Dynamically reconfigurable architectures for autonomic services and M2M applications

(Architectures dynamiquement reconfigurables pour les services autonomiques et les applications M2M)

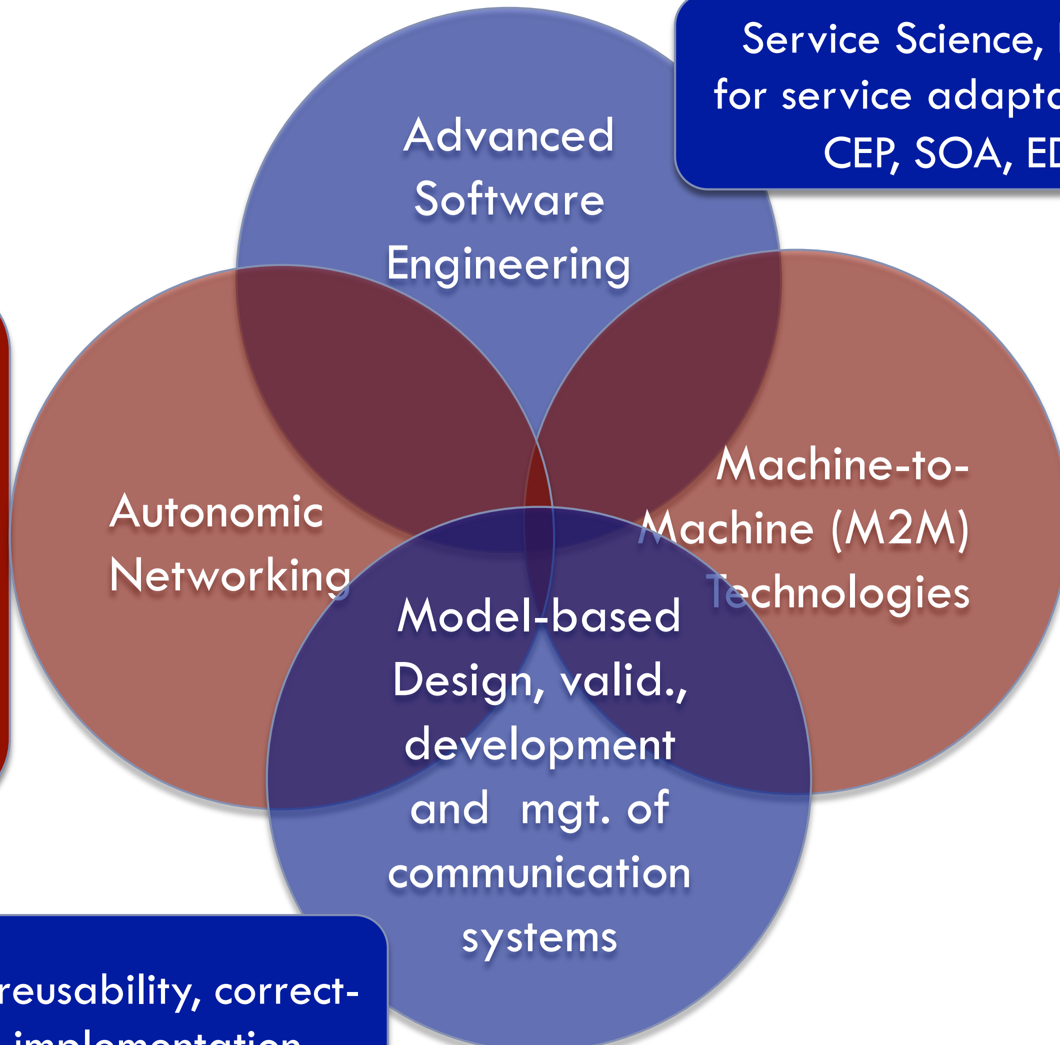
Khalil DRIRA

LAAS-CNRS

Introduction : the scientific and technologic perimeter of our investigations of this talk

Service Science, Dynamic architectures for service adaptation and composition, CEP, SOA, EDA, CBSE, SCA,WS,…

creating **self-managing** networks to overcome the rapidly growing complexity of the Internet and other networks,…



allow both wireless and wired systems to communicate with other devices of the same ability, …

PIM,PSM models, reusability, correct-by-design implementation,…

The context (1/2) : M2M (Machine-to-Machine) communication, services and applications

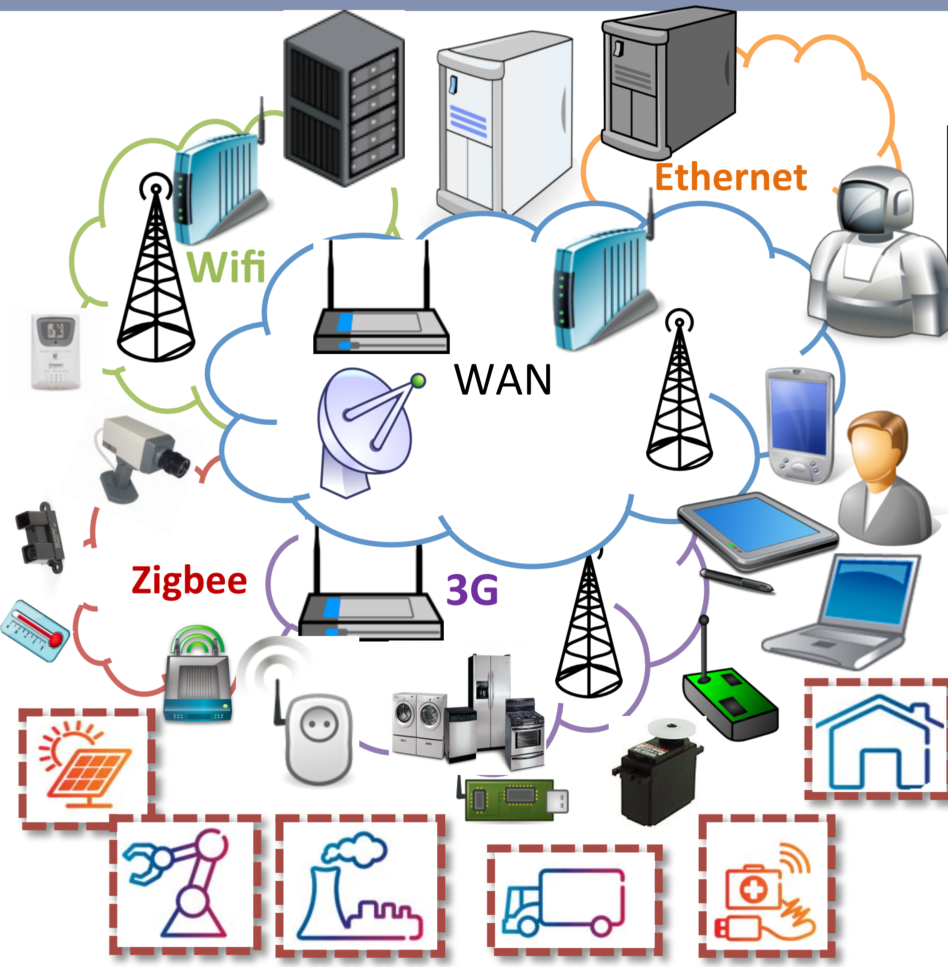
Computers (desktops, laptops)

Smart devices (phones, tablets, meters, cameras),

Remote sensors/actuators (force, presence, temperature, light, movement, acceleration, …)

Passive & active tags (NFC, RFID)

Any other communicating machines & objects (robots, vehicles, home appliances, …)



Locally or remotely communicating

cooperating actors

human or artificial

Various vertical domains

Mobile & Fixed	Standard& embedded	IP-enabled & Not	Short & long field communication	Wired & wireless connection

The Context (2/2): dynamically changing upper and lower surrounding contexts

Applications: Multimedia mobile group-enabled

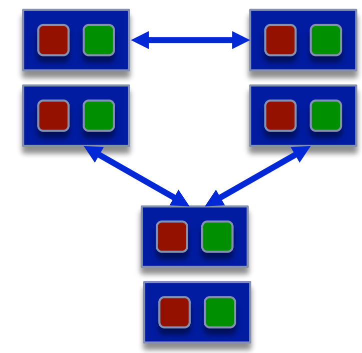
Evolving requirements:

- Dynamic groups: changing membership
- Mobile actors: changing access device
- Evolving missions: priorities/objectives

Communication protocols & services

Specific Architectures:

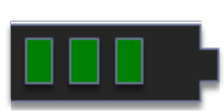
- **Multi-level**
- **Multi-component**
- **Distributed**



Access Networks: Heterogeneous wired & wireless

Variable constraints:

- energy
- bandwidth
- Const. & lted devices: storage, processing



The motivations behind our investigations

Applications with **Evolving requirements**

Communication protocols & services with **Multi-level multi-component distributed architectures**

Access Networks with **Variable constraints**

The Problem:

How to provide **service availability, & quality appropriateness** in **Dynamically changing contexts**

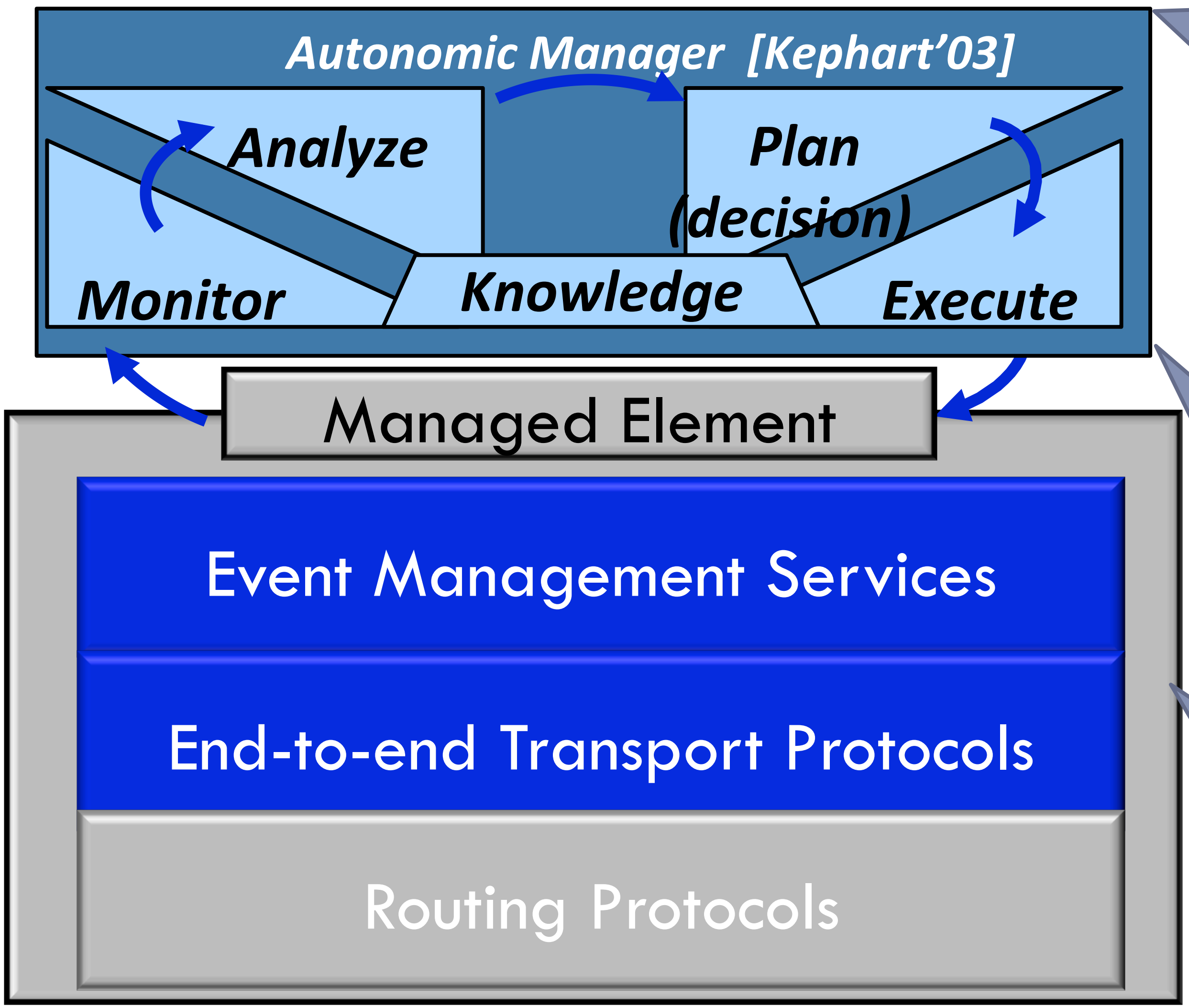
Our Solution:

Autonomic management of dynamic reconfiguration:

run-time adaptation of protocol behaviors and service compositions

Upon Service-Oriented, Component-based and Event-driven dynamic architectures

Model-based Autonomic Management



Models/form/theories : semantics (Onto), structural (GG), Event-based (chron) Behavioral (PNets, Proc. Algebra),

Tools/Frameworks:

GMTE (Dynamic archi), FACUS(NetQoS, Diamond, IMAP)

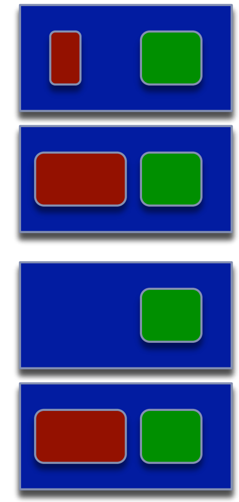
Methods/ technologies/ protocols: SOA, EDA, Publish/ subscribe (DDS, AMQP, CEP) MPTCP (ATP)

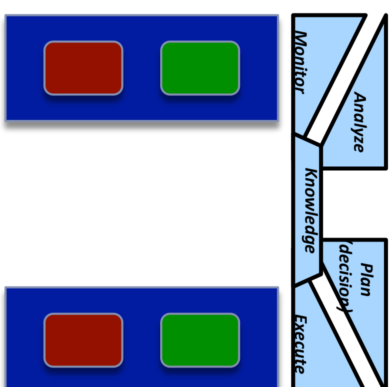
LAAS-CNRS

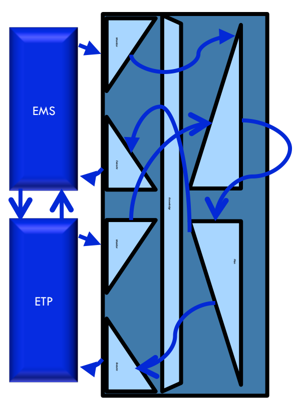
The challenges (1/2)

Vertical coordination for cross-levels Consistency Management

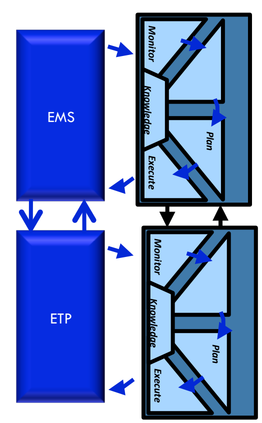
- How to avoid over-reactions between the levels
  - Application degrades the video codec
  - Network increases the video bandwidth
- How to avoid opposite reactions
  - Application disconnect video
  - Network increases resources for video connection
- Design the appropriate vertical coordination architecture







shared analysis & planning

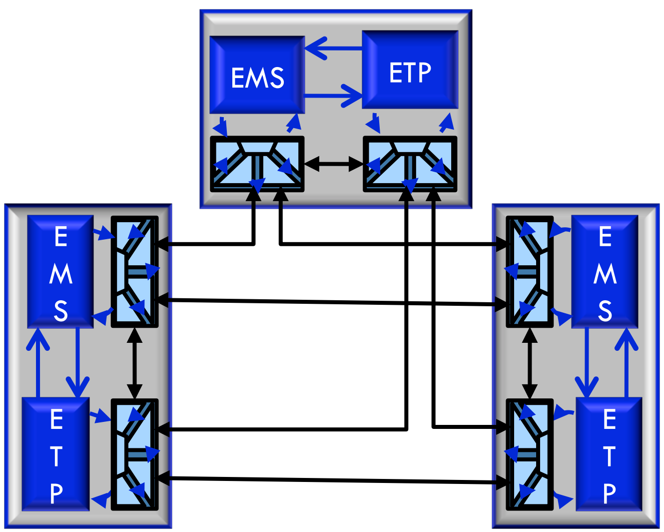


Cross-layer coordination

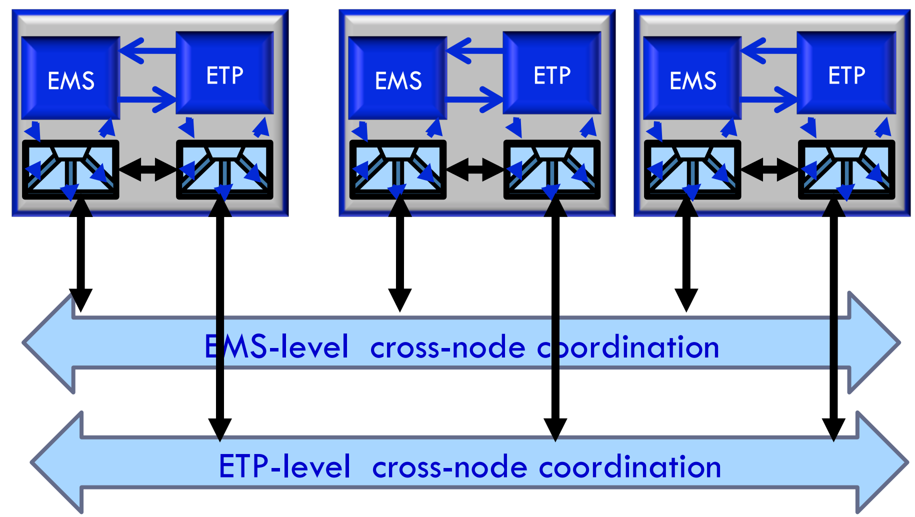
The challenges (2/2)

Horizontal coordination for cross-nodes Consistency Management

- Implement the distributed management : centralized/ decentralized monitoring/analysis/decision/execution
- Manage the ingoing/outgoing communication flows,



Federated architecture



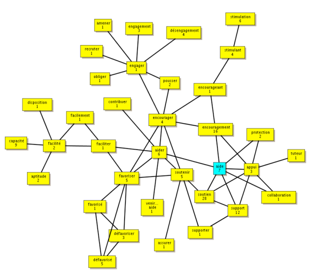
Orchestrated architecture

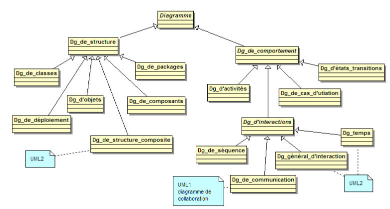
Method, Models (1/2)

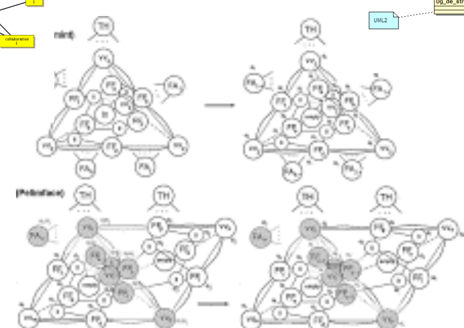
Model-based end-to-end Autonomic Management

- Theoretical investigations for elaborating models underlying analysis and decision

Semantic reasoning / Ontology models, SWRL rules







Dynamic architectures description / Extended UML models

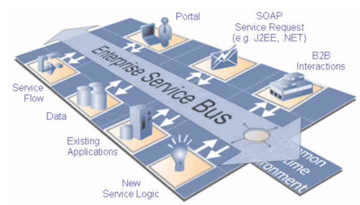
On-line execution / off-line simulation
Graph Matching and Transformation rules (GG)

Method, Models (2/2)

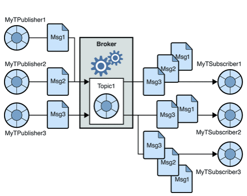
Model-based end-to-end Autonomic Management

- Advanced technology development for implementing monitoring and executing reconfiguration

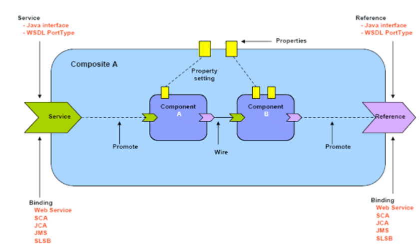
Event-driven architectures (AMQP, DDS)



Service Component Architecture (SCA, OSGI, IPOJO)



Service-oriented Architecture (WS, SOAP, ESB)



Method, Models

- Inter-levels consistency characterization GG rules: define the set of configurations of level n-1 that are valid implementations of a given level n configuration
- Intra-level reconfiguration GG rules : define the set of configurations of level n that are valid substitutes of a given level n configuration, select the most appropriate (avoid upper level perturbation, cost, distance)

structural

- Ontology for automated service deployment in cooperative activities
- Ontology for transport level adaptation and SLA management

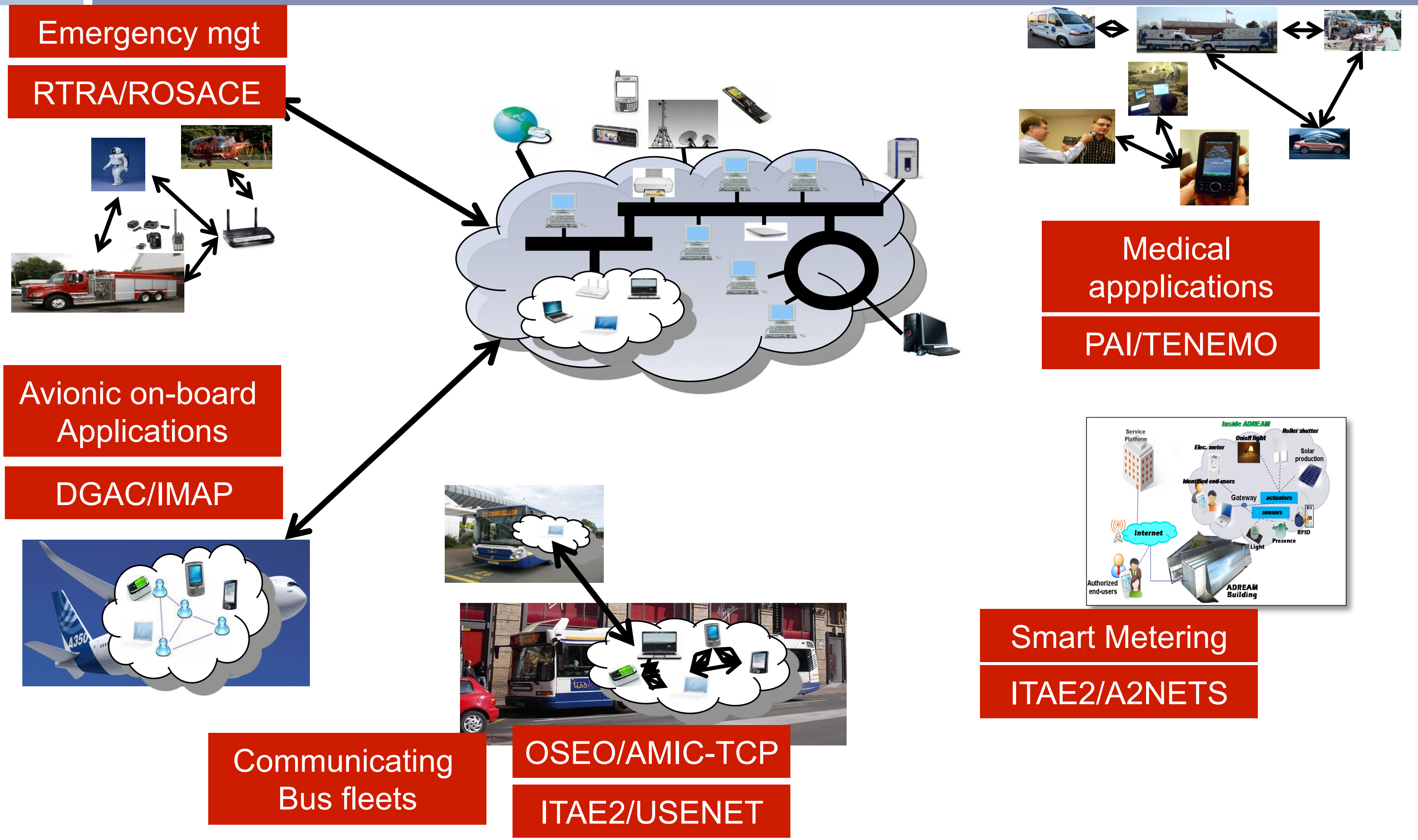
Semantics

Existing

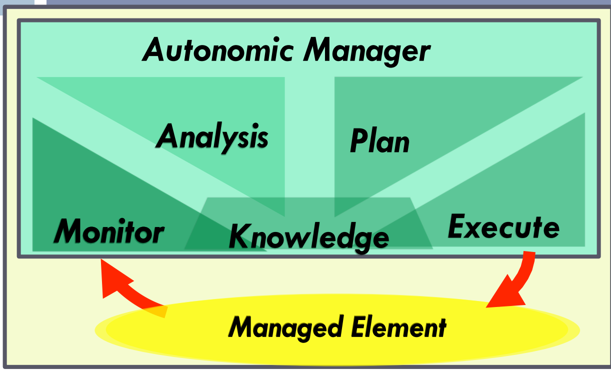
- Unify, integrate models (level-specific, level-independent)
- Scalability management, deterministic reconfiguration
- Additional scenarios : smart metering, networked control,…

Evolution

Main recent projects related to this talk

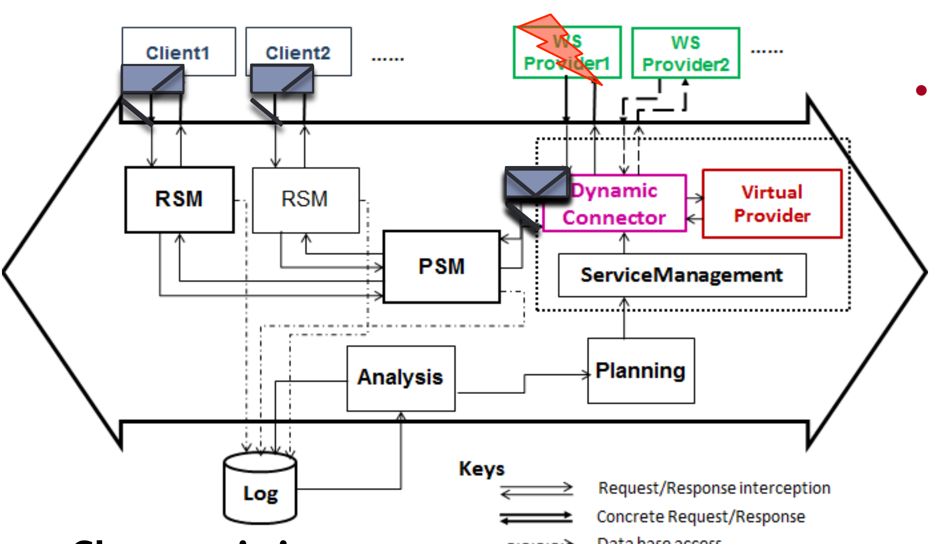


Dynamic Reconfiguration of Service Oriented Architecture, (DGAC/IMAP Project)



[Kephart'03]

The autonomic WS-based framework



- **Characteristics:**
  - **Providers are implemented as Web Services**
  - **Communication uses the SOAP protocol**

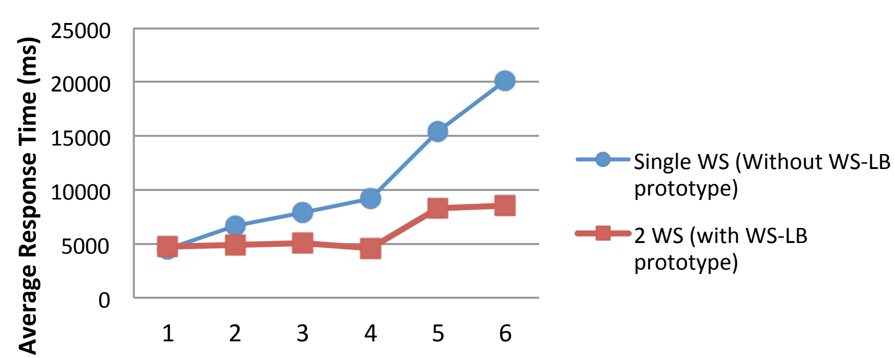
- **Two prototypes implemented**
- ***WS-Substitution***

Replacing a degraded provider by an equivalent

- ***WS-Load Balancing***

Sharing requests between different

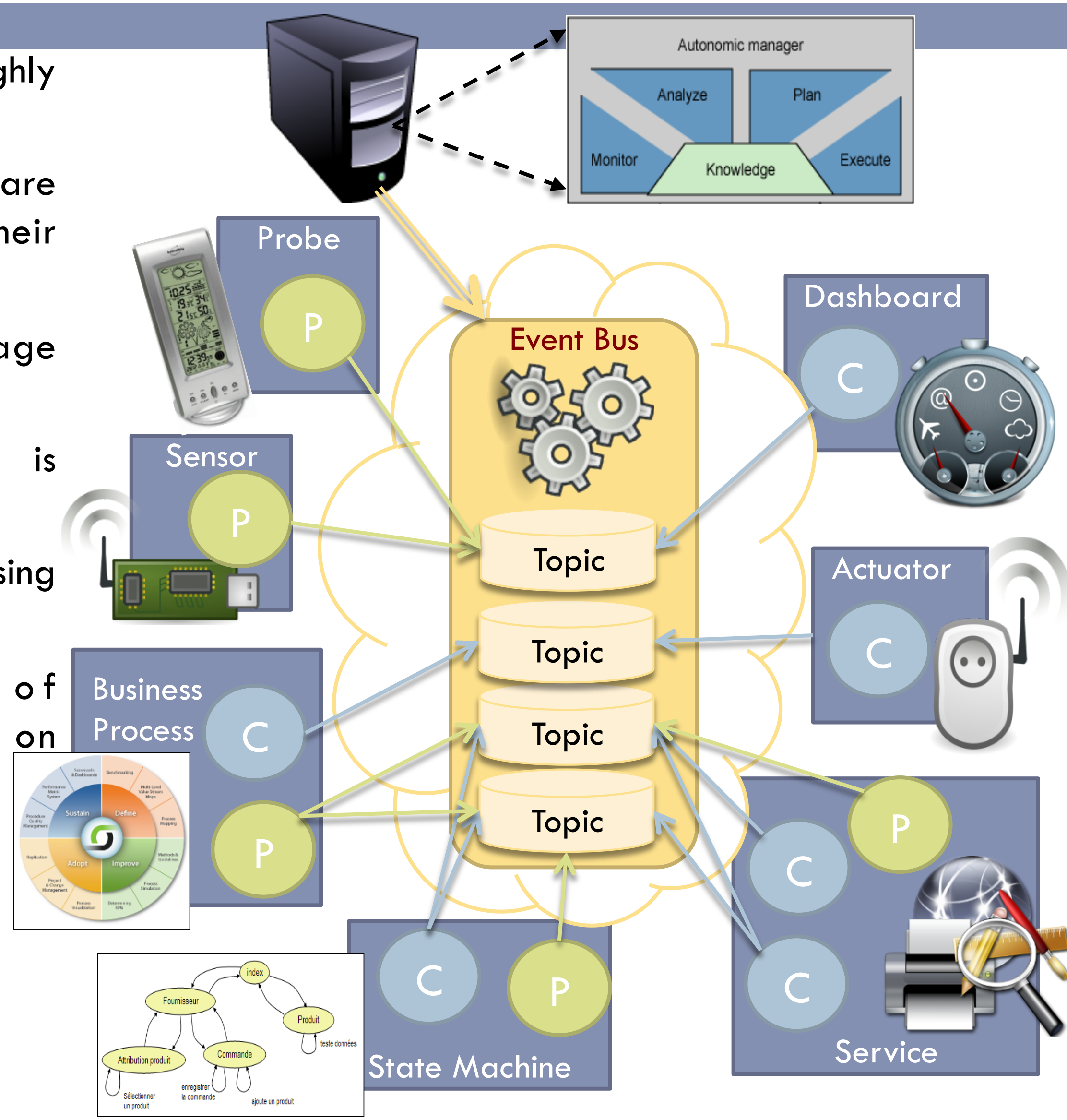
- **Reusability and Adaptability of our WS-Prototypes**
  - **Application-independent**
  - **Only provided interface described in WSDL is required**

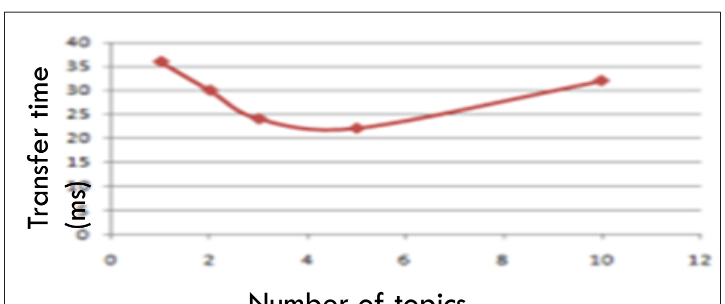


WS-LB prototype performance

Dynamic Reconfiguration for Event Driven Architecture (EDA) applied to Advanced Message Queuing Protocol (AMQP) (DGAC/IMAP Project)

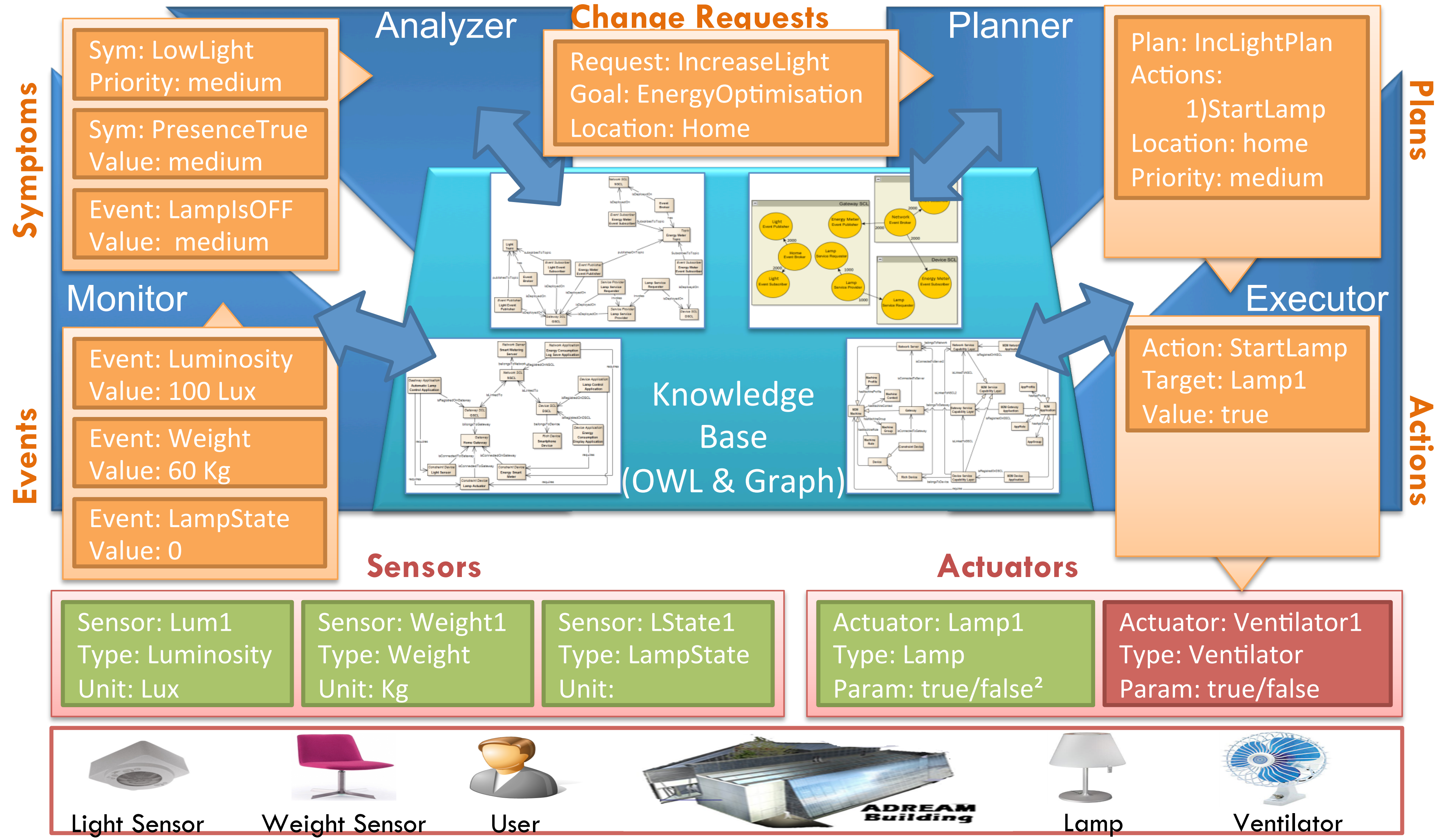
- An event-driven system is composed of highly distributed and heterogeneous nodes.
- Event producers and event consumers are deployed in nodes depending on their requirements.
- Communication is enabled by message exchange via an Event Bus through topics.
- Self-management of the event system is ensured by an autonomic manager for:
  - Dynamic reconfiguration of topics using load balancing techniques.
  - Dynamic deployment of of communication entities based on semantic models.





LAAS-CNRS

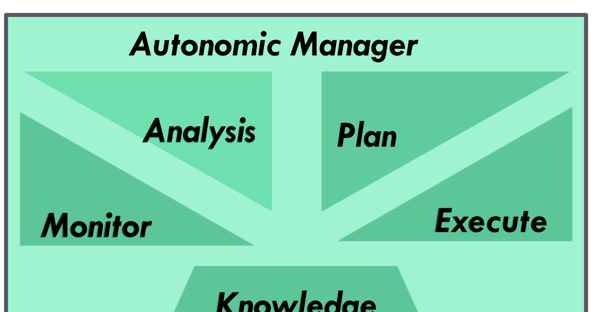
FRAMESELF Autonomic dynamic reconfigurtion for an M2M Aplication scenario (ITEA2/A2NETS Project, see Demo) ANR-SOP

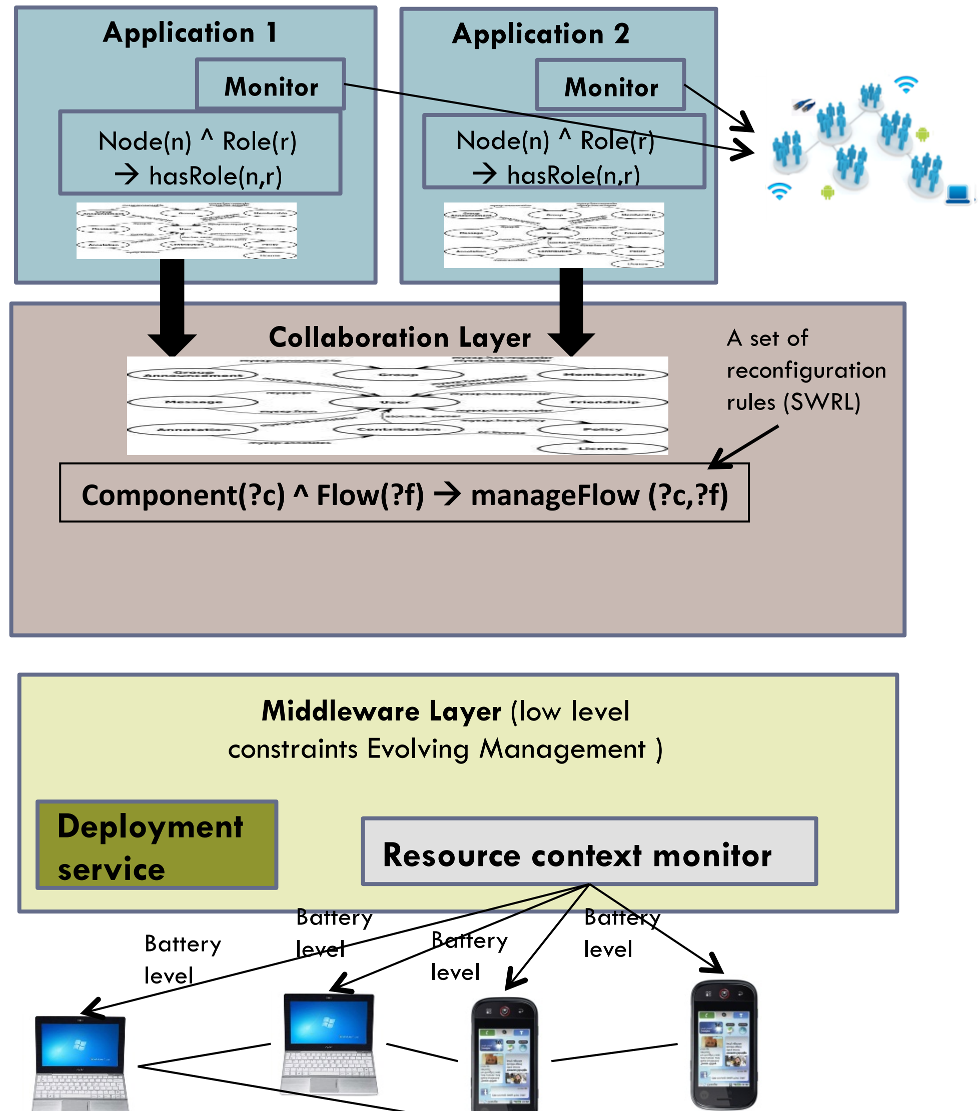


LAAS-CNRS

FACUS: A Semantic Adaptive Framework for collaborative systems (ITEA2/USENET, ANR/GALAXY) (1/2)

- Support of collaborative activities in distributed environment
- Multi-level generic approach to enable semantic multi-level adaptation
- Generic collaboration ontology + collaboration evolving management rules (SWRL)
- Domain ontology (specific concepts) + Application specific evolving management rules (SWRL)



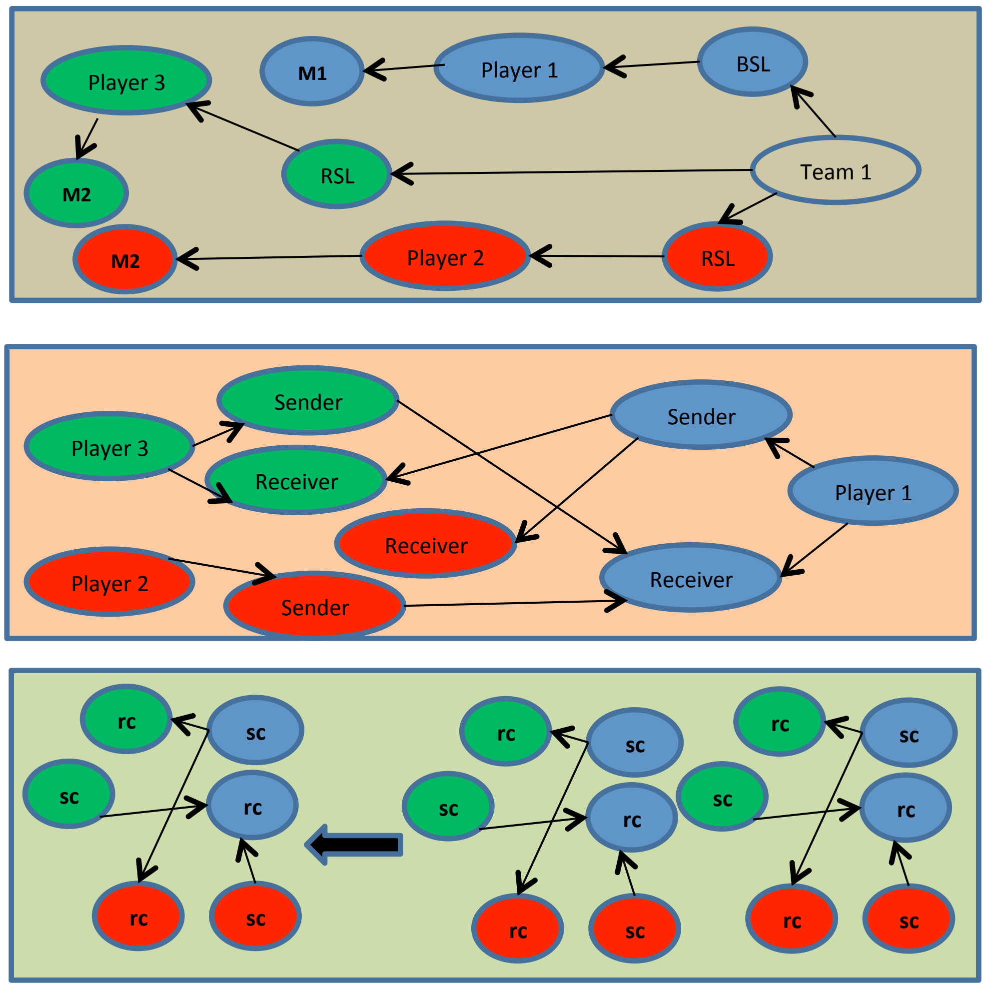


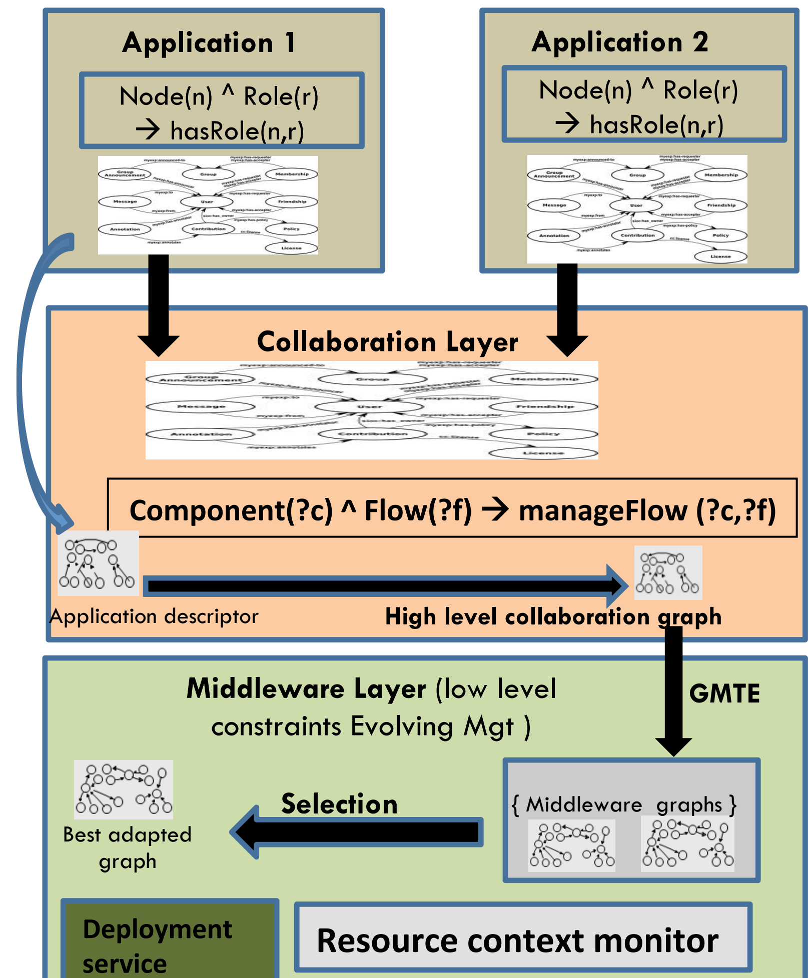
FACUS: A Semantic Adaptive Framework for collaborative systems (ITEA2/USENET, ANR/GALAXY) (2/2)

Adaptation process (after an application change)

S

Deploy the collaboration components





LAAS-CNRS

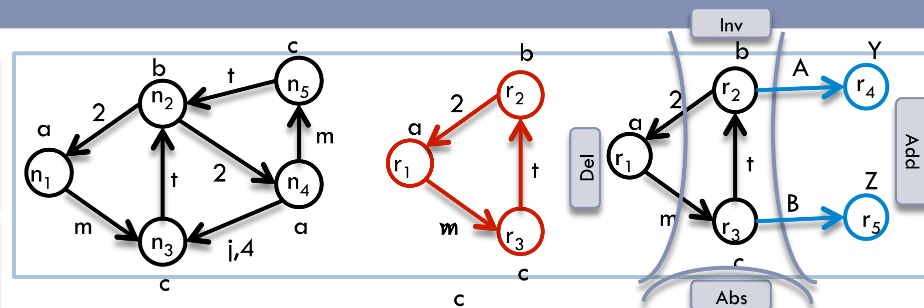
GMTE: Graph Matching & Transformation Engine

http://homepages.laas.fr/khalil/GMTE/ (CNRS Licence 2011)

Graphs

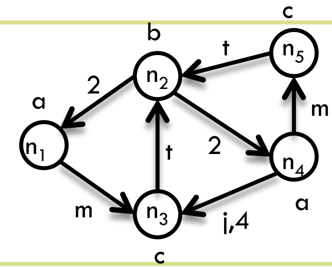
Directed
Multi-labeled (vertex & edge)
Multi-typed (scalar & string)

The "rule Graph" is a Graph partitioned in 4 zones



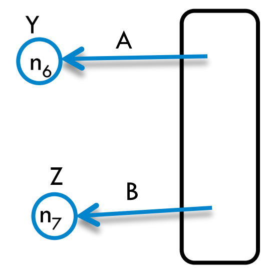
Matching

Mapping a graph (Rule G) to another (Host G)
Exact matching: same structure same labels
one occurrence / all occurrences
Inexact matching: differences are allowed



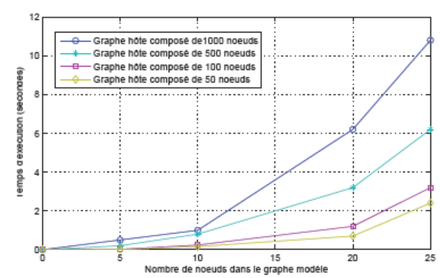
Transformation

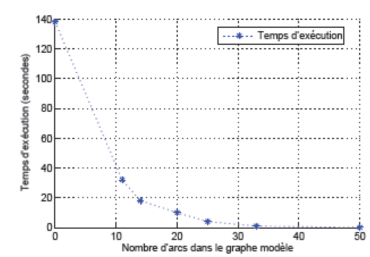
Delete what is in the « del » zone
Keep what is in the « Inv » zone invariant
Add what is in the « add » zone



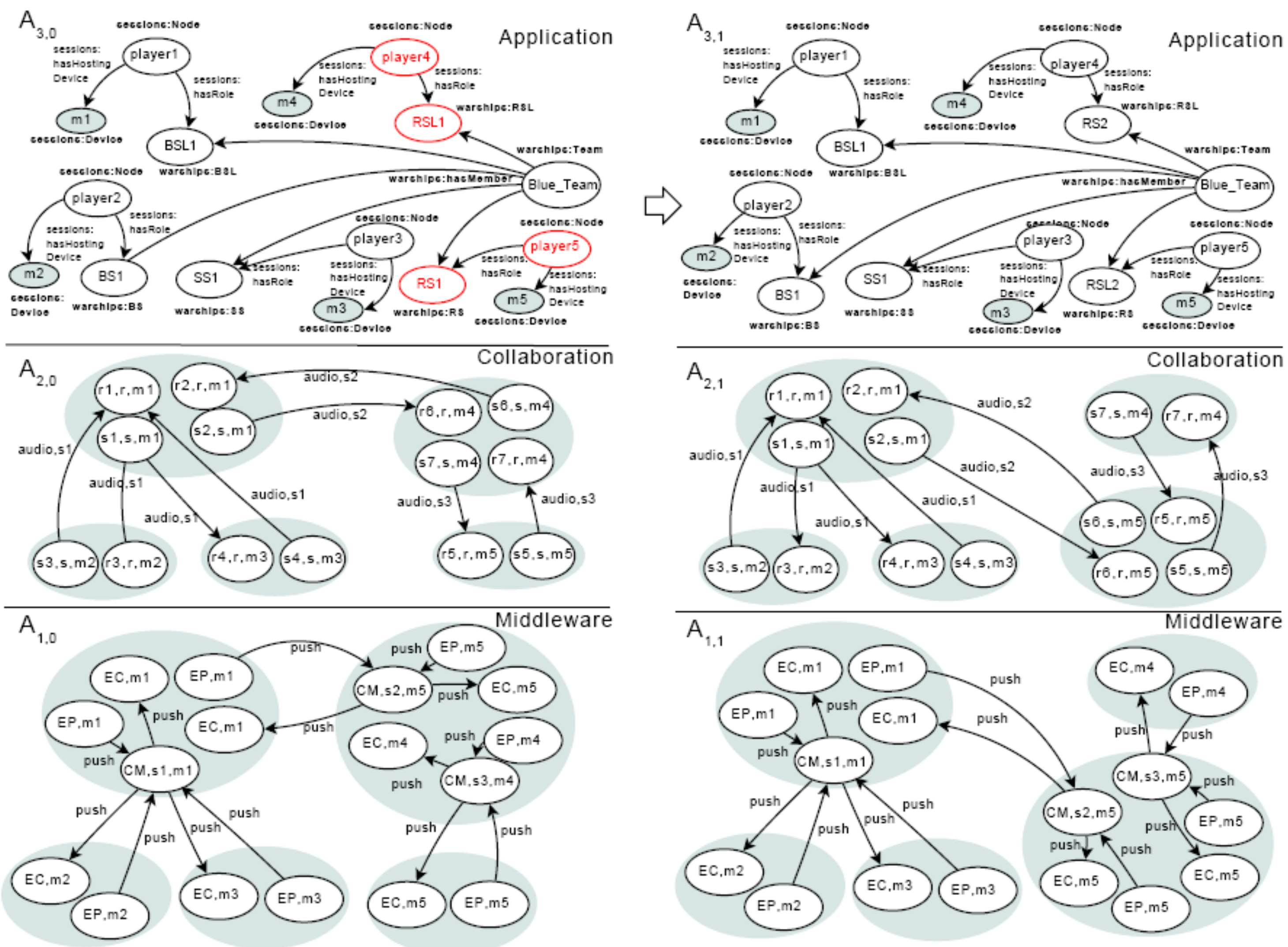
Performance

High speed matching small graph (<30 nodes) in huge graph (>1000 nodes)
More the model graph is connected more the matching is fast

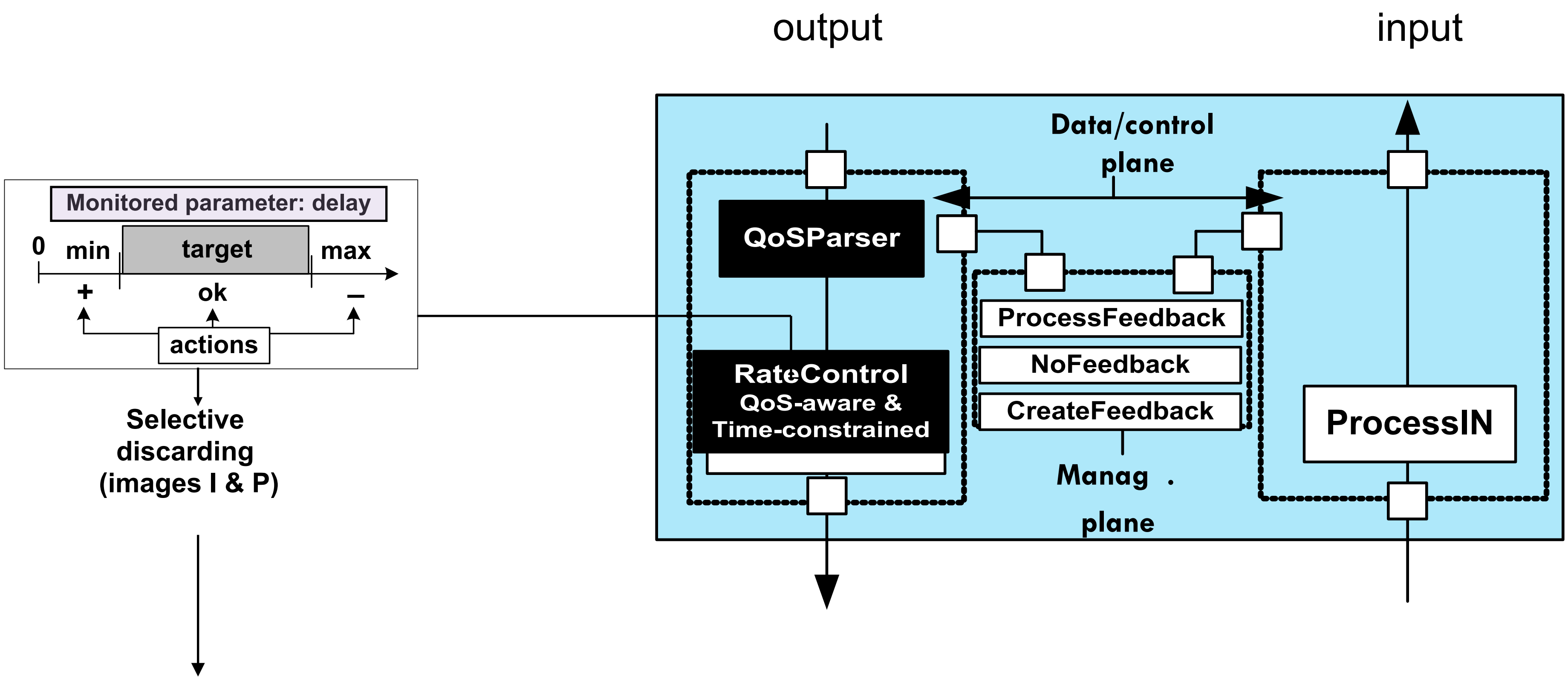




The intra-level reconfiguration and inter-level refinement models integrating FACUS and GMTE



Example for the transport level



H.263 VIDEO	Partial reliability (degradation)	Time constraints
Interactive :	(I,P) = {(100,100),(100,50),(50,0)}	(Tmin,Tmax)= (25,400)

LAAS-CNRS

Our ultimate perspective Autonomic Service Bus (ASB)

Sensing/actuating Services: meters (temp, pressure, energy), switches (light, pumps), detectors (mouv.)

Communication Services: routing and transmission protocols, Event services (CEP, DDS, AMQP)

Infrastructure-level services: monitoring, diagnostic, reconfiguration

System-level services: self-protection (RBAC, SAML), reliability, optimization

Dynamic discovery

Dynamic adaptation

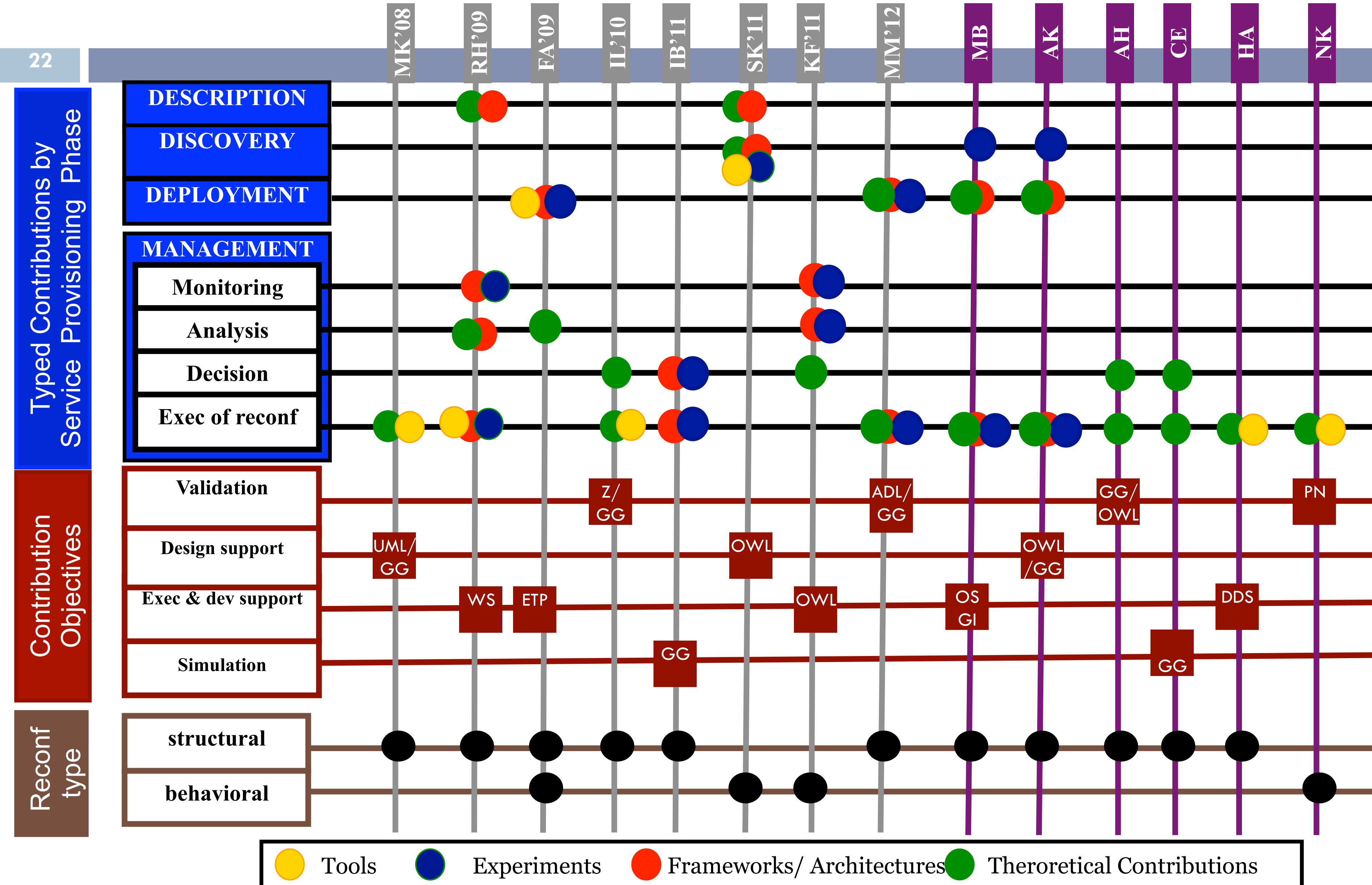
Dynamic composition

ASB: semantic-enabled, group-wide, secured e2e ubiquitous and autonomic service bus

Communicating objects: robots, smart devices (phones, lumps, meters), other machines (heating pumps, doors, sunblind,..), gateways

Automated management models: Structure (graph grammars, graphs), behavior (Petri Nets), function (Z), performance (queue models), semantic (ontology)

Recent PhD contributions



USENET(2007-2010) : 17 partners from France, Spain, Belgium, Finland

A2NETS (2010-2014) : 24 partners from France, Spain, Turkey, Finland

ITEA 2

INFORMATION TECHNOLOGY FOR EUROPEAN ADVANCEMENT

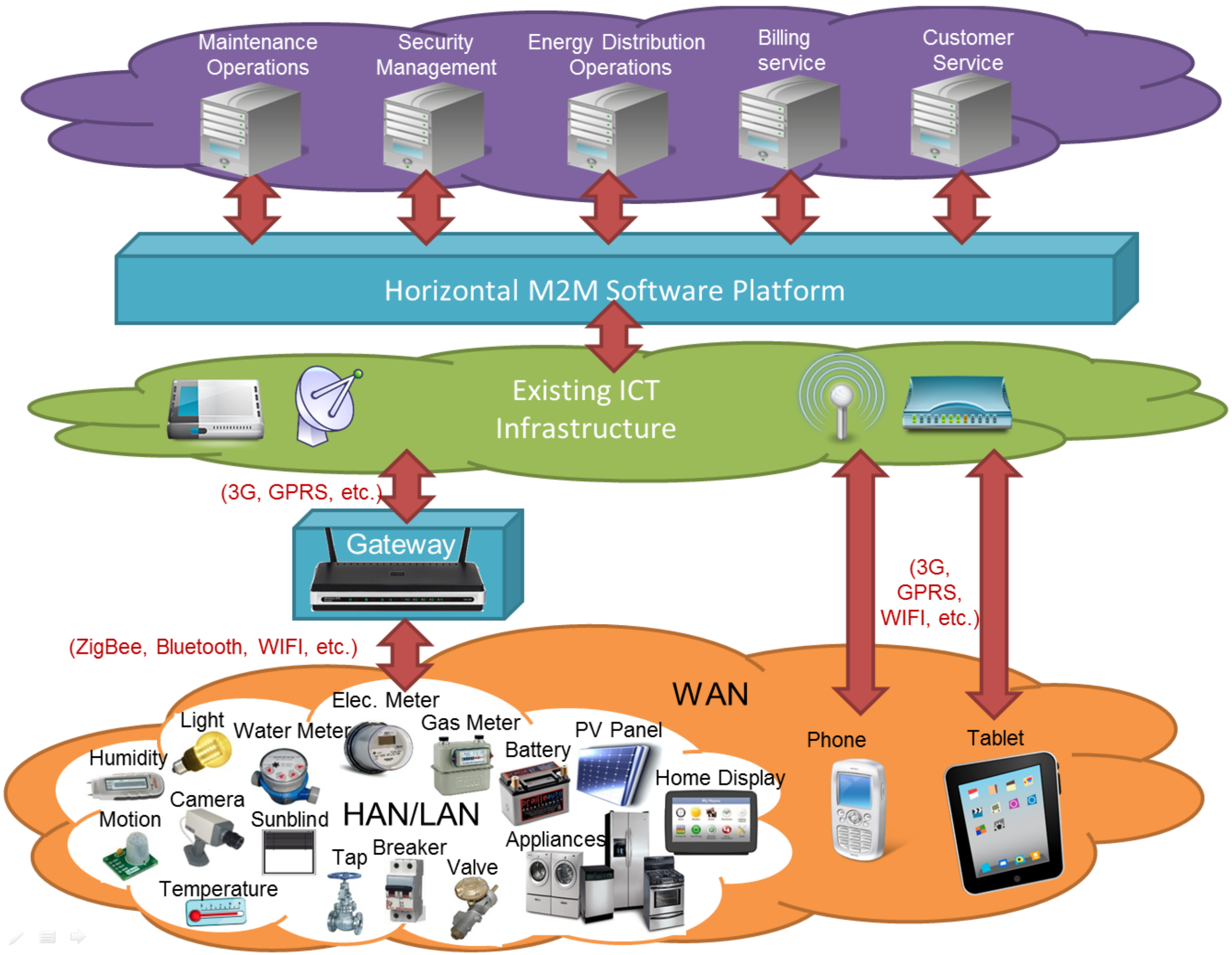
USENET: Ubiquitous M2M Service Networks

A2NETS: Autonomic services in M2M Networks

ITEA2-A2NETS project
Smart Metering business case

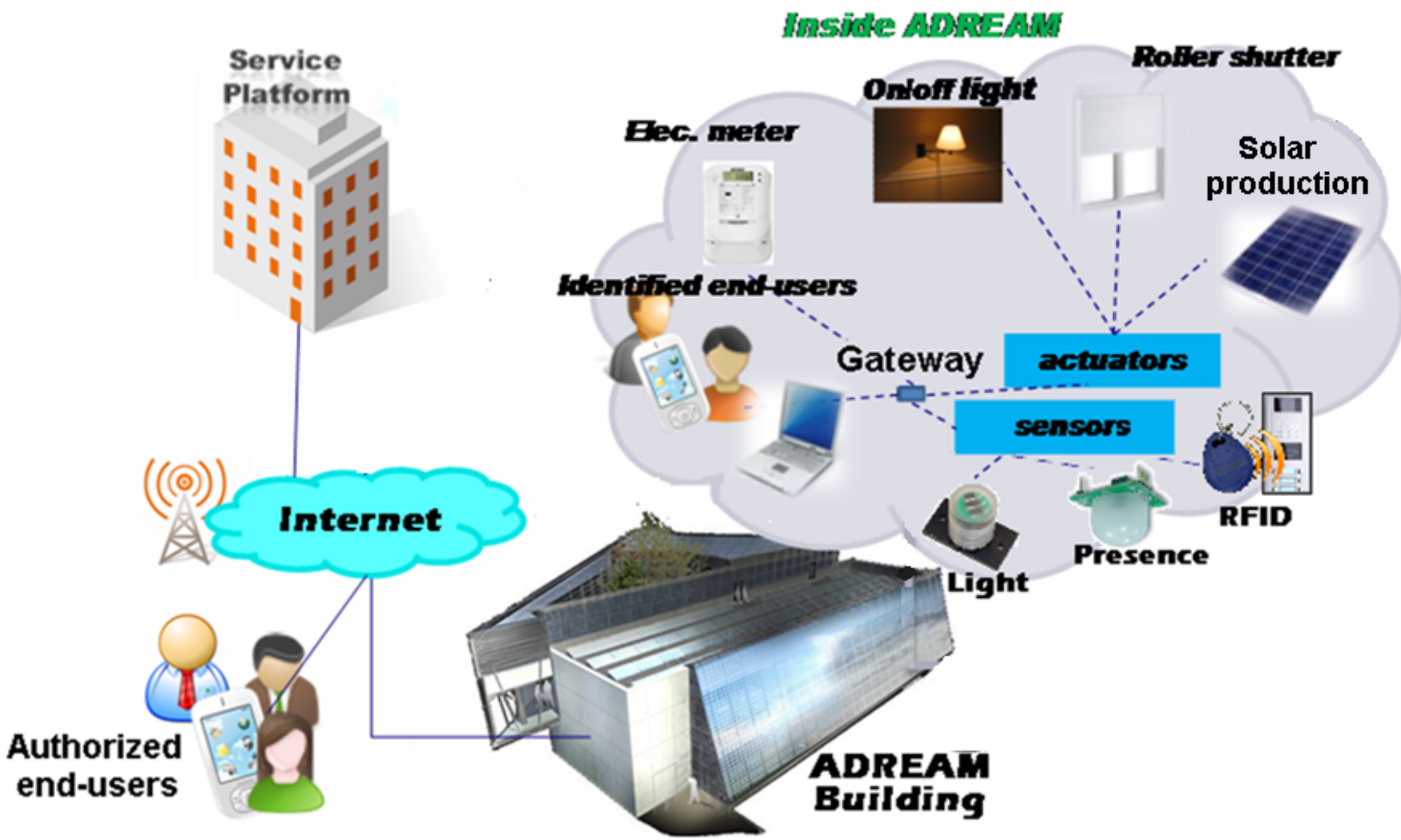
ITEA 2
INFORMATION TECHNOLOGY FOR EUROPEAN ADVANCEMENT

LAAS-CNRS A2NETS



Smart Metering M2M architecture

ITEA2-A2NETS project
Smart Metering business case



Smart Metering M2M plateforme

Smart Metering and M2M technologies and standards in Europe

ETSI

ESMIG
EUROPEAN SMART METERING INDUSTRY GROUP

Example of a M2M application (from 1st ITEA/Usenet project telemetry scenario : see tomorrow presentation of Johanna Kallio, VTT

protect the gateway of the ADREAM building from damage caused twice by oversized trucks, before and after putting traffic signs

Acknowledgments

Mahdi Ben Alaya

Ernesto Exposito

Ghada Gharbi

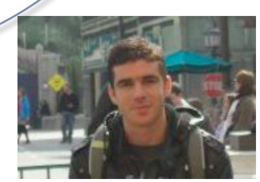
Amine Hannachi

Aymen Kamoun

Thierry Monteil

Emna Mezghani

Said Tazi



Thank you for your attention

Common definitions

Machine-to-machine (M2M) refers to technologies that allow both wireless and wired systems to communicate with other devices of the same ability.[1][2] M2M uses a *device* (such as a sensor or meter) to capture an *event* (such as temperature, inventory level, etc.), which is relayed through a *network* (wireless, wired or hybrid) to an *application* (software program), that translates the captured event into *meaningful information* (for example, items need to be restocked).[3] Such communication was originally accomplished by having a remote network of machines relay information back to a central hub for analysis, which would then be rerouted into a system like a personal computer.[4]

However, modern M2M communication has expanded beyond a one-to-one connection and changed into a system of networks that transmits data to personal appliances. The expansion of wireless networks across the world has made it far easier for M2M communication to take place and has lessened the amount of power and time necessary for information to be communicated between machines.[5] These networks also allow an array of new business opportunities and connections between consumers and producers in terms of the products being sold.[6]

Autonomic Computing refers to the self-managing characteristics of distributed computing resources, adapting to unpredictable changes while hiding intrinsic complexity to operators and users. Started by IBM in 2001, this initiative ultimately aims to develop computer systems capable of self-management, to overcome the rapidly growing complexity of computing systems management, and to reduce the barrier that complexity poses to further growth.

An autonomic system makes decisions on its own, using high-level policies; it will constantly check and optimize its status and automatically adapt itself to changing conditions. An autonomic computing framework is composed of autonomic components (AC) interacting with each other. An AC can be modeled in terms of two main control loops (local and global) with sensors (for self-monitoring), effectors (for self-adjustment), knowledge and planner/adapter for exploiting policies based on self- and environment awareness.