

Mule Demo

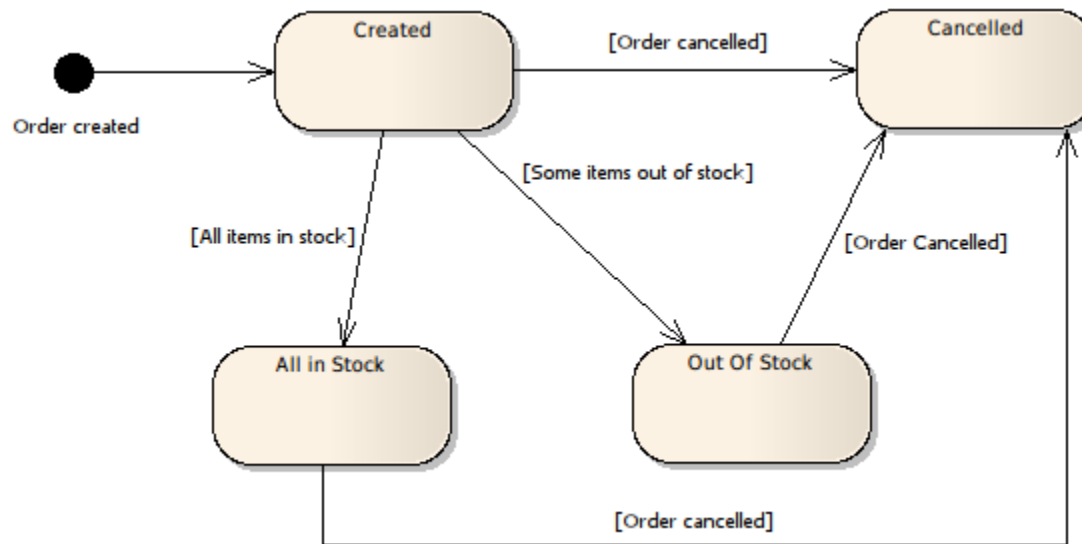
Overview



Premise of the demo

- A company has a web-based Order Management system (Orderman)
- They wish to integrate this system with their new Stock Control system (Stockcontrol)
- They also wish to monitor all the status change events that happen to an Order for BI purposes

Life-cycle of an Order



The Orderman app

- Web based application
- Allows orders for widgets to be created or cancelled
- All orders stored in database
- Mimics a legacy app, no services or API, only database access available

The Stockcontrol app

- Exposes a SOAP based web service that can be queried to see if items are in stock or not
- For the demo “widget2”s will always be out of stock

The Dashboard app

- Web based application
- Stores event data in database
- Provides event/per minute by type graphs
- Provides a REST(like) web service for logging events

Integration Strategy

- Glue applications together using ESB
- Add a database trigger to the Orderman app which creates order transaction records whenever an order is updated or created
- Have the ESB monitor the order transaction table and create OrderEvent messages for each record

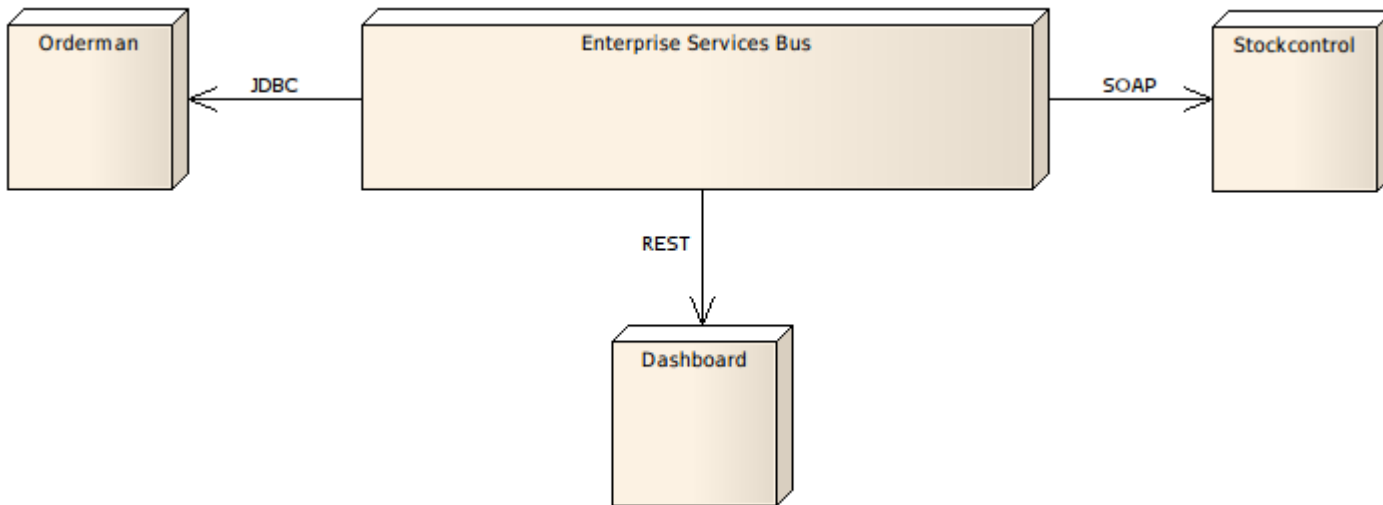
Integration Strategy cont.

- Have the ESB log all order events by calling the Dashboard's REST service for each OrderEvent that passes through the ESB
- For all “order created” OrderEvents have the ESB call the Stockcontrol's SOAP web service to check stock availability

Integration Strategy cont.

- Have the ESB update the status of the order in the Orderman database, to either “out of stock” or “in stock” based on the result of the SOAP call to the Stockcontrol system

Solution Deployment



Solution Demo

And now a demonstration of the solution....