



Networking for Communications Challenged Communities:  
Architecture, Test Beds and Innovative Alliances  
Grant Agreement: 223994

# System Integration Workshop

Coimbra, December 10-11, 2008

**N4C integration: an ESB perspective**

**Krzysztof Romanowski – ITTI**

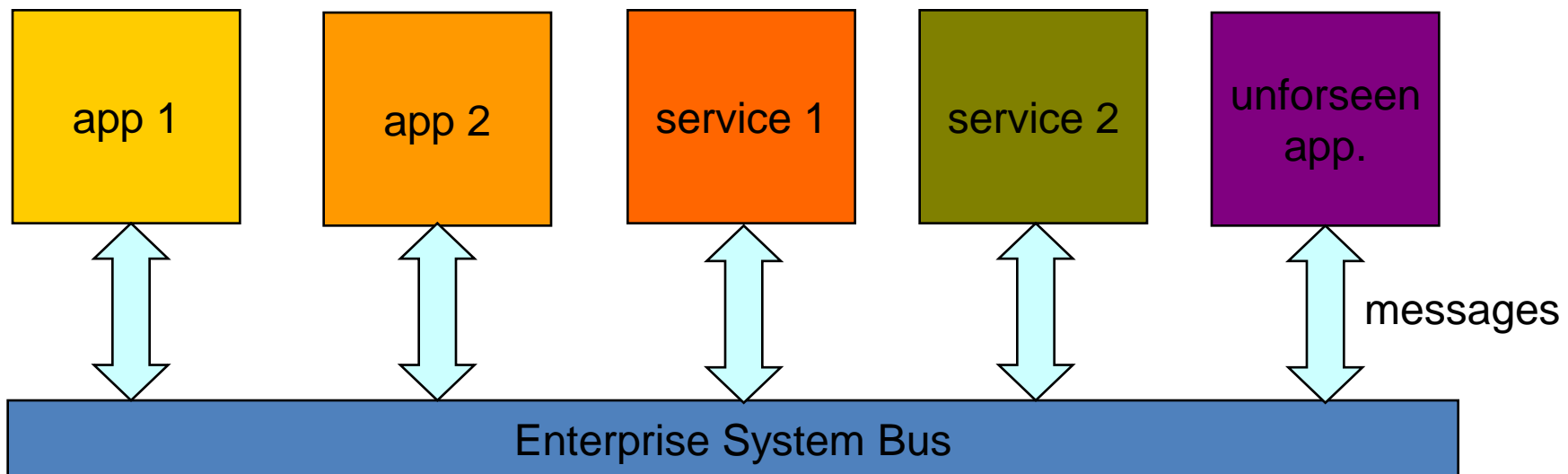
**[Krzysztof.Romanowski@itti.com.pl](mailto:Krzysztof.Romanowski@itti.com.pl)**

**<http://www.itti.com.pl>**



- WP7: System integration:
  - individual applications – mostly different usage domains, differences in regions, time, ...
    - hiker's PDA
    - reindeer tracking
    - meteo
  - common functionality:
    - middle layers:
      - transport
      - routing
      - management
      - ...

- What is in the middle?
  - DTN bundle layer
  - LTP transport protocol
  - PROPHET routing protocol
  - caching
  - storage
  - communication opportunity management
  - security
  - ...



- ESB functions:
  - invocation
  - message routing
  - mediation (transformations)
  - event processing
  - QoS (transactions, reliability, security)
  - ‘choreography’, ‘orchestration’ (implementation of complex processes, coordination and aggregation of multiple services)
  - management

- Useful features:
  - distributed
  - possibility of central management
  - extensible
  - standard-based
  - supports various communication patterns:
    - synchronous, asynchronous
    - one-way, two-way
  - supports various 'transport' protocols: file, FTP, HTTP, JMS, JDBC ...

- ESB is listed in WP7 as *possible* integration platform
  - makes sense only if there are *services* to be delivered over the bus
  - Is the *service* abstraction appropriate?
  - Are there common services?
  - What are they – see doc „M7.1”; apps list:
    - goal
    - initialization
    - assumptions and restrictions
    - inputs
    - outputs
    - test case pass/fail criteria
- which can be abstracted out as service?

- E.g. *System and Management Applications*:
  - awareness of DTN capability and available communication modes, location awareness
  - communication channel control
  - message storage and forwarding
  - notifications
  - security (authentication, authorization, integrity, confidentiality, non-repudiability)
  - addressing
  - routing control
  - time reference and stamping

- candidates for common services

## ■ Open source ESBs:

### ■ Infonatural ESB

<http://sourceforge.net/projects/infonatural-esb>

- MIT license
- no longer developed

### • Celtix Java ESB

<http://celtix.objectweb.org/>

- EPL (Eclipse Public License)
- active (v.1.0, 2006)

### • Mule ESB

<http://mule.mulesource.org>

- Common Public Attribution License
- active (v.2.1.2, December 2008)
- also paid supported Enterprise Edition

## ■ Open source ESBs (cont.):

### ■ Red Hat JBossESB

<http://www.jboss.org/jbossesb/>

- LGPL
- active (v.4.4, August 2008)
- also a paid supported version – a larger product: JBoss Enterprise SOA Platform

### • Sun OpenESB (GlassFish ESB)

<https://open-esb.dev.java.net>

- CDDL (Common Development and Distribution License)
- active (RC1, December 2008)

- Commercial ESBs are available as well. e.g.
  - Sonic ESB
  - BEA AquaLogic Service Bus
  - IBM WebSphere ESB
  - Oracle Fusion Middleware
  - Tibco ActiveMatrix Service Bus
  - Fiorano ESB
  - Cape Clear ESB
  - Software AG webMethods ESB
  - *[ Microsoft (BizTalk Server + Windows Communication Foundation)*  
*]*
  - ...

- Summary questions:
  - can the application needs be described in terms of common services?
  - can the functionality of the existing middleware layers (DTN) be exposed as common services over an ESB?

Thank you for your attention

**N4C speaker: Krzysztof Romanowski**

**Speakers organisation: ITTI**

**Speakers e-mail address: [Krzysztof.Romanowski@itti.com.pl](mailto:Krzysztof.Romanowski@itti.com.pl)**

**Speakers web page: [www.itti.com.pl](http://www.itti.com.pl)**