

OM2

and IESB

Roland Hedberg

<[roland.hedberg@adm.ur](mailto:roland.hedberg@adm.ur)

# OM2-The proje

<http://www.openmetadir.org/>

## Members

✦ SWAMI

✦ LiU, SU, UmU, UU

## Implementations

✦ Python

# What's ESB?

according to Wikipedia

ESB is an event-driven and standards-based messaging engine (the

ESB should be standards-based and flexible, supporting many tra

ESB allows integration architects to exploit the value of messagin

ESB **does not** implement a service-oriented architecture (SOA)

# What's OM2?

according to me

OM2 is an event-driven and standards-based messaging engine (1

OM2 is standards-based and flexible, and supports many transpoc

OM2 allows integration architects to exploit the value of messagi  
*of* code.

OM2 **does not** implement a service-oriented architecture (SOA

# Other nice features

Message syntax: RDF/XML

Ontology driven application development

Operations ontology: MIRO

Support for Dynamic Delegation Discovery  
(DDDS, RFC 3401-3)

# Eduroam

“eduroam is a RADIUS proxy server-based infrastructure that uses 802.1X standard to allow eduroam-enabled user to get network access at any institution connected to eduroam.”

<http://www.eduroam.org/>

# Euroam

## Information gap

### Demographics

\* <http://www.eduroam.org/?p=europe>

\* <http://www.eduroam.se/>

\* <http://www.eduroam.fr/en/>

# Eduroam

What a user wants to

Who provides eduroam services

How do one get access (SSID,..) ?

What do I get access to ?

Where (latitude,longitude) ?

# The Bologna Process

Higher education systems in European countries should be organized in such a way that:

- it is easy (for a student) to move from one country to another (within the European Higher Education Area)

# The Bologna Process

Two functions that needs information

- ⌘ Admission control
- ⌘ Results reporting

# Common patterns

## Dynamic membership

- Membership based on providing a service

## Loosely coupled

- Event based

## Application specific data

# Common problems

How to find the recipient

You have to handle version mismatch

You can not use a shoehorn and cram different applications information content into the

You must be able to represent operations

# Problems that we can

Transport the information

- ✦ to the correct address at all time by the use of DDDS
- ✦ by the transport protocol of the receivers choice

Provide a common syntax, while letting to  
define their own objects and operations c

Handle version control