



« Service Integration Goes Social »

Java Developer Track – Sept. 23rd – Open World Forum 2011



Marc Dutoo

- Head of R&D at Open Wide
- EasySOA project leader



Alain Boulze

- EasiFab Partner & Manager
- 25 years experience in IT for business



Cédric Carbone

- Talend CTO
- OW2 & Eclipse Board member



- I. EasySOA in a nutshell**
- II. Why collaborative (platform) ?**
 - Reaching out to business users
 - Reaching out to developers & IT staff
- III. EasySOA-aware development / EasiFab**
- IV. EasySOA-compatible ESB / Talend**
- V. Roadmap - Questions**

EasySOA Facts

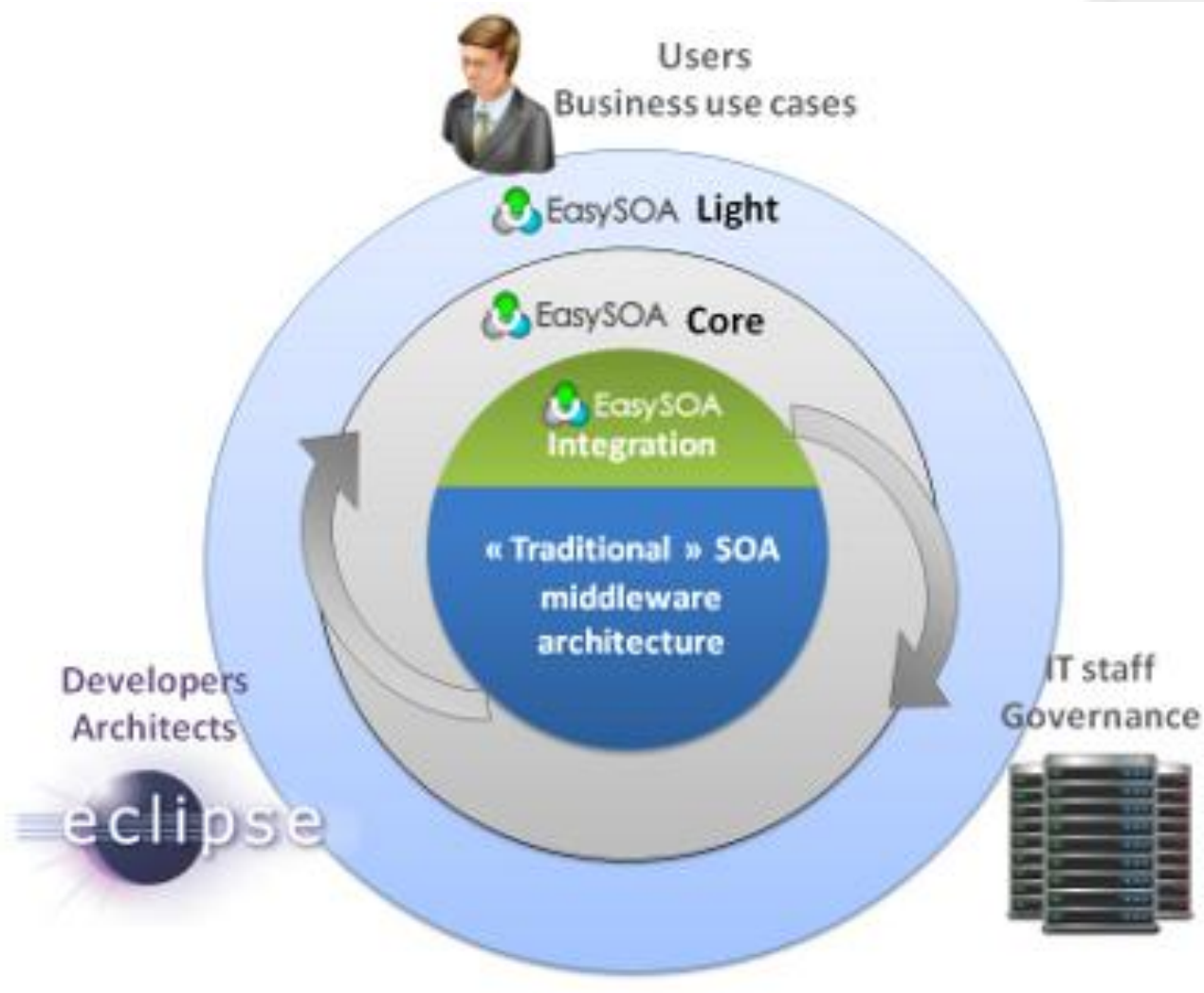
- 5 partners
- 2 years, started nov. 2010
- 4m€ budget
- System@tic label
- And an ambitious aim...



EasySOA

Making Service Oriented Architectures (SOA) simple to use

- Business use, development, production use, monitoring
- And **throttling up** the SOA engine in the enterprise !



- Add a lighter, agile SOA layer around “traditional” SOA
 - thanks to an online, social and collaborative approach, involving all actors of the SOA process
 - business users, SOA architects and developers, IT staff
 - Enabling
 - ex nihilo service **discovery, cartography and documentation**, all collaboratively
 - **sanitization** and protection of existing SOAs by tracking changes of outside services
 - help gathering and **fast-prototyping** business needs on top of existing applications, without hurting them
 - **reuse** requirements, architectural shell, tests and mockups to ease the transition to final implementations within the existing SOA solution

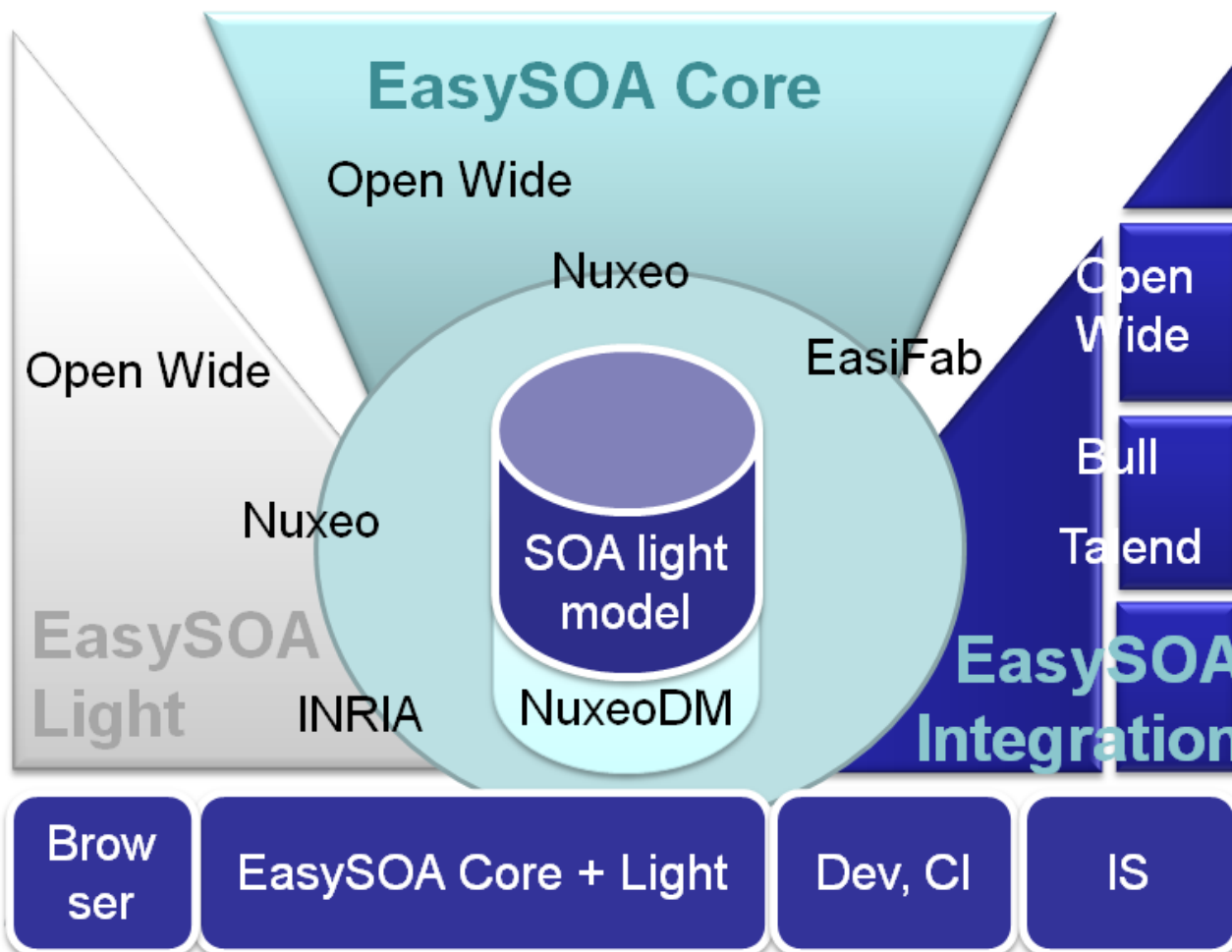
Behind, French partners but also worldwide leaders

- **INRIA labs** : service engine (OW2 FraSCAti)
- **EasiFab** : SOA Modeling (Eclipse SOA), monitoring (Galaxy)
- **Talend** (ETL/ESB) : SOA and data connectors to connect to existing business – but also Data Quality and MDM
- **Nuxeo** (ECM) : document management platform, to manage the SOA model, documents and artifacts
- **Bull** (service provider and middleware) : SOA administration with OW2 Jasmine and use case
- **Open Wide** : leader, global architecture and integration, BPM (with Eclipse JWT / OW2 Scarbo), use case

EasySOA – Architecture

EasySOA Core :
SOA collaborative management
and governance

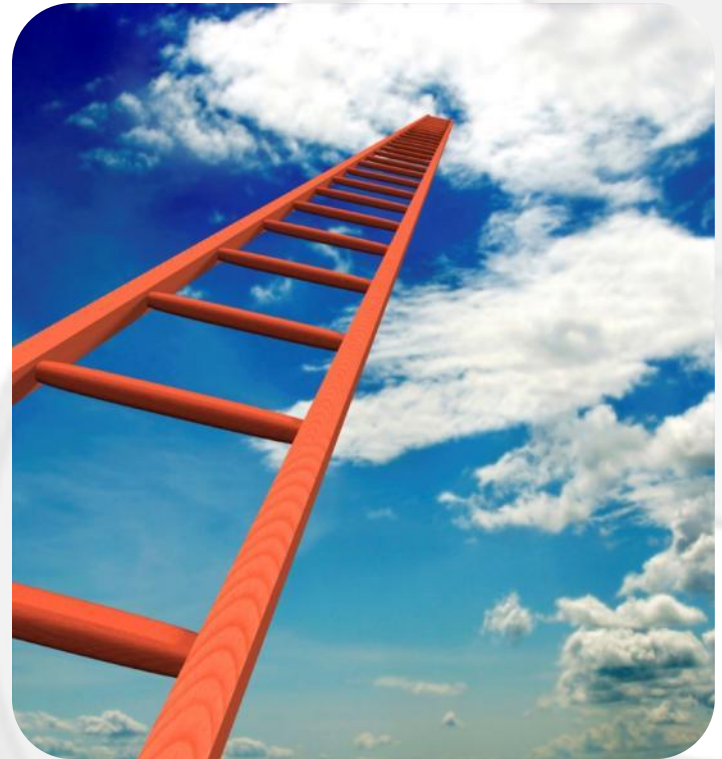
**EasySOA
Light :**
Scripted or
Point-and-
Click rapid
application
development



**EasySOA
Integration :**
Ecosystem of
compatible
SOA solutions
and
compatibility
extensions

Scaling up SOA to IT challenges

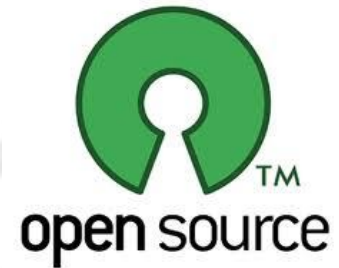
- New IT challenges require an efficient enterprise SOA process more than ever
 - Cloud
 - agility,
 - business relevant IT,
 - mobile,
 - green IT
- ...how to scale up ?



How ?

– We bet on :

- **open** source & open approach
 - For Information System heterogeneity, against vendor lock-in
- the **web**
 - The main drive of computer science democratization for the last 10 years
 - to get closer to end users - a.k.a. "normal" people
- **Collaborative...**



Why collaborative, isn't SOA an architect thing ?

- SOA = how to avoid the integration "noodle plate" problem at protocol and data format levels
- with SOA, integrating a new application with existing ones becomes making it talk with the right
 - protocol(s) : “connecting the lanes” (binding)
 - & data format : “translating the vocabulary” (mapping)
 - This requires **business knowledge** !
- Actually, architects and developers already need business users when changing features
 - To describe requirements, and ~~sometimes~~ often to make decisions on the functional perimeter (cost-driven...)



so we need them in an soa development process

– (this need also justifies business architecture)

but how and where to get them ?

– Rather than a formal, constrained, heavy process, provide the tools for **informal, social collaboration**


– Principle : to each his own...

- Knowledge - point of view - tools

– so look for business users at the place of their business : their **business applications** !

- 5' demo : service discovery – business user



 EasySOA Core

EasySOA > Core > Discovery by browsing
Service discovery by browsing

Logged as

http://localhost:8083/intranet

Export WSDL to the Service Registry

Selected WSDL

None.


Service details




Service Name:

Application:

Save service

Pure Air Flowers ~ Intranet



-  **Main page**
-  **CRM Application**
-  **Trip services**

Smart Travel

> Trip advisor

Your destination country

Phrase to translate

Amount of money in your budget

Found descriptors

- ☐ Create Summary
<http://localhost:9080/CreateSummary?wsdl>
- ☒ Airport information
<http://localhost:8200/esb/AirportService?wsdl>

Getting in also the other actors, just the same :

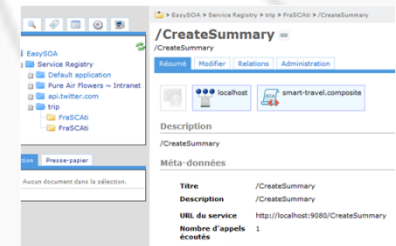
- Architects & developers

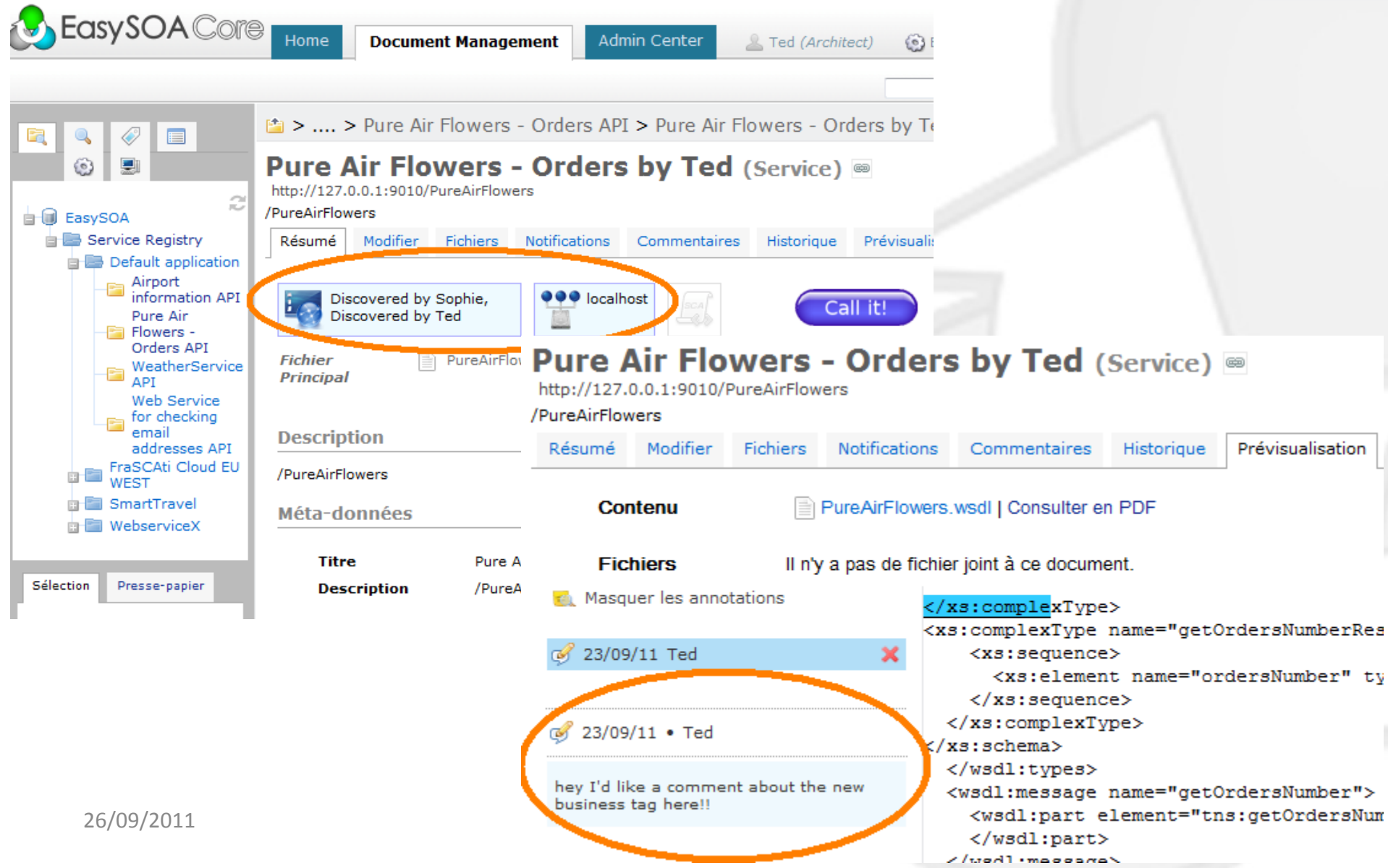
- using their own knowledge : technical architecture
- For that : Service Component Architecture (SCA)
 - an OASIS standard that describes services and **dependencies** between them - kinda like remote service injection / DI

- IT staff

- using what they manage : network

- 5' demo – collaborative discovery : archi & IT





The screenshot displays the EasySOA Core web application. The top navigation bar includes links for Home, Document Management, and Admin Center, along with a user profile for 'Ted (Architect)'. The left sidebar shows a tree view of the Service Registry, including a 'Default application' folder with various APIs like 'Airport information API' and 'WeatherService API'. The main content area is titled 'Pure Air Flowers - Orders by Ted (Service)' and shows the service's URL: 'http://127.0.0.1:9010/PureAirFlowers'. Below the title, there are tabs for 'Résumé', 'Modifier', 'Fichiers', 'Notifications', 'Commentaires', 'Historique', and 'Prévisualisation'. The 'Fichiers' tab is selected, showing a list of files with columns for 'Titre', 'Description', and 'Fichiers'. A comment is visible under the 'Fichiers' tab, dated '23/09/11' by 'Ted', stating: 'hey I'd like a comment about the new business tag here!!'. The right side of the interface shows the service's WSDL file, 'PureAirFlowers.wsdl', and a preview of the XML schema.

EasySOA Core

Home Document Management Admin Center Ted (Architect)

Pure Air Flowers - Orders API > Pure Air Flowers - Orders by Ted

Pure Air Flowers - Orders by Ted (Service)

http://127.0.0.1:9010/PureAirFlowers

/PureAirFlowers

Résumé Modifier Fichiers Notifications Commentaires Historique Prévisualisation

Discovered by Sophie, Discovered by Ted

localhost

Call it!

Pure Air Flowers - Orders by Ted (Service)

http://127.0.0.1:9010/PureAirFlowers

/PureAirFlowers

Résumé Modifier Fichiers Notifications Commentaires Historique Prévisualisation

PureAirFlowers.wsdl | Consulter en PDF

Titre	Description	Fichiers
Pure A	/PureA	Il n'y a pas de fichier joint à ce document.

Masquer les annotations

23/09/11 Ted

23/09/11 • Ted

hey I'd like a comment about the new business tag here!!

```

</xs:complexType>
<xs:complexType name="getOrdersNumberRes"
  <xs:sequence>
    <xs:element name="ordersNumber" ty
  </xs:sequence>
</xs:complexType>
</xs:schema>
</wsdl:types>
<wsdl:message name="getOrdersNumber">
  <wsdl:part element="tns:getOrdersNum
  </wsdl:part>
</wsdl:message>
  
```

Earlier discovery ?

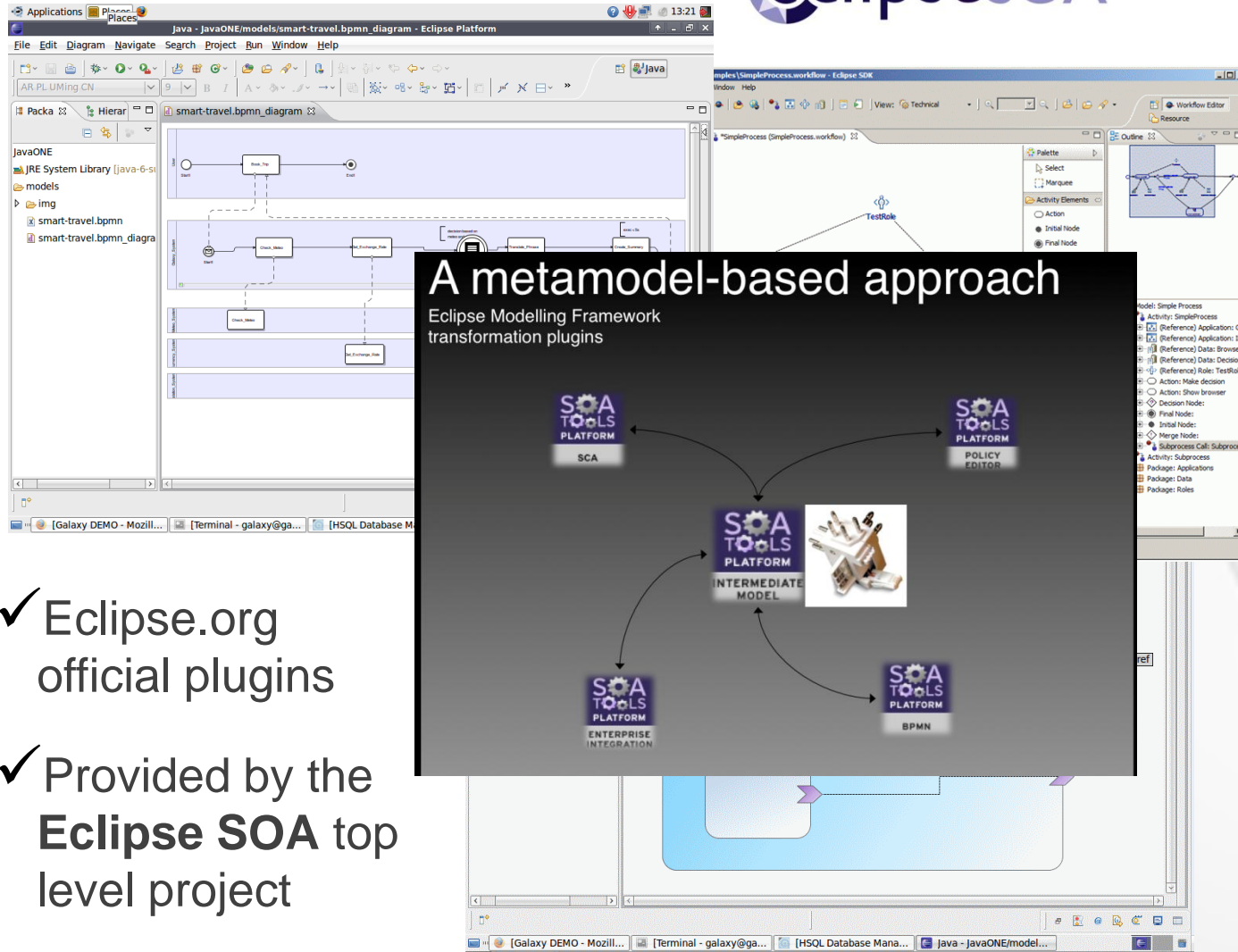
- So that's collaborative service discovery alright
- It can further support
 - Documents (specifications...),
 - Comments & commentable WSDLs,
 - Notifications (service info changed),
 - Etc. (workspaces, channels...)
- To fully benefit of all that, the earliest services are discovered and known the better
 - Could we detect them even before they're live ?!

Development time discovery

- You’ve noted that the architecture has been known by archis before service were in operation
 - services didn't appear out of nothing in operation !
 - rather before they were before in the SOA process,
 - i.e. in development, or even earlier in architecture
- How to detect them at development time ?
 - Again, we have to look for them in the right place
 - i.e. development tools ! ex. Eclipse SOA
- and before ? Business design!
 - are actually business stuff... so back in easysoa



EasySOA – The SOA tooling environment



- ✓ Eclipse.org official plugins
- ✓ Provided by the **Eclipse SOA** top level project

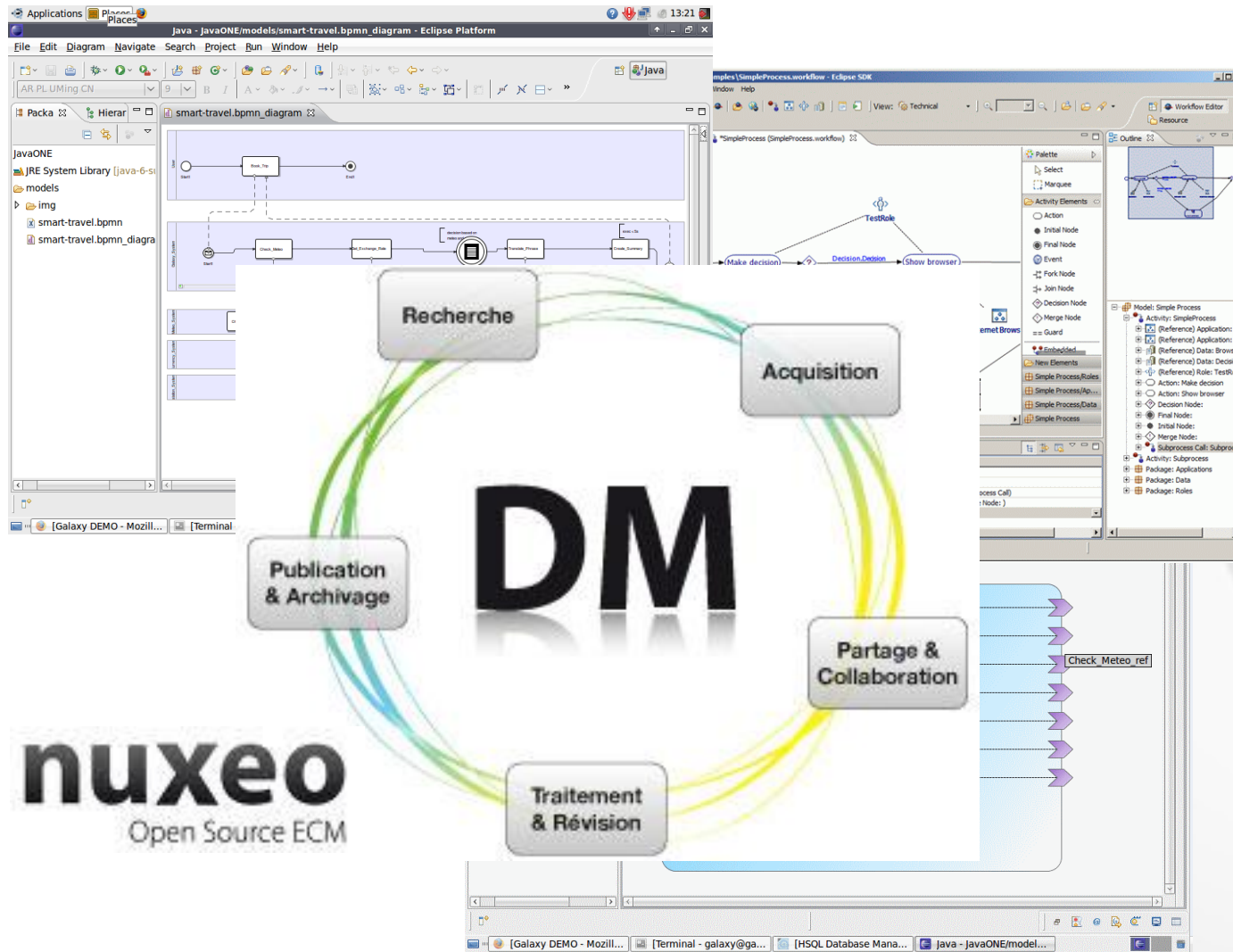
- ✓ **BPMN** for business process design
- ✓ **JWT** for implementation and execution of workflows
- ✓ **SCA** for assembling the service architecture
- ✓ and **Mangrove** as a glue around it all

- ✓ They give a pretty good picture of what is **assembled** in the target service oriented applications.
- However, they are not about the application that business users **actually see and use everyday**, and IT staff monitors
 - They are rather about its models in the development environment
 - Then about the live, executed architecture where the services live and strive

Too bad, these people are an important piece of the SOA puzzle !

- ✓ ...that's the place EasySOA Core expects to be in.

EasySOA – EasySOA Core case study

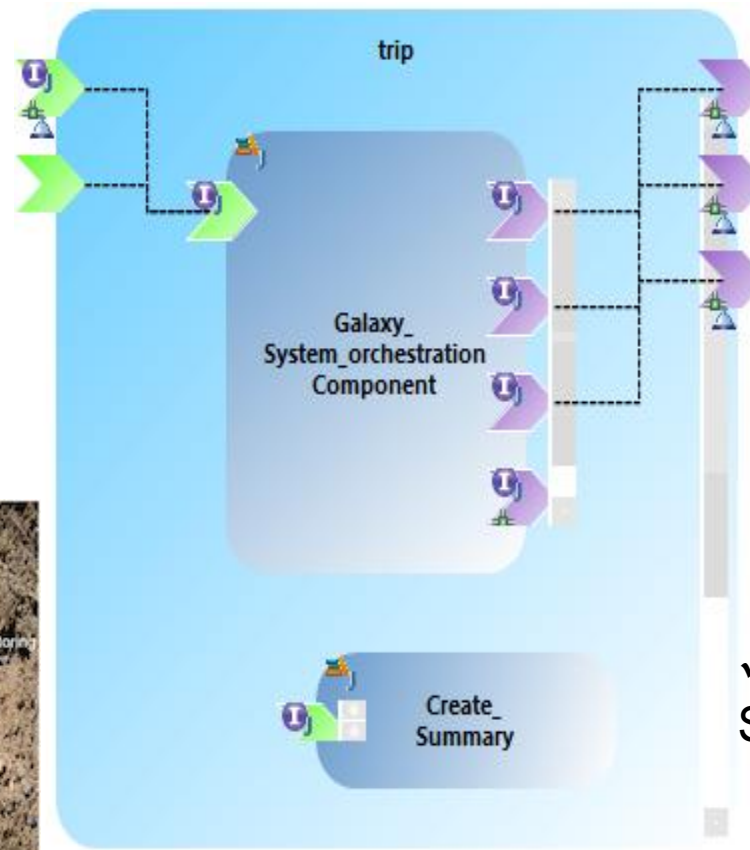


- ✓ EasySOA Core extends the collaboration from the Eclipse SOA Modelling / Mangrove design environment
- ✓ A shared repository with added value features
 - ✓ Business & Application Domain
 - ✓ Life Cycle Status
 - ✓ Authoring Info
- ✓ Multiple enrichment and customization capabilities

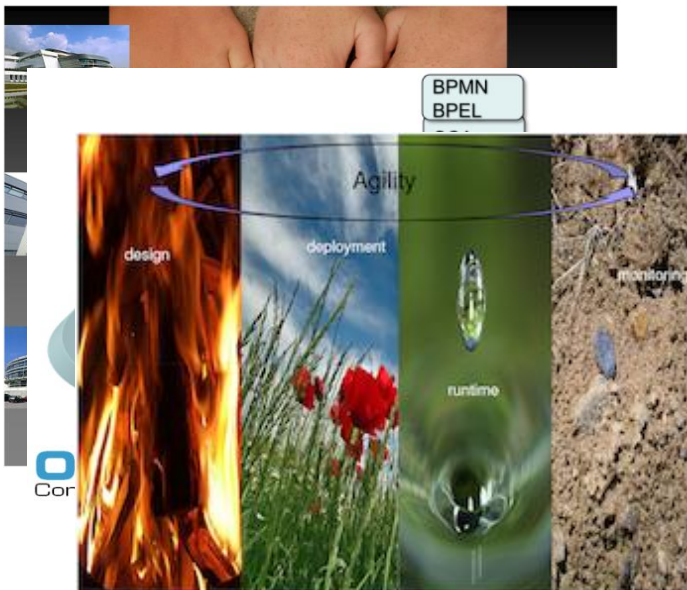
“SmartTravel”

Helping US people making a trip to a French city 😊

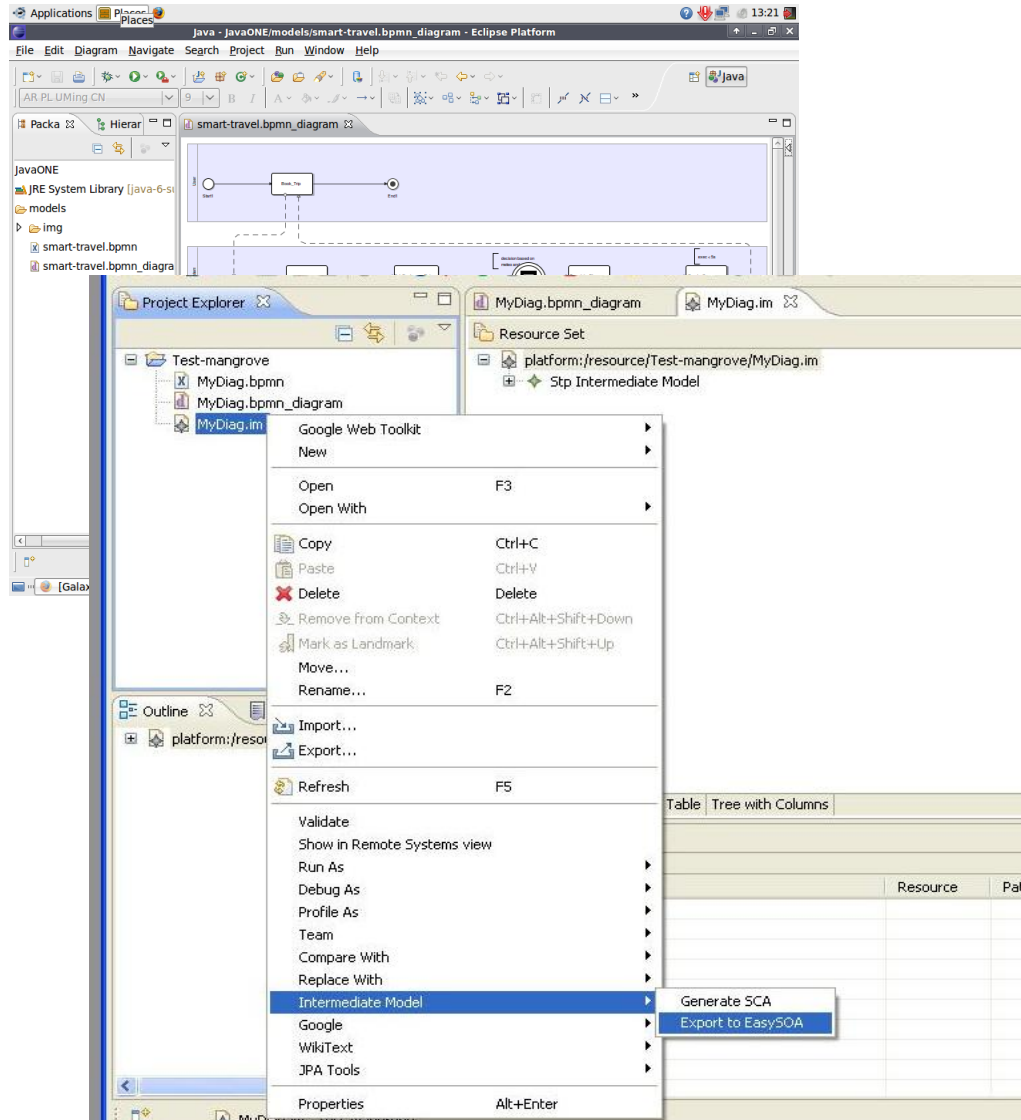
INRIA **galaxy** project
<http://galaxy.inria.fr>



- ✓ Get weather forecast
- ✓ Get current currency rate
- ✓ Translate visitor's advice
- ✓ Build the SmartTravel program

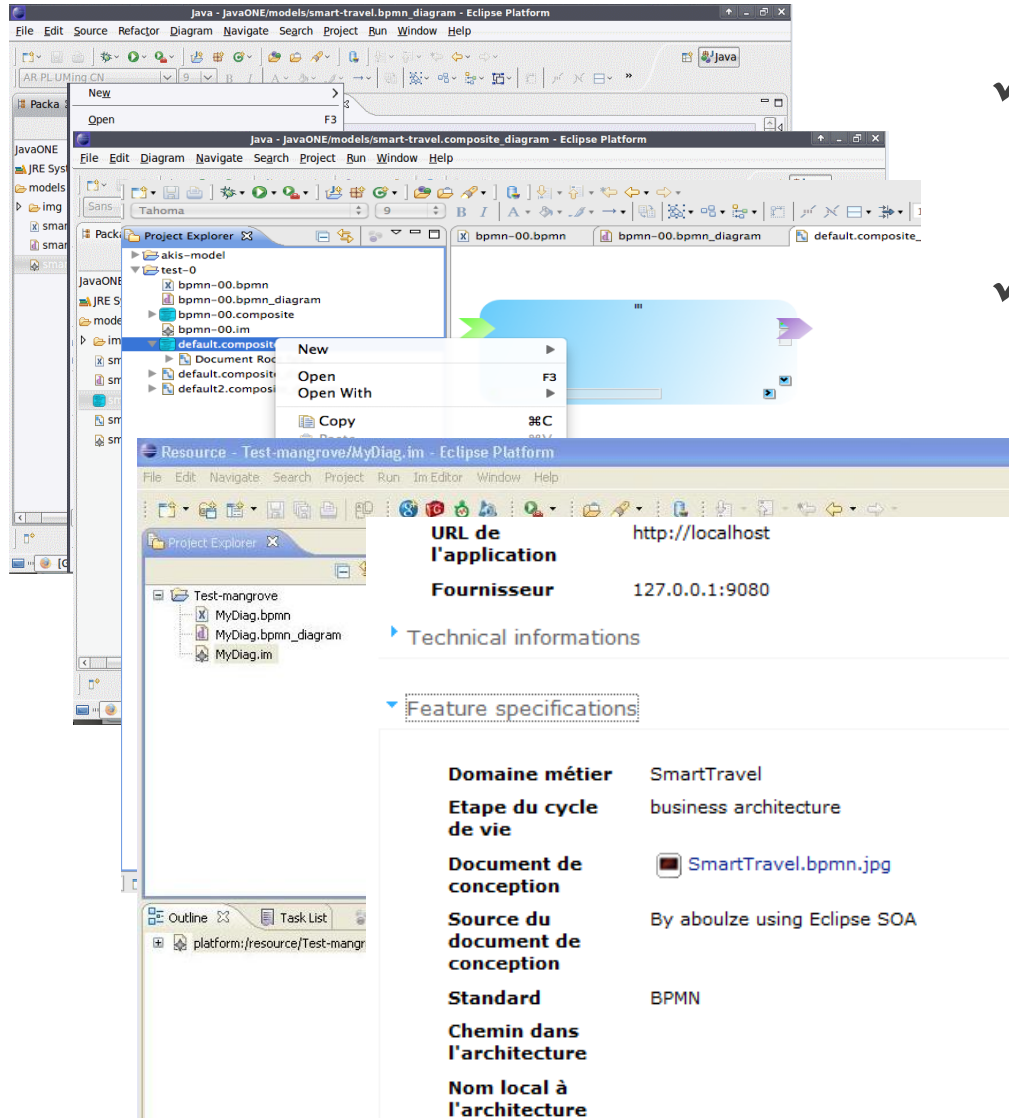


EasySOA – EasySOA Core: share Business Architecture features

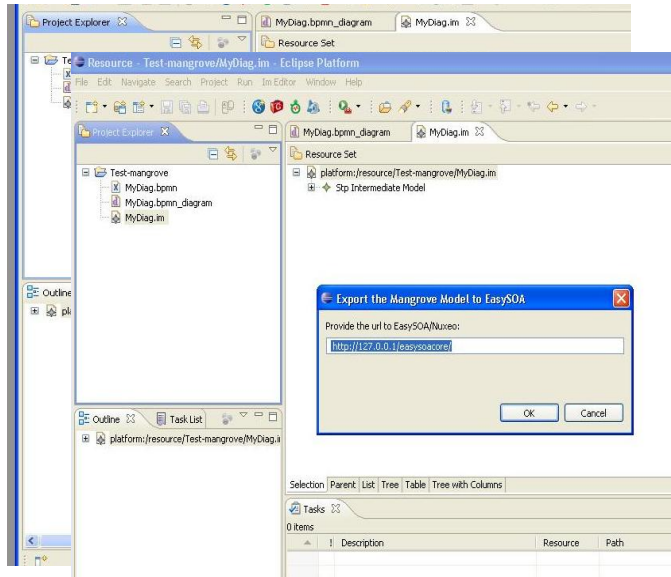


- ✓ Eclipse BPMN Editor
 - « Smart Travel » Business Domain
 - « Business Architecture » phase
 - « Business Architect » author
- ✓ Eclipse BPMN Editor
 - Export to Mangrove
- ✓ EasySOA Core ECM
 - Web shared b/w Actors
 - Access rights & collaboration rules
 - Multiple views
 - Information (Enrich)
 - Documents (Diagrams)
 - Business artefacts

EasySOA – EasySOA Core: share System Architecture features



- ✓ Eclipse BPMN Editor
 - From Mangrove generate SCA
- ✓ Eclipse SCA Editor
 - Graphical rearrangement
 - Refine technical aspects
 - Export to Mangrove
- ✓ EasySOA Core ECM
 - « Smart Travel Orchestration » Application Domain
 - « System Architecture » phase
 - « System Architect » author
 - System artefacts (services, components)



Export2EasySOA
Mangrove plug-in



Use/Specialization

Mangrove plug-in

- ✓ Eclipse workspace
 - Menu/click to Export to EasySOA Core
 - Enter EasySOA Core URL
 - Triggering an Eclipse Mangrove plug-in



- ✓ Export2EasySOA plug-in
 - Parsing Mangrove scheme
 - Relevant elements, element per element
 - Validation of EasySOA Core (NUXEO) structure
 - Interface with EasySOA Core (NUEXO) by invoking REST
 - Improvement of Mangrove plugin

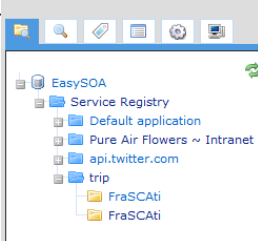
Import de données SCA

Fichier
".composite"☒ Aucun☐ Charger

Parcourir...

Impl.
d'application
parente

trip

Sélection Presse-papier
Aucun document dans la sélection.

trip

Contenu Modifier Administration

Nouveau document Importer un fichier

<input type="checkbox"/>	Titre ↕	Dernière modification ↕	A
<input type="checkbox"/>	/Check_Meteo_ref ↗	28/06/11 12:25	Ad
<input type="checkbox"/>	/Galaxy_System_orchestrationComponent/Create_Summary_genService_ref ↗	28/06/11 12:25	Ad
<input type="checkbox"/>	/Get_Exchange_Rate_ref ↗	28/06/11 12:25	Ad
<input type="checkbox"/>	/Translate_Phrase_ref ↗	28/06/11 12:25	Ad
<input type="checkbox"/>	FraSCaTi ↗	28/06/11 12:25	Ad
<input type="checkbox"/>	FraSCaTi ↗	28/06/11 13:19	Ad

Coller Ajouter à la sélection Copier Supprimer



Méta-données

État

- ✓ EasySOA Core
 - Import SCA (from file, SCM...)

EasySOA – Correlate with service discovery (Core) & monitoring (Integration) in the live application

- ✓ EasySOA Core
 - Service Discovery by browsing

Export WSDL to Nuxeo


Selected WSDL
None.

Service details

Service Name:

Application:

Pure Air Flowers ~ Intranet



Main page

Services list

List of available services

Order management
[PureAirFlowers Orders WSDL](#)

Flower travel management
[PureAirFlowers Create Summary WSDL](#)

EasySOA

- Service Registry
 - Default application
 - Pure Air Flowers ~ Intranet
 - api.twitter.com
 - trip
 - FraSCAti
 - FraSCAti

/CreateSummary

/CreateSummary

Résumé | Modifier | Relations | A

localhost
 smart-travel.composite

Description

/CreateSummary

Méta-données

Titre	/CreateSummary
Description	/CreateSummary
URL du service	http://localhost:9080/CreateSummary
Nombre d'appels écoutés	1
Utilisateurs liés	

- ✓ EasySOA Integration
 - Service Discovery by monitoring

And down to runtime

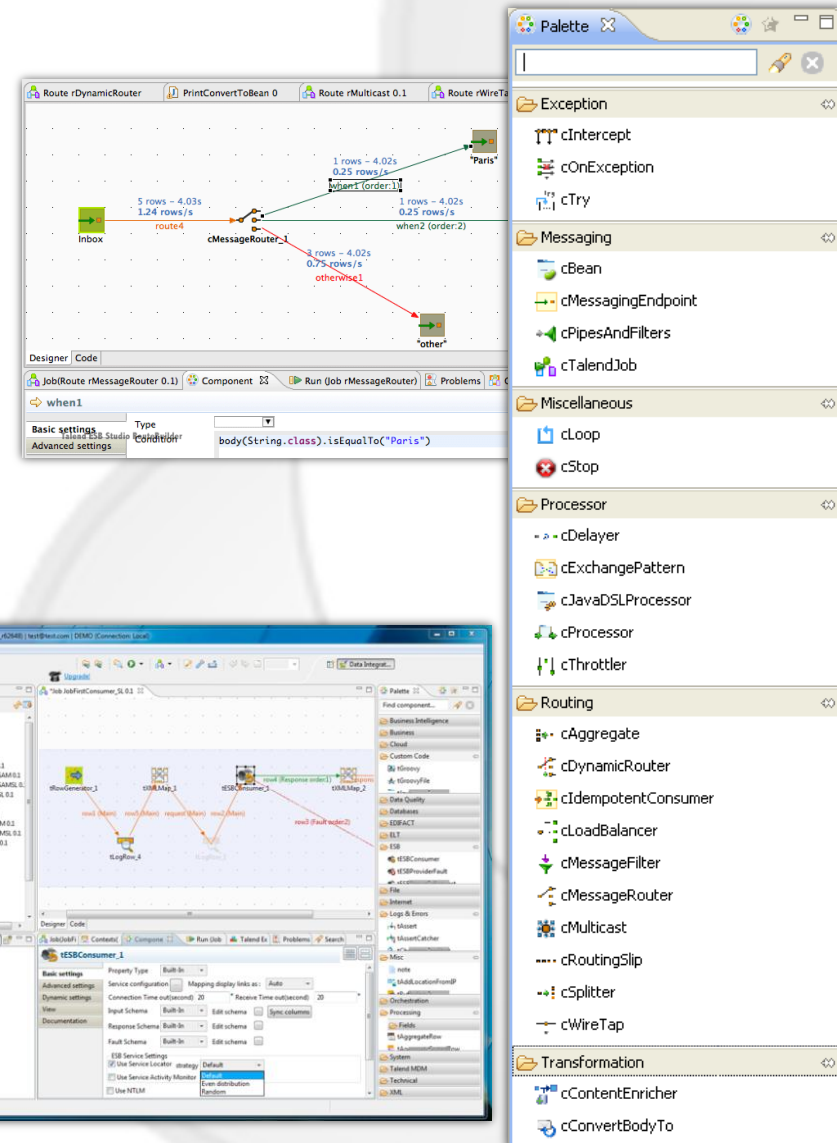
- We've seen development tools integrated with EasySOA, but what about runtimes ?
- Enter **Talend**, a real world SOA solution & ESB, with incidentally a business & data story
- Demo
 - 10' Talend ESB presentation & Airport demo
 - 3' Talend Airport in EasySOA
- ... and when on the other side EasySOA-discovered service will be available as endpoints in Talend Studio, the circle will be complete

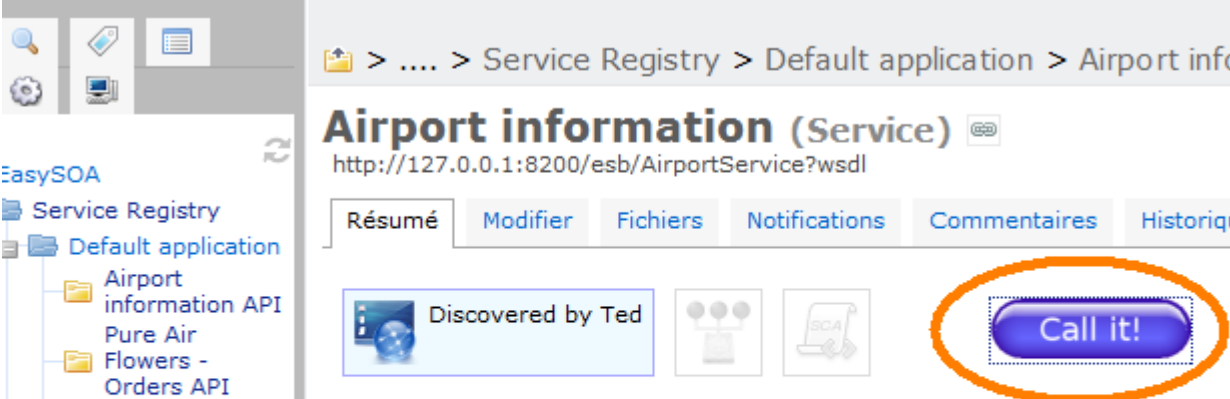
Easily integrate distributed systems

- Standards-based connectivity layer to integrate distributed systems
- Powered by Apache CXF, Camel, Karaf and ActiveMQ
- Includes messaging, Web services, intelligent routing, failover, monitoring and security

Talend ESB Studio

- Quickly and easily create data services and complex message routes without coding
- Leverage existing skillsets with Eclipse-based UI
- Drag-and-drop message routes using Enterprise Integration Patterns (EIPs)





Service Registry > Default application > Airport info


Airport information (Service)

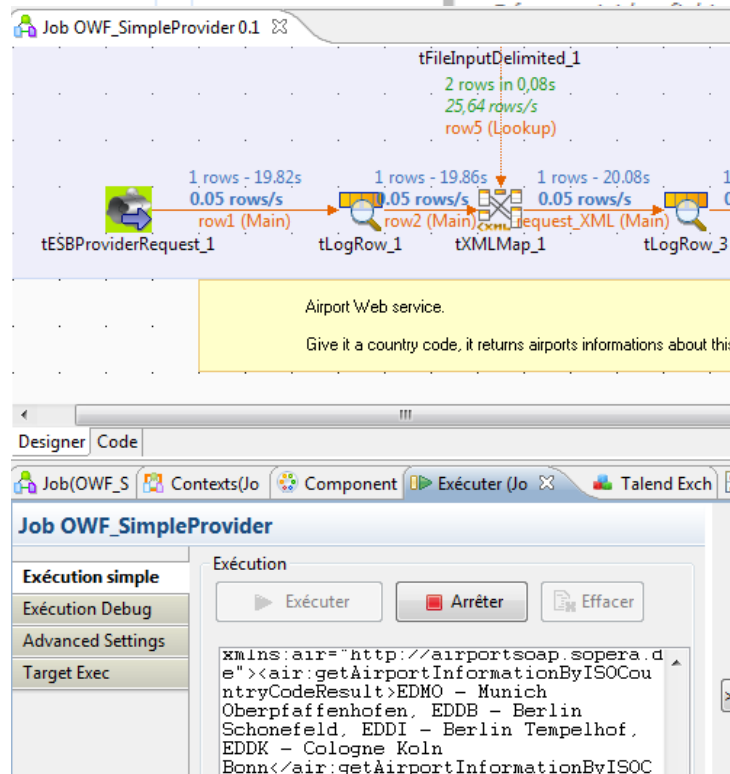
http://127.0.0.1:8200/esb/AirportService?wsdl

Résumé | Modifier | Fichiers | Notifications | Commentaires | Historique

Discovered by Ted

Call it!





Job OWF_SimpleProvider 0.1

tFileInputDelimited_1
2 rows in 0,08s
25,64 rows/s
row5 (Lookup)

tESBProviderRequest_1
1 rows - 19,82s
0,05 rows/s
row1 (Main)

tLogRow_1
1 rows - 19,86s
0,05 rows/s
row2 (Main)

tXMLMap_1
1 rows - 20,08s
0,05 rows/s
row1 (Main)

tLogRow_3
1 rows - 20,08s
0,05 rows/s
row1 (Main)

Airport Web service.
Give it a country code, it returns airports informations about this

Designer | Code

Job(OWF_S) | Contexts(Jo) | Component | Exécuter (Jo) | Talend Exch

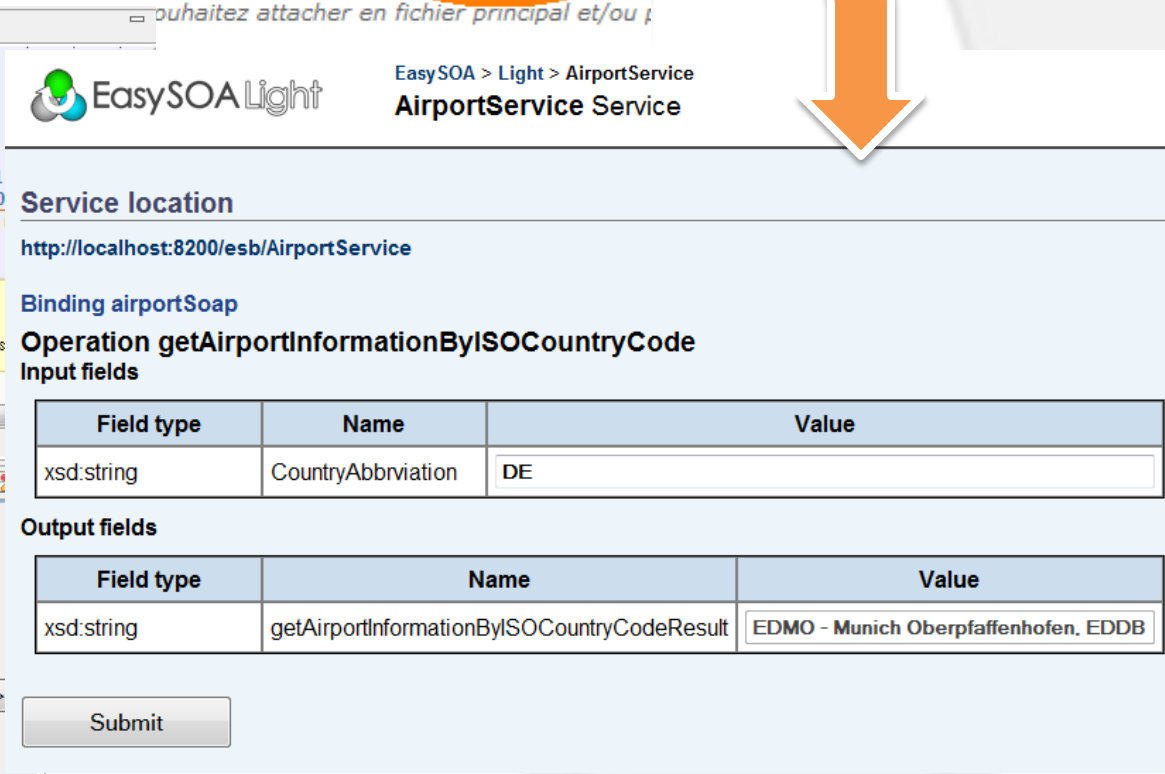
Job OWF_SimpleProvider

Exécution simple
Exécution Debug
Advanced Settings
Target Exec

Exécution
▶ Exécuter
■ Arrêter
🗑 Effacer

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<xm:ns:air="http://airportsoap.sopera.de" ?>
  <air:getAirportInformationByISOCountryCodeResult>EDMO - Munich
  Oberpfaffenhofen, EDDB - Berlin
  Schonefeld, EDDI - Berlin Tempelhof,
  EDDK - Cologne Koln
  Bonn</air:getAirportInformationByISOC
  
```



EasySOA > Light > AirportService
AirportService Service

Service location

http://localhost:8200/esb/AirportService

Binding airportSoap

Operation getAirportInformationByISOCountryCode

Input fields

Field type	Name	Value
xsd:string	CountryAbbriation	DE

Output fields

Field type	Name	Value
xsd:string	getAirportInformationByISOCountryCodeResult	EDMO - Munich Oberpfaffenhofen, EDDB

Submit

- Up to now :
 - Mar. 2011 – Started coding
 - May 2011 - **Linux Solutions** – business user demo
 - Jun. 2011 – **Eclipse DemoCamp** architect & IT demo
 - Sept. 2011 – **Open World Forum** – collaborative, talend
- finally, is it usable right now ?
 - Mostly prototype, however...
 - We seek to deliver ASAP features useful in the real world, and we've got a clue
- ... by asking real world people about them

A « from the trenches », « guerilla » approach

- To help gather recurring use cases and needs, around the project core, at our clients', our communities
- People we know, who trust us, that we ask to share their SOA issues

The idea : reciprocal sharing

- Share your issues with us,
- We'll enrich EasySOA to address the most promising and useful ones, and share our developments back in Open Source

Pay only custom work

- If you want some : install, config, custom devs

- Ex. Audit of an SOA use case (Partner X) :
 - use service discovery, then export wsdl
 - this can be useful to you right now, it has been for me !
 - 2' WSDL export demo

- Next :
 - API change detection (Partner X)
 - SOA documentation (Partner Y)
 - light cloud application platform (Partner Z)
 - BPM with OW2 Scarbo, monitoring with OW2 Jasmine



www.easysoa.org

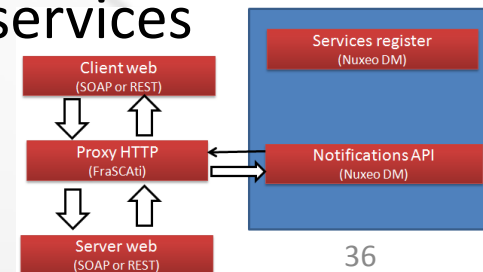
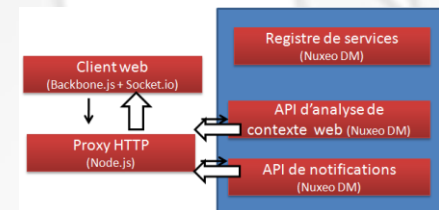
github.com/easysoa

easysoa-dev@googlegroups.com



Behind the curtain

- About the Service Registry...
 - aren't there already ESB registries, UDDI etc. ?
 - ESB registries' actual goal is to provide service resolution to their ESB runtime. And when they are provided as an application they show they're very **tied to their solution**
 - There are others : *as detected - as should be - as made...*
- Discovery by browsing
 - Web browser proxy + WSDL scraper
- Discovery by monitoring
 - HTTP proxy + Esper correlation on APIs / services
- Discovery by SCA architecture import
 - XML parsing of SCA composite file



- EasySOA branches out to an “on demand” Open Source ecosystem of compatible solutions
 - Starting with an SOA framework really simple to use
 - branching out to an on-demand Open Source solution
- Thanks to EasySOA Core plugins, integrating them
 - on : design, development (« SOA sandbox », continuous integration, functional testing & Q/A), monitoring, Business Process Management (BPM) and Cloud
- Built on best-of-breed Open Source components
 - Supported by EasySOA partners : Apache CXF & Camel, Nuxeo DM, Talend Data Integration, OW2 Jasmine, OW2 Frascati, OW2 Scarbo, Eclipse SOA et JWT

- we even plan an embedded SOA platform:
EasySOA Light, building on full web technologies
 - To democratize access and call of existing enterprise services
 - With the web as light, universal, end-user friendly service standard : REST(ful), javascript, JSON...
 - Provided to web developers : by scripted mashups, SOA scaffolding (“the Ruby on Rails of SOA”) or composition
 - but also business users : by point and click programming - or even Excel-driven...
- Yep, it goes way beyond the Service Scaffolder !

- cool things we're playing with :
 - node.js / backbone / connect / express / antinode
 - REST(ful) SOA, CXF
 - CEP / Esper
 - OSGi, SCA, FraSCAti - and all together
 - And eclipse, velocity, SOAPUI, sonar...
- and hope to get to play with :
 - weka, maven, OAuth, mongodb, SPoRE, restfulie