

MIDDLEWARE TRACK

A cura del Middleware Solution Architect team di Red Hat

#redhatosd

Agenda Middleware Track part 1

- **14.00 MicroServices, le dimensioni non contano**
 - Ugo Landini, Samuele Dell'Angelo
- **14.30 Wildfly Swarm, Spring Boot & Vertx.io: il nuovo che avanza**
 - Ugo Landini, Samuele Dell'Angelo
- **15.00 Microservices con JBoss EAP 7: innovare in continuità**
 - Giuseppe Bonocore
- **15.30 Microregole per grandi progetti con il BRMS**
 - Andrea Leoncini

Agenda Middleware Track part 2

- **16.00 Integrazione e microservizi: come un cammello può passare dalla cruna di un ago**
 - Filippo Calà, Ugo Landini
- **16.30 API Management con 3Scale nell'era dei microservizi**
 - Luca Bigotta
- **17.00 SALA PLENARIA**
- **17.15 ESTRAZIONE FINALE PREMI**



RED HAT
OPEN SOURCE DAY
Europe, Middle East & Africa



#redhatosd





Integrazione e Microservizi: come un cammello può passare dalla cruna di un ago

Filippo Calà & Ugo Landini

#redhatosd

Che cosa NON è Apache Camel?



Non è un
Enterprise Service Bus

Che cos'è Apache Camel?

Camel è un potente
Framework Open Source di **Integrazione**
basato sugli
Enterprise Integration Pattern

Che cos'è Apache Camel?

JBoss Fuse

Open Source Lightweight Integration Platform



Che cos'è Apache Camel?

JBoss Fuse

Architettura modulare e capacità di integrazione



Che cos'è Apache Camel?

JBoss Fuse

JBoss Fuse include Apache Camel



Che cos'è Apache Camel?

JBoss Fuse

Facile da estendere



Che cos'è Apache Camel?

JBoss Fuse

Facile da gestire



Che cos'è Apache Camel?

JBoss Fuse

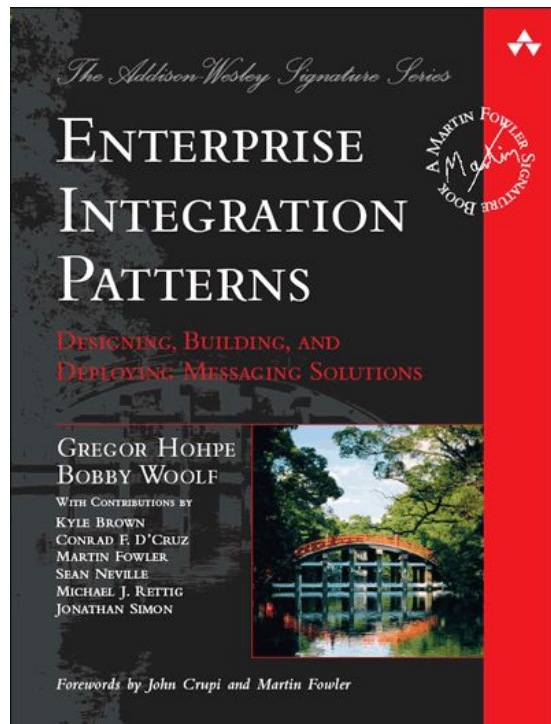
**Fa tutto quello che fanno gli ESB e molto di
più**



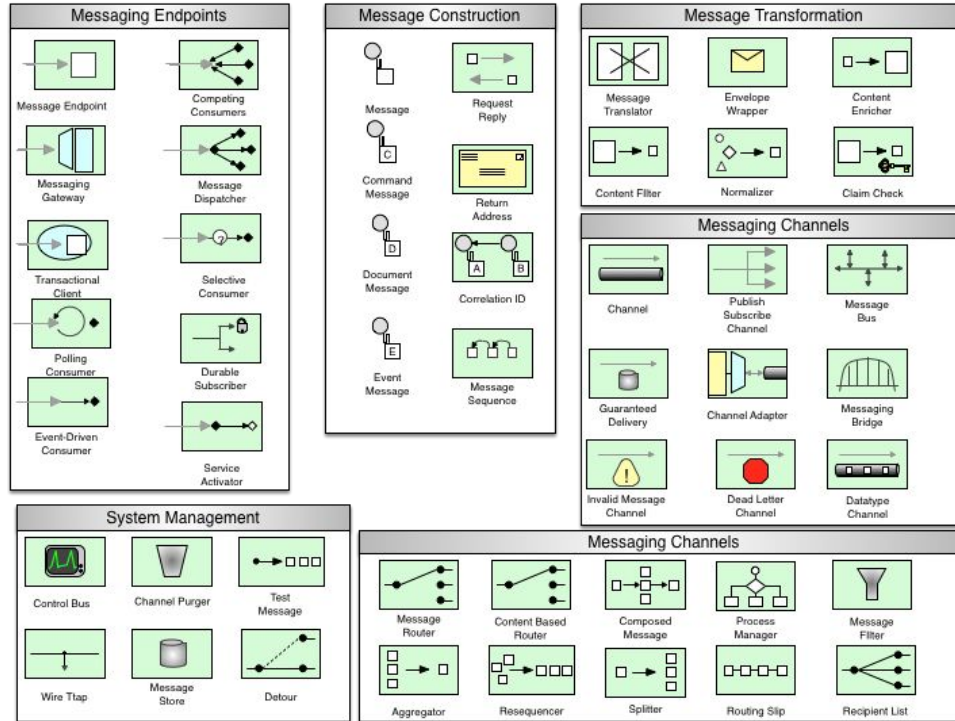
Integration Framework



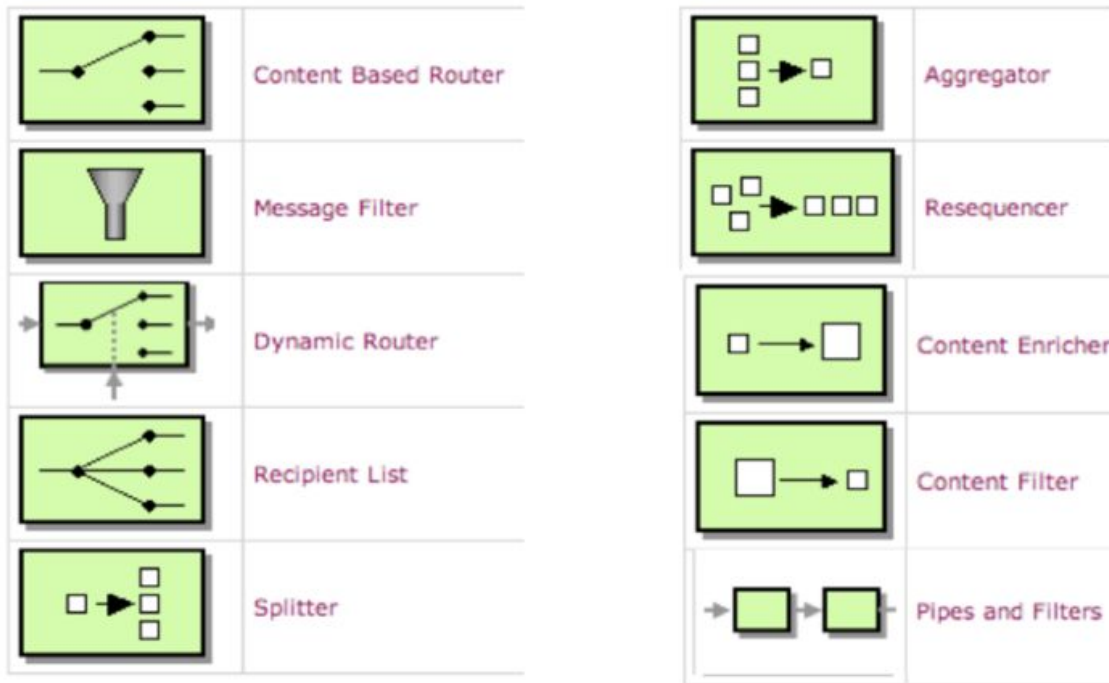
Enterprise Integration Patterns



Enterprise Integration Patterns

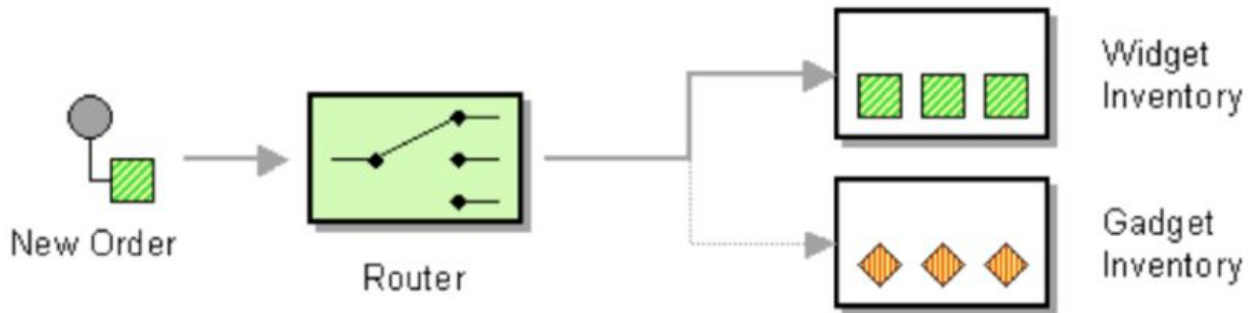


Enterprise Integration Patterns



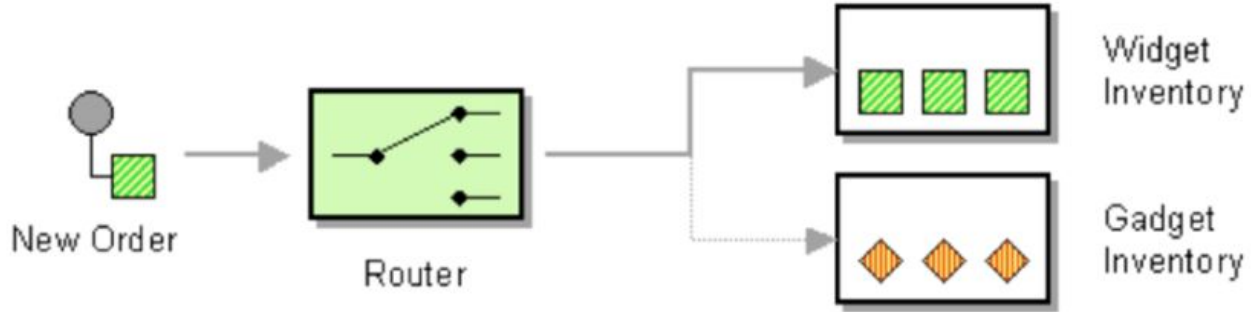
Enterprise Integration Patterns

Content Based Router



Enterprise Integration Patterns

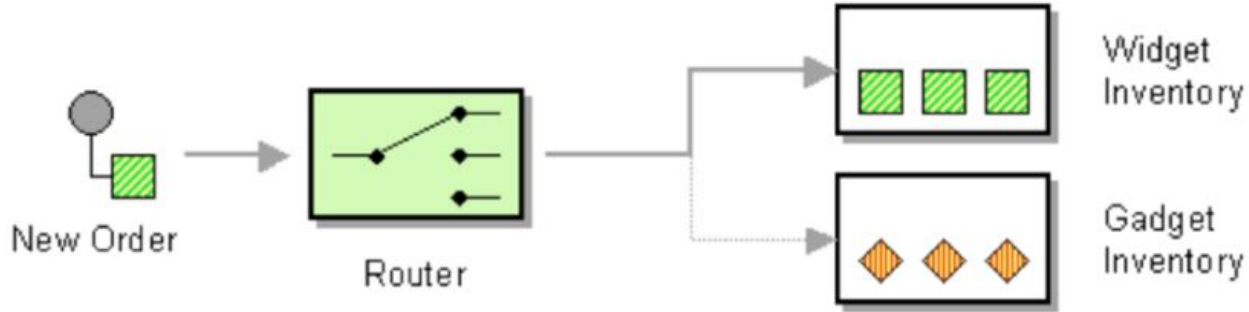
Content Based Router



from `newOrder`

Enterprise Integration Patterns

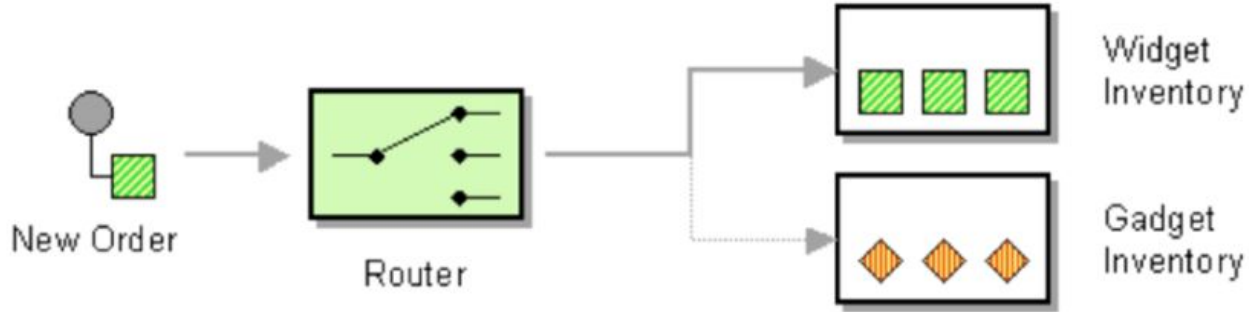
Content Based Router



from `newOrder`
choice

Enterprise Integration Patterns

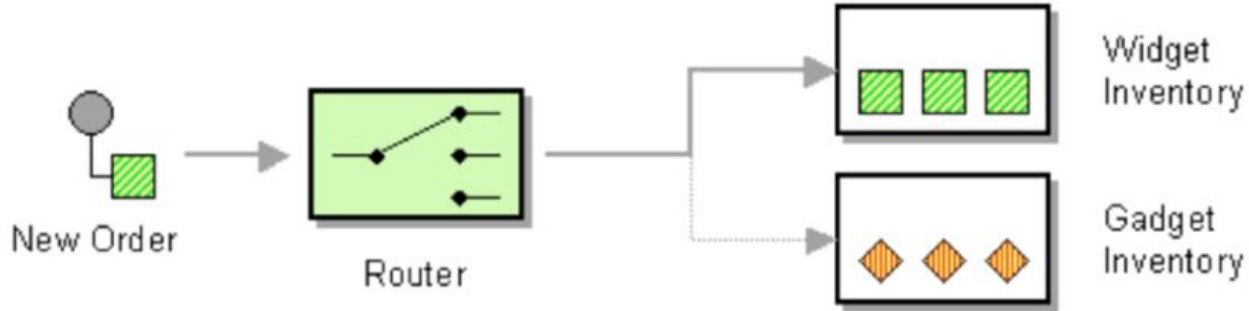
Content Based Router



from `newOrder`
choice
when `isWidget` to `widget`

Enterprise Integration Patterns

Content Based Router



from `newOrder`

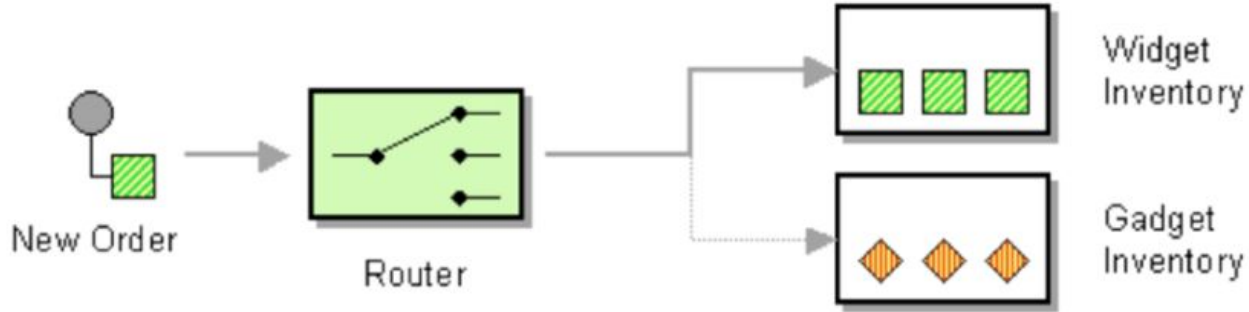
choice

when `isWidget` to `widget`

otherwise to `gadget`

Enterprise Integration Patterns

Content Based Router



from (`newOrder`)

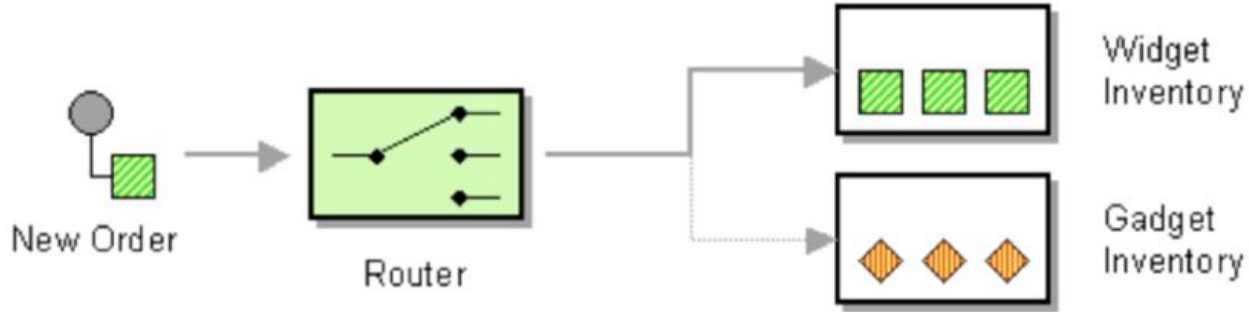
choice

when (`isWidget`) to (`widget`)

otherwise to (`gadget`)

Enterprise Integration Patterns

Content Based Router



```
from (newOrder)
    .choice()
    .when(isWidget) .to(widget)
    otherwise().to(gadget);
```

Enterprise Integration Patterns

Content Based Router

```
from (newOrder)
    .choice()
        .when(isWidget) .to(widget)
        otherwise().to(gadget);
```

Enterprise Integration Patterns

Content Based Router

```
Endpoint newOrder = endpoint("activemq:queue:newOrder");
```

```
from (newOrder)  
    .choice()  
        .when(isWidget) .to(widget)  
        otherwise().to(gadget);
```

Enterprise Integration Patterns

Content Based Router

```
Endpoint newOrder = endpoint("activemq:queue:newOrder");  
Predicate isWidget = xpath("/order/product = 'widget'");
```

```
from (newOrder)  
    .choice()  
        .when(isWidget) .to(widget)  
        otherwise().to(gadget);
```

Enterprise Integration Patterns

Content Based Router

```
Endpoint newOrder = endpoint("activemq:queue:newOrder");
Predicate isWidget = xpath("/order/product = 'widget'");
Endpoint widget = endpoint("activemq:queue:widget");
Endpoint gadget = endpoint("activemq:queue:gadget");

from (newOrder)
    .choice()
        .when(isWidget) .to(widget)
        otherwise().to(gadget);
```

Enterprise Integration Patterns

Java DSL

```
import org.apache.camel.builder.RouteBuilder;

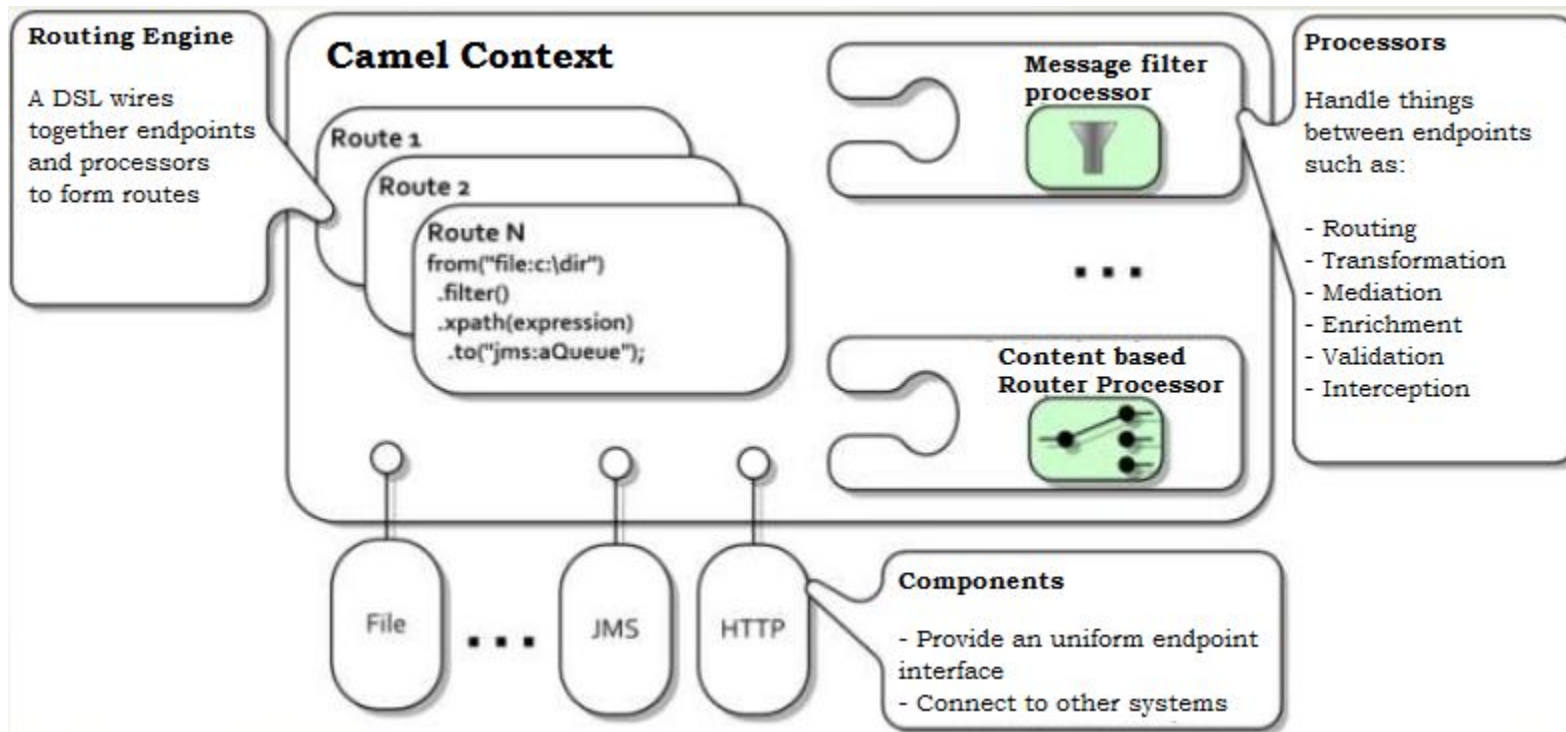
public class MyRoute extends RouteBuilder {
    public void configure() throws Exception {
        from("activemq:queue:newOrder")
            .choice()
                .when(xpath("/order/product = 'widget'"))
                    .to("activemq:queue:widget")
                .otherwise()
                    .to("activemq:queue:gadget");
    }
}
```

Enterprise Integration Patterns

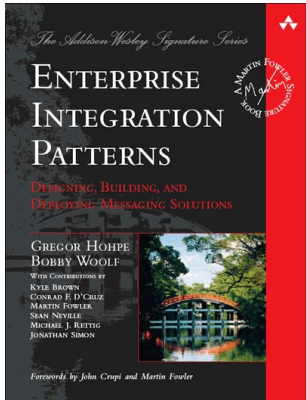
XML DSL

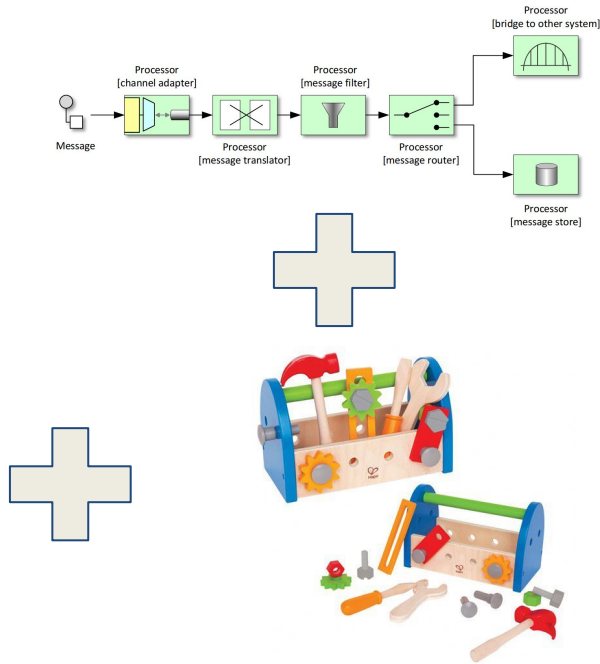
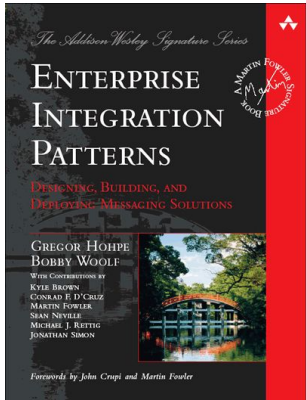
```
<route>
  <from uri="activemq:queue:newOrder"/>
  <choice>
    <when>
      <xpath>/order/product = 'widget'</xpath> <to uri="activemq:queue:widget"/>
    </when>
    <otherwise>
      <to uri="activemq:queue:gadget"/>
    </otherwise>
  </choice>
</route>
```

Architecture









Fuse **Brownfield**

I ferri del mestiere

- Runtime:

- Pure Java
- Karaf
- EAP

- Wiring:

- Blueprint
- CDI
- Spring

Fuse Greenfield

I ferri del mestiere

- Runtime:
 - Pure Java
 - Karaf
 - Spring Boot

- Wiring:
 - CDI
 - Spring

Fuse e Openshift

Camel Riding in the Cloud

- Fuse + Openshift = Fuse Integration Services
- Parte dell'offerta Red Hat xPaas
- Lavoro della community Fabric8

Fuse e Openshift

Modalità di deployment

- **Build then push:**

- Compilo usando il maven plugin di Fabric8
- Push del risultato (Container Image + Metadata) in OCP

- **Push then build:**

- Push del progetto su git
- Source 2 image in OCP

Fuse e Openshift

Un caso concreto

- **Soap2Rest:**
 - CXF, Freemarker
 - Pure Java, CDI
 - S2I

Fuse e Openshift

Camel
Context



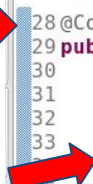
```
Soap2RestRoute.java soaprest/pom.xml
28 @ContextName("Soap2RestRoute")
29 public class Soap2RestRoute extends RouteBuilder {
30
31
32     @Inject
33     @Uri("log:output")
34     private Endpoint logEndpoint;
35
36     @Inject
37     @Uri("cxfrs://http://localhost:8080/?resourceClasses=it.redhat.osd.ShippingEndpoint")
38     private Endpoint restEndpoint;
39
40     @Inject
41     @Uri("http://{{SHIPPING_SOAP_ENDPOINT}}/ShippingService?bridgeEndpoint=true")
42     private Endpoint soapWsEndpoint;
43
44
45
46     @Override
47     public void configure() throws Exception {
48         from(restEndpoint)
49             .convertBodyTo(String.class)
50             .to("freemarker:it/redhat/osd/soapTemplate.ftl")
51             .removeHeaders("*")
52             .hystrix()
53                 .to(soapWsEndpoint)
54                 .setBody(xpath("//return",String.class))
55             .onFallback()
56                 .transform().constant("Shipping Info Unavailable")
57             .end()
58                 .marshal().json()
59                 .to(logEndpoint);
60     }
61
62 }
```

Fuse e Openshift

Camel
Context



Definizione
Endpoint



```
Soap2RestRoute.java soap2rest/pom.xml
28 @ContextName("Soap2RestRoute")
29 public class Soap2RestRoute extends RouteBuilder {
30
31
32     @Inject
33     @Uri("log:output")
34     private Endpoint logEndpoint;
35
36     @Inject
37     @Uri("cxfrs://http://localhost:8080/?resourceClasses=it.redhat.osd.ShippingEndpoint")
38     private Endpoint restEndpoint;
39
40     @Inject
41     @Uri("http://{{SHIPPING_SOAP_ENDPOINT}}/ShippingService?bridgeEndpoint=true")
42     private Endpoint soapWsEndpoint;
43
44
45
46     @Override
47     public void configure() throws Exception {
48         from(restEndpoint)
49             .convertBodyTo(String.class)
50             .to("freemarker:it/redhat/osd/soapTemplate.ftl")
51             .removeHeaders("*")
52             .hystrix()
53             .to(soapWsEndpoint)
54             .setBody(xpath("//return",String.class))
55             .onFallback()
56             .transform().constant("Shipping Info Unavailable")
57             .end()
58             .marshal().json()
59             .to(logEndpoint);
60     }
61
62 }
```

Fuse e Openshift

Camel
Context



```
28 @ContextName("Soap2RestRoute")  
29 public class Soap2RestRoute extends RouteBuilder {  
30
```

Definizione
Endpoint



```
32 @Inject  
33 @Uri("log:output")  
private Endpoint logEndpoint;
```



```
35  
36 @Inject  
37 @Uri("cxfrs://http://localhost:8080/?resourceClasses=it.redhat.osd.ShippingEndpoint")  
private Endpoint restEndpoint;
```



```
39  
40 @Inject  
41 @Uri("http://{{SHIPPING_SOAP_ENDPOINT}}/ShippingService?bridgeEndpoint=true")  
private Endpoint soapWsEndpoint;
```

Rotta in
Java DSL

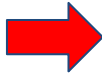


```
42  
43  
44  
45  
46 @Override  
47 public void configure() throws Exception {  
48     from(restEndpoint)  
49         .convertBodyTo(String.class)  
50         .to("freemarker:it/redhat/osd/soapTemplate.ftl")  
51         .removeHeaders("*")  
52         .hystrix()  
53         .to(soapWsEndpoint)  
54         .setBody(xpath("//return",String.class))  
55         .onFallback()  
56         .transform().constant("Shipping Info Unavailable")  
57         .end()  
58         .marshal().json()  
59         .to(logEndpoint);  
60 }  
61  
62 }
```

Fuse e Openshift

```
@Override
public void configure() throws Exception {
    from(restEndpoint)
        .convertBodyTo(String.class)
        .to("freemarker:it/redhat/osd/soapTemplate.ftl")
        .removeHeaders("*")
```

Camel
DSL



```
.to(soapWsEndpoint)
.setBody(xpath("//return",String.class))
```

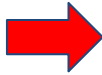
```
.marshal().json()
    .to(logEndpoint);
```

```
}
```

Fuse e Openshift

```
@Override
public void configure() throws Exception {
    from(restEndpoint)
        .convertBodyTo(String.class)
        .to("freemarker:it/redhat/osd/soapTemplate.ftl")
        .removeHeaders("*")
        .hystrix()
            .to(soapWsEndpoint)
            .setBody(xpath("//return",String.class))
        .onFallback()
            .transform().constant("Shipping Info Unavailable")
        .end()
        .marshal().json()
            .to(logEndpoint);
}
```

**Hystrix
Circuit
Breaker**



Fuse e Openshift

OPENSIFT CONTAINER PLATFORM

[osd-tests](#) > [Add to Project](#) > fis-java-openshift



Nome App



fis-java-openshift

Version 1.0

JBoss Fuse Integration Services 1.0 Java S2I images.

* Name

Identifies the resources created for this application.

* Git Repository URL

Git Reference

master

Optional branch, tag, or commit.

Context Dir

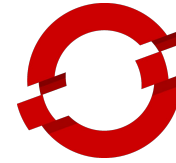
Optional subdirectory for the application source code, used as the context directory for the build.

Puntamenti
GIT

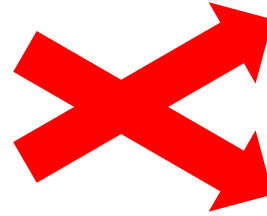


DEMO CAMEL + OSE

**App
tradizionali**



**RED HAT®
OPENS SHIFT**
Container Platform



Microservizi



Fisico



Virtuale



Cloud

Red Hat xPaas

Application Server

Pivotal tc Server, VMware vFabric, Oracle WebLogic Server, IBM WebSphere

In-memory data grid

IBM WebSphere eXtreme Scale, Oracle Coherence, Pivotal GemFire, Software AG Terracotta

Application integration

IBM Integration Bus, Oracle SOA Suite and ESB, MuleSoft AnyPoint Platform, Sonic ESB, Tibco ActiveMatrix

Business rules and process automation

IBM Operational Decision Manager and BPM, Oracle SOA Suite and BPM Suite, Pega BPM, FICO Blaze Advisor

Messaging

IBM WebSphereMQ, Pivotal RabbitMQ, VMware vFabric, Tibco Enterprise Message Service

Mobile

IBM MobileFirst, SAP Mobile Platform, Kony Mobility Platform, Telerik Platform, AnyPresence

Data virtualization and data services

IBM InfoSphere, Progress DataXtend SI, Composite Information Server, Denodo

JBoss EAP
JBoss Web Server

JBoss Data Grid

JBoss Fuse

JBoss BRMS
JBoss BPM Suite

JBoss A-MQ

Mobile Application Platform

JBoss Data Virtualization

Links

OSD Demo Applications

<https://github.com/redhat-italy/osd-microservices-apps>

Microservices Playground

bit.ly/msa-instructions

<http://developers.redhat.com/downloads>

Container Development Kit

Recorded Demo: <https://youtu.be/SPATMHP-xw8>