

Implementacja ESB przy użyciu Mule

Michał Majcher

michal.majcher@altkom.pl

Łukasz Krawczyk

lukasz.krawczyk@altkom.pl

slide 1

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

Tematy

- ESB
- Mule
- Obsługa zamówień – DEMO
 - Opis problemu
 - Przepływ komunikatów
 - Implementacja w Mule
 - Komponenty
 - Transport
 - Obsługa wyjątków

slide 2

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

ESB wprowadzenie

- Co to jest?
- Dlaczego i do czego tego potrzebujemy?
- Co z SOA ?
- Co to jest JBI?

slide 3

Altkom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

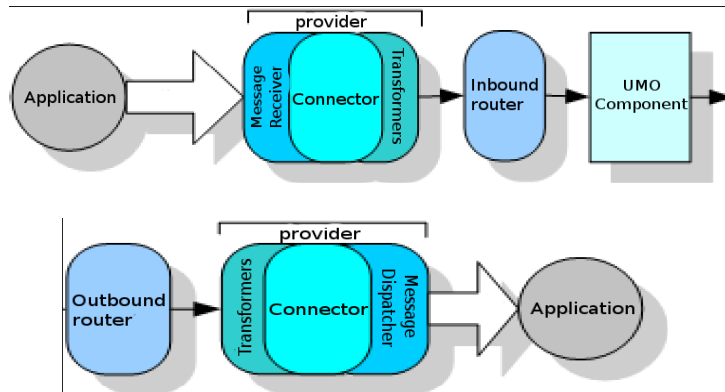
Mule

- Projekt opensource rozpoczęty w 2003
- 10+ deweloperów stale go rozwijających (Mule 2.0)
- Używany przez duże firmy takie jak HP, Sony, Citibank ...
- Dostępne komercyjne wsparcie

slide 4

Altkom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

Mule architektura, podstawowe komponenty



slide 5

Altikom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altikom.pl>, tel. +48 (22) 860 6060

Obsługa zamówień

DEMO

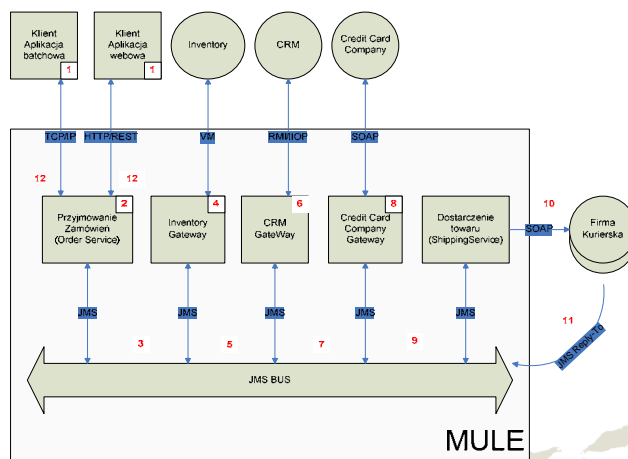
Obsługa zamówień założenia

- Aplikacja wspiera przetwarzanie request/response, klient zamawiając towar „od razu” dowiadyuje się o możliwości jego zrealizowania
- Możliwość przetwarzania dużej ilości zamówień, łatwa skalowalność
- Komponenty komunikują się między sobą poprzez JMSa
- Pobieramy dane z zewnętrznych systemów używając do tego różnych protokołów takich jak JMS, SOAP, HTTP, RMI/IIOP

slide 7

Altikom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altikom.pl>, tel. +48 (22) 860 6060

Obsługa zamówień, diagram



slide 8

Altikom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altikom.pl>, tel. +48 (22) 860 6060

Komponenty opis

Klient aplikacja webowa/batch	Aplikacja umożliwiająca prezentującą klientowi listy produktów i umożliwiającą składanie na nie zamówień.
OrderService	Obiera składane zamówienia, rozpoczyna przetwarzania, zwraca wyniki klientowi.
InventoryGateway	Sprawdza dostępność oraz pobiera dodatkowe informacje o produkcie.
Inventory	Zewnętrzna aplikacja obsługująca magazyn.
CRMGateway	Pobiera dodatkowe informacje o kliencie składającym zamówienie.
CRM	Zewnętrzna aplikacja zawierająca bazę klientów.
CreditCardCompanyGateway	Autoryzuje transakcję
Credit Card Company	Zewnętrzny system umożliwiający autoryzację transakcji
ShippingService	Wybiera sposób dostarczenia towaru.
Firmy kurierskie	Zewnętrzny system umożliwiający sprawdzenia kosztu i czasu dostarczenie towaru

slide 9

Altkom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

Przepływ komunikatów

1. Klient składa zamówienia wybierając identyfikator i ilość towaru.
2. Zaczynamy przetwarzanie zamówienia tworzymy komunikat z odebranymi informacjami, dodając dodatkowo identyfikator klienta.
3. Wysyłamy komunikat do magazynu.
4. Pobieramy szczegółowe informacje o produkcie (cenę, nazwę i typ produktu) i sprawdzamy jego dostępność, odebrane informacje doklejamy do komunikatu.
5. Wysyłamy komunikat do CRM.
6. Na podstawie identyfikatora klienta pobieramy dodatkowe informacje o kliencie takie jak jego adres, numer karty kredytowej i przysługujący mu rabat. Odebrane informacje ponownie doklejamy do komunikatu
7. Wysyłamy komunikat do firmy obsługującej karty kredytowe.
8. Znając numer karty kredytowej i kwotę na którą klient złożył zamówienie prosimy o autoryzację transakcji.
9. Przesyłamy komunikat aby wybrać sposób dostarczenia przesyłki.
10. Na podstawie typu produktu wybieramy możliwe sposoby dostarczenia i odpytujemy firmy kurierski o datę dostarczenia i koszt takiej usługi
11. Odpowiedzi zawierające datę i koszt dostarczenia towaru przesyłamy do OrderService.
12. Przesłanie i zaprezentowanie wyników klientowi.

slide 10

Altkom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

Obsługa zamówienia

Implementacja w Mule

Komunikat

- OrderRequest to podstawowy obiekt wymieniany pomiędzy wszystkimi komponentami w celu realizacji zamówienia. W tym przykładzie jest to zwykły JavaBean normalnie powinien to być XML.

```
public class OrderRequest implements Serializable {  
    /** ilość */  
    private int amount;  
    /** dane klienta */  
    private CustomerData;  
    /** dane zamawianego produktu */  
    private ProductData;  
    /** różne oferty dostarczenia */  
    private List shipOffers;  
    .....  
}
```

OrderService

- Odbiera identyfikator klienta, produktu oraz jego ilość następnie tworzy obiekt OrderRequest i wysyła go do skonfigurowanej kolejki rozpoczynając realizację zamówienia. Zbiera również wyniki przetwarzania i prezentuje je klientowi.
- Konfiguracja

```
<mule-descriptor name=„OrderService“
  implementation=„pl.altkom.order.OrderService“>
  <inbound-router>
    <endpoint address=„jetty:rest://localhost:8080/orderService,
      transformers=„RestRequestToOrderRequest“/>
    <endpoint address=„vm://order.service.request“/>
  </inbound-router>
  <outbound-router>
    <router className=„org.mule.routing.outbound.OutboundPassThroughRouter“>
      <endpoint address=„jms://esb.order.inventory “/>
    </router>
  </outbound-router>
  <response-router timeout=„1000000“>
    <endpoint address=„jms://esb.order.confirm“/>
    <router className=„pl.altkom.esb.routers.OrderResponseAggregator“/>
  </response-router>
</mule-descriptor>
```

slide 13

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

Inventory gateway

- Sprawdza czy istnieje zamawiany produkt w podanej ilości oraz pobiera dodatkowe informacje o towarze. Wynik ProductData zawierający identyfikator, nazwę, rodzaj i cenę produktu dołącza do OrderRequest i przesyła dalej.
- Konfiguracja

```
<mule-descriptor name=„InventoryGateway“
  implementation=„org.mule.components.simple.BridgeComponent.“>
  <inbound-router>
    <endpoint address=„jms://esb.order.inventory“/>
  </inbound-router>
  <outbound-router>
    <router className=„org.mule.routing.outbound.OutboundPassThroughRouter“>
      <endpoint remoteSync=„true“ address=„vm://order.inventory“/>
      <endpoint address=„jms://esb.order.crm“/>
    </router>
  </outbound-router>
</mule-descriptor>
```

slide 14

AltKom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altkom.pl>, tel. +48 (22) 860 6060

CRM gateway

- Na podstawie odebranego identyfikatora pobiera z bazy klientów informacje o nim. Interesują nas adres, numer karty kredytowej i wysokość rabatu. Wynikiem jest obiekt CustomerData który doklejamy do zamówienia i przesyłamy do dalszego przetwarzania

- Konfiguracja

```
<mule-descriptor name=„CrmGateway“
  implementation=„org.mule.components.builder.ReflectionMessageBuilder“>
  <inbound-router>
    <endpoint address=„jms://esb.order.crm“/>
  </inbound-router>

  <outbound-router>
    <router className=„org.mule.routing.outbound.FilteringOutboundRouter“>
      <endpoint remoteSync=„true“, transformers=„OrderRequestToCustomerDataArgs,
        responseTransformers=„CustomerDataXmlToCustomerData,
        address=„ejb://localhost:1099/local/crm?method=getCustomerData“/>
      <endpoint address=„jms://esb.order.ccc“/>
    </router>
  </outbound-router>
</mule-descriptor>
```

slide 15

Altikom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altikom.pl>, tel. +48 (22) 860 6060

Credit Card Company gateway

- Mając już kwotę zamówienia i numer karty kredytowej wysyłamy prośbę o autoryzację transakcji.

- Konfiguracja

```
<mule-descriptor name=„CCAGateway“
  implementation=„org.mule.components.simple.BridgeComponent“>

  <inbound-router>
    <endpoint address=„jms://esb.order.ccc“/>
  </inbound-router>

  <outbound-router>
    <router className=„org.mule.routing.outbound.OutboundPassThroughRouter“>
      <endpoint remoteSync=„true“ address=„axis:http://localhost:10080/mule“/>
      <endpoint address=„jms://esb.order.ship“/>
    </router>
  </outbound-router>
</mule-descriptor>
```

slide 16

Altikom Akademia S.A., ul. Stawki 2, 00-193 Warszawa, Poland
<http://www.altikom.pl>, tel. +48 (22) 860 6060

- W zależności do rodzaju produktu wybieramy możliwe sposoby dostarczenia. Następnie ustalamy listę firm do których wyślemy zapytanie o termin dostarczenia i cenę takiej usługi. Wyniki poszczególnych zapytań trafią bezpośrednio do OrderService w celu przetworzenia i zaprezentowania klientowi.
- Konfiguracja

```
<mule-descriptor name=„ShippingService“  
  implementation=„pl.altkom.order.ShippingService“>  
  <inbound-router>  
    <endpoint address=„jms://esb.order.ship“/>  
  </inbound-router>  
  
  <outbound-router>  
    <router className=“org.mule.routing.outbound.StaticRecipientList“>  
      <reply-to address=„jms://esb.order.confirm“/>  
    </router>  
  </outbound-router>  
</mule-descriptor>
```

- Obsługa wyjątków
 - poprzez definiowanie strategii na konektorach bądź komponentach
 - różne typy wyjątków możemy obsługiwać w inny sposób
 - możemy zmieniać routing komunikatów
- Obsługa błędów połączeń
 - kontrolujemy jak podłączamy się do zasobów w elastyczny sposób możemy definiować ponowne podłączenie
 - powiadomienia przy nieudanych połączeniach
 - możemy zmienić routing komunikatu na podstawie nieudanych połączeń

Dalsze informacje

- Mule
<http://www.muleumo.org>
- EIP Book (Enterprise Integration Patterns)
<http://www.eaipatterns.com>
- ESB
http://searchwebservices.techtarget.com/general/1,295582,sid26_gci1085711,00.html
- SEDA
<http://www.eecs.harvard.edu/~mdw/papers/mdw-phdthesis.pdf>

Pytania

Dziękujemy za uwagę

Michał Majcher

michal.majcher@alkom.pl

Łukasz Krawczyk

lukasz.krawczyk@alkom.pl

|