

RED HAT
SUMMIT

Deploying JBoss Fuse in Healthcare

Notes from the field

Quinn Stevenson - Solution Architect
Quinn@SpicoSolutions.com

Bryan Grant - Solution Architect
Bryan@SpicoSolutions.com

05/04/2017

Introductions



Quinn Stevenson

Quinn@SpicoSolutions.com

- Working in Clinical Integration for 15-yrs
User (5-yrs @ Intermountain Healthcare)
Vendor (2-yrs)
Consultant (8-yrs)
- Apache Contributor (Camel, ActiveMQ, Artemis)

Bryan Grant

Bryan@SpicoSolutions.com

- Working in Clinical Integration for 17-yrs
User (15-yrs @ DaVita)
Consultant (2-yrs)



- **The Lay of the Land**
- **Why use JBoss Fuse?**
- **Approach**
- **Strategies**
- **Case Study – Reduce Duplication**

The Lay of the Land



Quick Facts

- 24-hospitals
- ~450 MLLP Receivers
15-M messages/day
- ~2000 Translations
- ~800 MLLP Senders
30-M messages/day
- ~3500 Deployed components

Migration Path

- Re-use existing Java data structures
- 300 Translations migrated in 6-weeks (8 FTEs)
- Parallel for 3-months
- First Production deployment 09/2015
- Complete 05/2017

Quick Facts

- 3-hospitals
- ~120 MLLP Receivers
1.5-M messages/day
- ~200 Translations
- ~175 MLLP Senders
4-M messages/day
- ~800 Deployed components

Migration Path

- Convert Message Translators to HAPI
- Extended HAPI Terser
- Custom DSLs
- Production deployment
07/2017

Common Goals

The Lay of the Land

- **Portable Intellectual Property**
 - Stage for next migration
 - Resource availability (Java Developer)
- **Visible**
 - Source (debugging)
 - Configuration
 - Runtime Information
- **Auditable**
 - Configuration
 - Message History
- **Reproducible**
 - Environments (configuration)
 - Internal Standards
- **Maintainable**
 - Simple Configuration
 - Reuse
- **Separation of Duties**
 - HIPAA
 - Resource Utilization
 - Specialization

Common Characteristics

The Lay of the Land

- **Multiple “Applications”**

 - Receivers

 - Translations

 - Senders

 - Batch / Scheduled

 - Request-Response

- **Multiple Lifecycles**

 - MLLP Receivers

 - Translations

 - MLLP Senders

- **Multiple Approvals**

 - Integration QA

 - External System QA

 - External System Release Management

 - Enterprise Release management

- **Multiple Teams**

 - Integration

 - Development team for each system

 - QA Departments for each system

 - Release management for each system

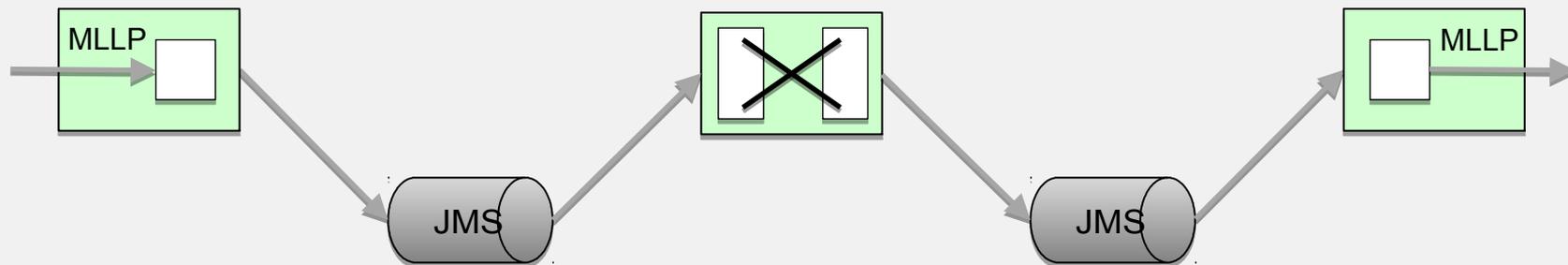
Common Requirements

The Lay of the Land

- **FIFO**
- **Simple Process for Developers**
 - Healthcare SMEs
 - Concurrently work on multiple components
- **Environment Auditing**
 - Current versions in runtime
 - What changed when
- **Component Auditing**
 - What changed between versions
- **Strict Change Control**
 - Human Validation
 - Human Approvals
 - Semantic Versioning
- **Partial “Releases”**
 - Every deployment doesn't change – most don't
 - New systems/interfaces
 - Updates/bug fixes
 - Retire systems/interfaces
- **Release Rollback**
 - Full Release
 - Individual Interfaces

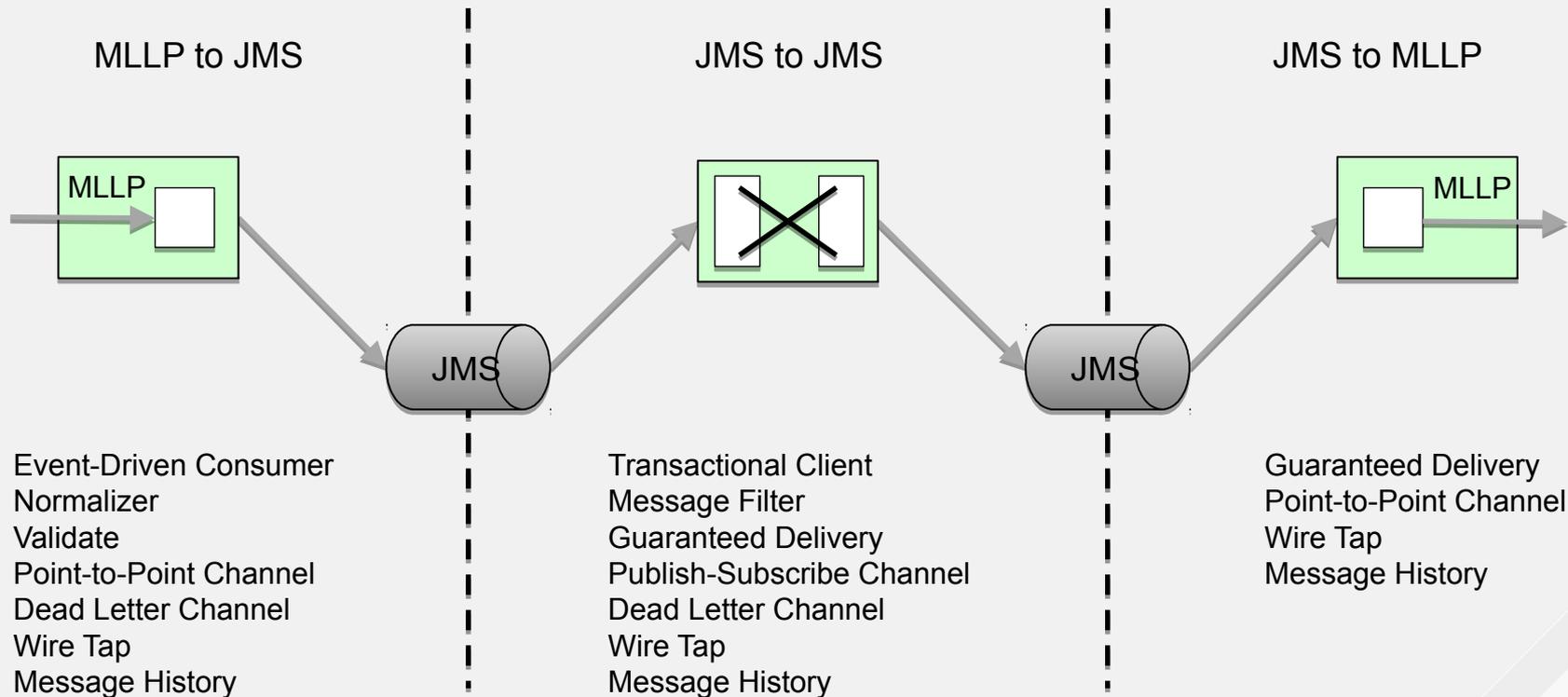
Common Pattern

The Lay of the Land



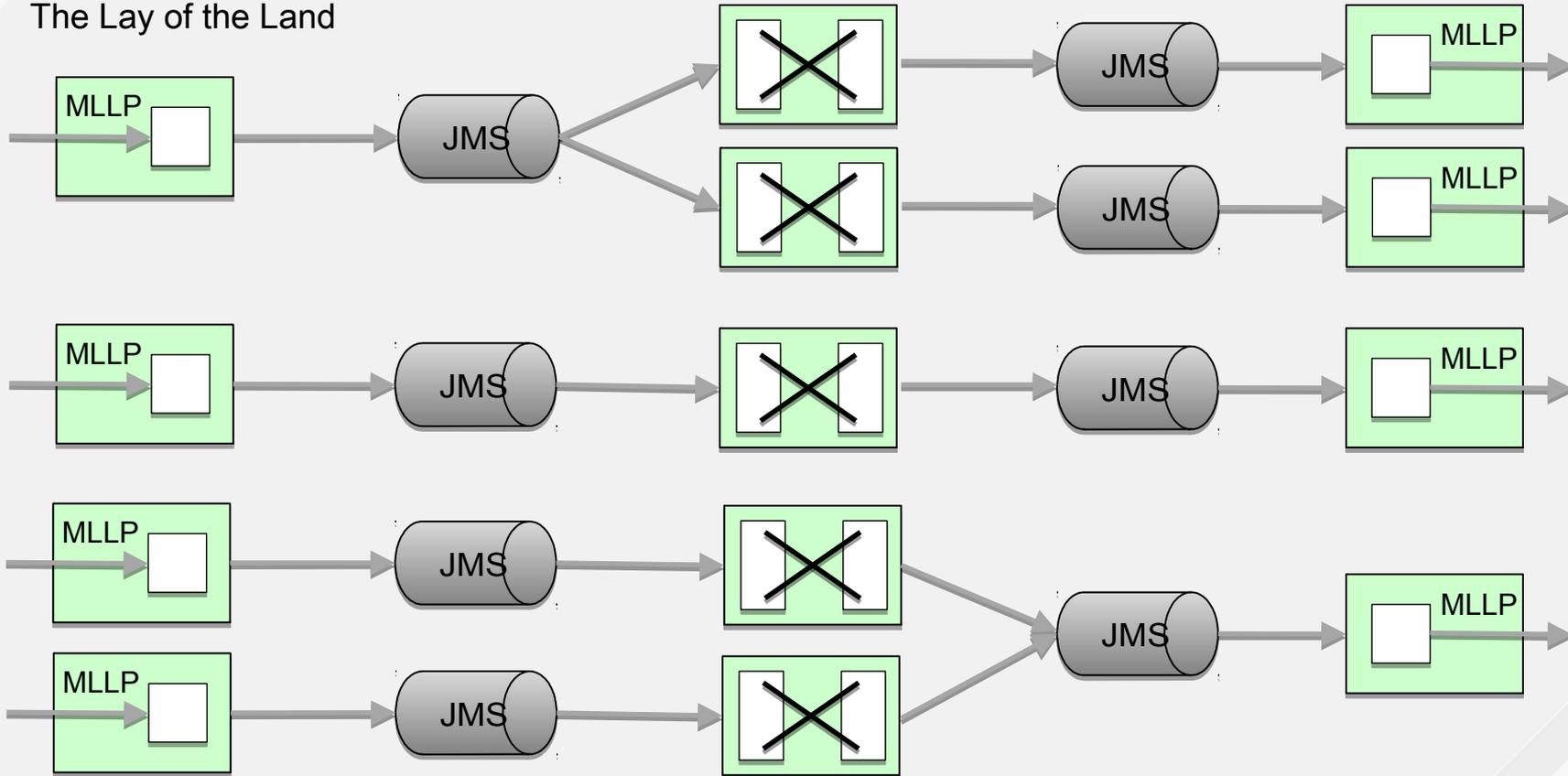
Common Pattern

The Lay of the Land



Common Pattern

The Lay of the Land



Lessons Learned

The Lay of the Land

- **Keep the end in mind (from the beginning)**
 - Automated Testing
 - Continuous Integration
 - Automated Delivery
- **Focus on reuse (avoid duplication)**
 - Code Duplication
 - Configuration Duplication
- **Testing policy**
 - 100% line-coverage for business logic
 - 100% path-coverage for routes
- **Lifecycle**
 - DEV => QA => STAGE => PROD => retirement
 - Move without change
- **Separation of Duties**
 - You can't do it all
 - Everybody can't know everything
- **Custom "DSL"s**
 - Camel-Blueprint
 - Camel-SCR
 - Blueprint Namespace Handlers

Why use JBoss Fuse?

JavaBeans for Business Logic

Why use JBoss Fuse?

- **Insulate from Camel Details**
Exchange, in-Message, out-Message
- **Business Logic**
Message Filter
Message Translator
- **Protect IP**
- **Separation of Duties**
Reduce dependency on product experts
Time-to-Value
- **Testable**
JUnit
Code coverage

Individual Component Features

Why use JBoss Fuse?

Apache Camel

- Reusable Routes
(RouteBuilder)
- Consistent error handling
- Simplified Testing

Apache ActiveMQ

- Virtual Topics
- Composite Destinations
- Exclusive Consumers

Individual Component Features

Why use JBoss Fuse?

Apache Karaf

- Multi-version support
- “Pull Deployment” (not push)
- Customizable
 - Custom commands
 - Namespace Handlers

fabric8

- Centralized container configuration
- Centralized deployment configuration
- Single-view (Management Console)

Approach

Planning – “Failure is not an Option”

Approach

Know your Milestones

- Upfront work is a must!
- Define your phases
- Rigid process

The right tool for the job

- Source Code Repository
- Binary Artifact Repository
- CI/CD
- Log Consolidation
- Monitoring
- Alerting

Small Wins => Big Success

Approach

Baby Steps

- Keep the Big Picture in-mind
- Build what you MUST
Don't re-invent the wheel
- Build it when you need it

Maximize Return

- Focus on Business Value
- Streamline prevalent patterns
- Deliver quickly (and frequently)

Strategies

Artifact Management

Strategies

Git Repositories

- One artifact per repository
- Branches for code lifecycle
(not deployment lifecycle)
- Tags for versions
- Match lifecycle of contents
- Enforce change comments

Maven Projects

- No Child Modules
- Flatten structure
- `<relativePath/>`
- SNAPSHOTs for xxx-test-support
- Semantic versioning
- mvn deploy to “DEV” only

Development – Business Logic

Strategies

Message Filters / Message Translators

- Business Logic in Java
- Prefer Bean Binding
- JavaDoc
- 100% code coverage
- Embedded Test Data
- Checkstyle

Development – Camel Routes

Strategies

Camel Routes

- XML DSL for unique routes
- Java DSL for common route templates
- Error Handlers
- Minimize dependencies

- HawtIO Rendering

- JavaDoc

Deployment

Strategies

Deployment Configuration

- Use fabric8 Profiles instead of Karaf features
- Container Profiles
- Deployment Profiles
- No SNAPSHOT versions in shared environments

Common Mistakes

Strategies

Know your rules

- **Application Access**

LDAP/AD

HTTPS

- **System Access**

SSH

“Service” accounts

- **Data Access**

- **Data Location**

secured network zones

encrypted storage

Know your procedures

- **Quality Measures**

Unit Test code-coverage

Checkstyle / PMD / FindBugs

- **Release Process**

QA signoff

Approvals

Scheduling

- **Production Support**

Automated Monitoring/Alerting

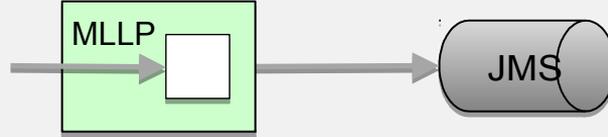
Add-hoc Monitoring

Data-Replay Restrictions

Case Study - Reduce Duplication

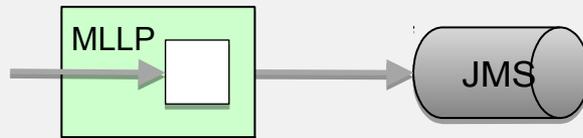
Common Pattern – MLLP Receiver

Reduce Duplication



XML DSL

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <reference id="connection-factory" interface="javax.jms.ConnectionFactory" />

  <bean id="jms-target" class="org.apache.camel.component.sjms.SjmsComponent" >
    <property name="connectionFactory" ref="connection-factory" />
  </bean>

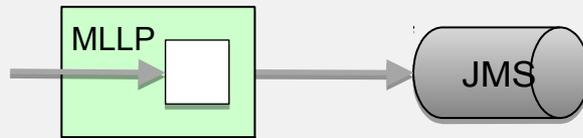
  <camelContext id="epic-mlp-to-jms" xmlns="http://camel.apache.org/schema/blueprint">
    <onCompletion onCompleteOnly="false" onFailureOnly="true">
      <to uri="controlbus://route?routeId=epic-mlp-receiver&action=stop&async=true" />
    </onCompletion>

    <route id="epic-mlp-receiver">
      <from uri="mllp://1234"/>
      <to uri="jms-target://queue:in.adt.epic" />
    </route>
  </camelContext>

</blueprint>
```

XML DSL

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <reference id="connection-factory" interface="javax.jms.ConnectionFactory" />

  <bean id="jms-target" class="org.apache.camel.component.sjms.SjmsComponent" >
    <property name="connectionFactory" ref="connection-factory" />
  </bean>

  <camelContext id="epic-mlp-to-jms" xmlns="http://camel.apache.org/schema/blueprint">
    <onCompletion onCompleteOnly="false" onFailureOnly="true">
      <to uri="controlbus://route?routeId=epic-mlp-receiver&action=stop&async=true" />
    </onCompletion>

    <route id="epic-mlp-receiver">
      <from uri="mllp://1234"/>
      <to uri="jms-target://queue:in.adt.epic" />
    </route>
  </camelContext>

</blueprint>
```

XML DSL

Reduce Duplication

Advantage's

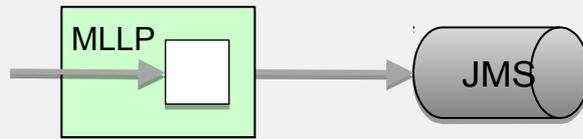
- Easy to read
- CamelBlueprintTestSupport
- Separate Bundle per Deployment
- Directly Deployable

Disadvantage's

- Duplication
- Can't derive values

Java DSL

Reduce Duplication



```
public void configure() throws Exception {
    verifyConfiguration();

    onCompletion().id("onFailureOnly Completion Handler")
        .onFailureOnly()
        .log(LoggingLevel.ERROR, "onFailureOnly Completion handler: ${header[CamelMllpAcknowledgement]}").id("Log Failure")
        .to(errorAuditUri).id("Audit Completion Error");

    RouteDefinition fromDefinition = (mllpHost != null) ? fromF("mllp://%s:%d", mllpHost, mllpPort) : fromF("mllp://%d", mllpPort);

    fromDefinition = fromDefinition.routeId(routeId);

    if (payloadProcessor != null) {
        fromDefinition = fromDefinition.process(payloadProcessor).id("Payload Processor");
    }

    fromDefinition.convertBodyTo(String.class).id("Convert body for Persistence")
        .toF("%s://%s?exchangePattern=InOnly&transacted=true", sjmsComponent, jmsDestination).id("Persist Message");
}
```

Java DSL

Reduce Duplication

Advantage's

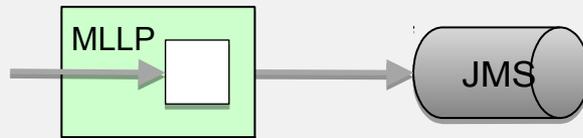
- Encapsulate common logic
- Derive values
- CamelTestSupport

Disadvantage's

- More difficult to read
- Boilerplate code
getters/setters
- Java DSL quirks
- Not directly deployable

Java DSL + Blueprint

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8" ?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <reference id="connection-factory" interface="javax.jms.ConnectionFactory" />

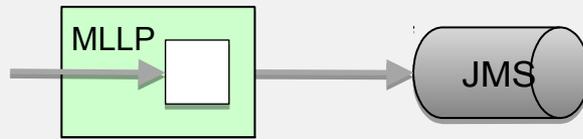
  <bean id="epic-mllep-to-jms-builder" class="edu.ucla.mednet.camel.mllp.jms.MllpToJmsRouteBuilder" >
    <property name="listenPort" value="1234"/>
    <property name="connectionFactory" ref="connection-factory"/>
    <property name="jmsDestination" value="queue://in.adt.epic"/>
  </bean>

  <camelContext id="epic-mllep-to-jms" xmlns="http://camel.apache.org/schema/blueprint">
    <routeBuilder ref="epic-mllep-to-jms-builder" />
  </camelContext>

</blueprint>
```

Java DSL + Blueprint

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8" ?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <reference id="connection-factory" interface="javax.jms.ConnectionFactory" />

  <bean id="epic-mllep-to-jms-builder" class="edu.ucla.mednet.camel.mllp.jms.MllpToJmsRouteBuilder" >
    <property name="listenPort" value="1234"/>
    <property name="connectionFactory" ref="connection-factory"/>
    <property name="jmsDestination" value="queue://in.adt.epic"/>
  </bean>

  <camelContext id="epic-mllep-to-jms" xmlns="http://camel.apache.org/schema/blueprint">
    <routeBuilder ref="epic-mllep-to-jms-builder" />
  </camelContext>

</blueprint>
```

Java DSL + Blueprint

Reduce Duplication

Advantage's

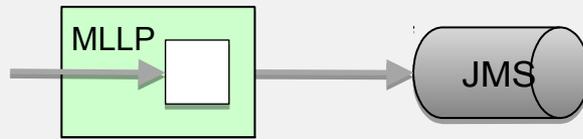
- Simple Configuration
- Directly deployable Configuration
- Separate Bundle per Deployment

Disadvantage's

- Duplication
- Camel-Blueprint + Java RouteBuilder bugs

Camel-SCR

Reduce Duplication

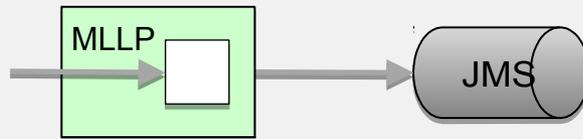


```
org.sutterhealth.eis.templates.mllpsender.SenderScrRunner-SSR_toECAPTURE_ad_ip.properties
```

```
camelContextId = epic-mlp-to-jms  
listeningPort = 1234  
targetDestination = queue:in.adt.epic
```

Camel-SCR

Reduce Duplication



```
org.sutterhealth.eis.templates.mllpsender.SenderScrRunner-SSR_toECAPTURE_ad_ip.properties
```

```
camelContextId = epic-mlp-to-jms  
listeningPort = 1234  
targetDestination = queue:in.adt.epic
```

Camel-SCR

Reduce Duplication

Advantage's

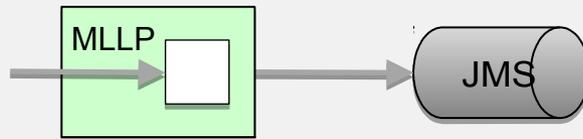
- Simple Configuration
- Directly Deployable Configuration

Disadvantage's

- Limited support for complex types
- PID => Filename dependency
- IDE Support for configurations
- Single Bundle for all configurations
- No CamelScrTestSupport
- Deprecated in 2.20 (?)

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

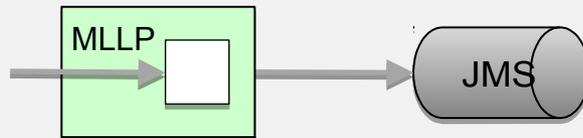
  <mllp-to-jms xmlns="http://edu.ucla.mednet/xmlns/mllp-to-jms/v1.0.0"
    application="epic" data-type="adt" port-offset="1" broker-name="inbound"/>

  <!-- jms destination => queue:vt.in.epic.adt -->
  <!-- camel-context name => epic-adt -->

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8" ?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

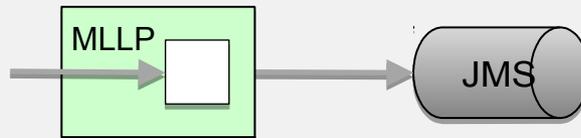
  <mllp-to-jms xmlns="http://edu.ucla.mednet/xmlns/mllp-to-jms/v1.0.0"
    application="epic" data-type="adt" port-offset="1" />

  <!-- jms destination => queue:in.epic.adt -->
  <!-- camel-context name => epic-adt -->

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8" ?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <mllp-to-jms xmlns="http://edu.ucla.mednet/xmlns/mllp-to-jms/v1.0.0"
    application="epic" data-type="adt" />

  <!-- jms destination => queue:in.epic.adt -->
  <!-- camel-context name => epic-adt -->

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication

Advantage's

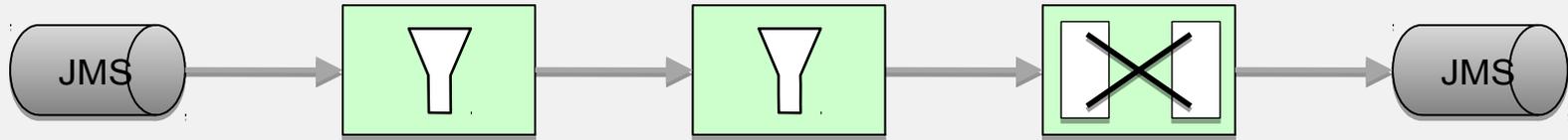
- Simple Configuration
- Minimal Duplication
- Directly Deployable Configuration
- Configuration Lookup
- Centralized Default values
- Separate Bundle or Shared Bundle options

Disadvantage's

- Immature spec - Blueprint / Apache Aries Only
- Additional Development Effort
- CamelKarafTestSupport is immature

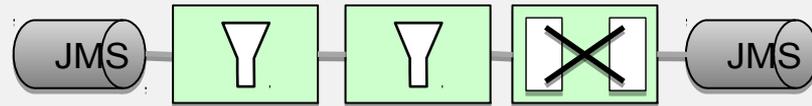
Common Pattern – Filter and Translate

Reduce Duplication



Camel-SCR

Reduce Duplication



AB_EPICadCBORD_ip.filter.properties

```
event = A01|A02|A03|A04|A06|A07|A08|A11|A12|A13|A21|A22|A52|A53
```

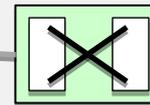
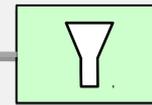
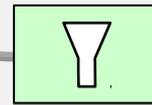
AB_EPICadCBORD_ip.qualifier.properties

```
event = A02|A03|A11|A12|A18
```

AB_EPICadCBORD_ip.translation.properties

Blueprint Namespace Handle

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt" target-application="lynx">
    <qualifier ref="qualifier-service" />
    <qualifier ref="qualifier-bean" />
    <translation ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

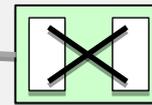
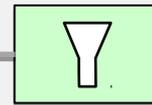
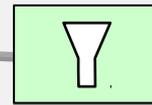
  <bean id="qualifier-bean" class="edu.ucla.mednet.iss.it.qualifier.common.HeaderBlacklistQualifier">
    <property name="blacklist">
      <map>
        <entry key="CamelHL7TriggerEvent">
          <set>
            <value>A02</value>
            <value>A03</value>
            <value>A11</value>
            <value>A12</value>
            <value>A18</value>
          </set>
        </entry>
      </map>
    </property>
  </bean>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8" ?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt" target-application="lynx">
    <qualifier ref="qualifier-service" />
    <qualifier ref="qualifier-bean" />
    <translation ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

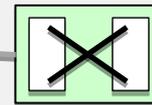
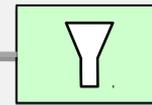
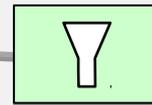
  <bean id="qualifier-bean" class="edu.ucla.mednet.iss.it.qualifier.common.HeaderBlacklistQualifier">
    <property name="blacklist">
      <map>
        <entry key="CamelHL7TriggerEvent">
          <set>
            <value>A02</value>
            <value>A03</value>
            <value>A11</value>
            <value>A12</value>
            <value>A18</value>
          </set>
        </entry>
      </map>
    </property>
  </bean>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8" ?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt" target-application="lynx">
    <qualifier ref="qualifier-service" />
    <qualifier ref="qualifier-bean" />
    <translation ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

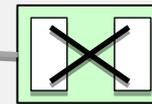
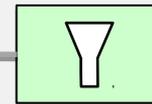
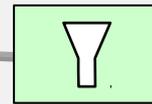
  <bean id="qualifier-bean" class="edu.ucla.mednet.iss.it.qualifier.common.HeaderBlacklistQualifier">
    <property name="blacklist">
      <map>
        <entry key="CamelHL7TriggerEvent">
          <set>
            <value>A02</value>
            <value>A03</value>
            <value>A11</value>
            <value>A12</value>
            <value>A18</value>
          </set>
        </entry>
      </map>
    </property>
  </bean>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt" target-application="lynx">
    <qualifier ref="qualifier-service" />
    <qualifier ref="qualifier-bean" />
    <translation ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

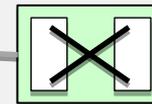
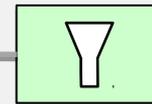
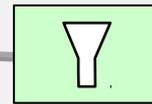
  <bean id="qualifier-bean" class="edu.ucla.mednet.iss.it.qualifier.common.HeaderBlacklistQualifier">
    <property name="blacklist">
      <map>
        <entry key="CamelHL7TriggerEvent">
          <set>
            <value>A02</value>
            <value>A03</value>
            <value>A11</value>
            <value>A12</value>
            <value>A18</value>
          </set>
        </entry>
      </map>
    </property>
  </bean>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt" target-application="lynx">
    <qualifier ref="qualifier-service" />
    <qualifier ref="qualifier-bean" />
    <translation ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

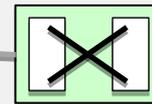
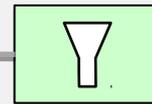
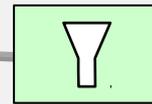
  <bean id="qualifier-bean" class="edu.ucla.mednet.iss.it.qualifier.common.HeaderBlacklistQualifier">
    <property name="blacklist">
      <map>
        <entry key="CamelHL7TriggerEvent">
          <set>
            <value>A02</value>
            <value>A03</value>
            <value>A11</value>
            <value>A12</value>
            <value>A18</value>
          </set>
        </entry>
      </map>
    </property>
  </bean>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt" target-application="lynx">
    <qualifier ref="qualifier-service" />
    <qualifier ref="qualifier-bean" />
    <translation ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

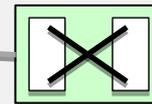
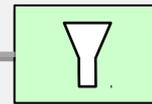
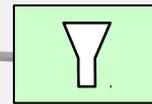
  <bean id="qualifier-bean" class="edu.ucla.mednet.iss.it.qualifier.common.HeaderBlacklistQualifier">
    <property name="blacklist">
      <map>
        <entry key="CamelHL7TriggerEvent">
          <set>
            <value>A02</value>
            <value>A03</value>
            <value>A11</value>
            <value>A12</value>
            <value>A18</value>
          </set>
        </entry>
      </map>
    </property>
  </bean>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

Blueprint Namespace Handler

Reduce Duplication



```
<?xml version="1.0" encoding="UTF-8"?>
<blueprint xmlns="http://www.osgi.org/xmlns/blueprint/v1.0.0">

  <jms-to-jms xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    source-application="epic" source-data-type="adt"
    target-application="lynx">
    <qualifier-ref ref="qualifier-service" />
    <qualifier-ref ref="qualifier-bean" />
    <translation-ref ref="translation-bean" />
  </jms-to-jms>

  <reference id="qualifier-service" interface="edu.ucla.mednet.iss.it.qualifier.spi.Qualifier" filter="application=epic" />

  <qualifier xmlns="http://edu.ucla.mednet/xmlns/jms-to-jms/v1.0.0"
    id="qualifier-bean" type="whitelist" source="message-header" key="CamelMllpTriggerEvent">
    <value>A02</value>
    <value>A03</value>
    <value>A11</value>
    <value>A12</value>
    <value>A18</value>
  </qualifier>

  <bean id="translation-bean" class="edu.ucla.mednet.iss.it.translation.epic.lynx.adt.EpicLynxADT" />

</blueprint>
```

RED HAT
SUMMIT

THANK YOU



plus.google.com/+RedHat



facebook.com/redhatinc



linkedin.com/company/red-hat



twitter.com/RedHatNews



youtube.com/user/RedHatVideos

The logo consists of a red speech bubble shape pointing downwards, containing the text "RED HAT" in a smaller font above "SUMMIT" in a larger, bold font.

RED HAT
SUMMIT

LEARN. NETWORK.
EXPERIENCE
OPEN SOURCE.