

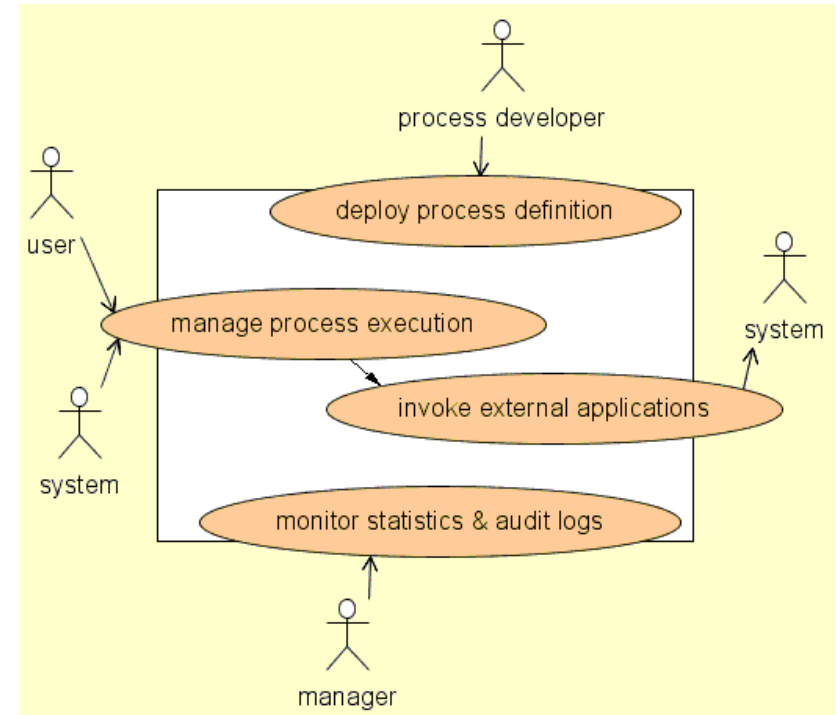
Business Process Management with Mule

Travis Carlson



What is BPM?

- ✓ “Programming in the large”
- ✓ Bridges the gap between business analysts and programmers
- ✓ Allows for BAM (Business Activity Monitoring) which gives managers a view into their operations
- ✓ Makes business operations more agile
- ✓ Can include human input (workflow) as part of a process



What does Mule bring to BPM?

- Business processes often require integration
- Integration is often part of a larger business process
- Managers need a unified view of their business processes, that includes integration



What does BPM bring to Mule?

- Easier modeling + management of long running integration processes
- Complex decision logic, possibly using Business Rules
- Human intervention for exception handling



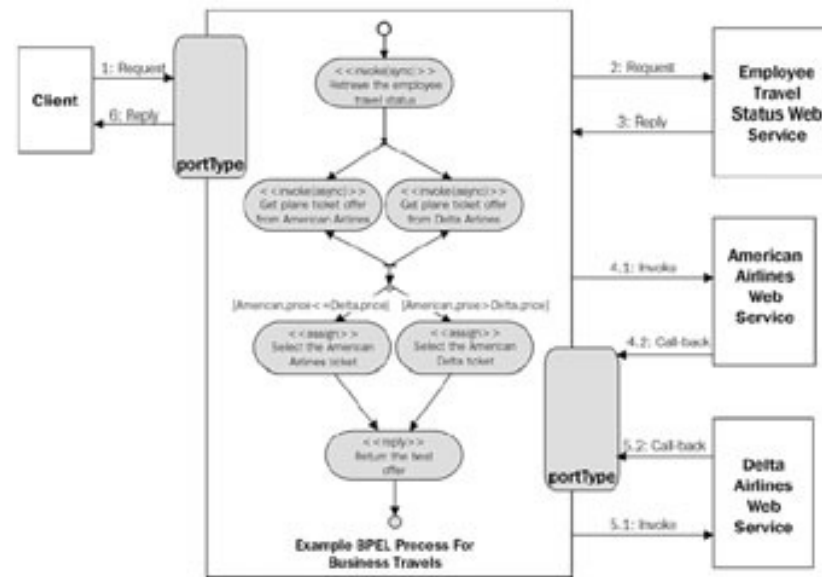
Some similar concepts / concerns

- **Process definition file**
- **Process Designer** – graphically model your process
- **BAM - analyze stats. from process transitions**
- **Transactions** between process events
- **Persistence/clustering** for failover
- **Mule configuration file**
- **Mule IDE** – graphically model your integration
- **MuleHQ** – analyze stats. from integration events
- **Transactions** between incoming/outgoing messages
- **Persistence/clustering** for failover (Mule 2.x)



What about BPEL?

BPEL tries to combine BPM and integration in a standard way, but...

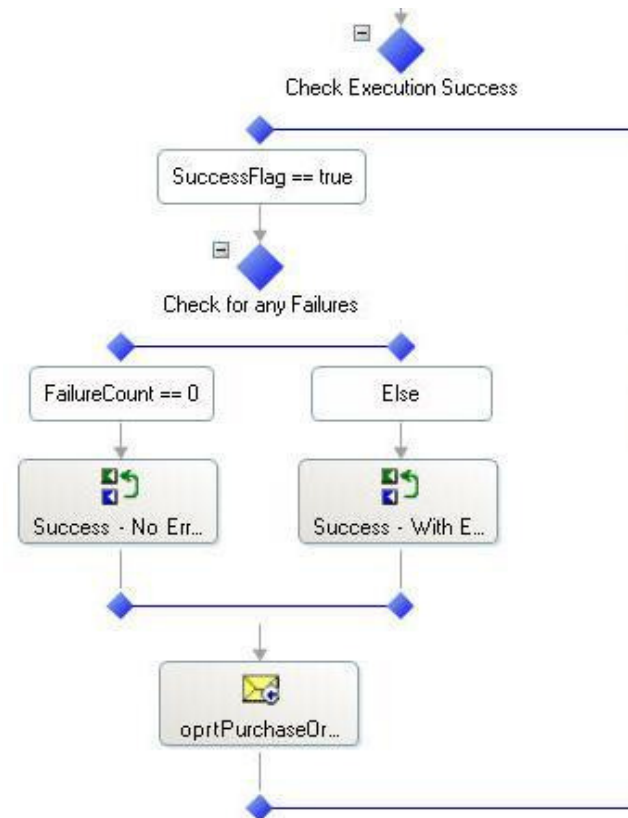


It only talks to Web Services and XML while Mule will integrate with anything



What about BPEL?

- Mule can integrate with a BPEL engine via SOAP endpoints (XFire or Axis)
- But a BPM engine with a Java API is a much better complement to Mule



Restrictive Standards vs. Flexible Java



Good idea

Better idea



Object Lifecycle

EJB 2.0

Spring

Systems Integration

JBIP

Mule

Process Management

BPEL

jBPM

POJOs!
Lightweight,
Flexible,
"Non-intrusive"

XML only, Web
services only,
lots of WSDL :-)



BPM Connector for Mule

Defines a simple interface to “plug in” any BPM engine

```
public interface BPMS
{
    public Object startProcess(Object processType, Map processVariables);
    public Object advanceProcess(Object processId, Object transition, Map processVariables);
    public Object updateProcess(Object processId, Map processVariables);
    public void setMessageService(MessageService msgService);
}

public interface MessageService
{
    public UMOMessage generateMessage(String endpoint, Object payloadObject,
                                     Map messageProperties, boolean synchronous) ;
}
```

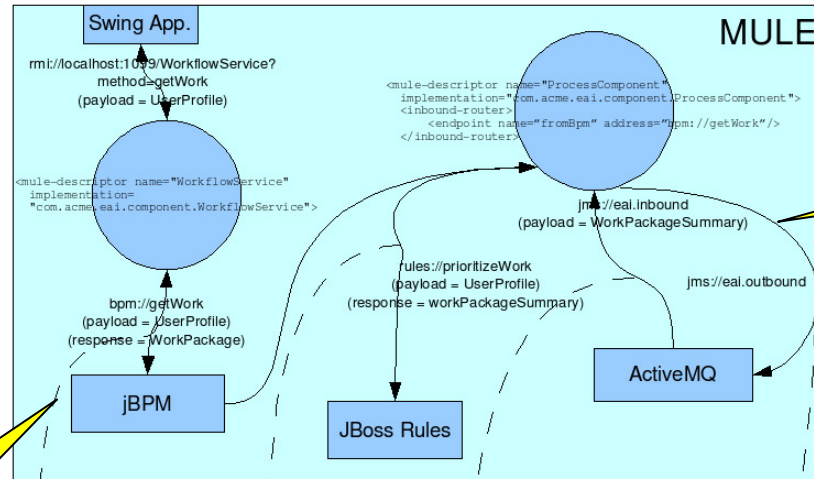


BPM Connector for Mule

- A running process can generate synchronous or asynchronous Mule events
- Incoming Mule events can launch new processes, advance or terminate running processes
- Endpoints of type “bpm://myProcess” are used to intelligently route process-generated events within Mule
- Mule can interact with different processes in parallel



Mule and BPM working together



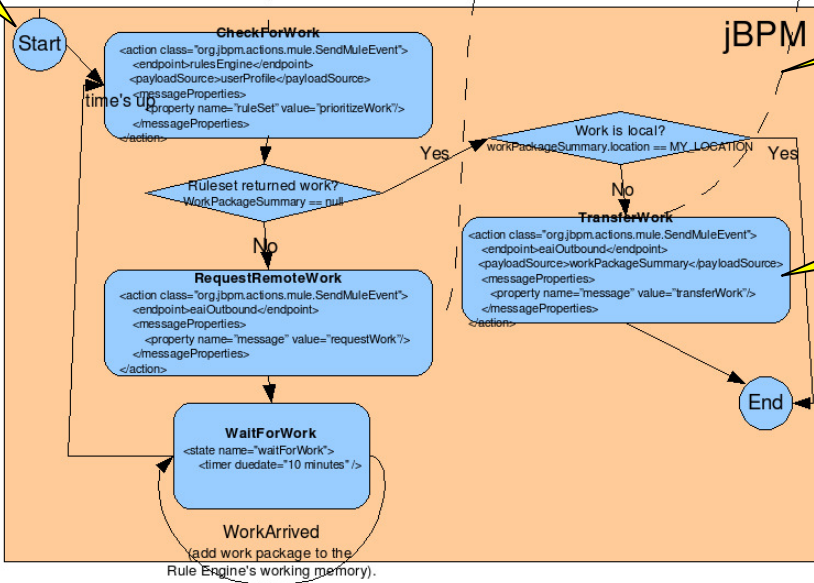
Mule messages...

A new process...

generated by...

triggered by Mule

the executing process



Coming soon

- Available in Mule 1.4
- Loan Broker BPM Example
- So far integrated with JBoss jBPM, but may be extended to “plug in” other BPM engines as well



Feedback / Questions

Your thoughts?

