## Taller de Patrones de Arquitectura-ESB

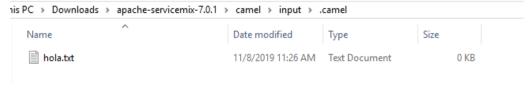
Instalando Apache Servicemix.

## Camel Example:

En la carpeta deploy de Apache Service Mix copiamos el siguiente archivo xml. Este va a enviar los archivos que se peguen en la carpeta camel/input a la carpeta camel/output.

```
<?xml version="1.0" encoding="UTF-8"?>
 <blueprint
    xmlns="http://www.osqi.org/xmlns/blueprint/v1.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="
       http://www.osqi.org/xmlns/blueprint/v1.0.0
Ξ
      http://www.osqi.org/xmlns/blueprint/v1.0.0/blueprint.xsd">
3
    <camelContext xmlns="http://camel.apache.org/schema/blueprint">
3
       <route>
         <from uri="file:camel/input"/>
         <log message="Moving ${file:name} to the output directory"/>
         <to uri="file:camel/output"/>
       </route>
     </camelContext>
-</blueprint>
```

Para probar lo que hicimos copiamos y pegamos el archivo hola.txt a camel/input. Después de hacerlo vemos cómo el archivo pegado en camel/input se copia automáticamente a camel/output.



Name	^	Date modified	Туре	Size
hola.txt		11/8/2019 11:27 AM	Text Document	0 KB

Vemos el log en Apache ServiceMix con log:display

📑 route xml 🗵 📙 activemgsender xml 🗵 📋 activemgreceiver xml 🗵

```
2019-11-08 11:23:15,740 | INFO | mix-7.0.1/deploy | BlueprintCamelContext | 43 - org.apache | camel |
```

## **ActiveMQ**

Apache ActiveMQ es un bróker de mensajería de código abierto, éste nos va a permitir enviar archivos y recibirlos, pero los almacena en una cola llamada events, luego el que recibe el mensaje los pasa de la cola events a la cola output.

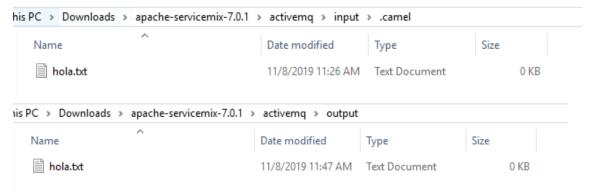
Enviando el mensaje.

```
<?xml version="1.0" encoding="UTF-8"?>
       <blueprint
  3
          xmlns="http://www.osqi.org/xmlns/blueprint/v1.0.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  5
          xsi:schemaLocation="
  6
            http://www.osqi.org/xmlns/blueprint/v1.0.0
  7
     http://www.osqi.org/xmlns/blueprint/v1.0.0/blueprint.xsd">
  8
          <camelContext xmlns="http://camel.apache.org/schema/blueprint">
 10
            <route>
 11
              <from uri="file:activemq/input"/>
 12
              <to uri="file:activemq/output"/>
 13
 14
              <setBody>
 15
                <simple>
 16
                 FileMovedEvent(file: ${file:name}, timestamp: ${date:now:hh:MM:ss.SSS})
 17
                </simple>
 18
              </setBodv>
 19
              <to uri="activemq://events" />
 20
            </route>
 21
           </camelContext>
      </blueprint>
 22
Recibiendo el mensaje.
<?xml version="1.0" encoding="UTF-8"?>
<blueprint
    xmlns="http://www.osqi.org/xmlns/blueprint/v1.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="
      http://www.osqi.org/xmlns/blueprint/v1.0.0
      http://www.osqi.org/xmlns/blueprint/v1.0.0/blueprint.xsd">
     <camelContext xmlns="http://camel.apache.org/schema/blueprint">
       <route>
         <from uri="activemq://events"/>
         <to uri="activemq://output"/>
```

Comprobamos una vez más que se pasen los archivos.

</route>
</camelContext>

</blueprint>



Después, corremos el cliente java que envía los mensajes. Se puede ver que ya el cliente está enviando los mensajes.

```
2019-11-08 11:55:01,092 | INFO | mix-7.0.1/deploy | BlueprintCamelContext | 43 - org.apache.camel.camel-core - 2.16.5 | Apache C amel 2.16.5 (CamelContext: camel-9) is starting | 43 - org.apache.camel.camel.core - 2.16.5 | JMX is e nabled | 2019-11-08 11:55:01,092 | INFO | mix-7.0.1/deploy | DefaultRuntimeEndpointRegistry | 43 - org.apache.camel.camel.core - 2.16.5 | JMX is e nabled | 2019-11-08 11:55:01,092 | INFO | mix-7.0.1/deploy | DefaultRuntimeEndpointRegistry | 43 - org.apache.camel.camel.core - 2.16.5 | Runtime endpoint registry is in extended mode gathering usage statistics of all incoming and outgoing endpoints (cache limit: 1800) | 2019-11-08 11:55:01,092 | INFO | mix-7.0.1/deploy | ActiveMQComponentResolver | 132 - org.apache.servicemix.activemq.camel - 7.0.1 | (creating an instance of the ActiveMQComponent (activemq:) | 2019-11-08 11:55:01,108 | INFO | mix-7.0.1/deploy | BlueprintCamelContext | 43 - org.apache.camel.camel.core - 2.16.5 | AllowUse OriginalMessage is enabled. If access to the original message is not needed, then its recommended to turn this option off as it may improve performance. | 43 - org.apache.camel.camel.core - 2.16.5 | StreamCa ching is not in use. If using streams then its recommended to enable stream caching. See more details at http://camel.apache.org/stream-caching.html | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started and consuming from: Endpoint[activemq://events] | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started and consuming from: Endpoint[activemq://events] | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 started | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 | 43 - org.apache.camel.camel.core - 2.16.5 | Route: route8 | 43 - org.apache.camel
```

Luego de correr el cliente java que envía los mensajes y el consumidor que los recibe, podemos ver que el mensaje fué enviado y recibido exitosamente.

Luego, desplegamos las carpetas que hicimos en una máquina virtual linux en AWS, y corremos Apache ServiceMix.

Corremos el HelloWorldSender y HelloWorldReceiver. Estos dos clientes java se conectan a la máquina linux desplegada en AWS, y se comunican a través del puerto 61617

```
public class App {
    public static void main(String[] args)
         HelloWorldSender sender = new HelloWorldSender();
         sender.run();
         HelloWorldConsumer consumer = new HelloWorldConsumer();
         consumer.run();
     1
public class HelloWorldSender {
   public void run() {
       CamelContext context = new DefaultCamelContext();
       //String brokerURL = args[0];
       ConnectionFactory connectionFactory = new ActiveMQConnectionFactory(
               "tcp://18.234.241.250:61616?jms.useAsyncSend=true");
       try {
               // Create a Connection
               Connection connection = connectionFactory.createConnection("smx", "smx");
               connection.start();
               // Create a Session
               Session session = connection.createSession(false, Session.AUTO ACKNOWLEDGE);
               // Create the destination (Topic or Queue)
               Destination destination = session.createQueue("events");
```

Vemos que el mensaje fué enviado y recibido exitosamente.

```
Building patronArqESB 1.0-SNAPSHOT

--- exec-maven-plugin:1.2.1:exec (default-cli) @ patronArqESB ---
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Sent message: 846238611 : main
Received: Hello world! From: main : 1740189450

BUILD SUCCESS

Total time: 2.555s
Finished at: Fri Nov 08 13:14:22 COT 2019
Final Memory: 8M/245M
```

