F4: External Integrations

Introducing Camel
What is Camel?

Good question. See: http://camel.apache.org/what-is-camel.html

So really you want see this: http://stackoverflow.com/questions/8845186/what-exactly-is-apache-camel
In short...

- Camel is a framework for creating small message based applications… and then some.
- Camel formalizes working with messages so well it can be described in multiple formats: Java, Spring/Blueprint XML, and Scala.
- Camel is all the code you should not have to write in order to work with queues, files, databases, RESTful APIs, common data formats, command line utilities, etc… in a consistent and reliable manner.
Available Camel Components

http://camel.apache.org/components.html

- ActiveMQ
- AWS SQS
- Dropbox
- System calls
- Local files
- FTP
- HTTP resources
- LDAP
- SMTP
- SQL
- Twitter
- etc, etc, etc
Camel can run...

- As a stand-alone Java application
- In a servlet container like Tomcat or Jetty
- In an OSGi runtime such as Karaf
What is OSGi?

- Open Service Gateway Initiative
- Framework for modularizing and deploying Java applications
  - Hot deployment
  - Automatic reloading of configuration
  - Sophisticated dependency resolution
  - XML scripting for complex deployments (features)
Hot Deployment

Bundles can be started, stopped, updated, etc… at runtime!

In other words:
YOU DO NOT HAVE TO RESTART YOUR SERVER TO UPDATE CODE OR CONFIGURATION
Terminology

● Apache Camel
  ○ Endpoints
  ○ Components
  ○ Messages
  ○ Routes

● Apache Karaf -- OSGi
  ○ Bundles
  ○ Features
Hands-On: Into the Vagrant

> vagrant ssh
  or:

> ssh -p 2222 vagrant@localhost
  password = vagrant
Detour: Fixity corruption revisited

> cd /var/lib/tomcat7/fcrepo4-data/fcrepo.binary.directory
> sudo su
> find . -name [cover.jpg-sha1]
> echo hello >> [full-path-from-previous-command]
> exit
Hands-On: Inspect features

> /opt/karaf/bin/client

>> feature:list | grep fcrepo

fcrepo-camel
fcrepo-indexing-triplestore
fcrepo-audit-triplestore
fcrepo-indexing-solr
fcrepo-reindexing
fcrepo-fixity
fcrepo-serialization
Hands-On: Helpful Commands

>> feature:install fcrepo-audit-triplestore
>> feature:stop <whichever>
>> camel:route-list
>> bundle:list | grep fcrepo
>> ctrl-d
Hands-On: Watch the log

In a new vagrant ssh terminal:

> sudo tail -f /opt/karaf/data/log/karaf.log
Hands-On: Indexing in triplestore

http://localhost:8080/fuseki
Hands-On: Indexing in triplestore

```
select * where {
}
```
Hands-On: Indexing in triplestore

PREFIX ldp: <http://www.w3.org/ns/ldp#>
PREFIX ebucore: <http://www.ebu.ch/metadata/ontologies/ebucore/ebucore#>

select * where {
  ?o ebucore:hasMime-Type ?m
}
Hands-On: Indexing in triplestore

Audit

PREFIX premis: <http://www.loc.gov/premis/rdf/v1#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

select ?s ?d where {
  ?s premis:hasEventDateTime ?d .
}
Hands-On: Indexing in Solr

http://localhost:8080/solr

collection1
Hands-On: Indexing in Solr
Hands-On: Reindexing - prep

> sudo service tomcat7 stop
> sudo rm -rf /etc/fuseki/databases/test_data/*
> sudo service tomcat7 start
Hands-On: Reindexing - check

In Fuseki: http://localhost:8080/fuseki

--------

select * where {
}

Hands-On: Reindexing

> curl -XPOST localhost:9080/reindexing/cover -H "Content-Type: application/json" -d '[]"
 ['-"activemq:queue:triplestore.reindex"]'
Hands-On: Fixity - setup

> sudo vi /opt/karaf/etc/org.fcrepo.camel.fixity.cfg

Change:

fixity.success=mock:fixity.success

To:

fixity.success=file:/tmp/?fileName=fixitySuccess.log&fileExist=Append
Hands-On: Fixity

> curl -XPOST localhost:9080/reindexing/cover
- H"Content-Type: application/json" -d
'"activemq:queue:fixity"]'

> less /tmp/fixitySuccess.log
> less /tmp/fixityErrors.log
Hands-On: Serialization - RDF

> tree /tmp/descriptions
Hands-On: Serialization - reconfig

> sudo vi /opt/karaf/etc/org.fcrepo.camel.serialization.cfg

Change:
serialization.includeBinaries=false

To:
serialization.includeBinaries=true
Hands-On: Serialization - Binaries

curl -XPOST localhost:9080/reindexing/cover -H "Content-Type: application/json" -d "["activemq:queue:serialization"]"

> tree /tmp/binaries
Success!