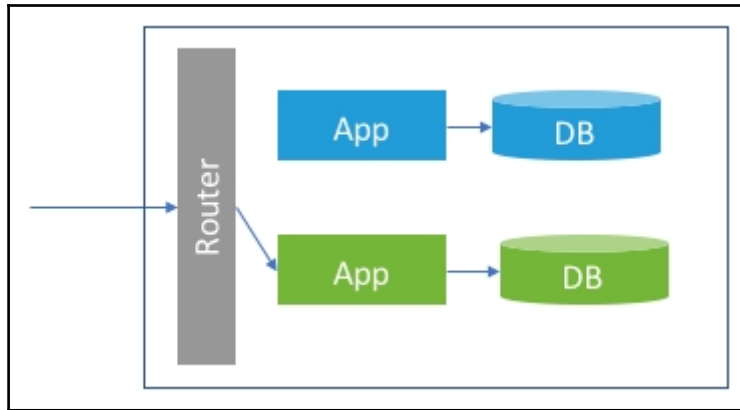
```
$ docker build -t cloudnativejava/datastore -f Dockerfile.postgres .
Sending build context to Docker daemon 61.06MB
Step 1/5 : FROM postgres:alpine
----> e9e9c4470522
Step 2/5 : ENV POSTGRES_USER dbuser POSTGRES_PASSWORD dbpass POSTGRES_DB product
----> Using cache
----> 20dcc938b14b
Step 3/5 : EXPOSE 5432
----> Using cache
----> aba503937193
Step 4/5 : RUN mkdir -p /docker-entrypoint-initdb.d
----> Using cache
----> b162459a436a
Step 5/5 : ADD import-postgres.sql /docker-entrypoint-initdb.d/import.sql
----> Using cache
----> d5def9256c7e
Successfully built d5def9256c7e
Successfully tagged cloudnativejava/datastore:latest
```

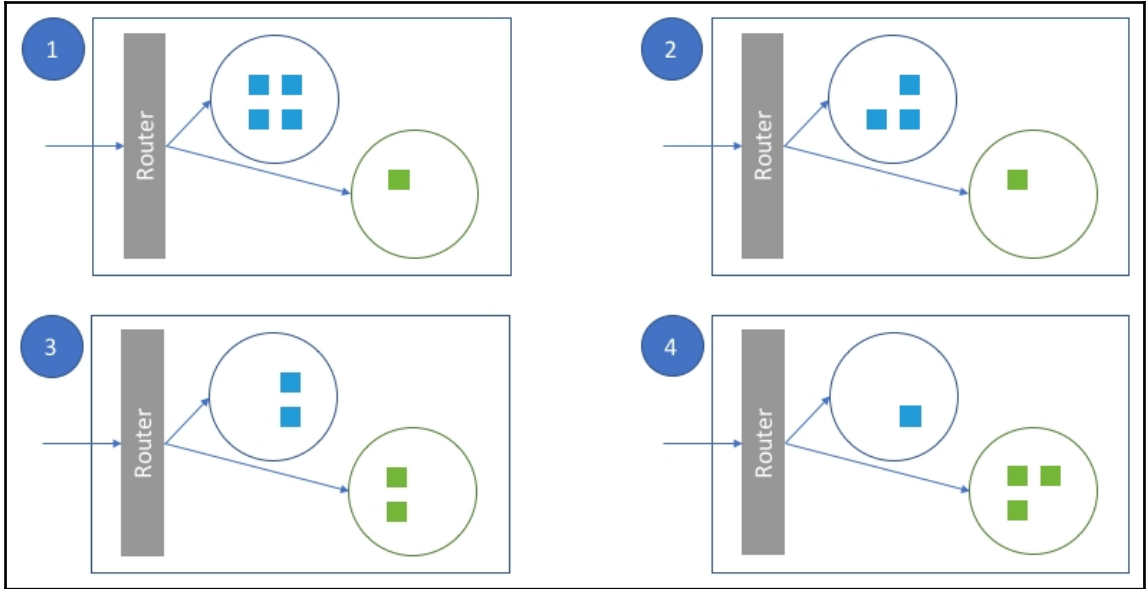
```
$ docker build -t cloudnativejava/product-api .
Sending build context to Docker daemon 61.06MB
Step 1/5 : FROM openjdk:8-jdk-alpine
----> 478bf389b75b
Step 2/5 : RUN mkdir -p /app
----> Using cache
----> 9a09e0b0d6f2
Step 3/5 : ADD target/product-0.0.1-SNAPSHOT.jar /app/app.jar
----> Using cache
----> 4facc6546ab1
Step 4/5 : EXPOSE 8080
----> Using cache
----> 4982d78f6443
Step 5/5 : ENTRYPOINT /usr/bin/java -jar /app/app.jar --spring.profiles.active=docker
----> Using cache
----> f3fe921a8988
Successfully built f3fe921a8988
Successfully tagged cloudnativejava/product-api:latest
```

```
$ docker run -d -p 5432:5432 \
> --network app_nw \
> --name datastore \
> cloudnativejava/datastore
0663e27115b1807b2924903c3996cd0f927c248624cf8f18ef617168fc31a53b
```

```
$ docker run -d -p 8011:8080 \  
>   --network app_nw \  
>   cloudnativejava/product-api \  
>   --spring.profiles.active=postgres  
a1b5e68cf251f3073961b75f91caf10bb39c321712114836fc474c002bc1d73a  
  
$ docker run -d -p 8012:8080 \  
>   --network app_nw \  
>   cloudnativejava/product-api \  
>   --spring.profiles.active=postgres  
6b9081236786544df3012d3d018c6e17003977d3e6b99715ac1eeb3931b1072b
```







Enter an item name

» Required field



Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.



Pipeline

Orchestrates long-running activities that can span multiple build slaves. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.



Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.



GitHub Organization

Scans a GitHub organization (or user account) for all repositories matching some defined markers.



Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.

OK

Pipeline

Definition

SCM

Repositories

Repository URL

Credentials

Branches to build

Branch Specifier (blank for 'any')

Repository browser

Additional Behaviours

Script Path

Lightweight checkout

[Pipeline Syntax](#)

Pipeline CloudNativeJava



Last Successful Artifacts

- [eureka-server-0.0.1-SNAPSHOT.jar](#) 38.06 MB [view](#)
- [product-0.0.1-SNAPSHOT.jar](#) 55.77 MB [view](#)



Recent Changes

Stage View

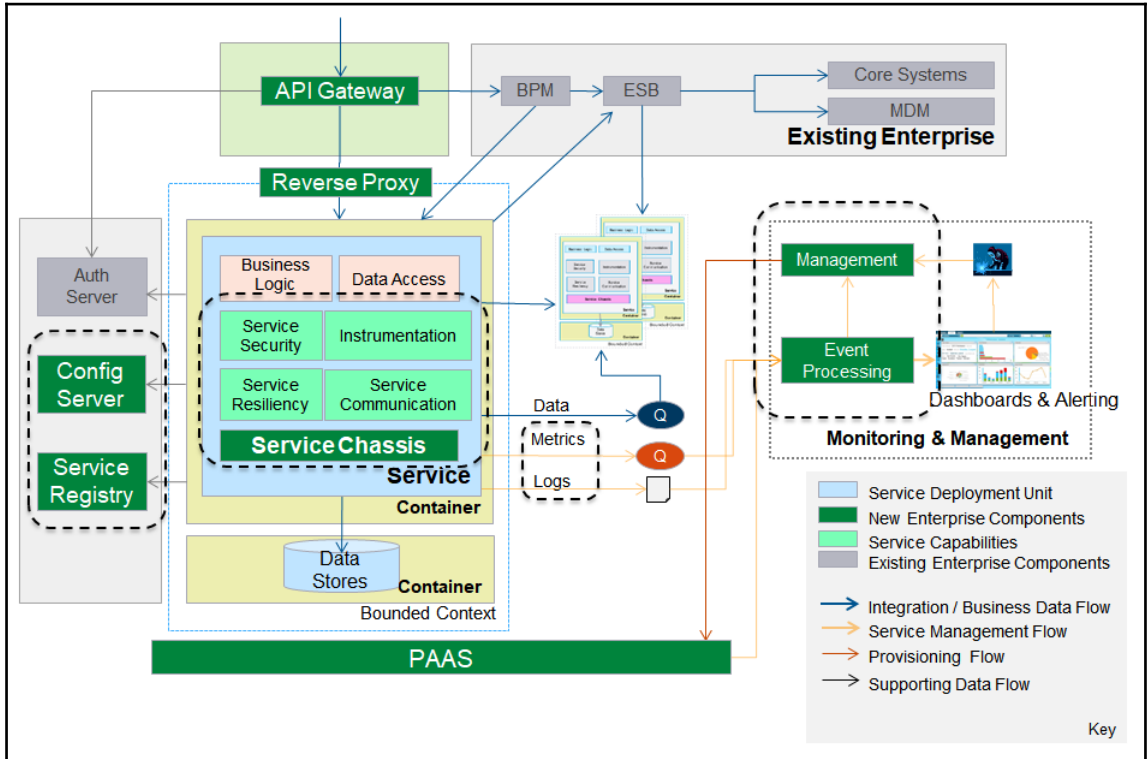
Average stage times:
(Average full run time: ~1min 58s)

	Preparation	Eureka Server	Build - Eureka Server	Results - Eureka Server	Product API	Build - Product API	Results - Product API
Average	2s	32ms	13s	263ms	90ms	1min 40s	183ms
#2 Dec 20 01:30 No Changes	2s	32ms	13s	263ms	90ms	1min 40s	183ms



[Latest Test Result](#) (no failures)

Chapter 7: Cloud-Native Application Runtime

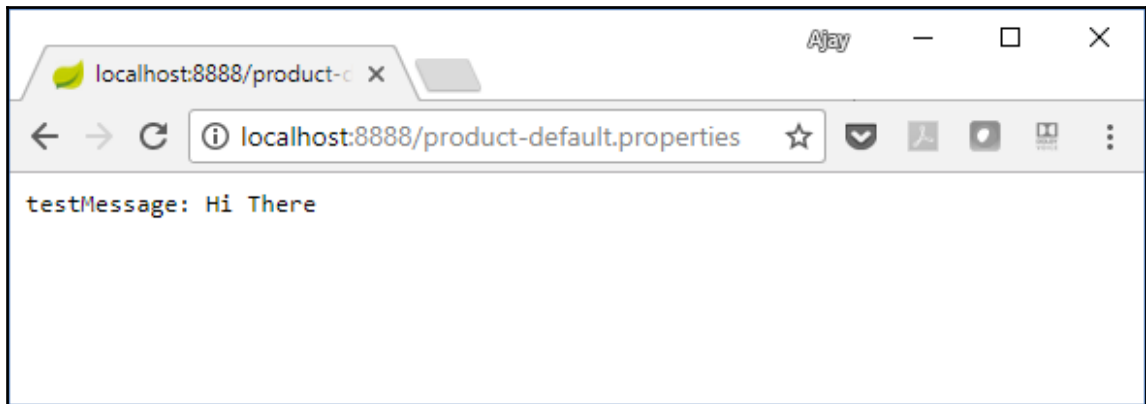


```
config-server/pom.xml
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
3   <modelVersion>4.0.0</modelVersion>
4
5   <groupId>com.mycompany.infra</groupId>
6   <artifactId>config-server</artifactId>
7   <version>0.0.1-SNAPSHOT</version>
8
9   <parent>
10    <groupId>org.springframework.boot</groupId>
11    <artifactId>spring-boot-starter-parent</artifactId>
12    <version>1.5.9.RELEASE</version>
13  </parent>
14
15  <dependencyManagement>
16    <dependencies>
17      <dependency>
18        <groupId>org.springframework.cloud</groupId>
19        <artifactId>spring-cloud-config</artifactId>
20        <version>1.4.0.RELEASE</version>
21        <type>pom</type>
22        <scope>import</scope>
23      </dependency>
24    </dependencies>
25  </dependencyManagement>
26
27  <dependencies>
28    <dependency>
29      <groupId>org.springframework.cloud</groupId>
30      <artifactId>spring-cloud-config-server</artifactId>
31    </dependency>
32  </dependencies>
33
34  <build>
35    <plugins>
36      <plugin>
37        <groupId>org.springframework.boot</groupId>
38        <artifactId>spring-boot-maven-plugin</artifactId>
39      </plugin>
40    </plugins>
41  </build>
42 </project>
```

```
config-server/pom.xml  ConfigServiceApplication.java
1 package com.mycompany.infra.configsvr;
2
3
4 import org.springframework.boot.SpringApplication;
5
6
7
8 @EnableConfigServer
9 @SpringBootApplication
10 public class ConfigServiceApplication {
11
12     public static void main(String[] args) {
13         SpringApplication.run(ConfigServiceApplication.class, args);
14     }
15 }
```

```
Registering beans for JMX exposure on startup
Bean with name 'configurationPropertiesRebinder' has been autodetected for JMX exposure
Bean with name 'refreshEndpoint' has been autodetected for JMX exposure
Bean with name 'environmentManager' has been autodetected for JMX exposure
Bean with name 'refreshScope' has been autodetected for JMX exposure
Located managed bean 'environmentManager': registering with JMX server as MBean [org.springfr
Located managed bean 'refreshScope': registering with JMX server as MBean [org.springfr
Located managed bean 'configurationPropertiesRebinder': registering with JMX server as MBean [org.springfr
Located managed bean 'refreshEndpoint': registering with JMX server as MBean [org.springfr
Starting beans in phase 0
Tomcat started on port(s): 8888 (http)
Started ConfigServiceApplication in 5.666 seconds (JVM running for 6.102)
```

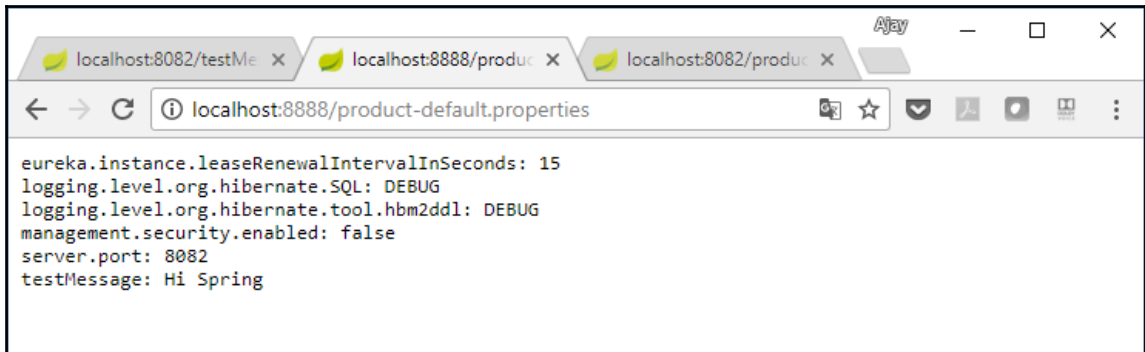
```
localhost:8888/product/c x
localhost:8888/product/default
{"name":"product","profiles":
["default"],"label":null,"version":"56a54504cb08fd0d8a2ee80cdcabeaa1bea71f29","state":null
,"propertySources":
[{"name":"file:../git/file:/D:/Apps/wkNeon/git/product.properties","source":
{"testMessage":"Hi There"}}]}
```



```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-eureka</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-config</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-ribbon</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
</dependencies>
```

```
: Fetching config from server at: http://localhost:8888
: Located environment: name=product, profiles=[default], label=null,
```

```
: Discovery Client initialized at timestamp 1518938358612 with initial
Registering application product with eureka with status UP
Saw local status change event StatusChangeEvent [timestamp=1518938358
DiscoveryClient_PRODUCT/localhost:product:8082: registering service..
DiscoveryClient_PRODUCT/localhost:product:8082 - registration status:
Tomcat started on port(s): 8082 (http)
Updating port to 8082
Started ProductSpringApp in 20.527 seconds (JVM running for 21.081)
```



The screenshot shows a web browser window with three tabs: localhost:8082/testMe, localhost:8888/product, and localhost:8082/product. The address bar shows localhost:8888/product-default.properties. The page content displays the following configuration properties:

```
eureka.instance.leaseRenewalIntervalInSeconds: 15
logging.level.org.hibernate.SQL: DEBUG
logging.level.org.hibernate.tool.hbm2ddl: DEBUG
management.security.enabled: false
server.port: 8082
testMessage: Hi Spring
```

```
: Refreshing org.springframework.context.annotation.AnnotationConfigApplicationContext@711ddb5: sta
: JSR-330 'javax.inject.Inject' annotation found and supported for autowiring
: Bean 'configurationPropertiesRebinderAutoConfiguration' of type [org.springframework.cloud.autocon
: Cannot determine local hostname
: Fetching config from server at: http://localhost:8888
```

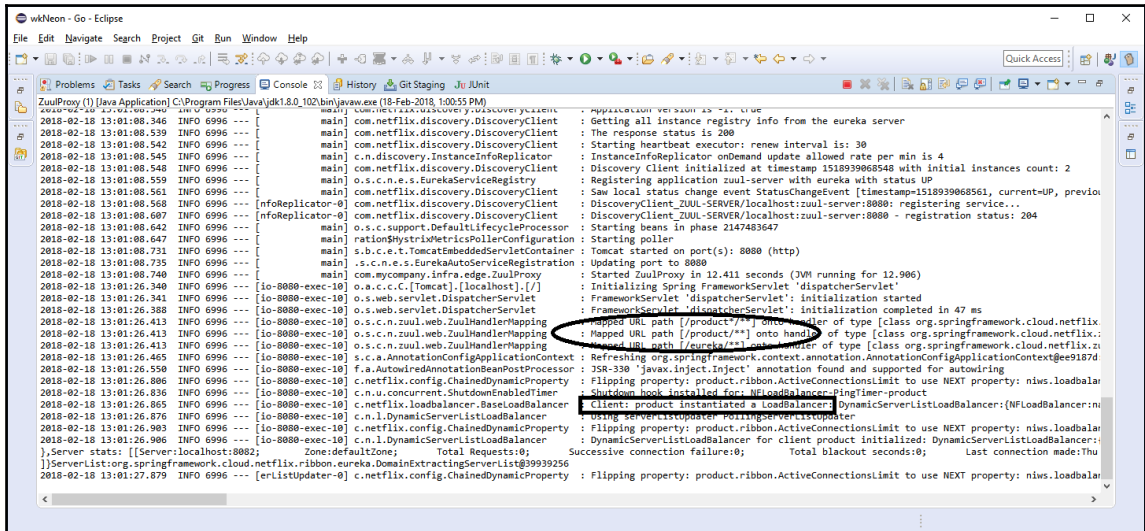

ch7-zuul/pom.xml

```
5 <groupId>com.mycompany.infra</groupId>
6 <artifactId>zuul-server</artifactId>
7 <version>0.0.1-SNAPSHOT</version>
8
9 <parent>
10 <groupId>org.springframework.boot</groupId>
11 <artifactId>spring-boot-starter-parent</artifactId>
12 <version>1.5.9.RELEASE</version>
13 </parent>
14
15 <dependencyManagement>
16 <dependencies>
17 <dependency>
18 <groupId>org.springframework.cloud</groupId>
19 <artifactId>spring-cloud-starter-netflix</artifactId>
20 <version>1.4.0.RELEASE</version>
21 <type>pom</type>
22 <scope>import</scope>
23 </dependency>
24 </dependencies>
25 </dependencyManagement>
26
27 <dependencies>
28 <dependency>
29 <groupId>org.springframework.cloud</groupId>
30 <artifactId>spring-cloud-starter-zuul</artifactId>
31 </dependency>
32 <dependency>
33 <groupId>org.springframework.boot</groupId>
34 <artifactId>spring-boot-starter-web</artifactId>
35 </dependency>
36 <dependency>
37 <groupId>org.springframework.cloud</groupId>
38 <artifactId>spring-cloud-starter-eureka</artifactId>
39 </dependency>
40 </dependencies>
```

```

ch7-zuul/pom.xml  ZuulProxy.java
1 package com.mycompany.infra.edge;
2
3
4 import org.springframework.boot.SpringApplication;
7
8 @EnableZuulProxy
9 @SpringBootApplication
10 public class ZuulProxy {
11
12     public static void main(String[] args) {
13         SpringApplication.run(ZuulProxy.class, args);
14     }
15 }

```

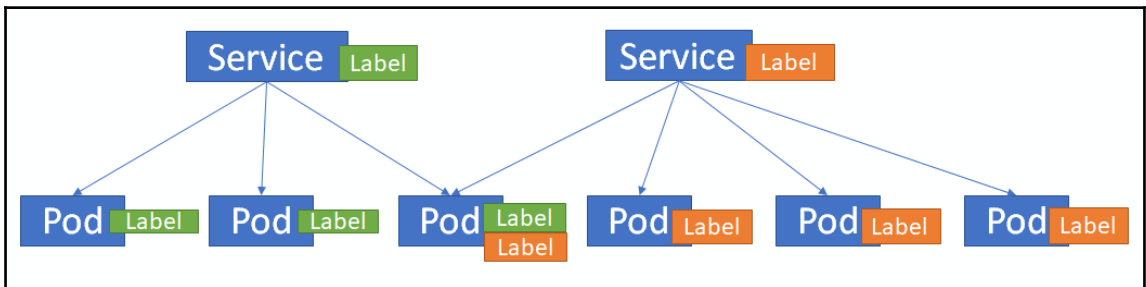
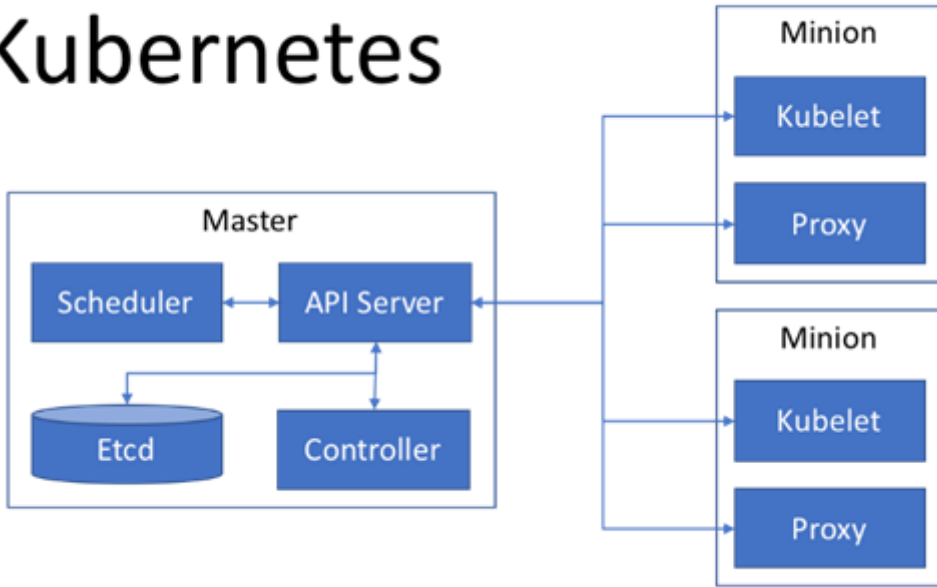


```

: FrameworkServlet 'dispatcherServlet': initialization started
: FrameworkServlet 'dispatcherServlet': initialization completed in 39 ms
: [172.21.185.129]:5701 [ProductCluster] [3.7.8] Initializing cluster partition table arrangement...
: HHH000397: Using ASTQueryTranslatorFactory
: select product0_.id as id1_0_, product0_.cat_id as cat_id2_0_, product0_.name as name3_0_ from product product0_ where product0_.cat_id=?
: Resolving eureka endpoints via configuration
: select product0_.id as id1_0_0_, product0_.cat_id as cat_id2_0_0_, product0_.name as name3_0_0_ from product product0_ where product0_.id=?

```

Kubernetes



```

$ minikube start
Starting local Kubernetes v1.9.0 cluster...
Starting VM...
Downloading Minikube ISO
142.22 MB / 142.22 MB [=====] 100.00% 0s
Getting VM IP address...
Moving files into cluster...
Downloading localkubernetes binary
162.41 MB / 162.41 MB [=====] 100.00% 0s
0 B / 65 B [=====] 0.00%
65 B / 65 B [=====] 100.00% 0s
Setting up certs...
Connecting to cluster...
Setting up kubeconfig...
Starting cluster components...
Kubectl is now configured to use the cluster.
Loading cached images from config file.
$

```

```

$ kubectl get svc

```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	1m

```

$ eval $(minikube docker-env)

```

```

$ docker build -t cnj/product-api .
Sending build context to Docker daemon 58.55MB
Step 1/5 : FROM openjdk:8-jdk-alpine
--> 224765a6bdbe
Step 2/5 : RUN mkdir -p /app
--> Using cache
--> b77b32c3d77b
Step 3/5 : ADD target/product-0.0.1-SNAPSHOT.jar /app/app.jar
--> aa5d22ada069
Step 4/5 : EXPOSE 8080
--> Running in 497306e36f52
--> 147b0640c16b
Removing intermediate container 497306e36f52
Step 5/5 : ENTRYPOINT /usr/bin/java -jar /app/app.jar --spring.profiles.active=docker
--> Running in 19d66666d16c
--> 5b294eb92199
Removing intermediate container 19d66666d16c
Successfully built 5b294eb92199
Successfully tagged cnj/product-api:latest
$

```

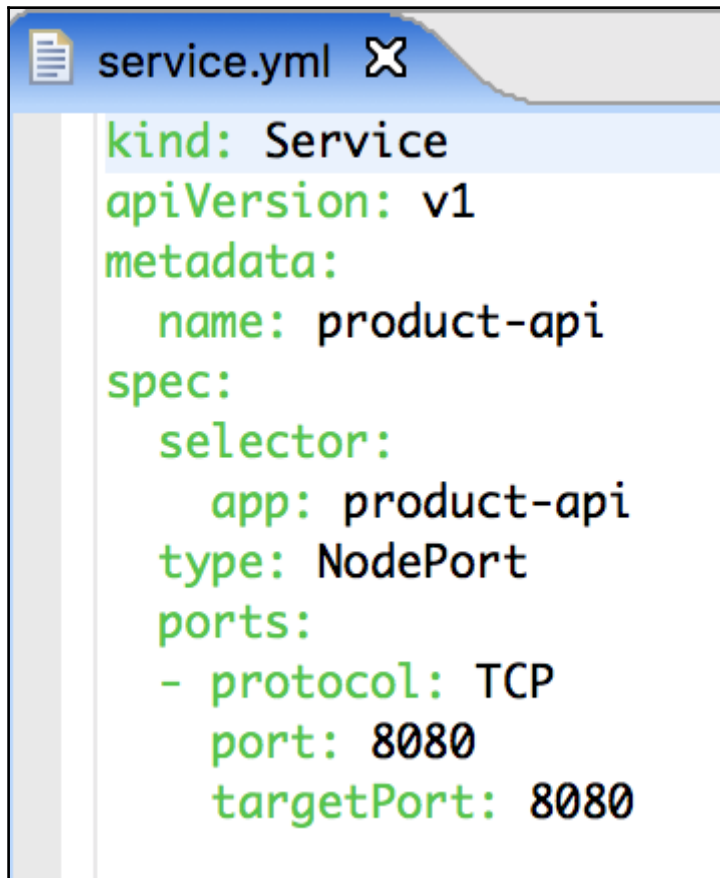


deployment.yml ✕

```
apiVersion: apps/v1beta2
kind: Deployment
metadata:
  name: product-api
  labels:
    app: product-api
spec:
  replicas: 3
  selector:
    matchLabels:
      app: product-api
  template:
    metadata:
      labels:
        app: product-api
    spec:
      containers:
      - name: product-api
        image: cnj/product-api:latest
        imagePullPolicy: Never
        ports:
        - containerPort: 8080
```

```
$ kubectl get deployment
NAME          DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
product-api   3         3         3             3           15m
```

```
$ kubectl get pod
NAME                                READY   STATUS    RESTARTS   AGE
product-api-7dcbbbf9b6-h2bjs       1/1    Running   0          13m
product-api-7dcbbbf9b6-mttcb       1/1    Running   0          13m
product-api-7dcbbbf9b6-qdbvq       1/1    Running   0          13m
```



```
service.yml X
kind: Service
apiVersion: v1
metadata:
  name: product-api
spec:
  selector:
    app: product-api
  type: NodePort
  ports:
  - protocol: TCP
    port: 8080
    targetPort: 8080
```

```

$ kubectl create -f service.yml
service "product-api" created
$ kubectl get service
NAME      TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
kubernetes ClusterIP      10.96.0.1       <none>           443/TCP         2h
product-api NodePort       10.108.166.132 <none>           8080:31709/TCP  5s
$

```

```

$ minikube service --url product-api
http://192.168.99.100:31709
$

```

```

$ curl http://192.168.99.100:31709/product/1
{"id":1,"name":"Apples","catId":1}
$ curl http://192.168.99.100:31709/products?id=1
[{"id":1,"name":"Apples","catId":1},{"id":2,"name":"Oranges","catId":1},{"id":3,"name":"Bananas","catId":1}]
$ curl -XDELETE http://192.168.99.100:31709/product/1
{"id":1,"name":"Apples","catId":1}
$ curl -v http://192.168.99.100:31709/product/1
* Trying 192.168.99.100...
* TCP_NODELAY set
* Connected to 192.168.99.100 (192.168.99.100) port 31709 (#0)
> GET /product/1 HTTP/1.1
> Host: 192.168.99.100:31709
> User-Agent: curl/7.54.0
> Accept: */*
>
< HTTP/1.1 200
< X-Application-Context: product:docker
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Date: Fri, 19 Jan 2018 21:00:45 GMT
* Connection #0 to host 192.168.99.100 left intact
{"id":1,"name":"Apples","catId":1}
$

```

```

$ curl -v http://192.168.99.100:31709/product/1
* Trying 192.168.99.100...
* TCP_NODELAY set
* Connected to 192.168.99.100 (192.168.99.100) port 31709 (#0)
> GET /product/1 HTTP/1.1
> Host: 192.168.99.100:31709
> User-Agent: curl/7.54.0
> Accept: */*
>
< HTTP/1.1 200
< X-Application-Context: product:docker
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Date: Fri, 19 Jan 2018 21:00:45 GMT
<
* Connection #0 to host 192.168.99.100 left intact
{"id":1,"name":"Apples","catId":1}
$

```

```

ProductService.java
package com.mycompany.product;

import java.util.List;

@RestController
public class ProductService {

    @Value("${version:v0.0.0}")
    String version;

    @Autowired
    ProductRepository prodRepo ;

    @RequestMapping(value="/product/{id}", method = RequestMethod.GET )
    ResponseEntity<Product> getProduct(@PathVariable("id") int id) {
        Product prod = prodRepo.findOne(id);
        if (prod == null)
            throw new BadRequestException(BadRequestException.ID_NOT_FOUND, "No product for id " + id) ;
        else {
            HttpHeaders responseHeaders = new HttpHeaders();
            responseHeaders.set("X-Application-Version", version);
            return new ResponseEntity<>(prod, responseHeaders, HttpStatus.OK);
        }
    }
}

```



```
$ docker build -t cnj/product-api:v1.0.1 .
Sending build context to Docker daemon 58.55MB
Step 1/5 : FROM openjdk:8-jdk-alpine
----> 224765a6bdbe
Step 2/5 : RUN mkdir -p /app
----> Using cache
----> b77b32c3d77b
Step 3/5 : ADD target/product-0.0.1-SNAPSHOT.jar /app/app.jar
----> 41625a85e526
Step 4/5 : EXPOSE 8080
----> Running in ab7936f82d38
----> 18e8b967a8f7
Removing intermediate container ab7936f82d38
Step 5/5 : ENTRYPOINT /usr/bin/java -jar /app/app.jar --spring.profiles.active=docker
----> Running in 5a84379a5912
----> 2a6c03bd5a15
Removing intermediate container 5a84379a5912
Successfully built 2a6c03bd5a15
Successfully tagged cnj/product-api:v1.0.1
$
```



deployment.yml ✕

```
apiVersion: apps/v1beta2
kind: Deployment
metadata:
  name: product-api
  labels:
    app: product-api
spec:
  replicas: 3
  selector:
    matchLabels:
      app: product-api
  template:
    metadata:
      labels:
        app: product-api
    spec:
      containers:
      - name: product-api
        image: cnj/product-api:v1.0.1
        imagePullPolicy: Never
        ports:
        - containerPort: 8080
```

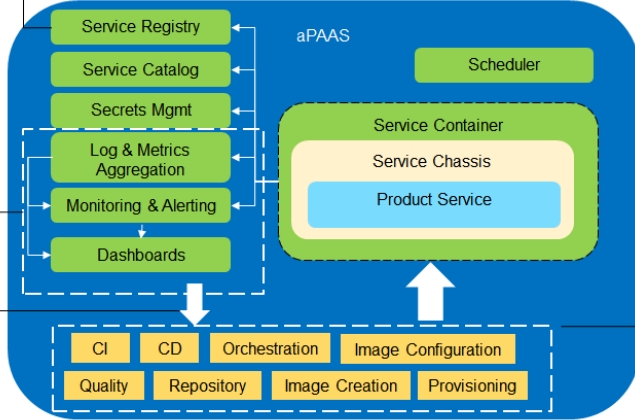
```

$ kubectl apply -f deployment.yml
deployment "product-api" configured
$ kubectl rollout status deployment/product-api
deployment "product-api" successfully rolled out
$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
product-api-55946666bc-2qddd        1/1     Running   0           14s
product-api-55946666bc-5bgrc        1/1     Running   0           11s
product-api-55946666bc-hd2fw        1/1     Running   0           8s
product-api-7dcbbbf9b6-6h7n4        0/1     Terminating   0           1m
product-api-7dcbbbf9b6-d79kk        0/1     Terminating   0           1m
product-api-7dcbbbf9b6-g7hm2        0/1     Terminating   0           1m
$ curl -v http://192.168.99.100:31709/product/1
* Trying 192.168.99.100...
* TCP_NODELAY set
* Connected to 192.168.99.100 (192.168.99.100) port 31709 (#0)
> GET /product/1 HTTP/1.1
> Host: 192.168.99.100:31709
> User-Agent: curl/7.54.0
> Accept: */*
>
< HTTP/1.1 200
< X-Application-Context: product:docker
< X-Application-Version: v1.0.1
< Content-Type: application/json; charset=UTF-8
< Transfer-Encoding: chunked
< Date: Fri, 19 Jan 2018 21:17:49 GMT
<
* Connection #0 to host 192.168.99.100 left intact
{"id":1,"name":"Apples","catId":1}
$

```

Resiliency is ensured by a robust monitoring and alerting platform.

Service Discovery via Registry is important for Microservices architecture



PAAS provides the necessary runtime capabilities to create application instances and run them. It also includes Enterprise capabilities such as monitoring, logging, discovery

PAAS also provides Dev Ops capability to ensure automation to start the containers, and terminate them increasing agility and reducing cost

Auto-scaling by enabling the feedback loop from Monitoring and Alerting to Service Creation



Application Code & Frameworks

Buildpacks / Spring Boot / Spring Cloud / Steeltoe



PAS

Pivotal Application Service



PKS

Pivotal Container Service



PFS

Pivotal Function Service



Pivotal Services Marketplace

Pivotal and Partner Products

Shared Services

Logging & Metrics / Services Brokers / API Management

Shared Security

Credhub / UAA / Single Sign On

Shared Networking

VMWare NSX

Concourse

CLOUDFOUNDRY

BOSH

Embedded Operating System (Windows / Linux)

vSphere

Openstack

AWS

Google Cloud

Azure & Azure Stack

```
Command Prompt
C:\pcfdev>cf dev resume
Resuming VM...
PCF Dev is now running.

C:\pcfdev>cf dev target
Successfully logged in to api.local.pcfdev.io as user.

C:\pcfdev>cf dev trust
***Warning: a self-signed certificate for *.local.pcfdev.io has been inserted into your OS certificate store. To remove this certificate, run: cf dev untrust***

C:\pcfdev>cf marketplace
Getting services from marketplace in org pcfdev-org / space pcfdev-space as user...
OK

service          plans          description
local-volume     free-local-disk Local service docs: https://github.com/cloudfoundry-
incubator/local-volume-release/
p-circuit-breaker-dashboard standard      Circuit Breaker Dashboard for Spring Cloud Applications
p-config-server  standard     Config Server for Spring Cloud Applications
p-mysql          512mb, 1gb   MySQL databases on demand
p-rabbitmq       standard     RabbitMQ is a robust and scalable high-performance m
ulti-protocol messaging broker.
p-redis          shared-vm    Redis service to provide a key-value store
p-service-registry standard     Service Registry for Spring Cloud Applications

TIP: Use 'cf marketplace -s SERVICE' to view descriptions of individual plans of a given service.

C:\pcfdev>
```

```
Select Command Prompt
D:\Apps\wkNeon\ch7-pcf-product>cf services
Getting services in org pcfdev-org / space pcfdev-space as user...
OK

name      service  plan  bound apps  last operation
prod-db   p-mysql  512mb pcf-product create succeeded

D:\Apps\wkNeon\ch7-pcf-product>
```

```
<artifactId>pcf-product</artifactId>
<version>0.1-SNAPSHOT</version>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-dependencies</artifactId>
      <version>Edgware.RELEASE</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>

<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-cloudfoundry-connector</artifactId>
  </dependency>
</dependencies>
```

```
@SpringBootApplication
public class ProductSpringApp {

    public static void main(String[] args) throws Exception {
        SpringApplication.run(ProductSpringApp.class, args);
    }

    @Bean
    public RestTemplate restTemplate() {
        return new RestTemplate();
    }
}
```

```
manifest.yml ☒
1 ---
2 applications:
3 - name: pcf-product
4   memory: 1G
5   random-route: true
6   path: target/pcf-product-0.1-SNAPSHOT.jar
7   services:
8     - prod-db
9
10
11
```



```
D:\Apps\wkNeon\ch7-pcf-product>cf push pcf-product
Using manifest file D:\Apps\wkNeon\ch7-pcf-product\manifest.yml

Creating app pcf-product in org pcfdev-org / space pcfdev-space as user...
OK

Creating route pcf-product-undedicated-spirkettling.local.pcfdev.io...
OK

Binding pcf-product-undedicated-spirkettling.local.pcfdev.io to pcf-product...
OK

Uploading pcf-product...
Uploading app files from: C:\Users\ajay\AppData\Local\Temp\unzipped-app422286615
Uploading 496.4K, 105 files
Done uploading
OK
Binding service prod-db to app pcf-product in org pcfdev-org / space pcfdev-space as user...
OK

Starting app pcf-product in org pcfdev-org / space pcfdev-space as user...
```

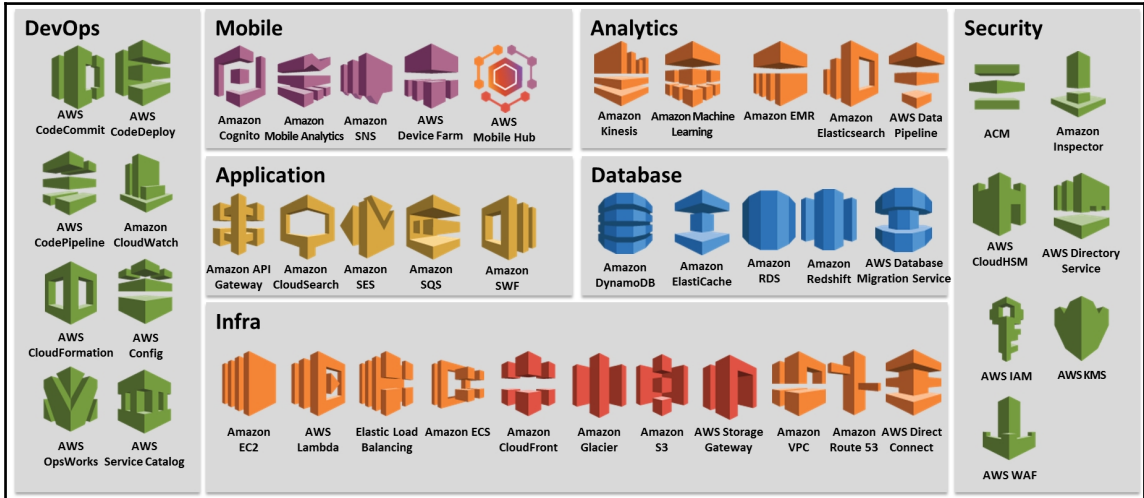
```
Staging complete
Uploading droplet, build artifacts cache...
Uploading build artifacts cache...
Uploading droplet...
Uploaded build artifacts cache (109B)
Uploaded droplet (71M)
Uploading complete
Destroying container
Successfully destroyed container

0 of 1 instances running, 1 starting
0 of 1 instances running, 1 starting
0 of 1 instances running, 1 starting
1 of 1 instances running

App started
```



Chapter 8: Platform Deployment – AWS



Elastic Beanstalk
Create New Application

Welcome to AWS Elastic Beanstalk

With Elastic Beanstalk, you can **deploy, monitor, and scale** an application quickly and easily. Let us do the heavy lifting so you can focus on your business.

To deploy your **existing web application**, create an [application source bundle](#) and then [create a new application](#). If you're using [Git](#) and would prefer to use it with our command line tool, please see [Getting Started with the EB CLI](#).

To deploy a **sample application**, click [Get started](#), choose a name, select a platform and click **Create app**.

By launching the sample application, you allow AWS Elastic Beanstalk to administer AWS resources and necessary permissions on your behalf. [Learn more](#)

Get started

Create a web app

Create a new application and environment with a sample application or your own code. By creating an environment, you allow AWS Elastic Beanstalk to manage AWS resources and permissions on your behalf. [Learn more](#)

Application information

Application name

Up to 100 Unicode characters, not including forward slash (/).

Base configuration

Platform


Choose **Configure more options** for more platform configuration options.

Application code Sample application

Get started right away with sample code.

Upload your code

Upload a source bundle from your computer or copy one from Amazon S3.

product-api-source 

[Cancel](#)

Environment properties

The following properties are passed in the application as environment properties. [Learn more](#)

Name	Value
<input type="text" value="GRADLE_HOME"/>	<input type="text" value="/usr/local/gradle"/> ✕
<input type="text" value="JAVA_HOME"/>	<input type="text" value="/usr/lib/jvm/java"/> ✕
<input type="text" value="M2"/>	<input type="text" value="/usr/local/apache-maven/bin"/> ✕
<input type="text" value="M2_HOME"/>	<input type="text" value="/usr/local/apache-maven"/> ✕
<input type="text" value="SERVER_PORT"/>	<input type="text" value="5000"/> ✕
<input type="text"/>	<input type="text"/>

Cancel

Save



Creating ProductApi-env

This will take a few minutes.

10:51am Created EIP: 50.18.168.90

10:51am Environment health has transitioned to Pending. Initialization in progress (running for 28 seconds). There are no instances.

10:50am Created security group named:
awseb-e-t9cr2tdmz9-stack-AWSEBSecurityGroup-1HERF4OTQZLQH

10:50am Using elasticbeanstalk-us-west-1-667811466675 as Amazon S3 storage bucket for environment data.

10:50am createEnvironment is starting.

Elastic Beanstalk product-api [Create New Application](#)


All Applications > product-api > [Actions](#)

ProductApi-env (Environment ID: e-t9ocr2tdmz9, URL: ProductApi-env.k7iyvudp7s.us-west-1.elasticbeanstalk.com)

Dashboard Overview [Refresh](#)

Configuration

Logs

Health  **Health** **Ok** [Causes](#)


Monitoring

Alarms

Managed

Updates

Running Version
product-api-source
[Upload and Deploy](#)


Configuration
64bit Amazon Linux 2017.09
v2.6.5 running Java 8
[Change](#)

productapi-env.k7iyvudp7s.us-west-1.elasticbeanstalk.com/product/1

[JSON](#) [Raw Data](#) [Headers](#)

[Save](#) [Copy](#)

```
id: 1
name: "Apples"
catId: 1
```

Roles > **aws-elasticbeanstalk-ec2-role**

Delete role

Summary

Role ARN arn:aws:iam::[redacted]:role/aws-elasticbeanstalk-ec2-role

Role description [Edit](#)

Instance Profile ARNs arn:aws:iam::[redacted]:instance-profile/aws-elasticbeanstalk-ec2-role

Path /

Creation time 2016-04-08 11:16 UTC+0530

Permissions





Trust relationships

Access Advisor

Revoke sessions

Attach policy

Attached policies: 4

	Policy name ▾	Policy type ▾	
▶	 AmazonEC2ContainerRegistryRead...	AWS managed policy	✕
▶	 AWSElasticBeanstalkWebTier	AWS managed policy	✕
▶	 AWSElasticBeanstalkMulticontainerD...	AWS managed policy	✕
▶	 AWSElasticBeanstalkWorkerTier	AWS managed policy	✕

Application information

Application name

Up to 100 Unicode characters, not including forward slash (/).

Base configuration

Platform

Choose **Configure more options** for more platform configuration options.

Application code

Sample application

Get started right away with sample code.

Upload your code

Upload a source bundle from your computer or copy one from Amazon S3.

 Upload

product-api-source 

Cancel

Configure more options

Create application

Elastic Beanstalk product-api Create New Application

All Applications > product-api > ProductApi-env-1 (Actions)

Environment ID: e-v5wx6pmhdp, URL: ProductApi-env-1.tbimub92qk.us-west-1.elasticbeanstalk.com

Dashboard Overview Refresh

Configuration

Logs

Health Ok Causes

Monitoring


Alarms

Managed Updates

Events

Running Version Upload and Deploy

product-api-source


Configuration

64bit Amazon Linux 2017.09
v2.8.4 running Docker 17.09.1-ce

Change

```

$ curl http://ProductApi-env-1.tbimub92qk.us-west-1.elasticbeanstalk.com/product/1 | jq .
{
  "id": 1,
  "name": "Apples",
  "catId": 1
}
$ curl http://ProductApi-env-1.tbimub92qk.us-west-1.elasticbeanstalk.com/products?id=1 | jq .
[
  {
    "id": 1,
    "name": "Apples",
    "catId": 1
  },
  {
    "id": 2,
    "name": "Oranges",
    "catId": 1
  },
  {
    "id": 3,
    "name": "Bananas",
    "catId": 1
  }
]

```

Deploying Spring Boot App to Elastic Container Service

AWS Elastic Container Service (ECS) is a service that allows a user to deploy

Getting Started with Amazon Elastic Container Service (ECS)

Select options to configure

Get started by running a sample app with Elastic Container Service (ECS), setting up a private image repository with Elastic Container Registry (ECR), or both.

- I want to Deploy a sample application onto an Amazon ECS Cluster
 Amazon ECS will set up an autoscaling group and help you create other resources to facilitate cluster management.
- Store container images securely with Amazon ECR
 Create and manage a new private image repository and use the Docker CLI to push and pull images. Access to the repository is managed through AWS Identity and Access Management.

Cancel **Continue**

Configure repository

The wizard guides you through creating a repository in Amazon ECR. [Learn more](#)

Repository name*



Namespaces are optional, and they can be included in the repository name with a slash (for example, namespace/repo)

Repository URI

████████████████████.dkr.ecr.ca-
central-
1.amazonaws.com/product-api

Permissions

As the owner, you have access to this repository by default. After completing this wizard, you can grant others permission to access this repository in the console.

*Required

[Cancel](#)

[Previous](#)

[Next step](#)

Build, tag, and push Docker image

Now that your repository exists, you can push a Docker image by following these steps:



Successfully created repository

667811466675.dkr.ecr.ca-central-1.amazonaws.com/product-api

To install the AWS CLI and Docker and for more information on the steps below, visit the [ECR documentation page](#).

1) Retrieve the `docker login` command that you can use to authenticate your Docker client to your registry:

Note:

If you receive an "Unknown options: --no-include-email" error, install the latest version of the AWS CLI.

[Learn more](#)

```
aws ecr get-login --no-include-email --region ca-central-1
```

2) Run the `docker login` command that was returned in the previous step.

Note:

If you are using Windows PowerShell, run the following command instead.

```
Invoke-Expression -Command (aws ecr get-login --no-include-email  
--region ca-central-1)
```

3) Build your Docker image using the following command. For information on building a Docker file from scratch see the instructions [here](#). You can skip this step if your image is already built:

```
docker build -t product-api .
```

4) After the build completes, tag your image so you can push the image to this repository:

```
docker tag product-api:latest 667811466675.dkr.ecr.ca-central-1.amazonaws.com/product-api:latest
```

5) Run the following command to push this image to your newly created AWS repository:

```
docker push 667811466675.dkr.ecr.ca-central-1.amazonaws.com/product-api:latest
```

Cancel

Next step

```
$ cmd=$(aws ecr get-login --no-include-email --region ca-central-1)
$ docker_cmd=$(echo $cmd | sed 's/https:\\/\\/'')
$ eval $docker_cmd
Login Succeeded
```

```
$ docker build -t product-api .
Sending build context to Docker daemon 29.14MB
Step 1/5 : FROM openjdk:8-jdk-alpine
--> 224765a6bdbe
Step 2/5 : RUN mkdir -p /app
--> Using cache
--> e06b3e1d15c2
Step 3/5 : ADD target/product.jar /app/app.jar
--> Using cache
--> 759113de40f8
Step 4/5 : EXPOSE 8080
--> Using cache
--> d57360fdfe1b
Step 5/5 : ENTRYPOINT [ "/usr/bin/java", "-jar", "/app/app.jar", "--spring.profiles.active=docker" ]
--> Using cache
--> a71d17d9fb63
Successfully built a71d17d9fb63
Successfully tagged product-api:latest
$ docker tag product-api:latest 667811466675.dkr.ecr.ca-central-1.amazonaws.com/product-api:latest
$ docker push 667811466675.dkr.ecr.ca-central-1.amazonaws.com/product-api:latest
The push refers to a repository [667811466675.dkr.ecr.ca-central-1.amazonaws.com/product-api]
b1139def75d5: Pushed
879a1a62c6c6: Pushed
685fdd7e6770: Pushed
c9b26f41504c: Pushed
cd7100a72410: Pushed
latest: digest: sha256:2eeba844bb82cd904e2b597029a8701a4a0ade7cc033ea3925e0afc45b01dc7f size: 1365
$
```

Create a task definition

An Amazon ECS task definition is a blueprint or recipe for containers. You can modify parameters in the task definition to suit your particular application (for example, to provide more CPU resources or change the port mappings). [Learn more](#)

Task definition name*



Container name*



Image*



Custom image format: [registry-url]/[namespace]/[image]:[tag]

Memory Limits (MiB)*

Hard limit ▼



Soft limit ▼



Define hard and/or soft memory limits in MiB for your container. Hard and soft limits correspond to the `memory` and `memoryReservation` parameters, respectively, in task definitions. ECS recommends 300-500 MiB as a starting point for web applications.

Port mappings

Host port

Container port

Protocol



tcp ▼



[+ Add port mapping](#)

STORAGE AND LOGGING

Read only root
file system



Mount points

Source volume

<none> ▾



Container path

Read only



[+ Add mount point](#)

Volumes from

Source container

Read only



[+ Add volumes](#)

Log configuration

Log driver

awslogs ▾



Log options

Key

Value

awslogs-group

product-api-log

awslogs-region

ca-central-1

awslogs-stream

product-api



Add key

Add value

The screenshot displays the AWS CloudWatch Logs console interface. On the left is a navigation sidebar with categories: CloudWatch, Dashboards, Alarms (with sub-items ALARM, INSUFFICIENT, and OK, each with a '0' badge), Billing, Events (with sub-items Rules and Event Buses), Logs (highlighted with an orange bar), Metrics, and Favorites (with a '+ Add a dashboard' link). The main content area is titled 'Welcome to CloudWatch Logs' and contains introductory text and a bulleted list of use cases. Below the text are two buttons: 'Quick Start Guide' and 'Create log group'. A modal dialog titled 'Create log group' is open in the foreground, featuring a close button (X) in the top right, a text input field for 'Log Group Name' containing 'product-api-logs', and two buttons at the bottom: 'Cancel' and 'Create log group'.

CloudWatch
Dashboards
Alarms
ALARM 0
INSUFFICIENT 0
OK 0
Billing
Events
Rules
Event Buses
Logs
Metrics
Favorites
[+ Add a dashboard](#)

Welcome to CloudWatch Logs

CloudWatch Logs helps you to aggregate, monitor, and store logs. For example, you can:

- Monitor HTTP response codes in Apache logs
- Receive alarms for errors in kernel logs
- Count exceptions in application logs

To start sending your logs to CloudWatch, click the Quick Start Guide and follow the instructions. To explore CloudWatch Logs before sending any data, click "Create Log Group" to create your first Log Group.

[Quick Start Guide](#) [Create log group](#)

Create log group

Log Group Name:

[Cancel](#) [Create log group](#)

Configure service

Create a name for your service and set the desired number of tasks to start with. A service auto-recovers any stopped tasks to maintain the desired number that you specify here. Later, you can update your service to deploy a new image or change the running number of tasks. [Learn more](#)

Service name*



**Desired number
of tasks***



Network access

Configure ports and load balancing for your service.

**Container name
: host port :
container port**



Load balancing

Use an Application Load Balancer

Create an Application Load Balancer and configure your service to run behind it. [Learn more](#)

*Required

Cancel

Previous

Next step

Configure cluster

Your Amazon ECS tasks run on container instances (Amazon EC2 instances that are running the ECS container agent). Configure the instance type, instance quantity, and other details of the container instances to launch into your cluster

Cluster name* ⓘ

EC2 instance type* ⓘ

Number of instances* ⓘ

Key pair ⓘ

You will not be able to SSH into your EC2 instances without a key pair. You can create a new key pair in the [EC2 console](#).

Security group

By default, your instances are accessible from any IP address. We recommend that you update the below security group ingress rule to allow access from known IP addresses only. ECS automatically opens up port 80 to facilitate access to the application or service you're running.

Allowed ingress source(s)* ⓘ

Container instance IAM role

The Amazon ECS container agent makes calls to the Amazon ECS API actions on your behalf, so container instances that run the agent require the `ecsInstanceRole` IAM policy and role for the service to know that the agent belongs to you. If you do not have the `ecsInstanceRole` already, we can create one for you.

Container instance IAM role ⓘ

*Required

[Cancel](#)

[Previous](#)

[Review & launch](#)

Amazon ECS status - 2 of 4 complete

Create cluster: product-api-cluster

✔ **Amazon ECS cluster created**
Amazon ECS cluster [product-api-cluster](#)

Create task definition: product-api-cluster product-api-task

✔ **Task definition created**
Task definition [product-api-task:3](#)

Create instances for:product-api-cluster

ℹ **Waiting for Amazon EC2 setup to complete...**
Amazon ECS instances Pending; see status below

Create service: sample-webapp

Service Pending

✔ **Subnet 2 association created**
Subnet 2 association [rtbassoc-f077e698](#)

✔ **Auto Scaling group created**
Auto Scaling group [EC2ContainerService-product-api-cluster-EcsInstanceAsg-1OH79L8G2C2AL](#)

✔ **Launch configuration created**
Launch configuration [EC2ContainerService-product-api-cluster-EcsInstanceLc-1MLT67KTW5BD3](#)

Filter:

1 to 1 of 1 Auto Scaling Groups

Name	Launch Configuration	Instances	Desired	Min	Max	Availability Zones
EC2Container...	EC2ContainerService-p...	1	1	0	1	ca-central-1b, ca-central-1a

Auto Scaling Group: EC2ContainerService-product-api-cluster-EcsInstanceAsg-1OH79L8G2C2AL

Details | Activity History | Scaling Policies | **Instances** | Monitoring | Notifications | Tags | Scheduled Actions

Actions

Filter: **Any Health Status** | **Any Lifecycle State** |

1 to 1 of 1 Instances

Instance ID	Lifecycle	Launch Configuration Name	Availability
<input type="checkbox"/> i-074c1b52483d1a69b	InService	EC2ContainerService-product-api-cluster-EcsInstanceLc-1MLT67KTW5BD3	ca-central-1

search : i-074c1b52483d1a69b

1 to 1 of 1

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
ECS Instanc...	i-074c1b52483d1a69b	t2.small	ca-central-1b	running	2/2 checks ...	None

Instance: i-074c1b52483d1a69b (ECS Instance - EC2ContainerService-product-api-cluster) **Public DNS:**

ec2-35-183-28-76.ca-central-1.compute.amazonaws.com

Description | Status Checks | Monitoring | Tags

Instance ID	i-074c1b52483d1a69b	Public DNS (IPv4)	ec2-35-183-28-76.ca-central-1.compute.amazonaws.com
Instance state	running	IPv4 Public IP	35.183.28.76
Instance type	t2.small	IPv6 IPs	-
Elastic IPs		Private DNS	ip-10-0-1-249.ca-central-1.compute.internal
Availability zone	ca-central-1b	Private IPs	10.0.1.249
Security groups	EC2ContainerService-product-api-cluster-	Secondary private IPs	

```

$ curl ec2-35-183-28-76.ca-central-1.compute.amazonaws.com:8080/product/1 | jq .
{
  "id": 1,
  "name": "Apples",
  "catId": 1
}
$ curl ec2-35-183-28-76.ca-central-1.compute.amazonaws.com:8080/products?id=1 | jq .
[
  {
    "id": 1,
    "name": "Apples",
    "catId": 1
  },
  {
    "id": 2,
    "name": "Oranges",
    "catId": 1
  },
  {
    "id": 3,
    "name": "Bananas",
    "catId": 1
  }
]

```

Select load balancer type

Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers (new), and Classic Load Balancers. Choose the load balancer type that meets your needs. [Learn more about which load balancer is right for you](#)

Application Load Balancer

Create

Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing, TLS termination and visibility features targeted at application architectures, including microservices and containers.

[Learn more >](#)

Network Load Balancer

Create

Choose a Network Load Balancer when you need ultra-high performance and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second while maintaining ultra-low latencies.

[Learn more >](#)

Classic Load Balancer

PREVIOUS GENERATION

for HTTP, HTTPS, and TCP

Create

Choose a Classic Load Balancer when you have an existing application running in the EC2-Classic network.

[Learn more >](#)

Step 1: Configure Load Balancer

Basic Configuration

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default configuration is an Internet-facing load balancer in the selected network with a listener that receives HTTP traffic on port 80.

Name ⓘ

Scheme ⓘ internet-facing
 internal

IP address type ⓘ ⌵

Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port
HTTP ⌵	8080 ✕
<input type="button" value="Add listener"/>	

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one subnet per Availability Zone. You must specify subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC ⓘ ⌵

Availability Zone	Subnet ID	Subnet IPv4 CIDR	Name
<input checked="" type="checkbox"/> ca-central-1a	subnet-d2977cba	10.0.0.0/24	
<input checked="" type="checkbox"/> ca-central-1b	subnet-6a797f11	10.0.1.0/24	

▸ Tags

[Cancel](#)

Step 3: Configure Security Groups

A security group is a set of firewall rules that control the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing one.

- Assign a security group:** Create a new security group
 Select an existing security group

Filter: VPC security groups

Security	Name	Description	Actions
<input type="checkbox"/>	sg-e1f9b789	default	default VPC security group Copy to new
<input checked="" type="checkbox"/>	sg-27f7b94f	EC2ContainerService-product-api-cluster-EcsSecurityGroup-5F0B8DVKCQO1	ECS Allowed Ports Copy to new

Step 5: Register Targets

Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and the target passes the initial health checks.

Registered targets

To deregister instances, select one or more registered instances and then click Remove.

Remove

Instance	Name	Port	State	Security groups	Zone
<input type="checkbox"/>	i-074c1b52483...	ECS Instanc...	8080	running	EC2ContainerService-product... ca-central-1b

Instances

To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered on the specified port, you must specify a different port.

Add to registered on port 8080

Search Instances

Instance	Name	State	Security	Zone	Subnet ID	Subnet CIDR
<input checked="" type="checkbox"/>	i-074c1b5...	ECS Insta...	running	EC2Cont...	ca-central...	subnet-6a797f11 10.0.1.0/24

Name	DNS name	State
product-api-lb	product-api-lb-146100789.ca...	provisioning

Load balancer: **product-api-lb**

Description Listeners Monitoring Tags

Basic Configuration

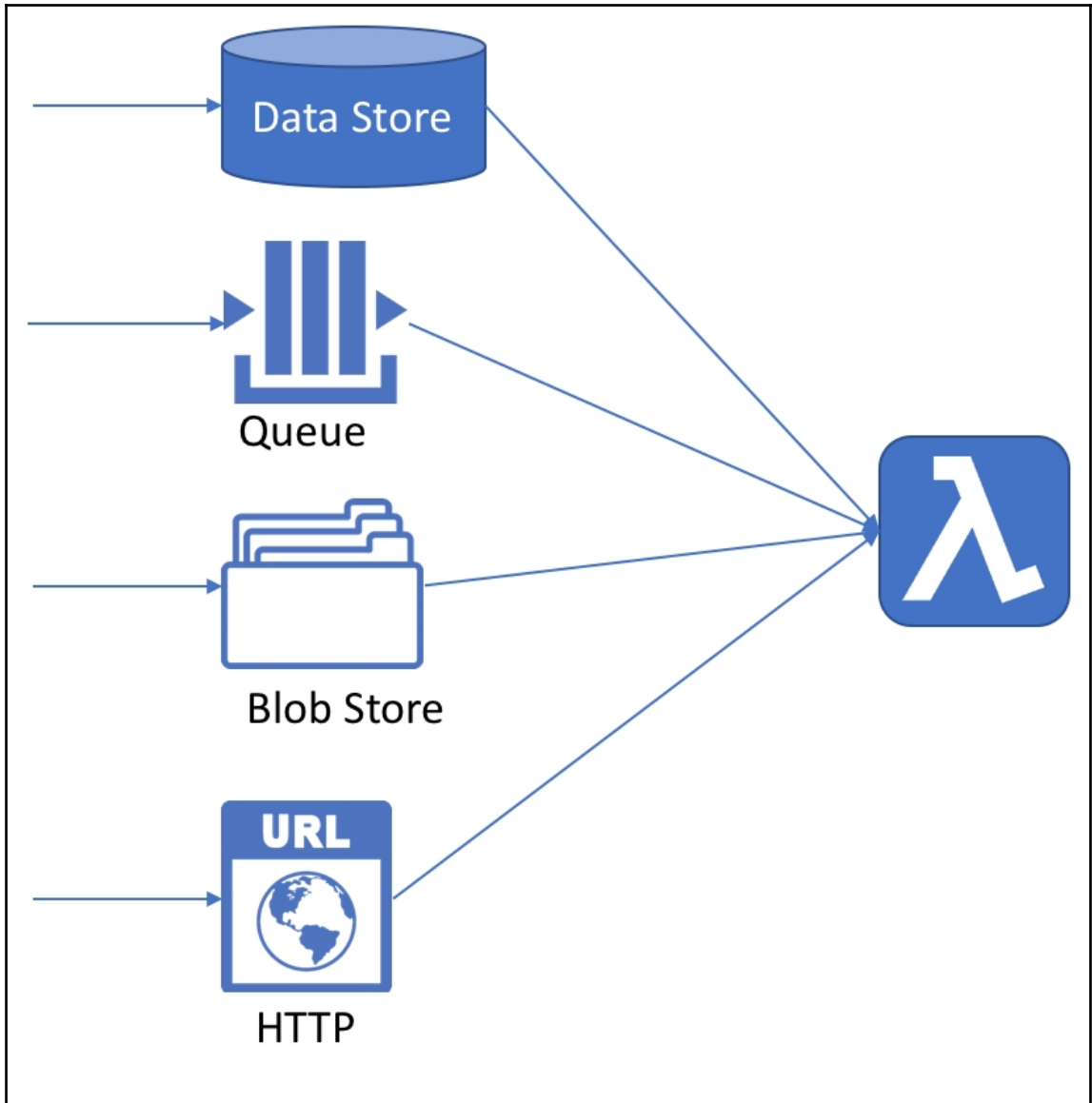
Name:	product-api-lb ↻	Creation time:	February 16, 2018 at 12:20:09 PM UTC+5:30
ARN:	arn:aws:elasticloadbalancing:ca-central-1:██████████:loadbalancer/app/product-api-lb/9fc45542a5020167	Hosted zone:	██████████
DNS name:	product-api-lb-146100789.ca-central-1.elb.amazonaws.com (A Record)	State:	provisioning
		VPC:	vpc-cf3dd9a7
Scheme:	internet-facing	IP address type:	ipv4
Type:	application		

```
$ curl http://product-api-lb-146100789.ca-central-1.elb.amazonaws.com:8080/product/1 | jq .
```

```
{
  "id": 1,
  "name": "Apples",
  "catId": 1
}
```

```
$ curl http://product-api-lb-146100789.ca-central-1.elb.amazonaws.com:8080/products?id=1 | jq .
```

```
[
  {
    "id": 1,
    "name": "Apples",
    "catId": 1
  },
  {
    "id": 2,
    "name": "Oranges",
    "catId": 1
  },
  {
    "id": 3,
    "name": "Bananas",
    "catId": 1
  }
]
```



Amazon API Gateway APIs > Create Show all hints ?

Create new API

In Amazon API Gateway, an API refers to a collection of resources and methods that can be invoked through HTTPS endpoints.

New API
 Import from Swagger
 Example API

Settings

Choose a friendly name and description for your API.

API name*
Description
Endpoint Type

* Required Create API

Amazon API Gateway APIs > hello-api (4uj2s1bue1) > Resources > / (vwrob0duuh) > Create Show all hints ?

New Child Resource

Use this page to create a new child resource for your resource.

[Configure as proxy resource](#) ⓘ

Resource Name*
Resource Path*

You can add path parameters using brackets. For example, the resource path **{username}** represents a path parameter called 'username'. Configuring **/{proxy+}** as a proxy resource catches all requests to its sub-resources. For example, it works for a GET request to /foo. To handle requests to /, add a new ANY method on the / resource.

Enable API Gateway CORS ⓘ

* Required Cancel Create Resource

Resources **Actions** **New Child Resource**

Use this page to create a new child resource for your resource. *

Configure as [proxy resource](#) ⓘ

Resource Name*

Resource Path*

You can add path parameters using brackets. For example, the resource path **{username}** represents a path parameter called 'username'. Configuring `/hello/{proxy+}` as a proxy resource catches all requests to its sub-resources. For example, it works for a GET request to `/hello/foo`. To handle requests to `/hello`, add a new ANY method on the `/hello` resource.

Enable API Gateway CORS ⓘ

* Required Cancel Create Resource

Resources **Actions** **/hello/{name} - GET - Setup**

Choose the integration point for your new method.

Integration type Lambda Function ⓘ

HTTP ⓘ

Mock ⓘ

AWS Service ⓘ

VPC Link ⓘ

Use Lambda Proxy integration ⓘ

Lambda Region

You do not have any **Lambda Functions** in **ca-central-1**. [Create a Lambda Function](#)

Use Default Timeout ⓘ

Save

Author from scratch [Info](#)

Name*

Runtime*

Role*

Defines the permissions of your function. Note that new roles may not be available for a few minutes after creation. [Learn more](#) about Lambda execution roles.

Lambda will automatically create a role with permissions from the selected policy templates. Note that basic Lambda permissions (logging to CloudWatch) will automatically be added. If your function accesses a VPC, the required permissions will also be added.

Role name*

Enter a name for your new role.

Policy templates

Choose one or more policy templates. A role will be generated for you before your function is created. [Learn more](#) about the permissions that each policy template will add to your role.

Cancel

Create function

Function code [Info](#)

Code entry type

Runtime

Handler [Info](#)

Function package*

hello-lambda.jar (2.2 MB)

com.mycompany.HelloHandler::handleRequest

For files larger than 10 MB, consider uploading via S3.

Resources **Actions** ● /hello/{name} - GET - Setup

Resources

- ▼ /
 - ▼ /hello
 - ▼ /{name}
 - GET**

Choose the integration point for your new method.

Integration type Lambda Function ⓘ

- HTTP ⓘ
- Mock ⓘ
- AWS Service ⓘ
- VPC Link ⓘ

Use Lambda Proxy integration ⓘ

Lambda Region us-west-1

Lambda Function myLambdaFunctionName ⓘ

Use Default Timeout hello

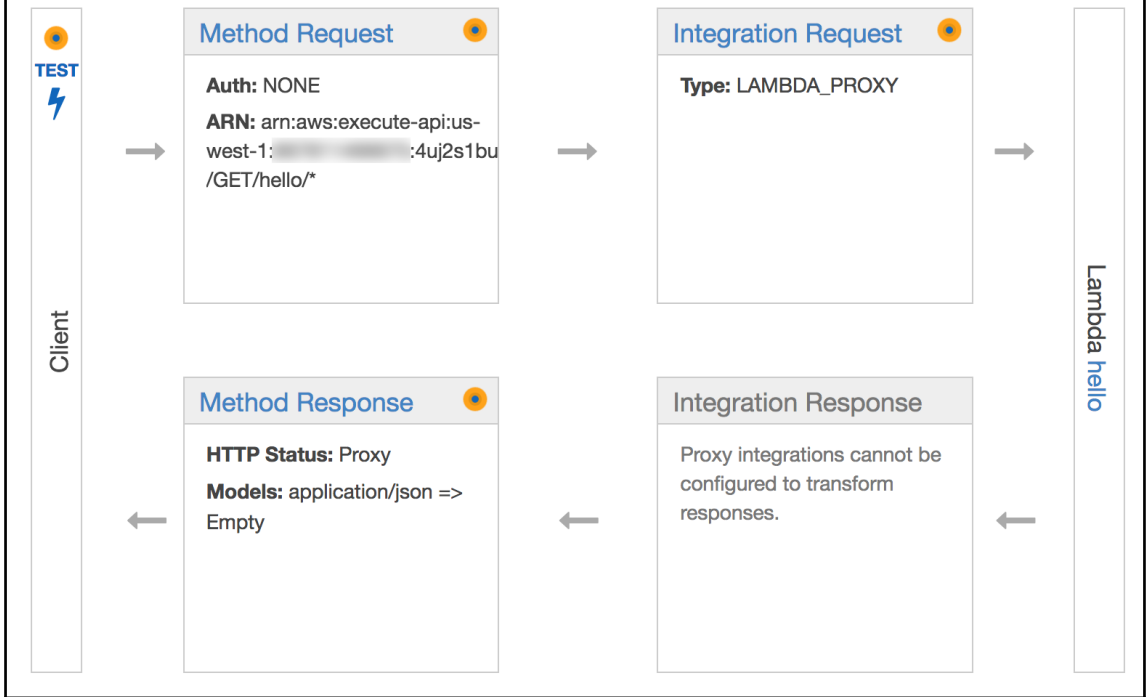
Add Permission to Lambda Function ×

You are about to give API Gateway permission to invoke your Lambda function:

arn:aws:lambda:us-west-1:██████████:function:hello

Cancel **OK**

/hello/{name} - GET - Method Execution



Method Execution /hello/{name} - GET - Method Test

Make a test call to your method with the provided input

Path

{name}

Universe

Query Strings

{name}

param1=value1¶m2=value2

Headers

Request: /hello/Universe

Status: 200

Latency: 118 ms

Response Body

Hello Universe!

Response Headers

```
{ "X-Amzn-Trace-Id": "sampled=0;root=1-5a86883a-61afb21166857ae3b6249e31" }
```

Deploy API ⦿ ✕

Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.

Deployment stage	<input type="text" value="[New Stage]"/>
Stage name*	<input type="text" value="prod"/>
Stage description	<input type="text" value="production"/>
Deployment description	<input type="text" value="production"/>

Cancel Deploy

prod Stage Editor Delete Stage

⦿ **Invoke URL:** https://4uj2s1bue1.execute-api.us-west-1.amazonaws.com/prod

Settings Logs Stage Variables SDK Generation Export Deployment History

Documentation History Canary

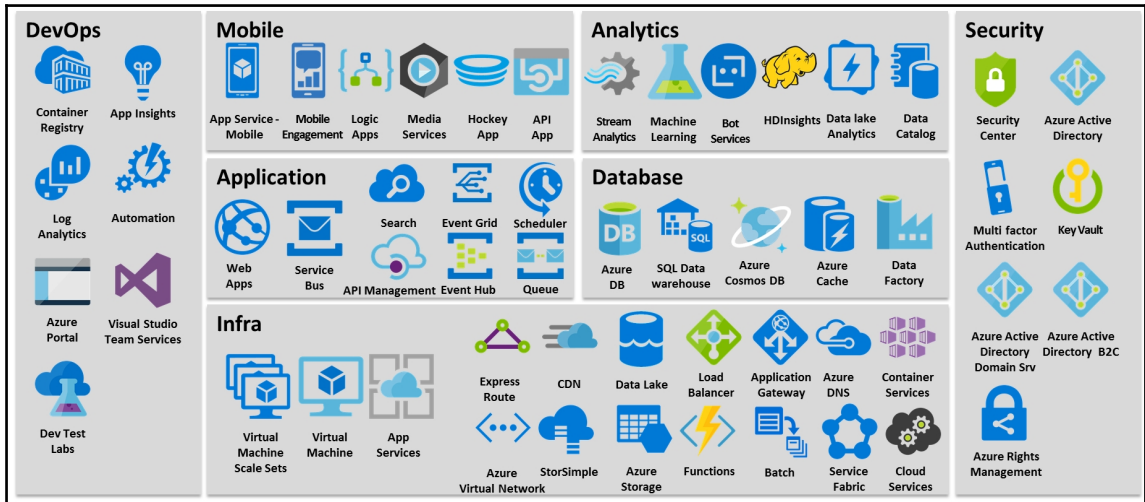
Configure the metering and caching settings for the **prod** stage.



Secure | <https://4uj2s1bue1.execute-api.us-west-1.amazonaws.com/prod/hello/Universe>

Hello Universe!

Chapter 9: Platform Deployment – Azure



```
Munish-Guptas-iMac:ch10-product admin$ ls -l
total 16
-rw-r--r--  1 admin  staff  3148 Jan 15 13:13 pom.xml
drwxr-xr-x  5 admin  staff   170 Jan 15 10:31 src
-rw-r--r--  1 admin  staff   530 Jan 15 13:48 web.config
Munish-Guptas-iMac:ch10-product admin$
```




+ New

- Dashboard
- All resources
- Resource groups
- App Services**
- Function Apps
- SQL databases
- Azure Cosmos DB
- Virtual machines
- Load balancers
- Storage accounts
- Virtual networks
- Azure Active Directory
- Monitor
- Advisor
- Security Center
- Cost Management + Billing
- Help + support
- More services >

All resources
ALL SUBSCRIPTIONS

[Refresh](#)

ServicePlan2acbd575-9980	App Service plan
--------------------------	------------------

Service Health

Marketplace

Azure getting started made easy!

Launch an app of your choice on Azure in a few quick steps

[Create DevOps Project](#)

Quickstart tutorials

- [Windows Virtual Machines](#)
Provision Windows Server, SQL Server, SharePoint VMs
- [Linux Virtual Machines](#)
Provision Ubuntu, Red Hat, CentOS, SUSE, CoreOS VMs
- [App Service](#)
Create Web Apps using .NET, Java, Node.js, Python, PHP
- [Functions](#)
Process events with a serverless code architecture
- [SQL Database](#)
Managed relational SQL Database as a Service

Microsoft Azure App Services

write2munish@hotmail.com (Default Directory)

+ Add Columns Refresh Assign Tags Start Restart Stop Delete

Subscriptions: Free Trial

Filter by name... All resource groups All locations No grouping

0 items

NAME	STATUS	APP TYPE	APP SER...	LOCATION	SUBSCRIP...
------	--------	----------	------------	----------	-------------

No App Services to display

Create, build, deploy, and manage powerful web, mobile, and API apps for employees or customers using a single back-end. Build standards-based web apps and APIs using .NET, Java, Node.js, PHP, and Python. [Learn more about App Service](#)

Create App Services

Dashboard
All resources
Resource groups
App Services
Function Apps
SQL databases
Azure Cosmos DB
Virtual machines
Load balancers
Storage accounts
Virtual networks
Azure Active Directory

Microsoft Azure

Search resources, services and docs

write2munish@hotmail.com

App Services Web + Mobile

Web + Mobile

Filter

Search Web + Mobile

0 items

No App Services to display

Create, build, deploy, and manage powerful web, mobile, and API apps for employees or customers using a single back-end. Build standards-based web apps and APIs using .NET, Java, Node.js, PHP, and Python. [Learn more about App Service](#)

Create App Services

Web Apps

- Web App (Microsoft)
- Web App + SQL (Microsoft)
- App Service Environment (Microsoft)
- WordPress on Linux (WordPress)
- Sitecore® Experience Cloud (Sitecore)
- Function App (Microsoft)

Mobile Apps

- Notification Hub (Microsoft)
- Mobile Engagement (Microsoft)
- Mobile App (Microsoft)

Microsoft Azure App Services > Web + Mobile > Web App

write2munish@hotmail...
WRITE2MUNISHHOTMAIL (DE...)

Web App

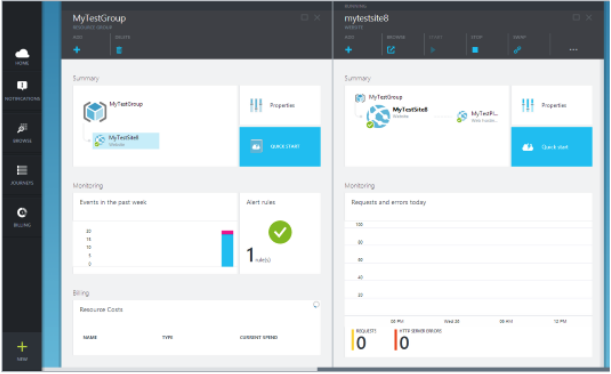
Microsoft

Create and deploy web sites in seconds, as powerful as you need them

Leverage your existing tools to create and deploy applications without the hassle of managing infrastructure. Microsoft Azure Web Sites offers secure and flexible development, deployment, and scaling options for any sized web application. Use frameworks and templates to create web sites in seconds. Choose from source control options like TFS, GitHub, and BitBucket. Use any tool or OS to develop your site with .NET, PHP, Node.js or Python.

- Fastest way to build for the cloud
- Provision and deploy fast
- Secure platform that scales automatically
- Great experience for Visual Studio developers
- Open and flexible for everyone
- Monitor, alert, and auto scale (preview)

[Twitter](#) [Facebook](#) [LinkedIn](#) [YouTube](#) [Google+](#) [Email](#)



Create



Web App

Create



+ New

Dashboard

All resources

Resource groups

App Services

Function Apps

SQL databases

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Azure Active Directory

Monitor

Advisor

Security Center

Cost Management + Billing

Help + support

More services >

* App name

ch10product
.azurewebsites.net

* Subscription

Free Trial

* Resource Group ⓘ

Create new Use existing

ch10product

* OS

Windows Linux

* App Service plan/Location

ServicePlan913234e0-8f54(Centr... >

Application Insights ⓘ

On Off

Pin to dashboard

Create Automation options

App Services
write2munishhotmail (Default Directory)

+ Add Columns Refresh Assign Tags Start Restart Stop Delete

Subscriptions: Free Trial

Filter by name... All resource groups All locations No grouping

1 items

NAME	STATUS	APP TYPE	APP SER...	LOCATION	SUBSCRIP...
ch10product	Running	Web app	ServicePlan91...	Central US	Free Trial

ch10product
App Service

Search (Ctrl+V)

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems

DEPLOYMENT
Quickstart
Deployment credentials
Deployment slots
Deployment options
Continuous Delivery (Preview)

SETTINGS
Application settings
Authentication / Authorization
Managed service identity
Backups
Custom domains
SSL certificates

Browse Stop Swap Restart Delete Get publish profile Reset publish profile

Click here to access our Quickstart guide for deploying code to your app →

Resource group (change)
ch10product

Status
Running

Location
Central US

Subscription (change)
Free Trial

Subscription ID
ee6e81da-c847-4957-9703-0a80636e50b5

URL
<https://ch10product.azurewebsites.net>

App Service plan/pricing tier
ServicePlan913234e0-8f54 (Standard: 1 Small)

FTP/deployment username
ch10product/write2munish

FTP hostname
<ftp://waws-prod-dm1-035.ftp.azurewebsites.windows.net>

FTPS hostname
ftps://waws-prod-dm1-035.ftp.azurewebsites.windows.net

OS name
Windows Server 2016

Http 5xx

0 HTTP SERVER ERRORS

Data In

DATA IN
8.94 kB

Data Out

3kB

Search (Ctrl+)

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

DEPLOYMENT

- Quickstart
- Deployment credentials
- Deployment slots
- Deployment options
- Continuous Delivery (Preview)

SETTINGS

- Application settings
- Authentication / Authorization
- Managed service identity

Save Discard

General settings

- .NET Framework version v4.7
- PHP version 5.6
- Java version Java 8
- Java Minor version Newest
- Web container Newest Tomcat 9.0

App Service supports installing newer versions of Python. Click here to learn more.

- Python version Off
- Platform 32-bit 64-bit
- Web sockets Off On
- Always On Off On
- Managed Pipeline Version Integrated Classic

ch10product - Deployment credentials
App Service

Search (Ctrl+/)

Save Discard

Deployment Credentials

Local Git and FTP can't authenticate using the credentials used for the portal. Create a new username and password to use with those deployment methods. This username and password will be the same across all apps in all subscriptions associated with your Microsoft Azure account. [Learn more](#)

* FTP/deployment username ⓘ

* Password ⓘ

* Confirm password

Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems

DEPLOYMENT

Quickstart
Deployment credentials
Deployment slots
Deployment options
Continuous Delivery (Preview)

SETTINGS

Dashboard - Microsoft Azure x https://ch10product.azureweb: x

Secure | https://ch10product.azurewebsites.net/product/4



```
{"id":4,"name":"Carrot","catId":2}
```

Dashboard + New dashboard ✎ Edit dashboard 🔗 Share 🔍 Fullscreen 📄 Clone 🗑️ Delete

New






- Dashboard
- All resources
- Resource groups
- App Services
- Function Apps
- SQL databases
- Azure Cosmos DB
- Virtual machines
- Load balancers
- Storage accounts
- Virtual networks
- Azure Active Directory
- Monitor
- Advisor
- Security Center
- Cost Management + Billing
- Help + support



All resources
ALL SUBSCRIPTIONS 🔄 Refresh

 ServicePlan825b0c07-b09e	App Service plan
 ServicePlanbd87cec8-85c8	App Service plan

Azure getting started made easy!
Launch an app of your choice on Azure in a few quick steps
[Create DevOps Project](#)

Quickstart tutorials

-  [Windows Virtual Machines](#) 🔗
Provision Windows Server, SQL Server, SharePoint VMs
-  [Linux Virtual Machines](#) 🔗
Provision Ubuntu, Red Hat, CentOS, SUSE, CoreOS VMs
-  [App Service](#) 🔗
Create Web Apps using .NET, Java, Node.js, Python, PHP
-  [Functions](#) 🔗
Process events with a serverless code architecture
-  [SQL Database](#) 🔗
Managed relational SQL Database as a Service

 Service Health  Marketplace

New ×

☰

+

Dashboard

All resources

Resource groups

App Services

Function Apps

SQL databases

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Azure Active Directory

Monitor

Advisor

Security Center

Cost Management + Billing

Help + support

More services >

Azure Marketplace See all Popular

Get started

Recently created

Compute

Networking

Storage

Web + Mobile

Containers

Databases

Data + Analytics

AI + Cognitive Services

Internet of Things

Enterprise Integration

Security + Identity

Developer tools

Monitoring + Management

Add-ons

Blockchain

Windows Server 2016 VM

[Quickstart tutorial](#)

Ubuntu Server 16.04 LTS VM

[Quickstart tutorial](#)

Web App

[Quickstart tutorial](#)

SQL Database

[Quickstart tutorial](#)

Cosmos DB

[Quickstart tutorial](#)

DevOps Project

[Quickstart tutorial](#)

Storage Account


[Quickstart tutorial](#)

Serverless Function App

[Quickstart tutorial](#)

[Show recently created items](#)

☰



Web App for Containers







Microsoft

✦ ✕

- + New
- Dashboard
- All resources
- Resource groups
- App Services
- ⚡ Function Apps
- SQL databases
- Azure Cosmos DB
- Virtual machines
- + Load balancers
- Storage accounts
- ↔ Virtual networks
- Azure Active Directory
- Monitor
- Advisor
- Security Center
- Cost Management + Billing
- Help + support
- More services >


Web App for Containers from Azure App Service allows customers to use their own containers and deploy them to App Service as a web app running on Linux. Similar to the Web App solution, Web App for Containers eliminates time-consuming infrastructure management tasks during container deployment, updating, and scaling to help developers focus on coding and getting their apps in front of more end users faster. Furthermore, it provides integrated CI/CD capabilities with DockerHub, Azure Container Registry, and VSTS, as well as built-in staging, rollback, testing-in-production, monitoring, and performance testing capabilities to boost developer productivity.

For Operations, Web App for Containers also provides rich configuration features so developers can easily add custom domains, integrate with AAD authentication, add SSL certificates and more—all crucial to web app development and management. It is an ideal environment to run web apps that do not require extensive infrastructure control.

MyTestGroup
MyTestSite8

Summary




Properties

Go to start

Monitoring

Events in the past week



Alert rules

1

✓


Billing

Resource Costs

NAME	TYPE	CURRENT SPEND

mytestsite8

Summary




Properties

Go to start

Monitoring

Requests and errors today



PUBLISHER

USEFUL LINKS

Microsoft

[Documentation](#)
[Solution Overview](#)
[Solutions you can deliver](#)
[Pricing Details](#)

Create

Web App for Containers
Create

* App name
ch10productContainer ✓
.azurewebsites.net

* Subscription
Free Trial

* Resource Group ⓘ
 Create new Use existing
ch10productContainer ✓

* App Service plan/Location
ServicePlan1a3e8304-8160(West...)

* Configure container

Pin to dashboard

Create Automation options

Docker Container

Docker Container

Web App for Containers lets you bring your own Docker formatted container images and easily deploy and run them at scale with Azure.

Image source
Azure Container Registry **Docker Hub** Private registry

Repository Access
 Public Private

* Image and optional tag (eg 'image:tag')
cloudnativejava/ch10productap[] ✓

OK

App Services
writ2zmandib@hotmail (Default Directory)

+ Add Columns Refresh Assign Tags Start Restart Stop Delete

Subscriptions: Free Trial

Filter by name... All resource groups All locations

1 items

NAME	STATUS	APP TYPE	APP SERVICE PLAN
ch10productContainer	Running	Web app	ServicePlan1a3e8304-8160

ch10productContainer
App Service

Search (Ctrl+F)

Overview

- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

DEPLOYMENT

- Quickstart
- Deployment credentials
- Deployment slots
- Deployment options
- Continuous Delivery (Preview)

SETTINGS

- Application settings

Resource group (change)
ch10productContainer

Status
Running

Location
West Europe

Subscription (change)
Free Trial

Subscription ID
eee6e81da-c847-4957-9703-0a80636e50b5

URL
<https://ch10productcontainer.azurewebsites.net>

App Service plan/pricing tier
ServicePlan133e8304-8160 (Standard: 1 Small)

FTP/deployment username
ch10productContainer\write2munish

FTP hostname
ftp://waws-prod-am2-167.ftp.azurewebsites.windows.net

FTPS hostname
ftps://waws-prod-am2-167.ftp.azurewebsites.windows.net

Http 5xx

Data In

Data Out

HTTP SERVER ERRORS 0

DATA IN 3.55 kB

DATA OUT 993 B

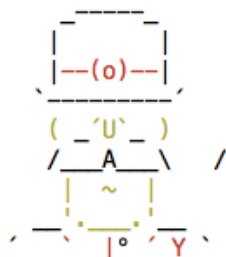
https://ch10productcontainer.e x

Secure | https://ch10productcontainer.azurewebsites.net/product/1

```
{ "id": 1, "name": "Apples", "catId": 1 }
```

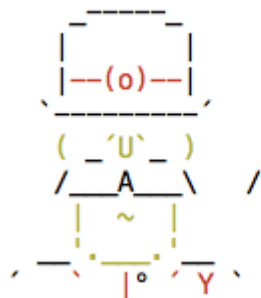
```
Munish-Guptas-iMac:ch10-product admin$ yo azuresfguest
```

```
Just found a `.yo-rc.json` in a parent directory.  
Setting the project root at: /Users/admin/Documents/workspace/CloudNativeJava
```



Welcome to Service
Fabric guest application
generator

```
? Name your application (ProductServiceFabric) █
```



Welcome to Service
Fabric guest application
generator

```
? Name your application ProductServiceFabric  
? Name of the application service: ProductAPI  
? Source folder of guest binary artifacts: complete/build/libs/  
? Relative path to guest binary in source folder: entryPoint.sh  
? Parameters to use when calling guest binary:  
? Number of instances of guest binary: 3
```

Microsoft Azure Service Fabric x

Secure | <https://try.servicefabric.azure.com>



Microsoft Azure Service Fabric Party Clusters

Try Service Fabric on Azure for free!

Download SDK >

Get started with a time-limited Service Fabric cluster in Azure free of charge, no subscriptions required.

Sign in to join a Party Cluster

 GitHub  Facebook

Please refer to our [Privacy & Cookies](#) policy and [Terms of Use](#) for details.

The image shows a web browser window with the following content:

- Browser title: Microsoft Azure Service Fabric x
- Address bar: Secure | https://try.servicefabric.azure.com
- Header: Microsoft Azure Service Fabric Party Clusters
- Sub-header: Try Service Fabric on Azure for free!
- Button: Download SDK >
- User greeting: Welcome, write2munish. Sign out
- Main message: Party clusters are available!
- Buttons: Join a Windows cluster, Join a Linux cluster (highlighted with a red border)

Microsoft Azure Service Fabric Party Clusters

Try Service Fabric on Azure for free!

Download SDK >

Welcome, write2munish.

[Sign out](#)

Party clusters are available!

Join a Windows cluster

Join a Linux cluster

Your secure Service Fabric cluster is ready!

ReadMe: [How to connect to a secure Party cluster?](#)

Certificate required to connect:

PFX

Service Fabric Explorer

<https://zlnxyngsvzoe.westus.cloudapp.azure.com:19080/Explorer/index.html>

Connection endpoint

zlnxyngsvzoe.westus.cloudapp.azure.com:19000

Expires on:

January 27 at 6:50:54 UTC

Time remaining

0 hours, 59 minutes, and 59 seconds

Available ports

80, 8081, 8080, 20000, 20001, 20002, 20003, 20004, 20005

Service Fabric Explorer

Not secure | <https://zlnxyngsvzoe.westus.cloudapp.azure.com:19080/Explorer/index.html#/>

Microsoft Azure Service Fabric Explorer REFRESH RATE 15s OFF FAST

Cluster <https://zlnxyngsvzoe.westus.cloudapp.azure.com>

ESSENTIALS DETAILS CLUSTER MAP METRICS MANIFEST

Cluster Health State: OK
 System Application Health State: OK
 Healthy Grid Nodes: 3 (100%)

Upgrade Domains: 3
 Fault Domains: 3

DASHBOARD

3 NODES	0 APPLICATIONS	0 SERVICES	0 REPLICAS
ERROR: 0 WARNING: 0 HEALTHY: 3	ERROR: 0 WARNING: 0 HEALTHY: 0	ERROR: 0 WARNING: 0 HEALTHY: 0	ERROR: 0 WARNING: 0 HEALTHY: 0
		0 PARTITIONS	0 UPGRADE IN PROGRESS
		ERROR: 0 WARNING: 0 HEALTHY: 0	

```

Munish-Gupta@-JMac:ProductServiceFabric admin$ ./install.sh
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[1/9] files, ApplicationManifest.xml
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[2/9] files, _dir
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[3/9] files, ProductAPIPkg/ServiceManifest.xml
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[4/9] files, ProductAPIPkg/_dir
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[5/9] files, ProductAPIPkg/code/entryPoint.sh
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[6/9] files, ProductAPIPkg/code/product-0.0.1-SNAPSHOT.jar
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[7/9] files, ProductAPIPkg/code/_dir
[8/9] files, ProductAPIPkg/config/Settings.xml
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
[9/9] files, ProductAPIPkg/config/_dir
Complete
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)
usr/local/lib/python3.5/site-packages/urllib3/connectionpool.py:858: InsecureRequestWarning: Unverified HTTPS request is being made. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#ssl-warnings
InsecureRequestWarning)

```

Service Fabric Explorer | zlnxyngsvzoe.westus.cloudapp.azure.com

Cluster: https://zlnxyngsvzoe.westus.cloudapp.azure.com

ESSENTIALS | DETAILS | CLUSTER MAP | METRICS | MANIFEST

Cluster Health State: OK
 Application Health State: OK
 Healthy Seed Nodes: 3 (100%)

Upgrade Domains: 3
 Fault Domains: 3

DASHBOARD

3 NODES	1 APPLICATION	1 SERVICE	1 REPLICA
ERROR: 0 WARNING: 0 HEALTHY: 3	ERROR: 0 WARNING: 0 HEALTHY: 1	ERROR: 0 WARNING: 0 HEALTHY: 1	ERROR: 0 WARNING: 0 HEALTHY: 1

UPGRADE IN PROGRESS: 0

UNHEALTHY EVALUATIONS

Kind	Health State	Description
[Empty table]		

Activate Windows
Go to Settings to activate Windows.

Service Fabric Explorer | zlnxyngsvzoe.westus.cloudapp.azure.com

URL: https://zlnxyngsvzoe.westus.cloudapp.azure.com:8080/product/2

```
["Id":2,"name":"Oranges","catId":1]
```

Activate Windows
Go to Settings to activate Windows.

Microsoft Azure Service Fabric Explorer

Cluster: <https://zlnxyngsvzoe.westus.cloudapp.azure.com>

Refresh Rate: 15s OFF FAST

Cluster Health Status: OK

System Application Health State: OK

Healthy Seed Nodes: 3 (100%)

Upgrade Domains: 3

Fault Domains: 3

ESSENTIALS | DETAILS | CLUSTER MAP | METRICS | MANIFEST

Cluster: ProductServiceFabricType

- fabric/ProductServiceFabric
 - fabric/ProductServiceFabric/ProductAPI
 - a432adaa-f667-424d-bb1f-4ac1a7e9ccd8
 - _Invm_2

Nodes: _Invm_0, _Invm_1, **_Invm_2** (selected)

Dashboard:

- 3 NODES: ERROR 0, WARNING 0, HEALTHY 3
- 1 APPLICATION: ERROR 0, WARNING 0, HEALTHY 1
- 1 SERVICE: ERROR 0, WARNING 0, HEALTHY 1
- 1 REPLICA: ERROR 0, WARNING 0, HEALTHY 1
- 1 PARTITION: ERROR 0, WARNING 0, HEALTHY 1
- UPGRADE IN PROGRESS: 0

Microsoft Azure Service Fabric Explorer

Cluster: <https://zlnxyngsvzoe.westus.cloudapp.azure.com>

Refresh Rate: 15s OFF FAST

Cluster Health Status: OK

System Application Health State: OK

Healthy Seed Nodes: 3 (100%)

Upgrade Domains: 3

Fault Domains: 3

ESSENTIALS | DETAILS | CLUSTER MAP | METRICS | MANIFEST

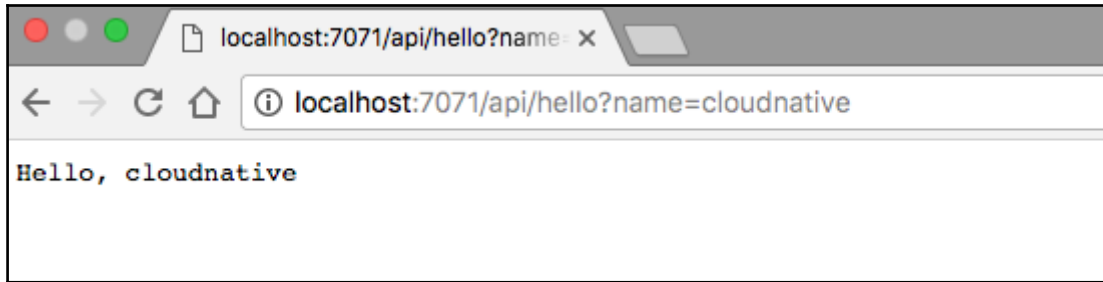
Cluster: ProductServiceFabricType

- fabric/ProductServiceFabric
 - fabric/ProductServiceFabric/ProductAPI
 - a432adaa-f667-424d-bb1f-4ac1a7e9ccd8
 - _Invm_2
 - fabric/ProductService
 - Activate
 - Deactivate (pause)
 - Deactivate (restart)
 - Deactivate (remove data)
 - Remove node state
 - Restart

Nodes: _Invm_0, _Invm_1, **_Invm_2** (selected)

Dashboard:

- 3 NODES: ERROR 0, WARNING 0, HEALTHY 3
- 1 APPLICATION: ERROR 0, WARNING 0, HEALTHY 1
- 1 SERVICE: ERROR 0, WARNING 0, HEALTHY 1
- 1 REPLICA: ERROR 0, WARNING 0, HEALTHY 1
- 1 PARTITION: ERROR 0, WARNING 0, HEALTHY 1
- UPGRADE IN PROGRESS: 0



```
package productAPI;

import java.util.*;

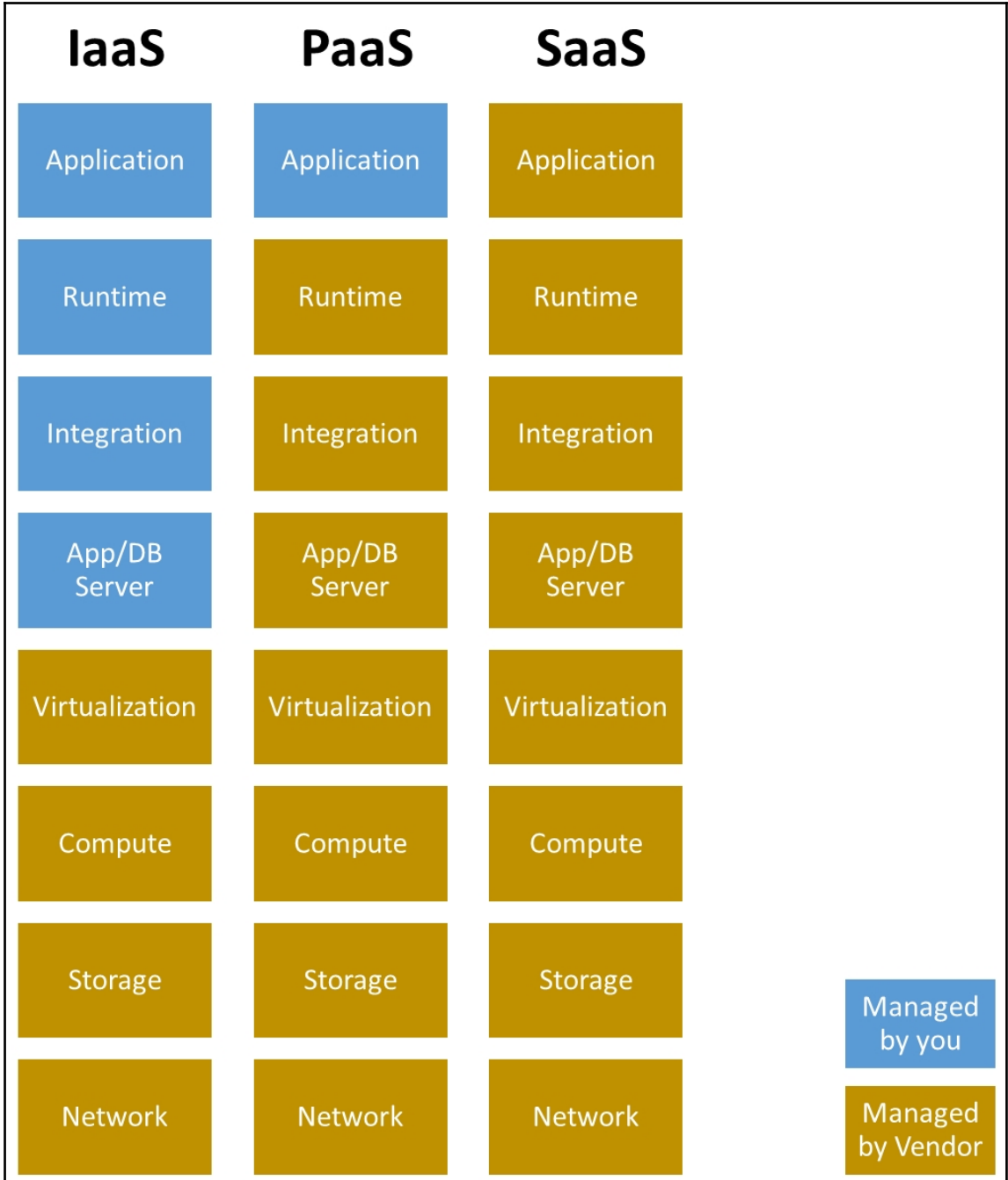
/**
 * Azure Functions with HTTP Trigger.
 */
public class Function {

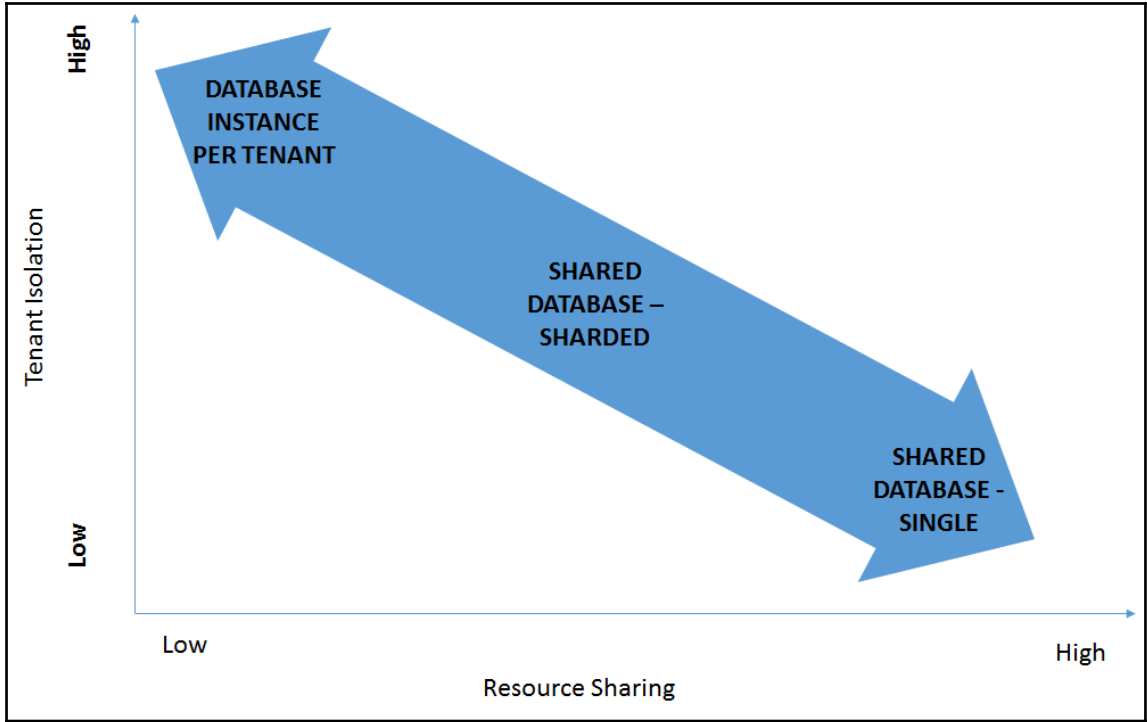
    @FunctionName("hello")
    public HttpResponseMessage<String> hello(
        @HttpTrigger(name = "req", methods = { "get",
            "post" }, authLevel = AuthorizationLevel.ANONYMOUS) HttpRequestMessage<Optional<String>> request,
        final ExecutionContext context) {
        context.getLogger().info("Java HTTP trigger processed a request.");

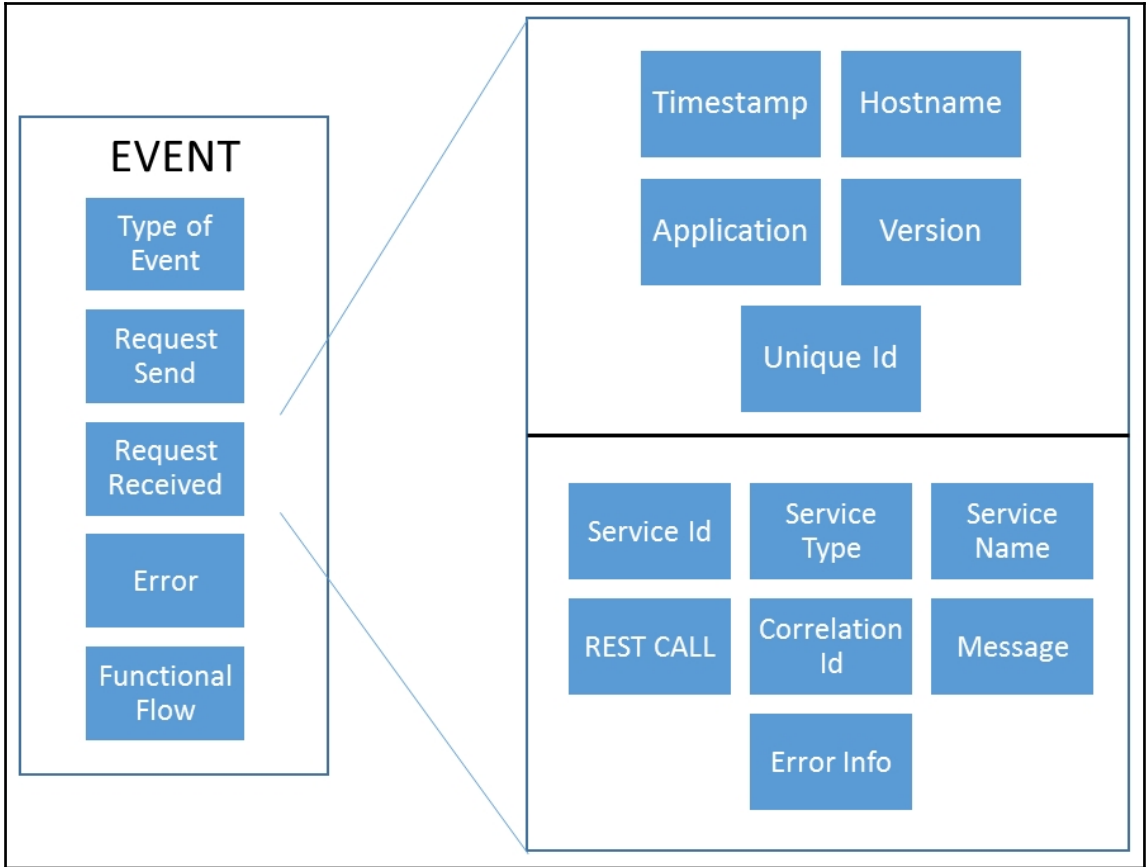
        // Parse query parameter
        String query = request.getQueryParameters().get("name");
        String name = request.getBody().orElse(query);

        if (name == null) {
            return request.createResponse(400, "Please pass a name on the query string or in the request body");
        } else {
            return request.createResponse(200, "Hello, " + name);
        }
    }
}
```

Chapter 10: As a Service Integration







Relational

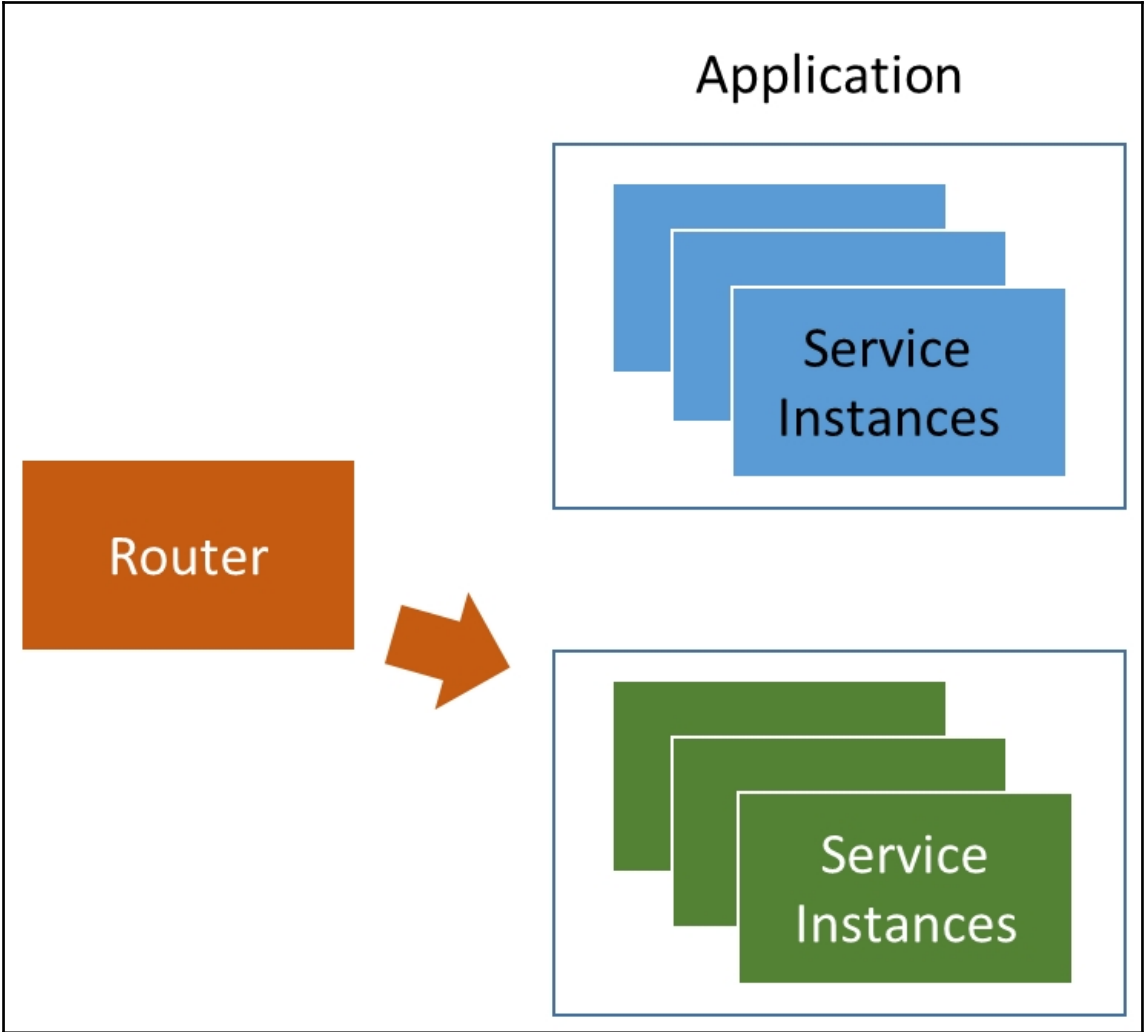
Columnar

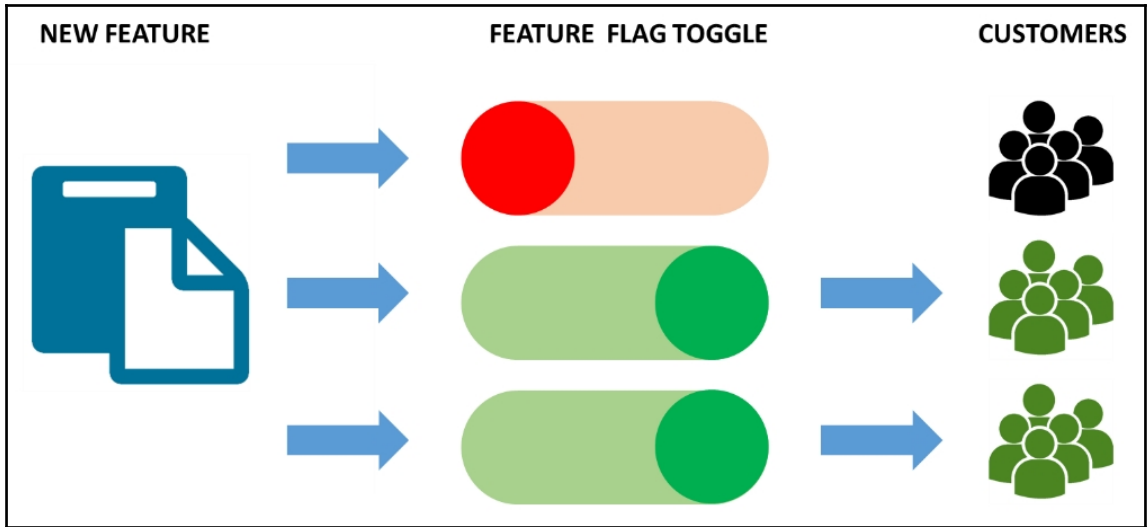
Key Value

Document

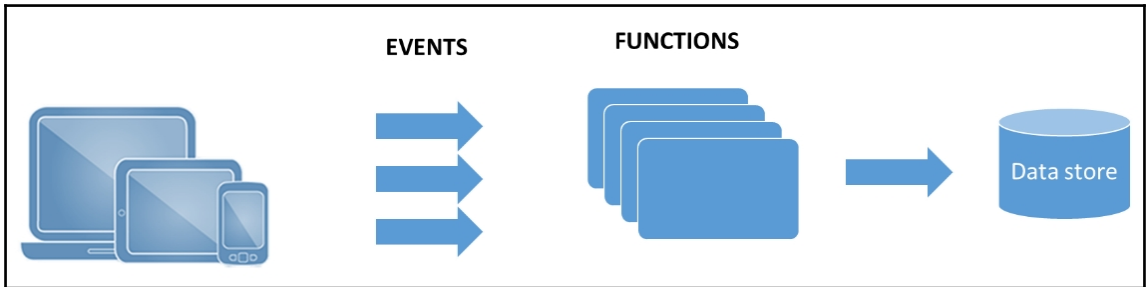
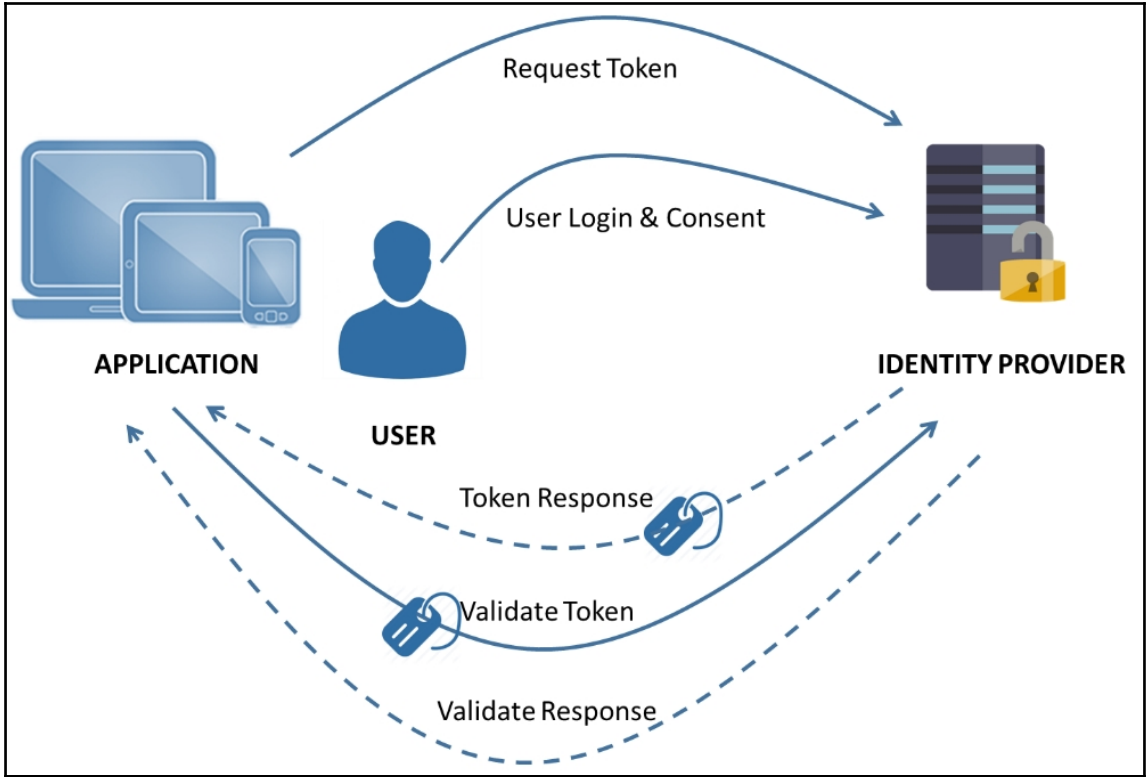
Graph

Blob

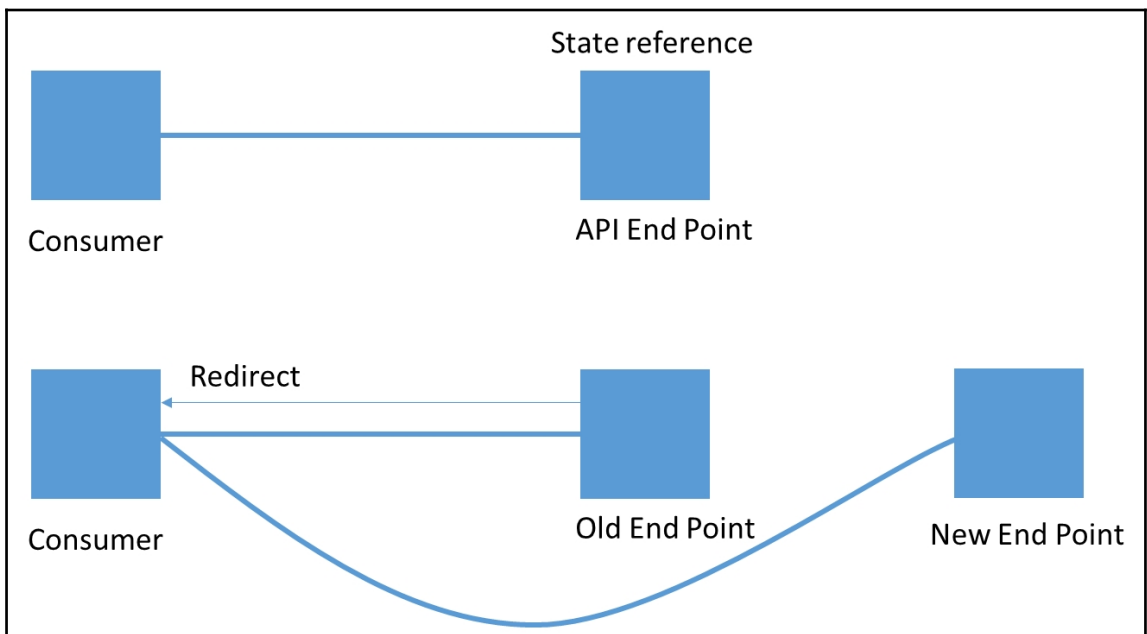
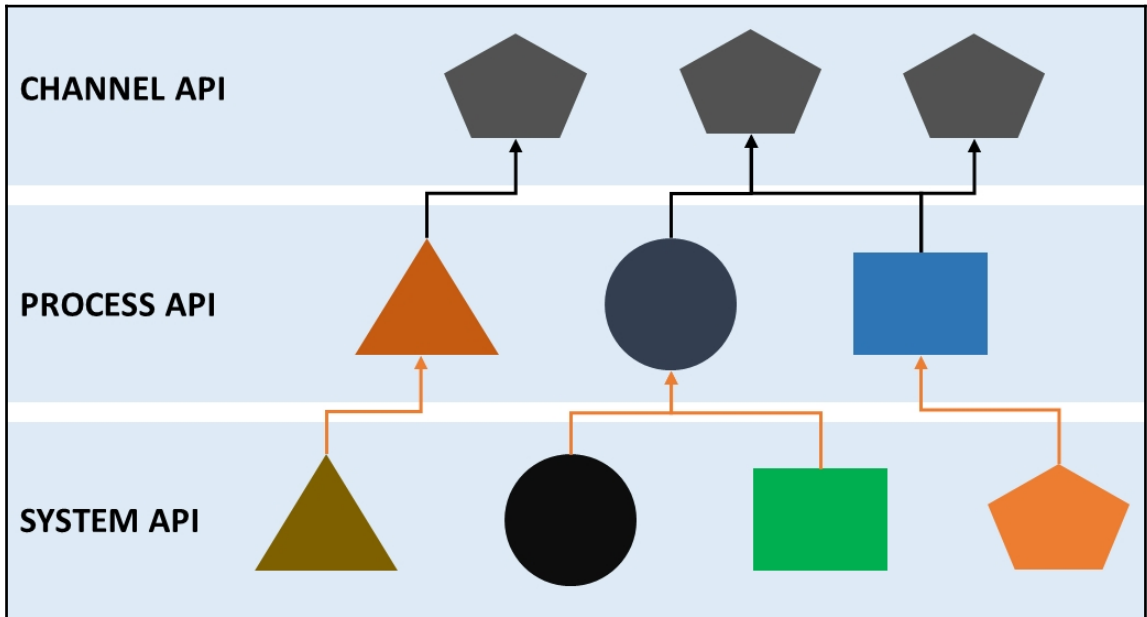




Multi tenancy	Faster Provisioning	Auditing	Security
Data Storage	Monitoring	Error Handling	Automated Build/Deployment
Customer Tiers	Feature Flags	Self Service Portal	SDKs
	Documentation & Community Support	Product roadmap	



Chapter 11: API Design Best Practices



For given API, describe the version of Swagger, API title, Terms of Service, Licensing Model, Hostname, basePath and tags to identify and discover the API

```
1  swagger: '2.0'
2  info:
3    description: This API is for building music playlist
4    version: 1.0.0
5    title: Music Playlist API
6    termsOfService: 'http://swagger.io/terms/'
7    contact:
8      email: rockstar_team@api.org
9    license:
10     name: Apache 2.0
11     url: 'http://www.apache.org/licenses/LICENSE-2.0.html'
12  host: example.api.org
13  basePath: /v1
14  tags:
15    - name: playlist
16      description: Everything about your Music Playlists
17    externalDocs:
18      description: Find out more
19      url: 'http://swagger.io'
20  schemes:
21    - http
```

Definition of resources and methods with details wrt method calls like

- tag to identify the method
- description for the methods
- Operation name the method maps to
- Out put types
- Parameters definition along with description and type (array, string, integer etc)
- Expected response types
 - 200 along with expected output
 - 400 along with expected output
- Security policy applicable to these methods

```
22 paths:
23   /playlist:
24     get:
25       tags:
26         - playlist
27       summary: get playlist by listening mood
28       description: Multiple mood values can be provided with comma separated strings
29       operationId: findPlaylistByMood
30       produces:
31         - application/xml
32         - application/json
33       parameters:
34         - name: mood
35           in: query
36           description: Mood values that need to be considered for filter
37           required: true
38           type: array
39           items:
40             type: string
41             enum:
42               - heartbroken
43               - sad
44               - ecstatic
45             default: available
46           collectionFormat: multi
47       responses:
48         '200':
49           description: successful operation
50           schema:
51             type: array
52             items:
53               $ref: '#/definitions/Playlist'
54         '400':
55           description: Invalid mood value
56       security:
57         - musicstore_auth:
58           - 'write:playlist'
59           - 'read:playlist'
```


Defines the security policy that need is applicable to the API along with the authorization model

Define the object structure that are applicable for post and get methods when using the API

```
60 securityDefinitions:
61   musicstore_auth:
62     type: oauth2
63     authorizationUrl: 'http://musicstore.swagger.io/oauth/dialog'
64     flow: implicit
65     scopes:
66       'write:playlist': modify playlist in your account
67       'read:playlist': read your playlist by mood
68   api_key:
69     type: apiKey
70     name: api_key
71     in: header
72 definitions:
73   Playlist:
74     type: object
75     properties:
76       id:
77         type: integer
78         format: int64
79       playlistId:
80         type: integer
81         format: int64
82       mood:
83         type: array
84         items:
85           type: string
86       xml:
87         name: Playlist
88 externalDocs:
89   description: Find out more about Swagger
90   url: 'http://swagger.io'
```

For given API, describe the version of RAML, API title, baseURI and version

Import the name spaced, reusable libraries

Add annotations for any vendor specific functionality e.g. monitoring

Traits allow code reuse and design patterns

Definition of resources and methods with applied traits and other patterns, details wrt to method calls

Describe expected response for different media types with data types. Schema and example can be defined here

```
1 #RAML 1.0
2 title: Music Playlist API
3 baseUri: http://example.api.org/{version}
4 version: v1
5
6 uses:
7   PlayLists: !include libraries/songs.raml
8
9 annotationTypes:
10  monitoringInterval:
11    parameters:
12      value: integer
13
14 traits:
15   secured: !include secured/accessToken.raml
16
17 /playlist:
18   is: secured
19   get:
20     (monitoringInterval): 30
21     queryParameters:
22       mood:
23         description: get playlist by listening mood
24   post:
25     /{playlistId}:
26       get:
27         responses:
28           200:
29             body:
30               application/json:
31                 type: PlayLists.Playlist
```

Design



Build



Test



Document



Integrate

Chapter 12: Digital Transformation

