Multi-agent Self-organization and Reorganization to Adapt M2M Infrastructures

Gauthier Picard\textsuperscript{*†} C. Persson\textsuperscript{*†} O. Boissier\textsuperscript{*†} F. Ramparany\textsuperscript{‡}

\textsuperscript{*} MINES Saint-Étienne
\textsuperscript{†} LaHC UMR CNRS 5516
\textsuperscript{‡} Orange Labs/COSY

September 24th, 2015
Machine-to-Machine (M2M)

⚠️ Used by multiple applications
M2M Infrastructures

- **Vertical M2M Infrastructure**
  - ✓ Control over Devices
  - ✗ Expensive Cost
  - ✗ Limited reuse between Apps
  - ✗ Small Scale Deployment
M2M Infrastructures

- **Vertical M2M Infrastructure**
  - ✔ Control over Devices
  - ✗ Expensive Cost
  - ✗ Limited reuse between Apps
  - ✗ Small Scale Deployment

- **Multi-Vertical M2M Infrastructure**
  - ✔ Control over Devices
  - ✔ Shared Infrastructure Cost
  - ✗ Limited reuse between Apps
  - ✗ Medium Scale Deployment
M2M Infrastructures

- **Vertical M2M Infrastructure**
  - ✓ Control over Devices
  - ✗ Expensive Cost
  - ✗ Limited reuse between Apps
  - ✗ Small Scale Deployment

- **Multi-Vertical M2M Infrastructure**
  - ✓ Control over Devices
  - ✓ Shared Infrastructure Cost
  - ✗ Limited reuse between Apps
  - ✗ Medium Scale Deployment

- **Horizontal M2M Infrastructure**
  - ✗ Less Control over Devices
  - ✓ Shared Infrastructure Cost
  - ✓ Reuse of Devices between Apps
  - ✓ Large Scale Deployment
M2M Infrastructures

- Vertical M2M Infrastructure
  - ✔ Control over Devices
  - ✗ Expensive Cost
  - ✗ Limited reuse between Apps
  - ✗ Small Scale Deployment

- Multi-Vertical M2M Infrastructure
  - ✔ Control over Devices
  - ✔ Shared Infrastructure Cost
  - ✗ Limited reuse between Apps
  - ✗ Medium Scale Deployment

- Horizontal M2M Infrastructure
  - ✗ Less Control over Devices
  - ✔ Shared Infrastructure Cost
  - ✔ Reuse of Devices between Apps
  - ✔ Large Scale Deployment
Problem Statement

■ How to balance Vertical and Horizontal Benefits?
  ▶ How to control the horizontal M2M infrastructure to guarantee Devices Constraints and Application Requirements?
  ▶ How to adapt the M2M infrastructure Dynamics and Scale?

■ Definition of a Machine-to-Machine specific Governance
  ▶ Automated Control and Adaptation Governance processes
  ▶ Decentralized and Distributed Governance processes
Outline

M2M, Governance and Adaptation

AGAMEMNON : an Adaptive Governance MAS for M2M

Self-reorganization in AGAMEMNON

Implementation and Evaluation

Conclusion
Governance Principles in ICT

COBIT \((HBC^+ 07)\), IBM SOA Governance \((BC06)\)

- **3 Governance Levels**
  - **Strategy**: *Functional / Non-Functional Requirements Specification*
    e.g. IT Goals, Requirements Planning
  - **Tactic**: *Intermediate Goals Definition*
    e.g. Process Goals, Approach Definition
  - **Policy**: *Resource Usage Management*
    e.g. Activity Goals, Model Enabling

- **Governance Process**
  - **Governance Refinement**: *Performance Analysis and Redefinition*
    e.g. Measurement and Realignment, Measurement and Refinement

G. Picard et al.  
Multi-agent Self-organization and Reorganization to Adapt M2M Infrastructures  
6
Governance Principles in M2M

- **Strategy**: Functional / Non-Functional Requirements Specification
  - Horizontal: ETSI TC M2M – Functional Architecture (ETS11)
  - Vertical: Service Level Agreement (SLA) Ad-hoc Application Requirements (We09, SK09+)

- **Tactic**: Intermediate Goals Definition

- **Policy**: Resource Usage Management
  - Device Management Recommendations (MB09)
  - Device Management Protocols (ASSC02, LBABV12)
  - Platform’s Device Abstraction (She10, GGB08, BB08)

- **Governance Process**: Governance Adaptation: Performance Analysis and Redefinition
Governance Principles in M2M

3 Governance Levels

- **Strategy**: Functional / Non-Functional Requirements Specification
  - Horizontal: ETSI TC M2M – Functional Architecture
  - Vertical: Service Level Agreement (SLA)
  - Ad-hoc Application Requirements

- **Tactic**: Intermediate Goals Definition

- **Policy**: Resource Usage Management
  - Device Management Recommendations
  - Device Management Protocols
  - Platform’s Device Abstraction

Governance Process

- **Governance Adaptation**: Performance Analysis and Redefinition
Governance Approaches in M2M Control and Adaptation Systems

Endogenous Control and Adaptation

- Mix the Business Logic processes and the Governance processes

- **Ad-Hoc Adaptative Algorithms**
  - Stochastic/Probabilistic Algorithms
  - Swarm Intelligence
  - Clustering Algorithm

- **Governance Pros/Cons**
  - Requirements Definition: Not Explicit, Less Dynamic, Procedural
  - Control: Bottom-Up, No Global State, No Guaranty
  - Adaptation: Fast Reactivity, Locally

Exogenous Control and Adaptation

- Separate the Business Logic processes and the Governance processes

- **Requirement-based Control Loop**
  - Middleware
  - Autonomic Computing
  - Multi-Agent Systems (MAS)

- **Governance Pros/Cons**
  - Requirements Definition: Explicit, Dynamic, Declarative
  - Control: Top-Down, Global State, Better Guaranty
  - Adaptation: Less Reactive, Globally
  - Combined with Endogenous
Governance Approaches in M2M
Control and Adaptation Systems

Endogenous Control and Adaptation

- Mix the Business Logic processes and the Governance processes

  - **Ad-Hoc Adaptative Algorithms**
    - Stochastic/Probabilistic Algorithms
    - Swarm Intelligence
    - Clustering Algorithm

  - Governance Pros/Cons
    - Requirements Definition: Not Explicit, Less Dynamic, Procedural
    - Control: Bottom-Up, No Global State, No Guaranty
    - Adaptation: Fast Reactivity, Locally

Exogenous Control and Adaptation

- Separate the Business Logic processes and the Governance processes

  - **Requirement-based Control Loop**
    - Middleware
    - Autonomic Computing
    - Multi-Agent Systems (MAS)

  - Governance Pros/Cons
    - Requirements Definition: Explicit, Dynamic, Declarative
    - Control: Top-Down, Global State, Better Guaranty
    - Adaptation: Less Reactive, Globally
    - **Combined with Endogenous**
Multi-Agent Oriented Programming

Organization
- Collective Behavior: *Role–Social Goals–Norms*
- Functional / Non-Functional Requirements

Agent
- Reactive and Proactive Processes (e.g. BDI)
- Requirement → Control Decisions

Environment
- First-class Abstraction: *Resources, Communication*
- External Business Entities Abstraction

Interaction
- Protocols to bind all MAS entities

Reorganization
- Redefine the Organization Specification/Entity
- Top-Down / Bottom-Up
Multi-Agent Oriented Programming

Organization ................. Strategy
- Collective Behavior: Role–Social Goals–Norms
- Functional / Non-Functional Requirements

Agent ............................. Tactic
- Reactive and Proactive Processes (e.g. BDI)
- Requirement → Control Decisions

Environment .................... Policy
- First-class Abstraction: Resources, Communication
- External Business Entities Abstraction

Interaction
- Protocols to bind all MAS entities

Reorganization ............... Adaptation
- Redefine the Organization Specification/Entity
- Top-Down / Bottom-Up

G. Picard et al.
Multi-agent Self-organization and Reorganization to Adapt M2M Infrastructures

VOWELS dimensions (Dem95)
Outline

M2M, Governance and Adaptation

AGAMEMNON: an Adaptive Governance MAS for M2M

Self-reorganization in AGAMEMNON

Implementation and Evaluation

Conclusion
Overview of the AGaMEMNON Framework

- **Organization-based Governance Strategy**
  - Normative
  - MOISE (HB02)

- **Agent-based Governance Tactic**
  - Declarative
  - Jason (BHW08)

- **Artifact-based Governance Policy**
  - Resource Abstraction
  - CArtAgO (ORV08)

- **Synergistic Integration**
  - JaCaMo (PBRH10)

- **MAS Adaptation Governance**
  - Refinement cycle
    - Top-Down Enactment
    - Horizontal Adaptation
    - Bottom-Up Reorganization
Organization-based M2M Governance Strategy
A Multi-Organizational Specification

1 Horizontal Organization
ETSi TC M2M Functional Architecture

Several Vertical Organizations
M2M Vertical Contract
  - End-to-End communication schema
  - Between M2M entities
  - Service Level Agreement (SLA)

- Horizontal Flexibility
- Vertical Maintenance
Agent-based Governance Tactic

Agent *governs 1 M2M Entity*: binds Horizontal and Vertical Strategy

- **Application Agents**
  - Application point of view
  - (O1) Maximize utility of contract
  - (O2) Minimize subscription cost

- **Infrastructure Agents**
  - Network point of view
  - (O1) Ensure end-to-end communication
  - (O2) Minimize infrastructure cost

- **Device Agents**
  - Device Group point of view
  - (O1) Maximize #subscriptions
  - (O2) Maximize device lifetime
Agent-based Governance Tactic

Agent *governs 1 M2M Entity*: binds Horizontal and Vertical Strategy

- **Application Agents**
  - Application point of view
    - (O1) Maximize utility of contract
    - (O2) Minimize subscription cost

- **Infrastructure Agents**
  - Network point of view
    - (O1) Ensure end-to-end communication
    - (O2) Minimize infrastructure cost

- **Device Agents**
  - Device Group point of view
    - (O1) Maximize #subscriptions
    - (O2) Maximize device lifetime
Agent-based Governance Tactic

Agent *governs 1 M2M Entity*: binds Horizontal and Vertical Strategy

- **Application Agents**
  - *Application point of view*
  - (O1) Maximize utility of contract
  - (O2) Minimize subscription cost
- **Infrastructure Agents**
  - *Network point of view*
  - (O1) Ensure end-to-end communication
  - (O2) Minimize infrastructure cost
- **Device Agents**
  - *Device Group point of view*
  - (O1) Maximize #subscriptions
  - (O2) Maximize device lifetime
Agent-based Governance Tactic

Agent governs 1 M2M Entity: binds Horizontal and Vertical Strategy

- **Application Agents**
  - Application point of view
  - (O1) Maximize utility of contract
  - (O2) Minimize subscription cost

- **Infrastructure Agents**
  - Network point of view
  - (O1) Ensure end-to-end communication
  - (O2) Minimize infrastructure cost

- **Device Agents**
  - Device Group point of view
  - (O1) Maximize #subscriptions
  - (O2) Maximize device lifetime
Artifact-based Governance Policy

Governance Artifact Architecture
Outline

M2M, Governance and Adaptation

AGAMEMNON : an Adaptive Governance MAS for M2M

Self-reorganization in AGAMEMNON

Implementation and Evaluation

Conclusion
Coupling Bottom-up and Top-down Dynamics

Organization modifies Agents

constrains

Agents modifies Environment

constrains

modifies

Organization
Coupling Bottom-up and Top-down Dynamics

Reorganization =
An organization-centered adaptation process where agents are aware of the state of the organization.
Coupling Bottom-up and Top-down Dynamics

Reorganization =
An organization-centered adaptation process where agents are aware of the state of the organization

Self-organization =
An agent-centered process where the implicit state of the organization is the result of their activities in the environment
Coupling Bottom-up and Top-down Dynamics

Reorganization =
An organization-centered adaptation process where agents are aware of the state of the organization

Self-organization =
An agent-centered process where the implicit state of the organization is the result of their activities in the environment

Self-reorganization =
Agents self-organize to define the reorganization process with respect to the organizational and environmental constraints
Agent-based Governance Tactic

- Governance Tactic: 3 phases
  - Strategy Enactment
  - Horizontal Reorganization
  - Vertical Reorganization

- Strategy Enactment
  - Realization of strategy objectives
  - M2M Infrastructure Configuration
  - Enforce objectives via policies
  - Using Governance Artifact

- Horizontal/Vertical Org. Adaptation
  - Adapt Governance to the M2M Infrastructure Dynamics
Adaptation by Horizontal Self-Organization

- Adapt number of agents to the governance load
  - 1 Agent for 1 M2M Entity (initially)
  - Overloaded Agent → Delegation

- Scale-Out: Governance Coalition
  - Informal Organization for 1 M2M Entity
  - Each agent = 1 subset of Governance Tasks
  - Coordination
    - Heuristic before task commitment
      (new M2M Entity, new SLA, ...)

- Scale-Down: Agent Self-termination
  - Minimal task threshold
  - Delegates remaining tasks
  - Terminates itself
Adaptation by Horizontal Self-Organization

- Adapt number of agents to the governance load
  - 1 Agent for 1 M2M Entity (initially)
  - Overloaded Agent → Delegation

- Scale-Out: Governance Coalition
  - Informal Organization for 1 M2M Entity
  - Each agent = 1 subset of Governance Tasks
  - Coordination
    Heuristic before task commitment
    (new M2M Entity, new SLA, ...)

- Scale-Down: Agent Self-termination
  - Minimal task threshold
  - Delegates remaining tasks
  - Terminates itself
Adaptation by Vertical Reorganization

1. Handle new Vertical Contracts
   - Before adopting a role in a vertical organization
   - Evaluate feasibility of the SLA
   - Potentially propose an Alternative SLA

2. Refine Vertical SLA to M2M Constraints
   - When failing SLA
   - When critical state (e.g. M2M Devices)
Outline

M2M, Governance and Adaptation

AGAMEMNON: an Adaptive Governance MAS for M2M

Self-reorganization in AGAMEMNON

Implementation and Evaluation

Conclusion
Governing SensCity with AGAMEMNON

G. Picard et al. Multi-agent Self-organization and Reorganization to Adapt M2M Infrastructures
Evaluation with a Simulated M2M Infrastructure

Simulation Scenario: Smart Parking Management

- **Smart Parking Scenario**
  - Parking Sensors in different areas
    - Business Area (Peak Hour: 7AM–9AM)
    - Residential Area (Peak Hour: 17PM–20PM)

- **Measure Governance Impact over several (6) years**
  - Average **accuracy** of collected data
  - Average devices’ battery **lifetime**

- **Different Application Requirement profiles**
  - `ON_EVENT`: Notification for each parking state change
  - `NOTIFY(X)`: Notification of parking state each X minutes
  - `ADAPTIVE`: Combines `ON_EVENT` and `NOTIFY(X)` profiles
    - Adaptation to end-user demand

- **Device Battery Constraints**
  - Homogeneous and Affine consumption model (Scenario#1)
  - Heterogeneous and Exponential consumption model (Scenario#2)
Evaluation with a Simulated M2M Infrastructure

Experiment #1: Application Vertical Refinement

- Benefits of an explicit SLA declaration
  - Enable Governance Management

- Application point of view:
  - #1 – ON_EVENT
  - #2 – ADAPTIVE
  - #3 – NOTIFY(X), (X ≤ 20)

- Device point of view:
  - #1 – NOTIFY(X), (X ≥ 30)
  - #2 – ADAPTIVE

- ADAPTIVE profile: best trade-off between application satisfaction and device lifetime
Evaluation with a Simulated M2M Infrastructure

Experiment #2 : Device Regulation and Vertical Refinement

- Battery consumption improvement by Governance actions:
  - Improve Average Device Lifetime

- Maintain an acceptable data accuracy during 5 years
- Improvement to be done after 5 years
  - Finer SLA negotiation tactics
Outline

M2M, Governance and Adaptation

AGAMEMNON: an Adaptive Governance MAS for M2M

Self-reorganization in AGAMEMNON

Implementation and Evaluation

Conclusion
1. A Machine-to-Machine Governance Model
   - Horizontal and Vertical concerns
   - Step to horizontal M2M infrastructure deployment
   - Dynamic, decentralized and adaptive governance framework

2. A Multi-Agent Oriented Programming approach for M2M Governance Adaptation and Control
   - Separate governance concerns along MAS dimensions
   - Multi-organizational specifications
   - MAS integration within a component-based platform
   - Couples an agent-centered vision and an organization-centered vision of the adaptation: **self-reorganization**
Synthesis

1. A Machine-to-Machine Governance Model
   - Horizontal and Vertical concerns
   - Step to horizontal M2M infrastructure deployment
   - Dynamic, decentralized and adaptive governance framework

2. A Multi-Agent Oriented Programming approach for M2M Governance Adaptation and Control
   - Separate governance concerns along MAS dimensions
   - Multi-organizational specifications
   - MAS integration within a component-based platform
   - Couples an agent-centered vision and an organization-centered vision of the adaptation: self-reorganization

What’s next?

- Integrative framework for self-reorganization to design self-adaptive systems
Ian F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci.  
Wireless sensor networks: a survey.  

Tim Baugé and Jesus Bernat.  
Components for End to End Networking, Management and Security and Accounting.  

William A. Brown and Murray Cantor.  
SOA governance: how to oversee successful implementation through proven best practices and methods.  
White Paper, IBM Rational Center, August 2006.

Rafael H. Bordini, Jomi F. Hübner, and Michael Wooldridge.  

Dominique Barthel, Giyyarpuram Madhusudan, and Nicolas Dejean.  
Application characteristics – An applicative framework for the research work conducted in ARESA2.  
http://aresa-project.insa-lyon.fr/.
Bibliography (contd)

Yves Demazeau.
From Interactions To Collective Behaviour In Agent-Based Systems.

ETSI.
Technical Committee Machine-to-Machine Communications (M2M).

State of The Art - Sensor Frameworks and Future Internet.

Jomi F. Hübner and Olivier Boissier.
Bibliography (contd)


Bibliography (contd)

Z. Shelby.
Embedded Web Services.

Patrick Spiess, Stamatis Karnouskos, Dominique Guinard, Domnic Savio, Olivier Baecker, Luciana Moreira Sá Souza, and Vlad Trifa.
SOA-based Integration of the Internet of Things in Enterprise Services.

Wefer, Gerold, editor.