# (Re) Organization model for decentralized management and optimization of green data center: ParaMoise

Mateusz Guzek, Grégoire Danoy, Pascal Bouvry

University of Luxembourg

January 29, 2013

1. Introduction
    - Context and Motivation
    - State of the art
2. ParaMoise
    - Description
    - Examples
3. Summary

1. Introduction
   - Context and Motivation
   - State of the art

2. ParaMoise
   - Description
   - Examples

3. Summary

# Cloud computing and Autonomic computing

- Cloud computing – virtualized data center(s) that processes any type of workload. (It is a large-scale dynamic distributed system.)
- Autonomic computing – bringing self-* properties to system (*={management,optimization,healing,...}).

Both approaches include an underlying organization, e.g.:

- Communication patterns
- Authority
- Responsibilities

# Purpose of (re)organization

The purpose of organization is to facilitate reaching optimal states. In this way, we can define organization's[1]:

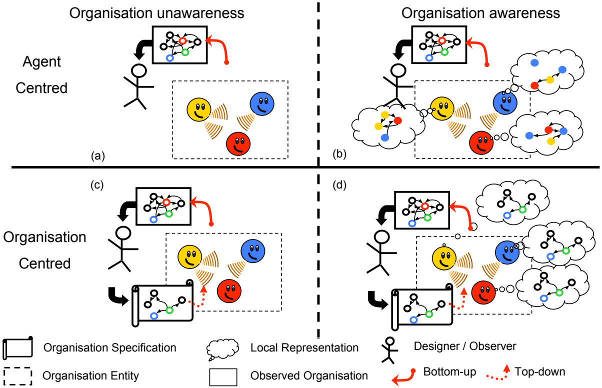
- *Cost*
- *Benefit*
- $Efficiency = Benefit - Cost$

*Efficiency* of an organization depends on the system and environment. In a dynamic system (cloud) both system and environment changes. There is a need to reorganize to achieve optimal *Efficiency*.

[1] Ramachandra Kota, Nicholas Gibbins, and Nicholas R. Jennings. "Decentralised Structural Adaptation in Agent Organisations". In: *AAMAS-OAMAS*. 2008, pp. 54–71.

# MAS organizations classification[2]

In context of MAS, we can always investigate its organization, even in a systems that does not explicitly define it.



source: [2]

[2] G. Picard et al. "Reorganisation and self-organisation in multi-agent systems". In: *International Workshop on Organizational Modeling*. OrgMod'09. 2009, pp. 66–80.

1 Introduction

- Context and Motivation
- State of the art

2 ParaMoise

- Description
- Examples

3 Summary

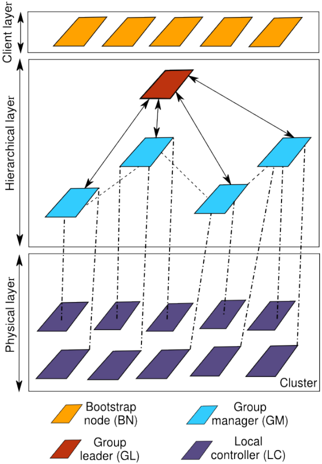
# Distributed management system – Snooze[3]

Snooze is an example of a Multi-Agent System (MAS) with hierarchical organization for:

- Management
- Optimization
- Reliability

Limitations:

- One organization schema
- Reorganization limited to behavior predefined in the schema



source: snooze.inria.fr

[3] E. Feller, L. Rilling, and C. Morin. “Snooze: A Scalable and Autonomic Virtual Machin Management Framework for Private Clouds”. In: *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*. 2012, pp. 482 –489.

# Organizational frameworks – state of the art

Organizational frameworks:

- Centralized (AgentCoRe, GPGP/STAEM)
- Limited reorg. capabilities
- Offline or not specified
- Sequential

Cloud system needs:

- Decentralization (Performance, Reliability)
- Wide reorg. capabilities
- Online reorg.
- Parallelism and concurrency

We have chosen a MOISE+ framework as a base, because of its advancement and its elasticity.

# MOISE+

MOISE+ is an organizational framework. It defines Organization Specification (OS) as:

- Structural Specification (SS)(Roles and their interactions)
- Functional Specification (FS) (Missions and goals)
- Deontic Specification (DS) (Obligation and permissions to commit to missions by roles), binds SS and FS.

The OS is a template used to create Organization Entity (OE), which is a description of a deployed organization (essentially, agents and their relations with organization).

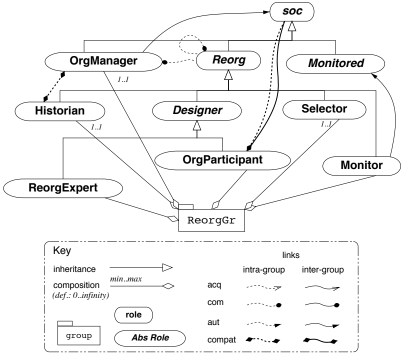
# MOISE+ Reorganization I

Reorganization is a process of chaining organization either on level of:

- OS – e.g. changes in definitions of roles, interaction patterns, missions, obligations.
- OE – e.g. adopting roles, creating missions

# MOISE+ Reorganization II

MOISE+ provides a reorganization capabilities by a set of special roles.



source:[4]

However, the only proposed reorganization implementation (JaCaMo)[a] requires:

1. halting an organization
2. applying changes to organization
3. resuming the organization

It is not feasible in a cloud system.

[a] O. Boissier et al. "Multi-agent oriented programming with JaCaMo". In: *Science of Computer Programming* (2011).

[4] J. Hübner, J. Sichman, and O. Boissier. "Using the $\mathcal{M}$oise$^+$ Model for a Cooperative Framework of MAS Reorganisation". In: *Advances in Artificial Intelligence, SBIA 2004*. Vol. 3171. Springer Berlin / Heidelberg, 2004, pp. 481–517.

1 Introduction
- Context and Motivation
- State of the art

2 ParaMoise
- Description
- Examples

3 Summary

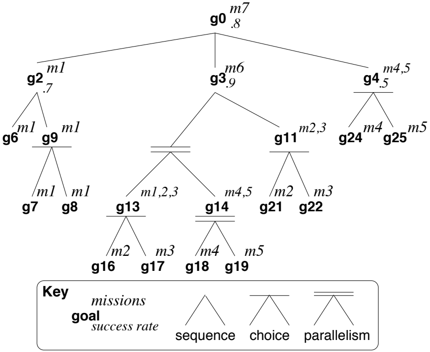
# ParaMoise

ParaMoise enables parallel, distributed reorganization on OS or OE level at runtime. It is based on the following changes:

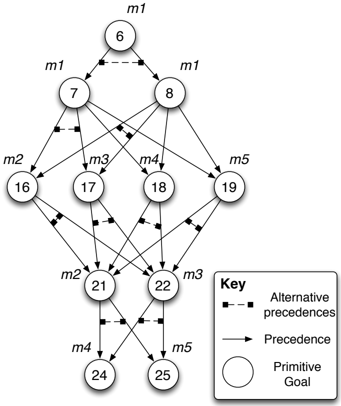
- Introducing a new functional specification
- Adding the lock mechanism
- Enabling multiple organization managers

# Workflow Specification – novel FS

Basic MOISE+ FS is based on tree structure:



ParaMoise introduces Workflow Specification (WFS):



source:[5]

[5] J. Hübner, J. Sichman, and O. Boissier. "A Model for the Structural, Functional, and Deontic Specification of Organizations in Multiagent Systems". In: *Advances in Artificial Intelligence*. Vol. 2507. Springer Berlin / Heidelberg, 2002, pp. 439–448.

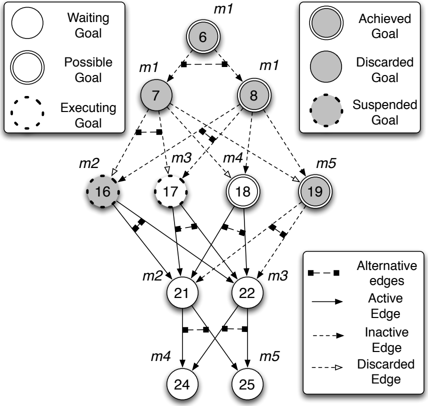
# Workflow execution

Workflow enables:

- arbitrary dependencies
- parallel execution
- correct execution: mutual exclusion,progress, deadlock free
- repetition of goals

Additional features:

- progress tracking
- alternative execution paths
- repetitions (failure handling)



# Locks

A reorganization modifies the state of the organization.

The proposed method of minimal interaction with organization are *locks*:

A lock defines elements of organization that must remain unchanged as long as the lock exists.

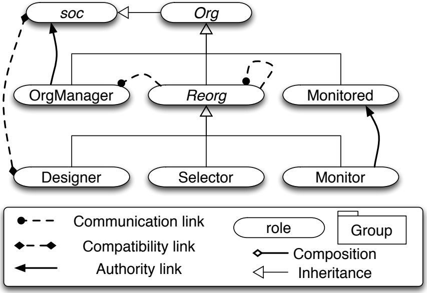
There are two types of locks:

1. read – can overlap with other read locks
2. write – can not overlap with other locks

Write locks are used when an element is modified or removed. Read locks are used when an element existence is needed (e.g. definition of a role if reorganization includes the adoption of that role).

# Concurrent and parallel reorganization

- Locks enable concurrent reorganizations (the only requirement is that existing locks do not overlap).
- Additional requirement: multiple organization managers (as OrgManager is the only role that has permission to change elements of organization).
- Multiple OrgManagers: removal of the potential bottleneck and single point of failure.

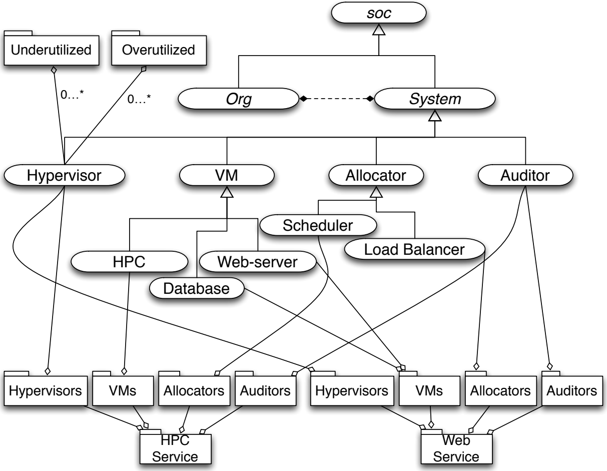


1. Introduction
   - Context and Motivation
   - State of the art

2. ParaMoise
   - Description
   - **Examples**

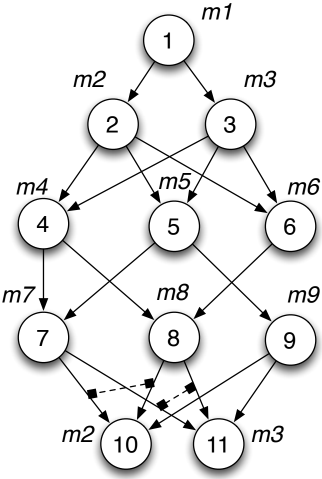
3. Summary

# Sample Organization of Cloud Computing system



1. The tree structure presents the roles and their inheritance.
2. The boxes presents the groups.
3. Group can consist of roles or other groups.

# Small HPC Job – sample workflow

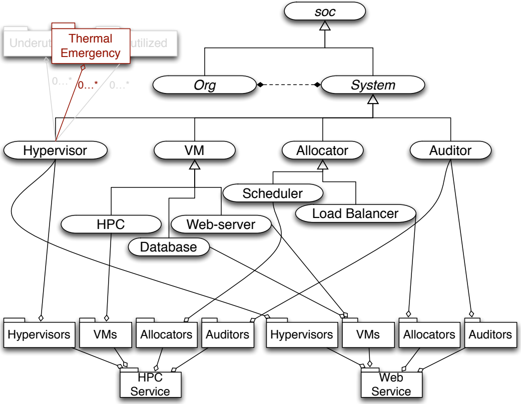


Goals description:

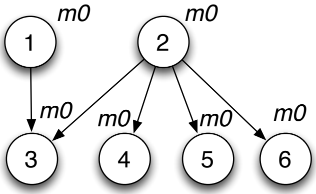
- $g_1$ is the job acceptance goal
- $g_2$ is the preparation of VMs
- $g_3$ is preparation of the audit
- $g_4 - g_6$ are three data preprocessing goals,
- $g_7 - g_9$ are three Monte Carlo simulation goals
- $g_{10}$ is deactivation of unnecessary VMs
- $g_{11}$ finishes the audit and creates its result

# Sample structural reorganization

In case of thermal emergency stop consolidation and decrease utilization of affected nodes.



# Thermal emergency – sample reorganization



Goals description:

- $g_1$: Identify all affected hypervisors (WFS),
- $g_2$: Create a group Thermal Emergency for affected hypervisors,
- $g_3$: The affected hypervisors join Thermal Emergency group,
- $g_4$: Notify VMs
- $g_5$: Notify Auditors
- $g_6$: Notify Allocators

1 Introduction
- Context and Motivation
- State of the art

2 ParaMoise
- Description
- Examples

3 Summary

# Summary

ParaMoise allows to model:

- Parallel execution
- Concurrent and parallel reorganization
- Distributed organization management

# Future work

- Designing exact implementation of organization description and locks
- Designing reorganization phase: monitoring, design, selection.
- Distributed optimization of cloud system

# The End

Thank you for your attention.

Any questions?