

Integrating messaging middleware with application servers

Component Computing 2000

Johan Eltes

e-business architect

Callista Enterprise AB

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23



Content

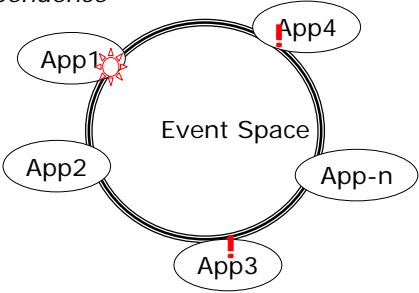
- ✍ Background / introduction
- ✍ Adapting business components to MOM
- ✍ Adapting inbound messages
- ✍ Adapting outbound messages
- ✍ Error handling and operations
- ✍ Products and tools

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23




Background

- ✍ EAI in a perfect world
 - ✍ *Autonomous applications*
 - ✍ *Publish and Subscribe*
 - ✍ *Common message format and syntax*
 - ✍ *Network independence*



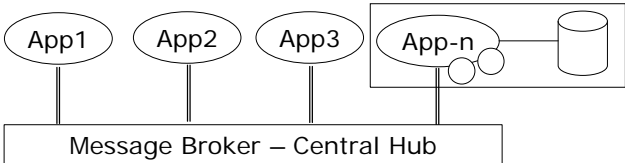
The diagram shows a central circle labeled "Event Space". Four ovals, labeled "App1", "App2", "App3", and "App-n", are arranged around the circle. App1 is at the top left, App2 at the bottom left, App3 at the bottom, and App-n at the bottom right. App4 is at the top right. Red arrows point from App1 and App3 towards the Event Space.

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23




Background...

- ✍ EAI in the real world
 - ✍ *Transactional store-and-forward*
 - ✍ *Different operations cycles*
 - ✍ *No common message formats or syntax*
 - ✍ *Application extensions*
 - ✍ *Important to preserve the principals!*



The diagram shows a central rectangular box labeled "Message Broker - Central Hub". Four ovals, labeled "App1", "App2", "App3", and "App-n", are positioned above the box. Vertical lines connect each oval to the top of the box. App-n is enclosed in a larger box that also contains a cylinder icon representing a database.


Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23



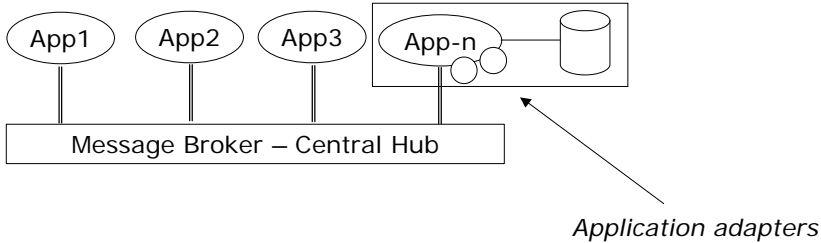
Levels of Pub/Sub

- ✍ Application level
 - ✍ App Server hosted Pub/Sub
 - Transactional
 - Synchronous
 - Application Extensions
 - Often within in ONE appserver
- ✍ Enterprise level
 - ✍ Message Broker Pub/Sub
 - Series of transactions (over MOM)
 - Asynchronous
 - Different technologies

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23




Left alone...

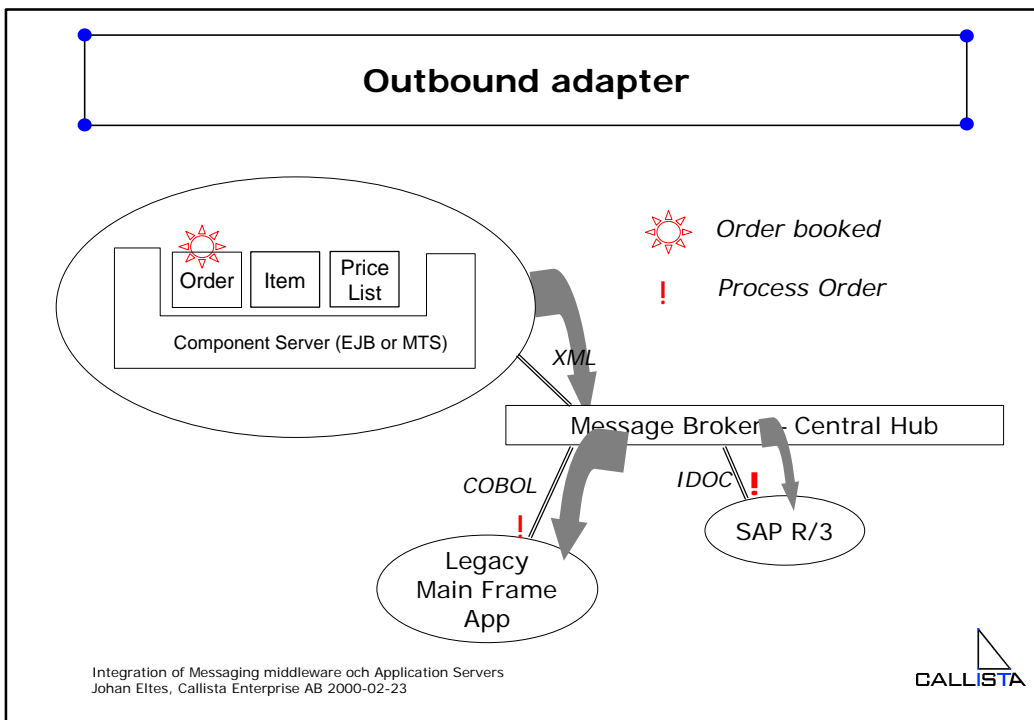
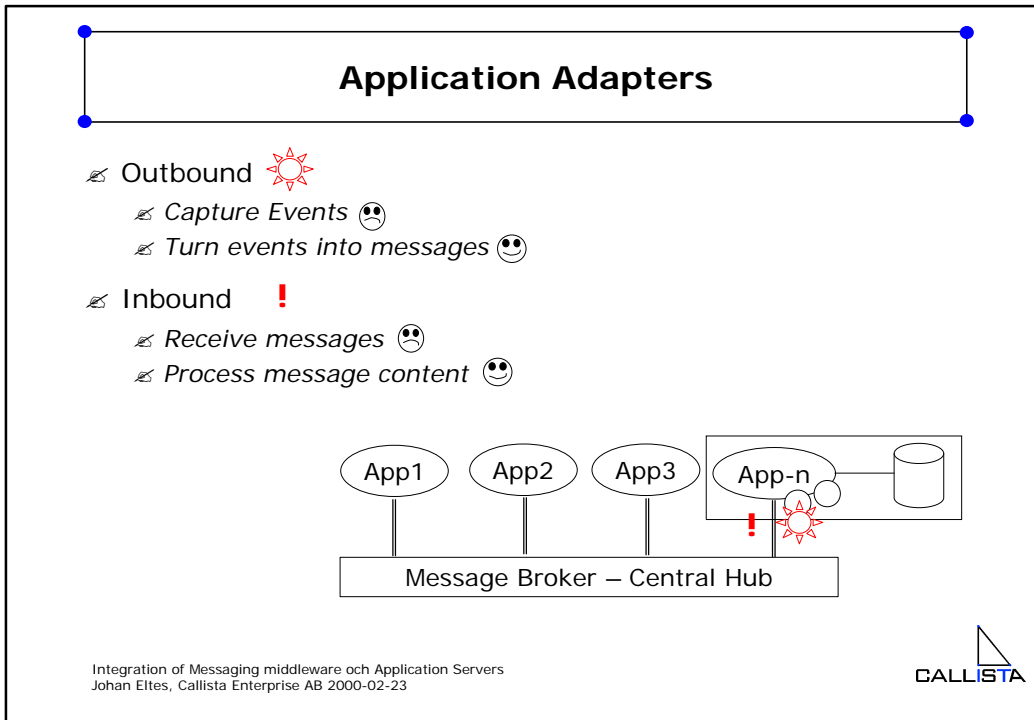


Message Broker – Central Hub

Application adapters

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23







Events vs Messages

- ✎ Events (in this context) ☀
 - ✎ Synchronous
 - ✎ Transactional
 - ✎ 0..3 subscribers
 - ✎ No subscriber – virtualy no cost
 - ✎ Local to app server
- ✎ Subscribers
 - ✎ Extends the application
 - ✎ MAY produce a message
- ✎ Messages
 - ✎ Asynchronous
 - ✎ ALL data the message broker needs

Integration of Messaging middleware och Application Servers
 Johan Eltes, Callista Enterprise AB 2000-02-23



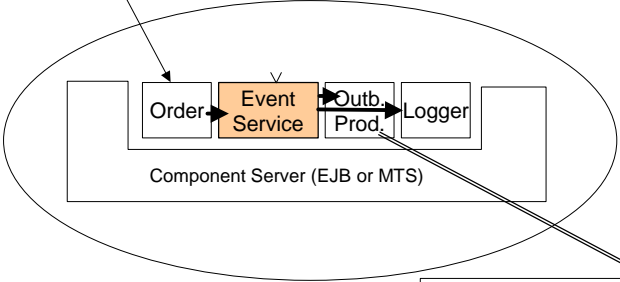
Events vs. Messages...



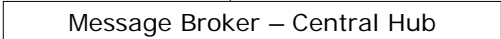
Subscriptions
(XML)

```

            <?xml version="1.0" encoding="UTF-8" ?>
            - <EventSubscription>
              <Subscriber>OrderMessageProducer</Subscriber>
              <EventMask>OrderStatus_Booked</EventMask>
              <EventMask>OrderStatus_Canceled</EventMask>
            - <SubscriptionData>
              <PROPERTY key="QueueName" value="OUT.MESSAGEBROKER.QUEUE" />
            </SubscriptionData>
            </EventSubscription>
            
```




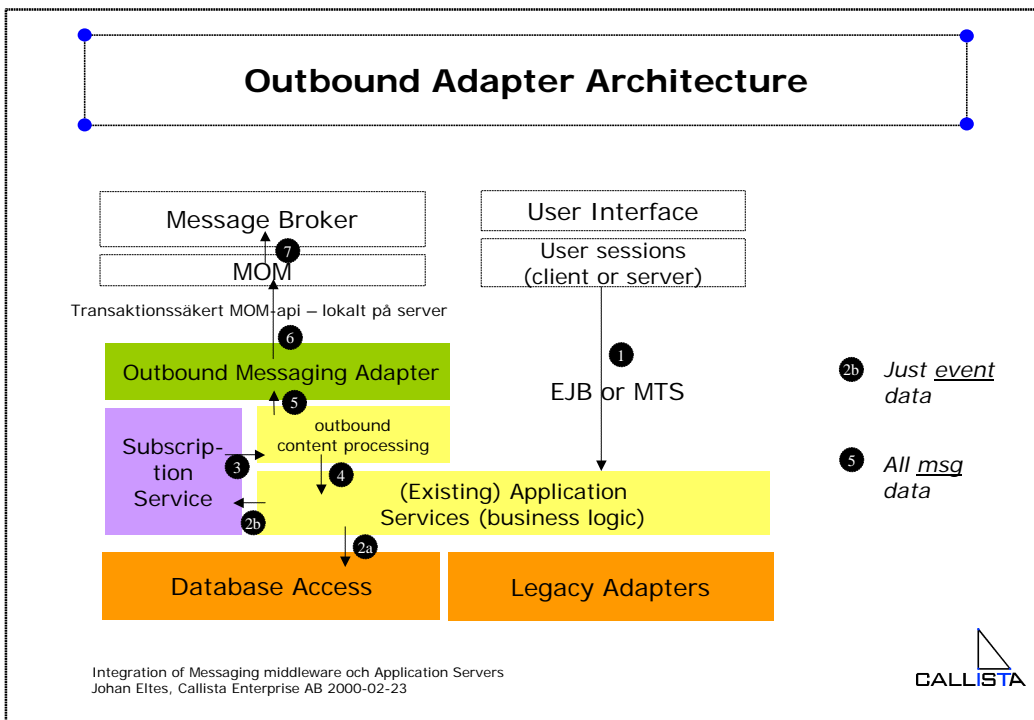
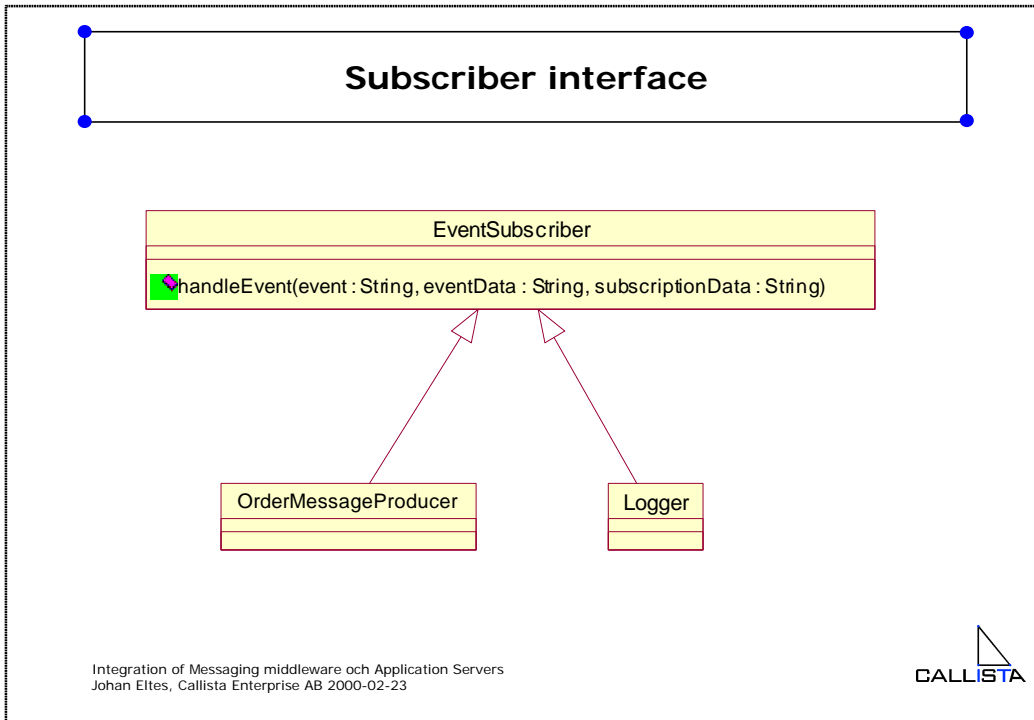
Component Server (EJB or MTS)

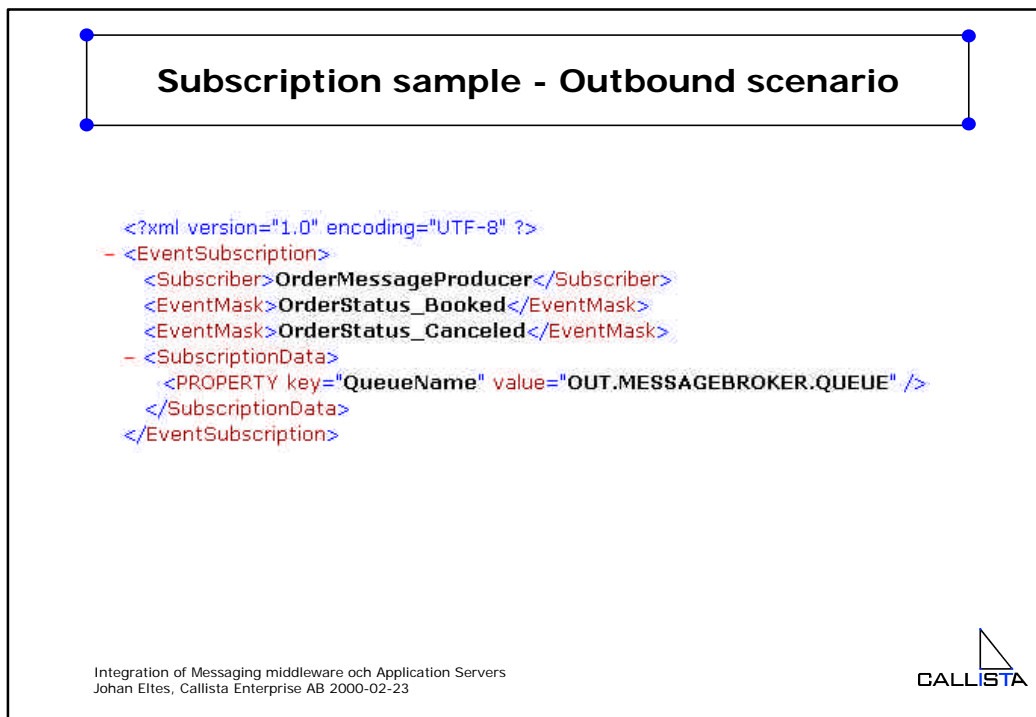
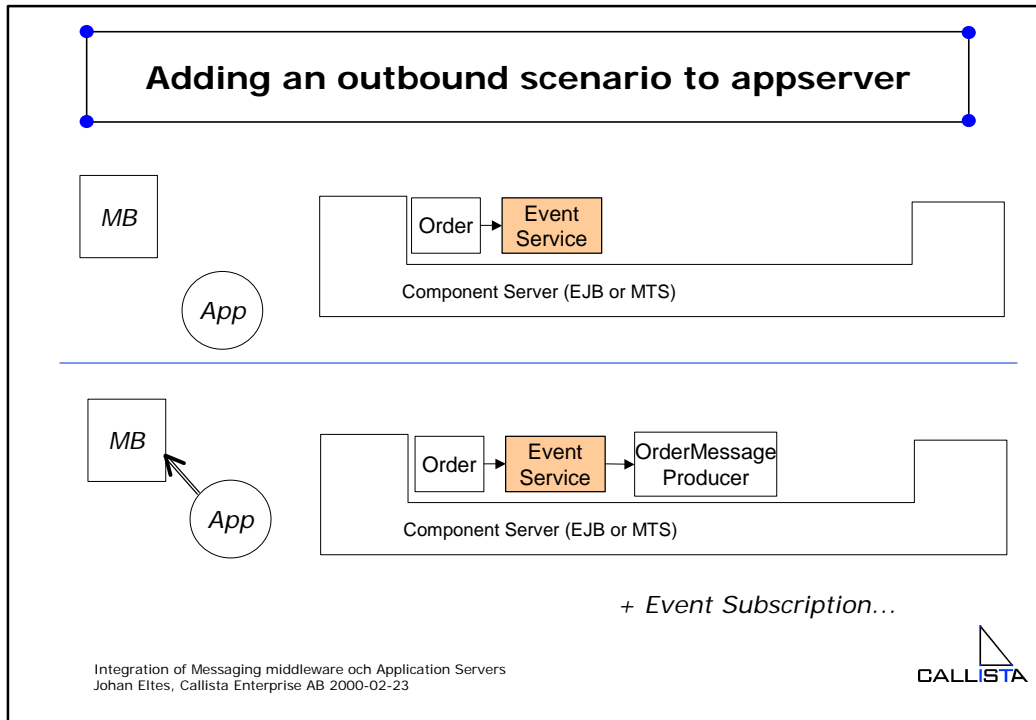


Message Broker – Central Hub

Integration of Messaging middleware och Application Servers
 Johan Eltes, Callista Enterprise AB 2000-02-23






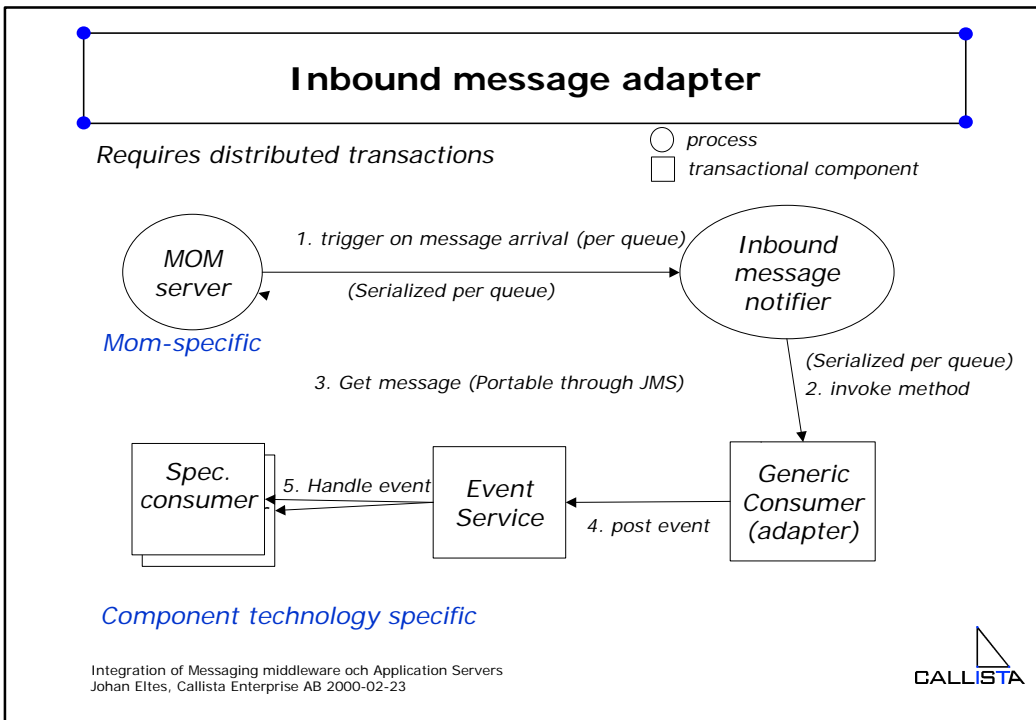


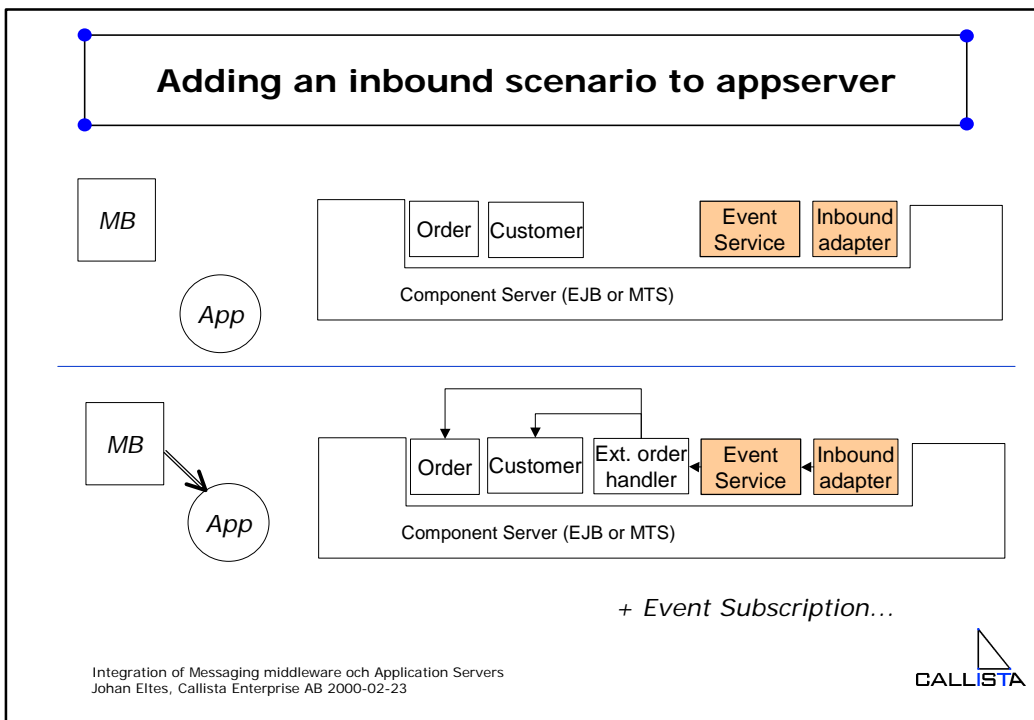
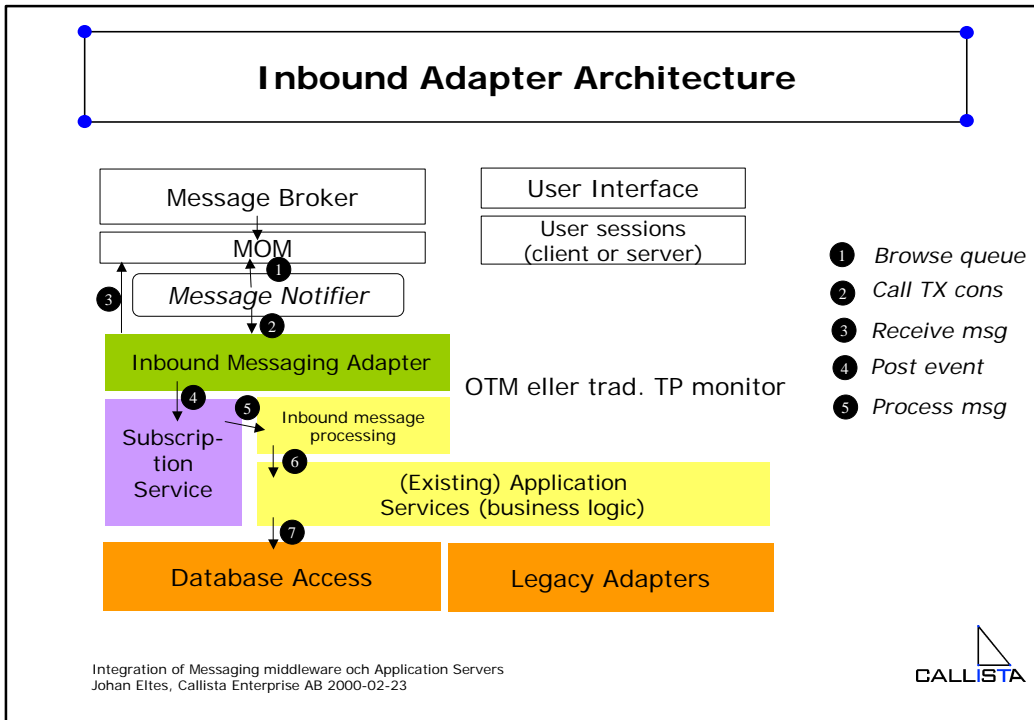
Inbound Messages

- ✂ Trigger/Consume
 - ✂ How to consume a message under transaction?
- ✂ Service/daemon/AppServer startup-class
 - ✂ Listens with BROWSE option (JMS QueueBrowser) or trigger
 - ✂ When message arrives, call:
 - ✂ Message Consumer component, declared to start transaction
 - ✂ Message Consumer gets (receives) message and posts event
 - ✂ Event Handler processes the event

Integration of Messaging middleware och Application Servers
 Johan Eltes, Callista Enterprise AB 2000-02-23







Inbound Order Subscription

- ✍ InboundAdapter component
 - ✍ Get (XML) message from queue
 - ✍ call `EventService.postEvent("order.web.in.queue",xmlOrderData)`
- ✍ EventService
 - ✍ Look-up subscribers on "order.web.in.queue"
 - ✍ Instanciate each subscriber and call `handleEvent("order.web.in.queue",xmlOrderData, subscriptionData)`

```
<?xml version="1.0" encoding="UTF-8" ?>
- <EventSubscription>
  <Subscriber>ExternalOrderHandler</Subscriber>
  <EventMask>order.web.in.queue</EventMask>
</EventSubscription>
```

Error handling

- ✍ Inbound
 - ✍ Message processing code should give hints...
 - Logical error
 - Action: User problem
 - Resource problem (ev. temporary)
 - Action: Shut down – notify operations
 - Inconistency problem
 - Maybe temporary, due to sequencing problems
 - Action: Put back at end, try once more, then User Problem.
- ✍ Outbound
 - ✍ With end user – no big deal
 - ✍ With other application as orig. trigger; same as for inbound

Error handling Strategies

- ✗ Application component may never reject a message
 - ✗ *Message logging part of application*
- ✗ Application may reject a message
 - ✗ *Adapter must implement logging and user correction workflow*
 - ✗ *Often delegates back to message broker*

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23



Products and tools

- ✗ Messaging api
 - ✗ *Native a'la MQSeries, MSMQ*
 - ✗ *J2EE: Java Message Service*
 - No added value, except portability
 - Only asynchronous messaging
- ✗ Event Service
 - ✗ *COM+ Event System*
 - Transactional, synchronous pub/sub
 - No asynch messaging (special case: queued components)
- ✗ Messaging adaptors for comp. technologies
 - ✗ *We need products!*

Integration of Messaging middleware och Application Servers
Johan Eltes, Callista Enterprise AB 2000-02-23



Links

- ✎ MTS resource manager for MQSeries
✎ <http://www.dynamo-ny.com/products.htm>
- ✎ JMS Spec
✎ <http://java.sun.com/products/jms/>
- ✎ IBM MQSeries JMS implementation
✎ <http://www-4.ibm.com/software/ts/mqseries/library/manuals99/csqzaw/csqzaw28.htm#HDRJMS770X>
- ✎ COM+ Event System
✎ http://msdn.microsoft.com/library/psdk/cos sdk/pgservices_events_20rp.htm
- ✎ This presentation
✎ <http://www.callista.se/cc2000/Presentation>