

Enterprise Application Integration patterns for Java EE cloud applications

JavaOne 2012

Alexander Heusingfeld

Stefan Reuter

Speakers

§ Alexander Heusingfeld

§ Senior Consultant, Cyber:con GmbH & Freelancer

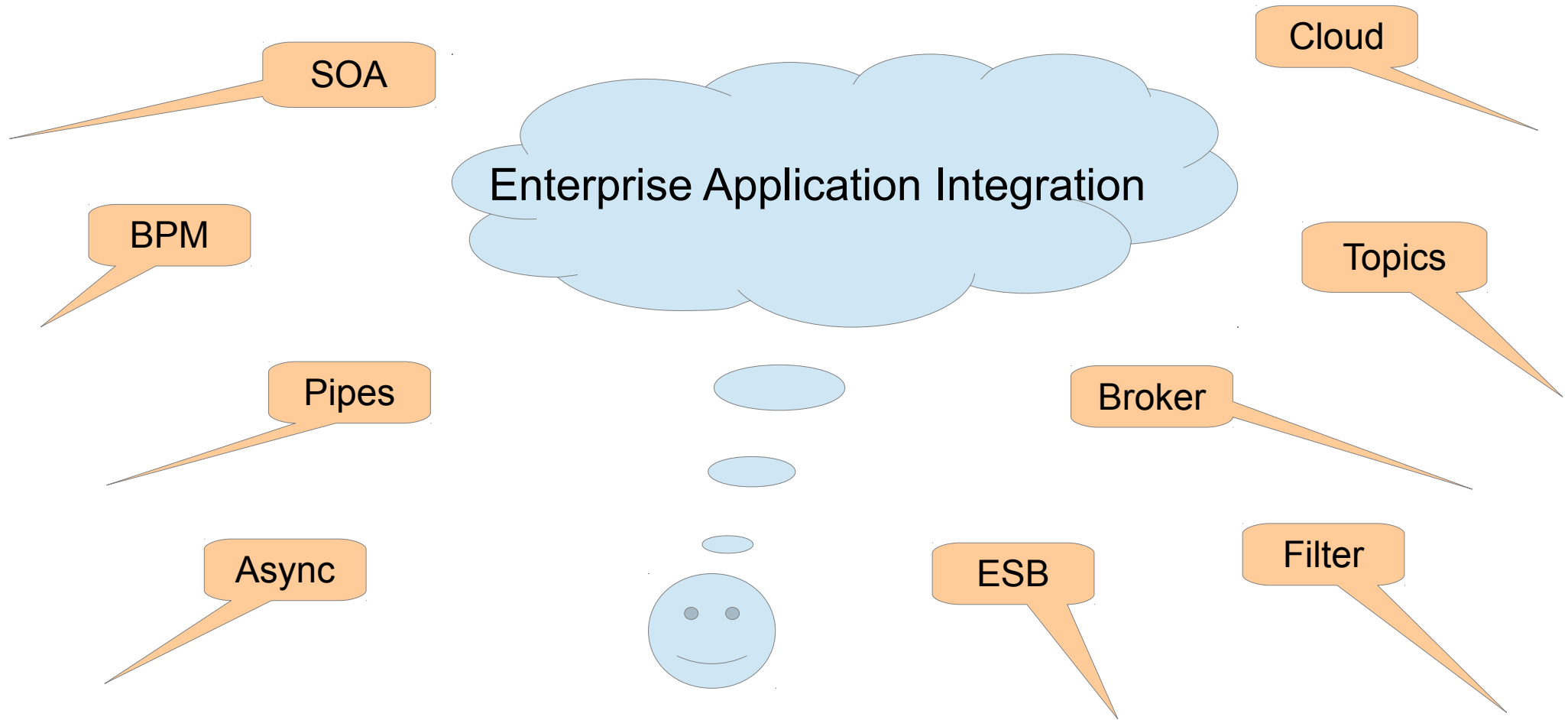
§ [@goldstift](#), alex(at)firstpoint.de

§ Stefan Reuter

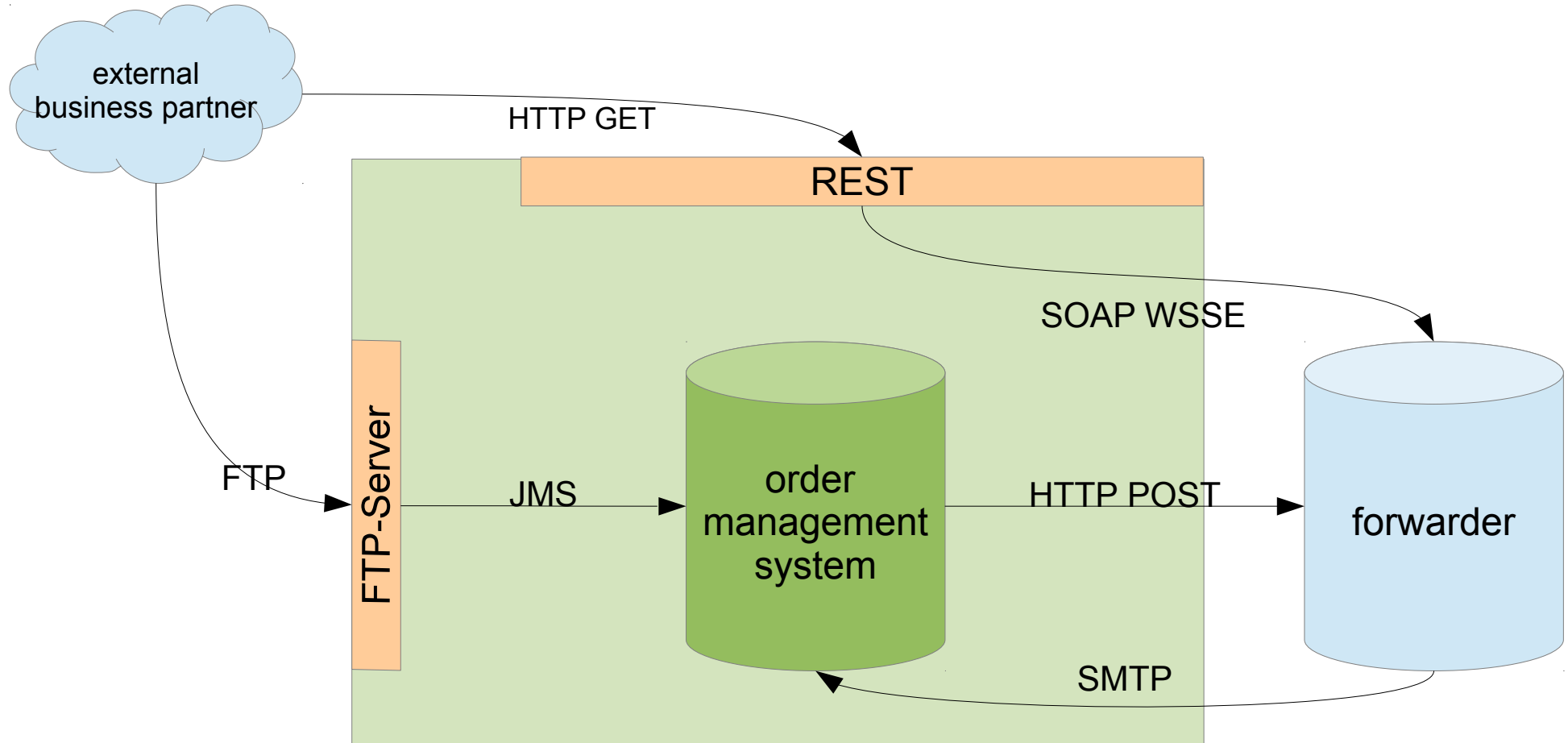
§ Software Architect, Freelancer

§ [@stefanreuter](#), stefan.reuter(at)reucon.com

Handling the buzz



Real life scenario - a logistics service provider



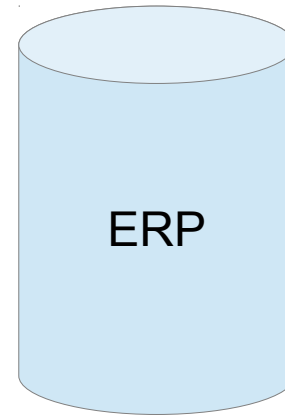
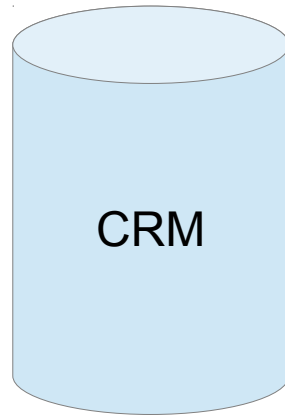
Definition of 'Enterprise Application Integration'

“Enterprise application integration (EAI) is defined as the use of software and computer systems architectural principles to integrate a set of enterprise computer applications.”

- Wikipedia

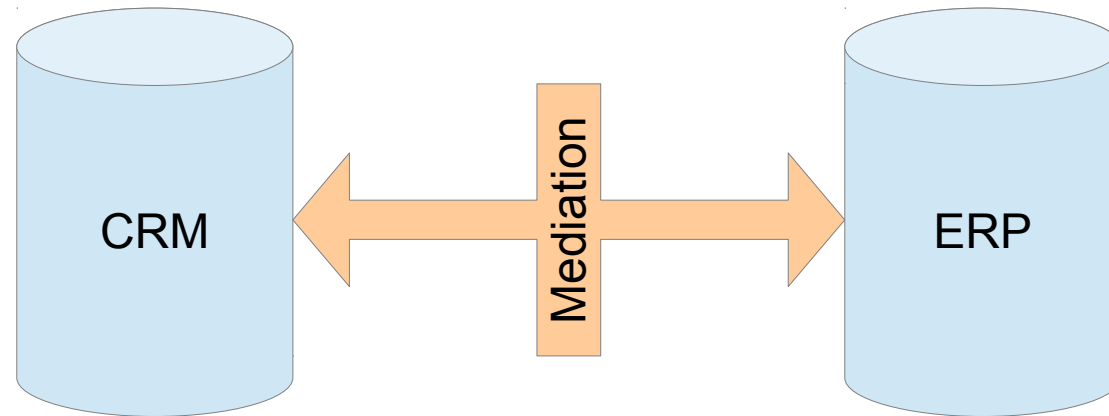
Enterprise Application Integration

is set of “computer applications” a.k.a. “information silos”



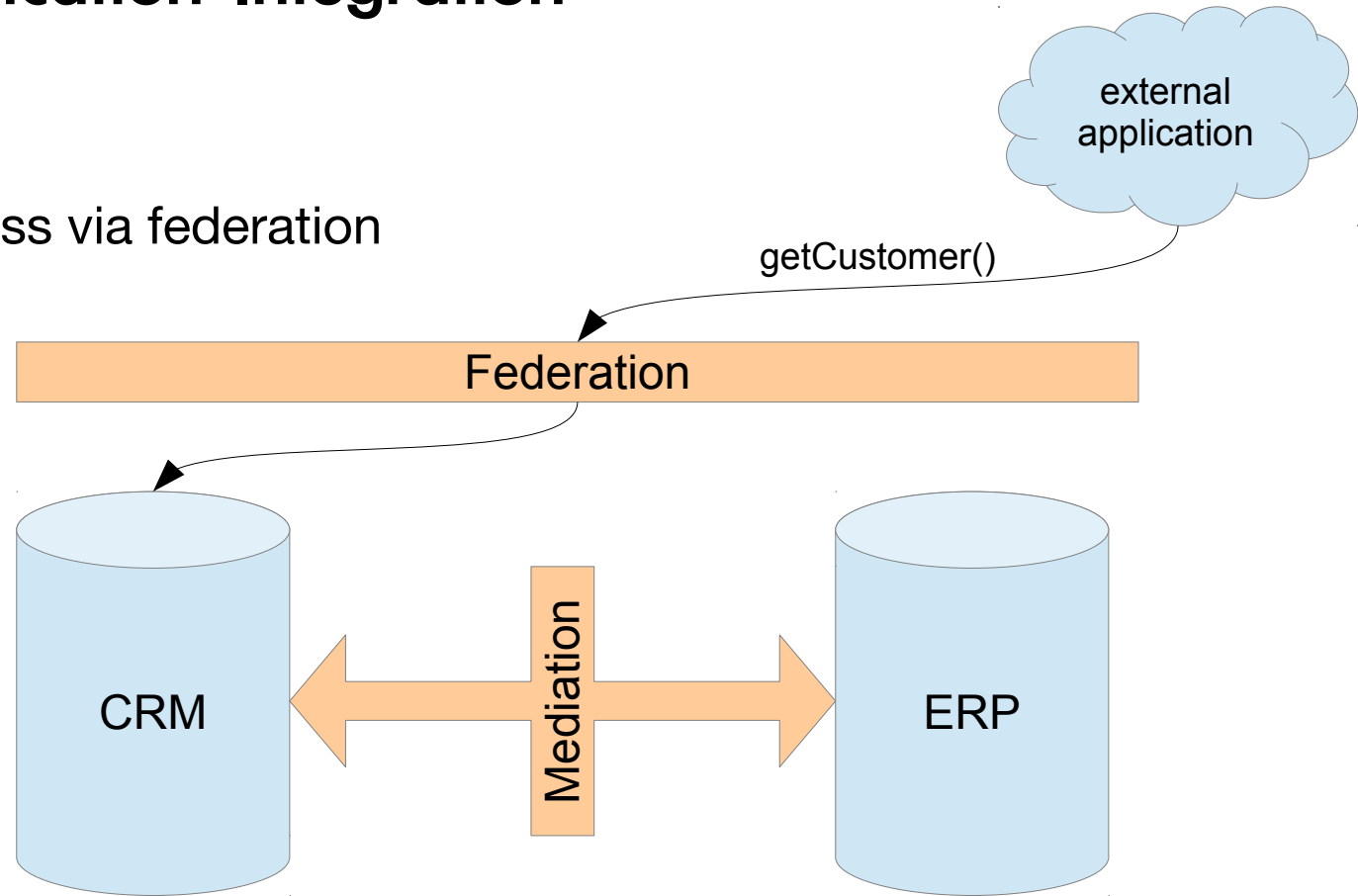
Enterprise Application Integration

≈ linking 'information silos' via mediation

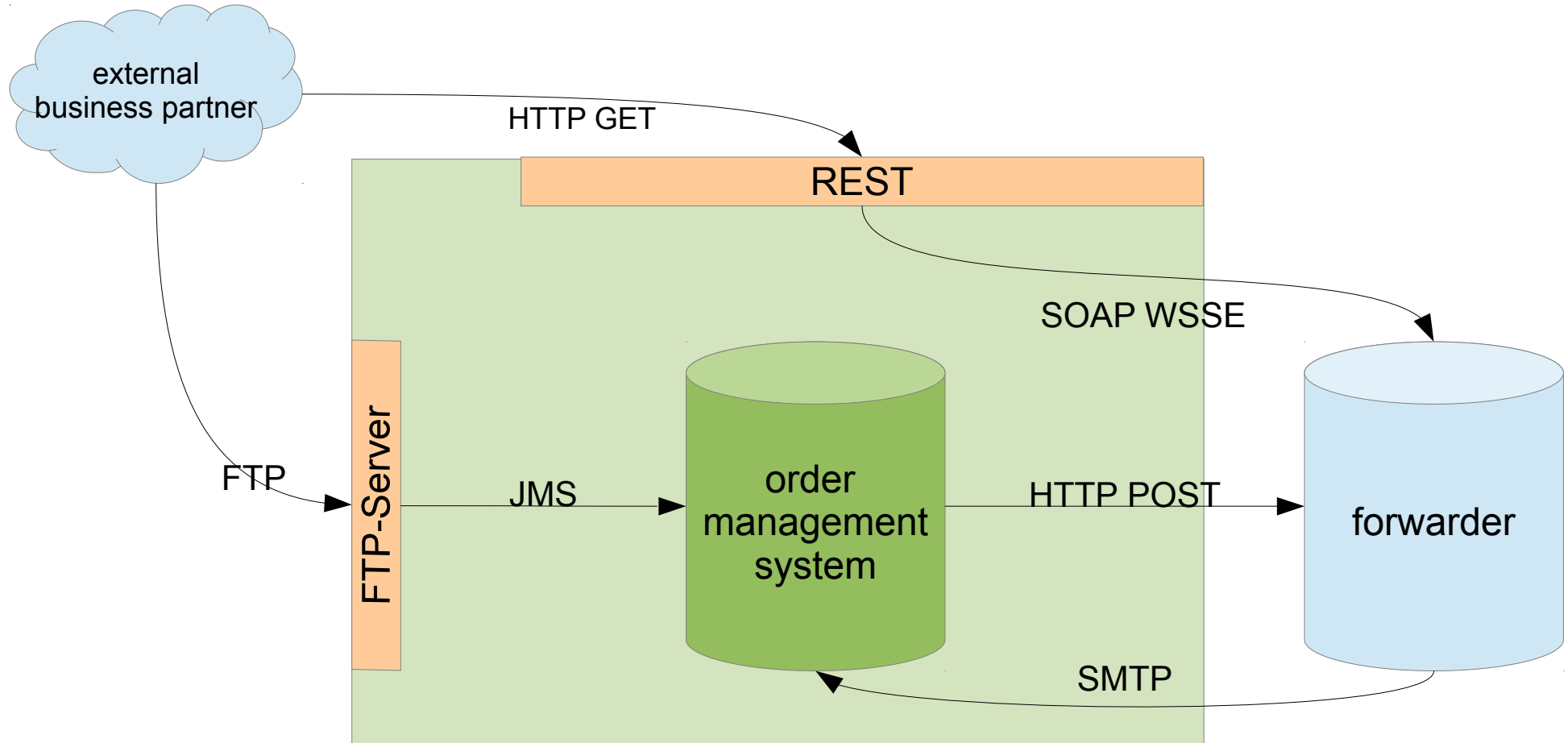


Enterprise Application Integration

is providing access via federation



Real life scenario - a logistics service provider



Enterprise Application Integration

- § linking 'information silos' via
 - § mediation – explicitly connecting multiple applications
 - § federation – providing access for external applications

“Various technologies have been around (...). We all believe that asynchronous messaging carries the greatest promise.”

- Martin Fowler (Enterprise Integration Patterns, 2003)

EAI in the real world

As every application has a kind of mailbox imagine a ...

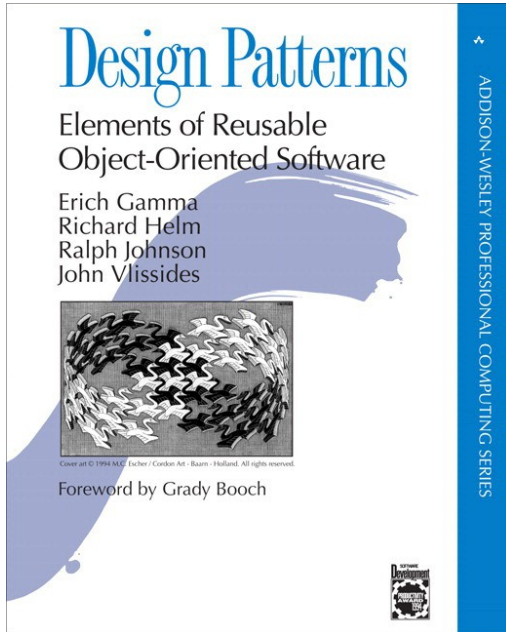
... **reliable** postal service



Benefits of using messaging

- is message-based communication allows decoupling
- is integrate heterogenous platforms/ languages
- is variable timing & throttling – every application works at its pace
- is reliable communication
- is disconnected operation

Patterns applied to EAI

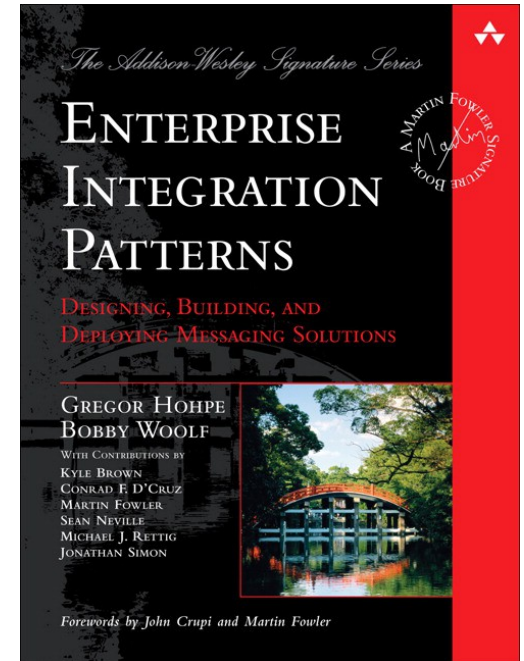


Design Patterns (Gamma et al), 1994

Proven solutions for common problems

Enterprise Integration Patterns (Hohpe & Woolf), 2003

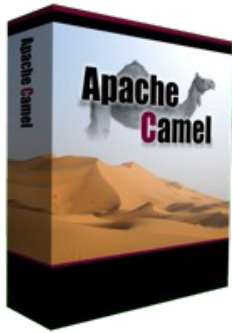
Swiss-army knife for asynchronous messaging



How to use EAI patterns in JavaEE

Multiple approaches possible

- is Do-it-yourself by leveraging the JEE 6 APIs
- is Use a mediation framework



Apache Camel

<http://camel.apache.org/>



Spring Integration

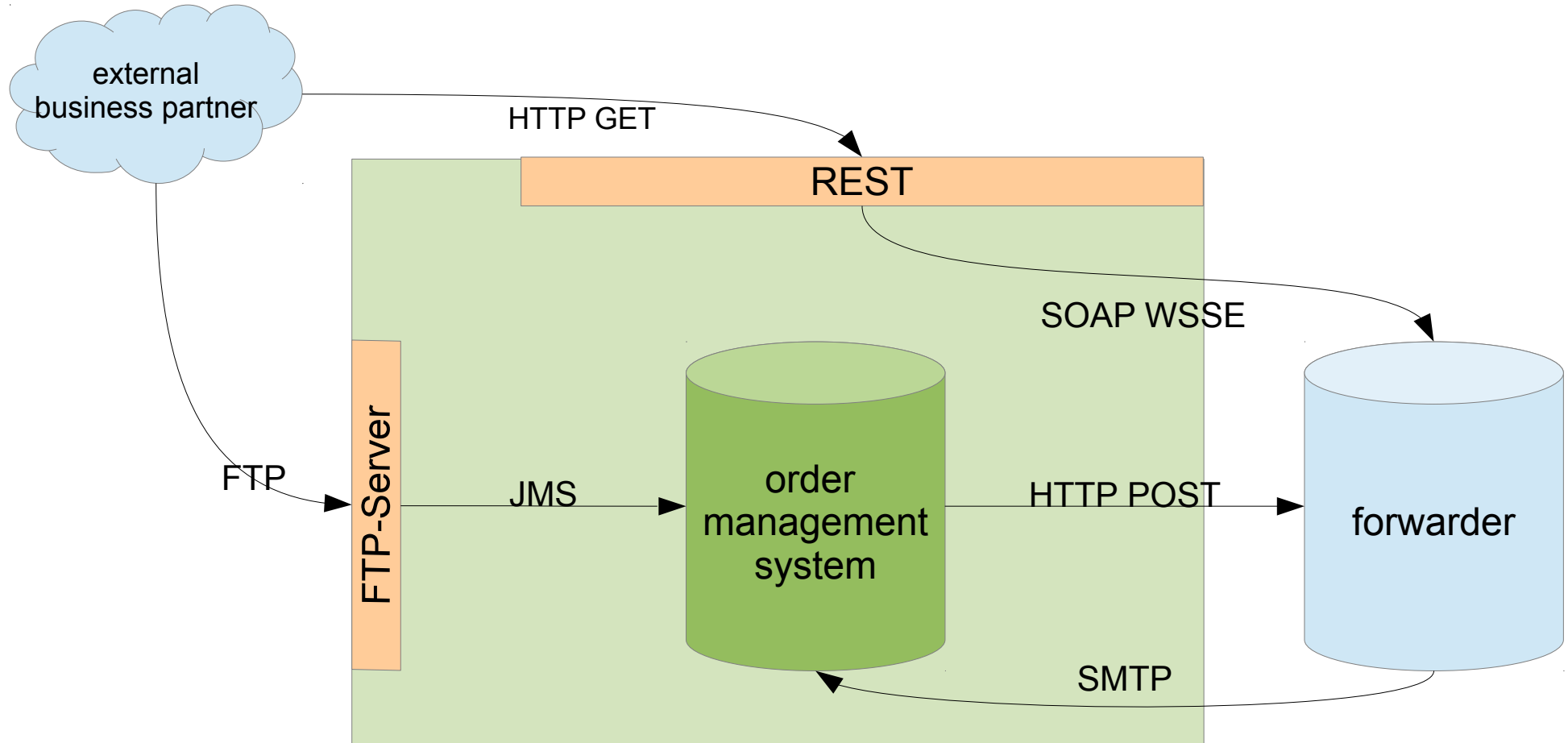
<http://www.springsource.org/spring-integration>

Samples for common EAI patterns

- is adapters
 - is pipes and filters
 - is router
 - is transformer
 - is splitter
 - is aggregator
- ...and many more



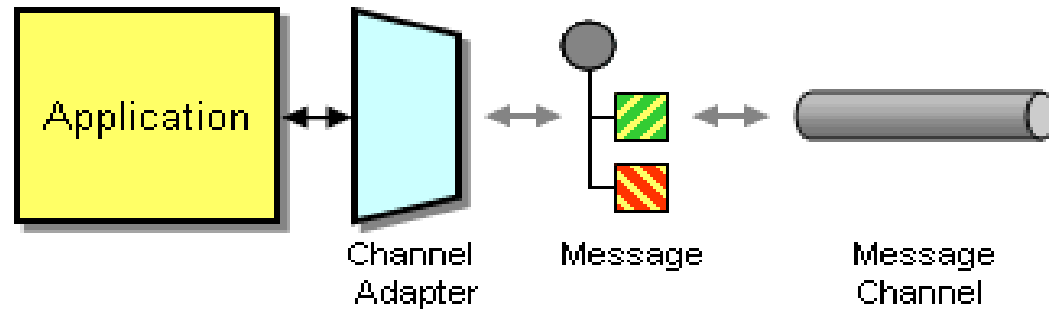
EAI in real life - domain model



EAI Pattern: Inbound- and Outbound-Adapter (Scenario)

- § every application has a specific interface
 - § business partner's system provides data via FTP
 - § forwarder's system only provides SOAP WSSE for tracking data
 - § forwarder's system sends E-Mail notification for successful shipping

EAI Pattern: Inbound- and Outbound-Adapter



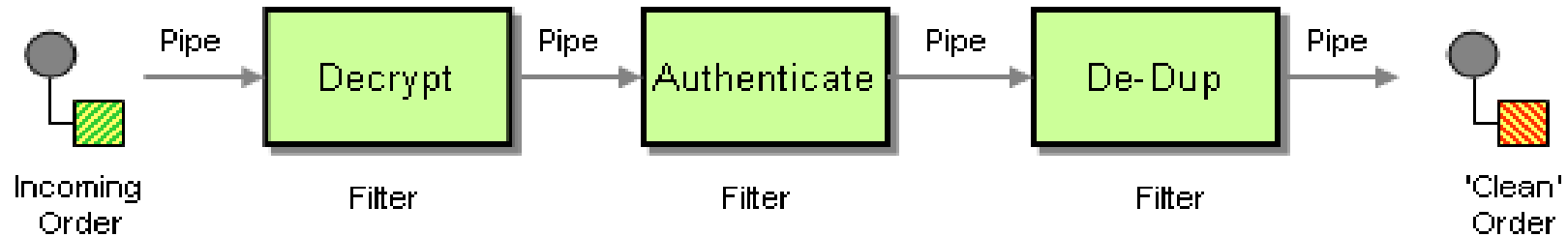
- § adapter = endpoint fitting the specific capabilities of the remote system's API
- § inbound adapter = from application to EAI system
- § outbound adapter = from EAI system to application

<http://www.eaipatterns.com/ChannelAdapter.html>

EAI Pattern: Pipes and Filters (Scenario)

- § decouple processing into simple steps unaware of each other
 - § new purchase order arrives as a message
 - § order is encrypted to insure integrity → decrypt
 - § trusted customer's deserve special treating → authenticate
 - § we don't want duplicated orders → check

EAI Pattern: Pipes and Filters



- § divide complex processing tasks into simple steps
- § easier maintenance, reusable and exchangeable
- § steps (Filters) are connected by channels (Pipes)

<http://www.enterpriseintegrationpatterns.com/PipesAndFilters.html>

EAI Pattern: Gateway Sample

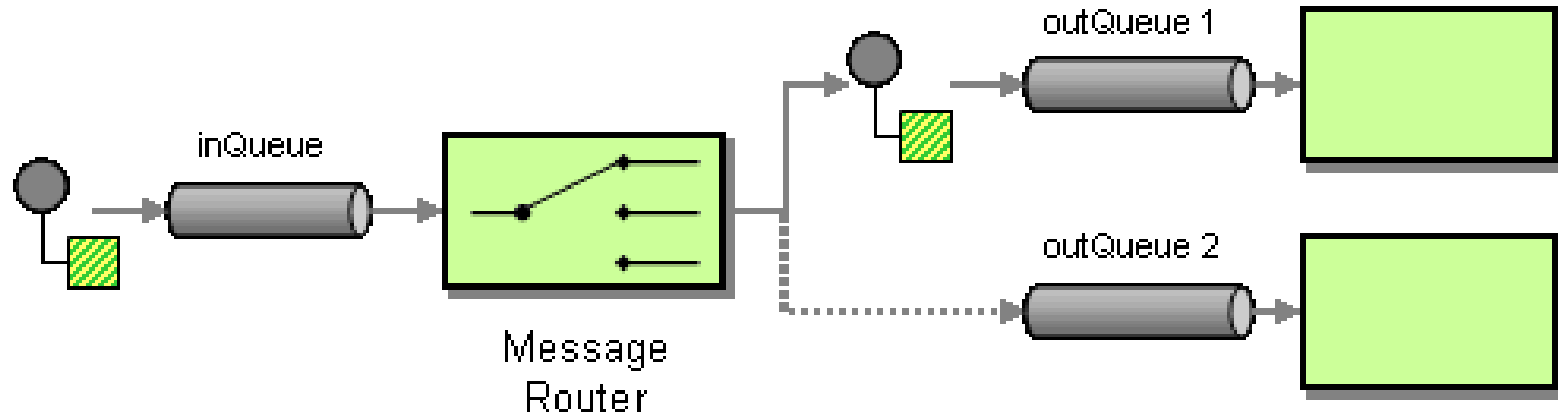
§ Demo

- § configuration of pipes and filters and howto use in your application
- § see “gateway-sample.xml” for details

EAI Pattern: Message Router (Scenario)

- § forward messages depending on conditions
 - § product type: downloadable software cannot be shipped
 - § payment method: credit card transactions vs. direct debit

EAI Pattern: Message Router



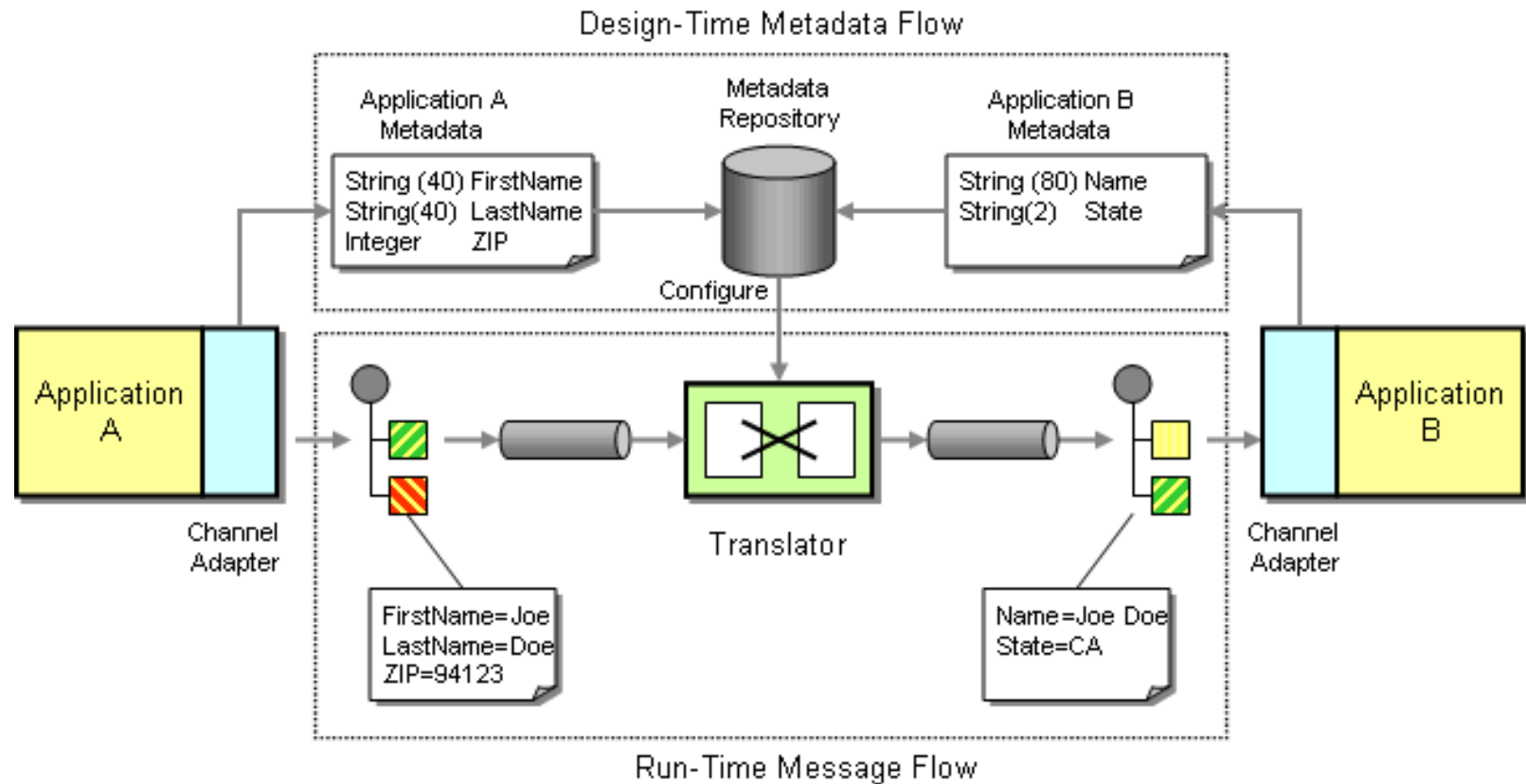
- § a message router forwards but doesn't modify message
- § a message router can have multiple output channels (difference to filter)
- § decoupled: surrounding components are unaware of its existence

<http://www.enterpriseintegrationpatterns.com/MessageRouter.html>

EAI Pattern: Transformer (Scenario)

- § communicating applications have a different data model
 - § customer data from ERP (A) needs to be send to Forwarder (B)
 - § different data model
 - § fields have different semantics
 - § fields have different length

EAI Pattern: Transformer

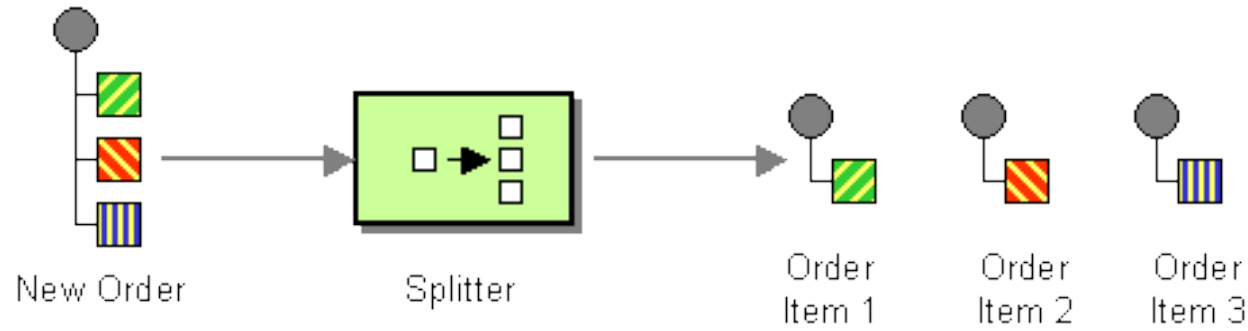


<http://www.enterpriseintegrationpatterns.com/MessageTransformationIntro.html>

EAI Pattern: Splitter (Scenario)

- § message shall be split for further processing
 - § a CSV file shall be imported line by line
 - § warehouse system sends picking information for orders in a large CSV file

EAI Pattern: Splitter



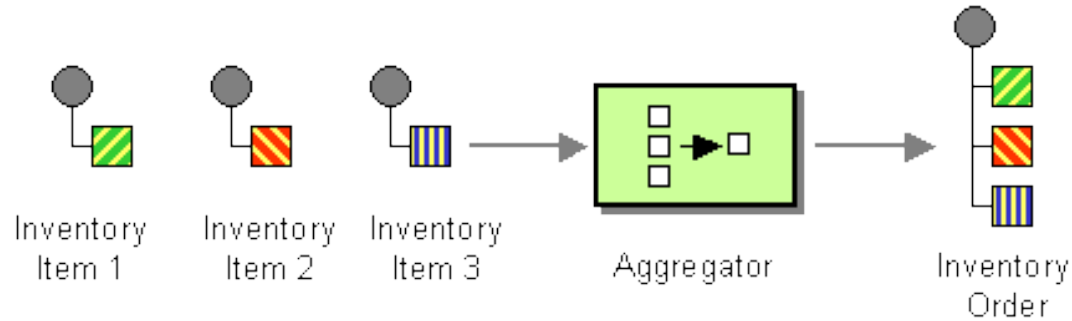
- 2 splitter publishes one message for each item of the original message

<http://www.enterpriseintegrationpatterns.com/Sequencer.html>

EAI Pattern: Aggregator (Scenario)

- § single messages shall be combined for further processing
 - § state of order positions is aggregated to report order state
 - § aggregate order state depending on shipping state of order positions

EAI Pattern: Aggregator



- 2 aggregator collects messages until a set of related messages is complete

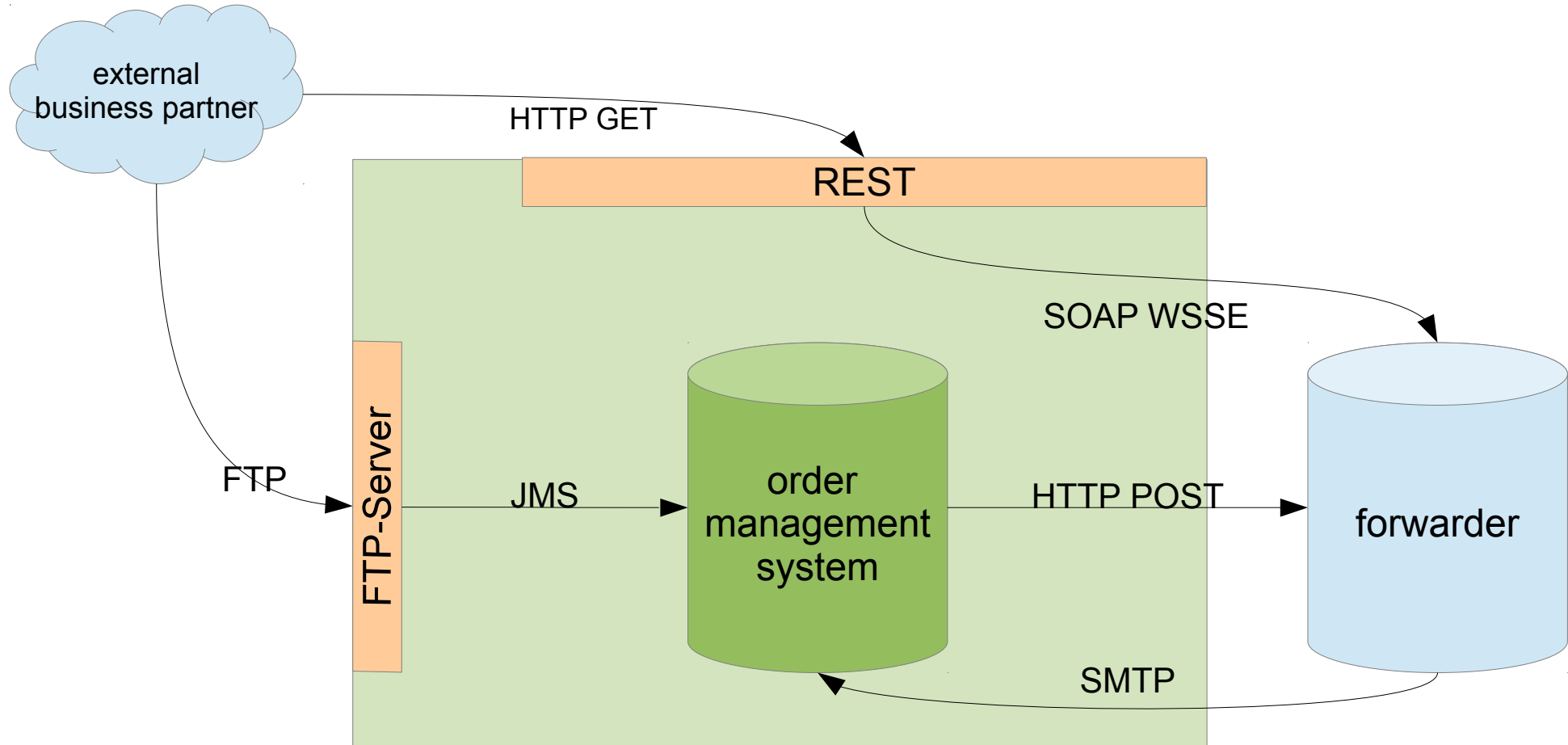
<http://www.enterpriseintegrationpatterns.com/Aggregator.html>

EAI Pattern: Demo

§ Demo

- § scenario: an importer for CSV files with different content
- § configuration of file to string transformer
- § configuration of a router
- § configuration of an expression based splitter
- § see “csv2db-route-context.xml” for details

Real life scenario - the cloud



EAI in the cloud

- § specific environment
- § limited number of I/O gateways
 - § no filesystem → need specific storage adapter
 - § mostly no open ports → no basic TCP to custom ports
- § scalable but unique endpoints are needed
 - § transport mostly via JMS, AMQP or HTTP
- § keep an eye on traffic and network I/O

Q & A

Feel free to

- ask questions now
- contact us on twitter @goldstift & @stefanreuter
- post issues on github:
<https://github.com/aheusingfeld/javaone2012/issues>

Thanks for your attention

If you have any questions afterwards

- ↳ contact us on twitter

- ↳ @goldstift

- ↳ @stefanreuter

- ↳ post issues on github so everyone can benefit:

- <https://github.com/aheusingfeld/javaone2012/issues>



Image copyrights

- USB Power adapter courtesy of ChinBuye Limited. <http://bit.ly/xBKwGw>
- 3newmessages.jpg from <http://www.photooutpost.com>
- Deutsche Post postboxes by <http://bit.ly/PtrhWy>
- EAI pattern graphics are courtesy of Addison Wesley (<http://eaipatterns.com>)
- Photo of Lego bricks by <http://www.sxc.hu/photo/109896>
- Photo of LEGO candles by
<http://www.livbit.com/article/2009/06/11/brighten-it-up-with-colorful-lego-candles/>
- LEGO is a trademark of The Lego Group (<http://aboutus.lego.com/en-us/legal-notice>)