

EMSE - séminaires G2I

Embodying Organisations

an introduction

Michele Piunti

Olivier Boissier, Jomi Fred Hubner, Alessandro Ricci

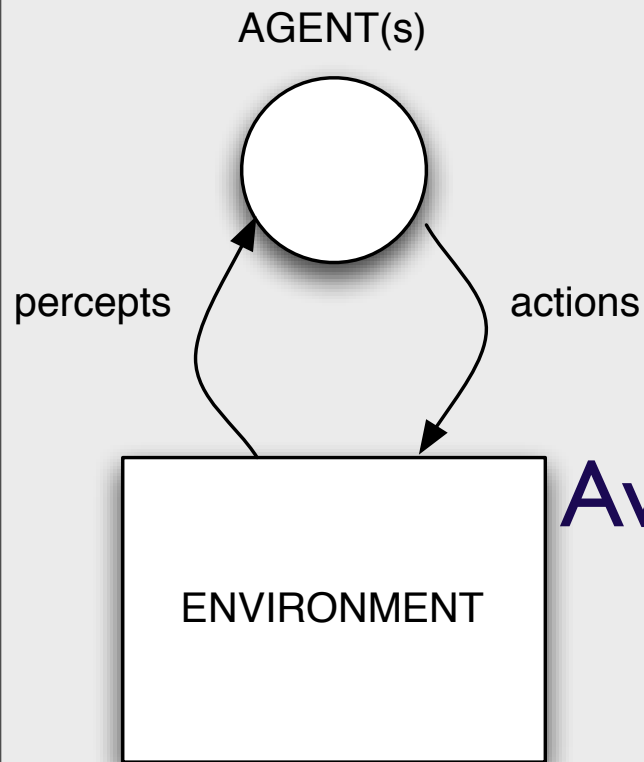
OUTLINE

- Artifact Based Organisations
- Organisation, Agents, Awareness
- Embodied Organisational Artifacts (EOA)
- A Language for Programming Embodied Organisations

PART 1

Artifact Based Organisations (for MAS)

A&A INTERACTIONS

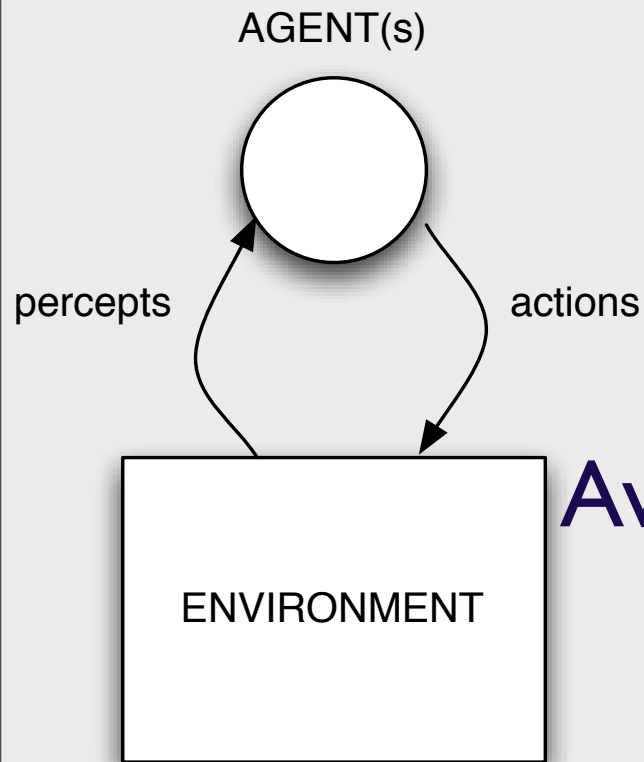


Ease designers (and agents) to tackle problems
at the right domain level

Avoid facing with low level mechanisms and constructs

Maintain the same level of ABSTRACTION
in conceiving interactions from A to A

A&A INTERACTIONS



Ease designers (and agents) to tackle problems
at the right domain level

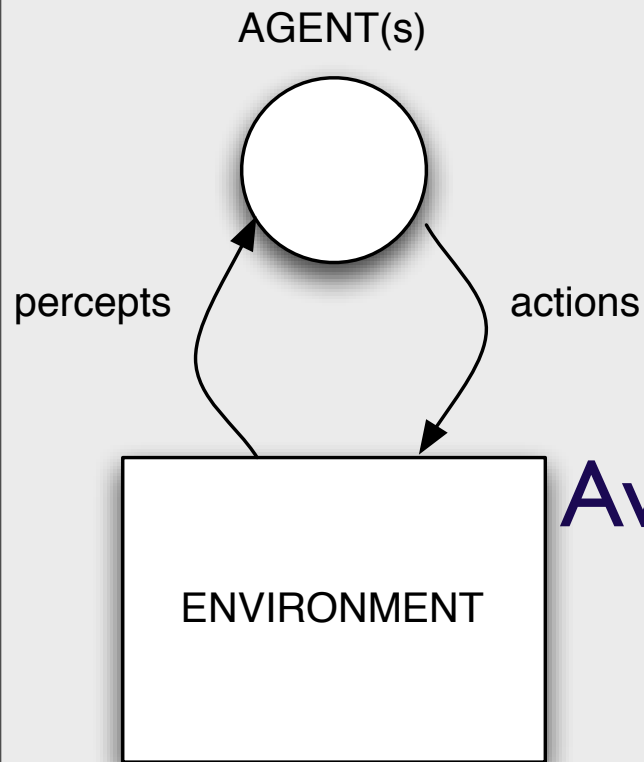
Avoid facing with low level mechanisms and constructs

Maintain the same level of ABSTRACTION
in conceiving interactions from A to A

Agents do not operate upon “component based” middleware, nor Java Objects....

Shaping interaction based on agents *basic activities*: **Action and Perception**

A&A INTERACTIONS



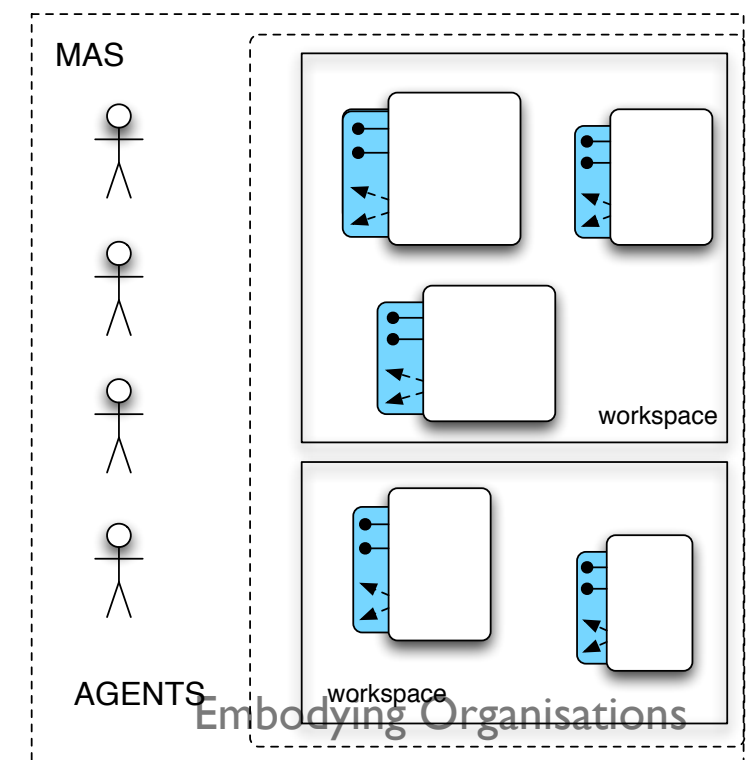
Ease designers (and agents) to tackle problems
at the right domain level

Avoid facing with low level mechanisms and constructs

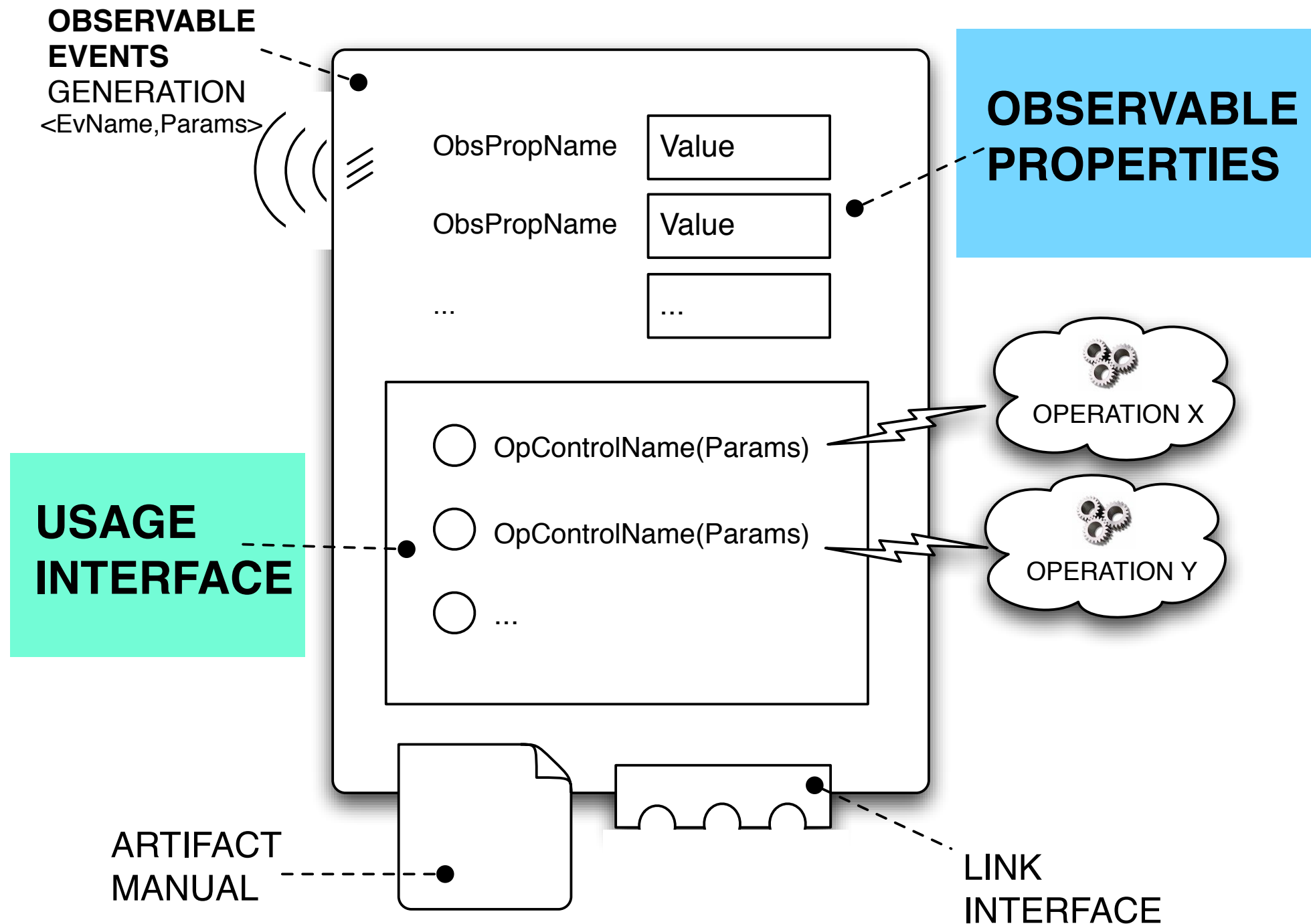
Maintain the same level of **ABSTRACTION**
in conceiving interactions from A to A

Agents do not operate upon “component based” middleware, nor Java Objects....

Shaping interaction based on agents *basic activities*: **Action and Perception**



ARTIFACT COMPUTATIONAL MODEL



A&A Functional Approach

```
(1) joinWorkspace(WName,-Node)
(2) moveToWorkspace(WName,-Node)
(3) quitWorkspace(WName)


---


(4) use(AName,UICntrlName(Params),?SName,?Timeout,?Filter)
(5) sense(SName,?Filter,?Timeout,+Perception)


---


(6) focus(AName,?SName,?Filter)
(7) stopFocussing(AName)


---

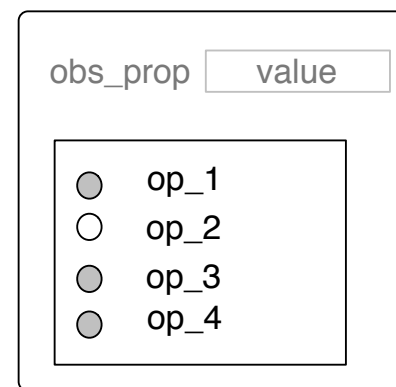
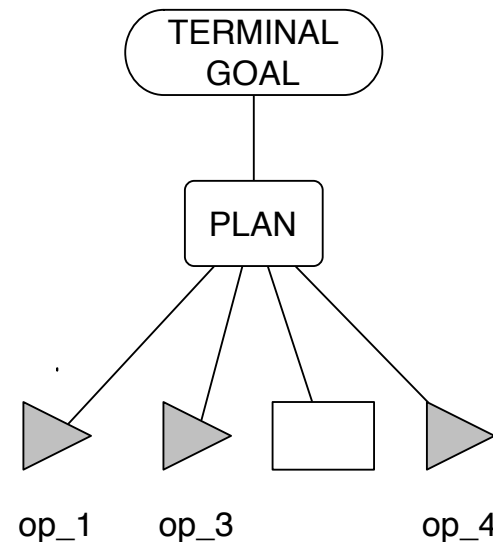

(8) observeProperty(AName,?SName,?PFilter,+Property)
```

A&A Functional Approach

```

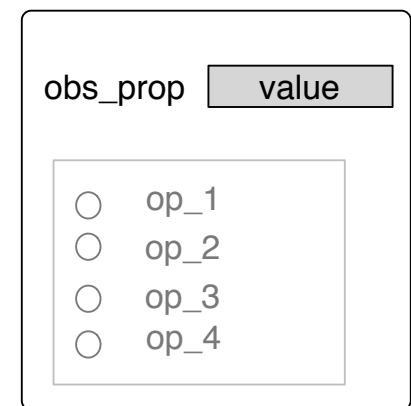
(1) joinWorkspace(WName,-Node)
(2) moveToWorkspace(WName,-Node)
(3) quitWorkspace(WName)
(4) use(AName,UICntrlName(Params),?SName,?Timeout,?Filter)
(5) sense(SName,?Filter,?Timeout,+Perception)
(6) focus(AName,?SName,?Filter)
(7) stopFocussing(AName)
(8) observeProperty(AName,?SName,?PFilter,+Property)
  
```

AGENT PRACTICAL REASONING



ARTIFACT PRACTICAL FUNCTION

AGENT EPISTEMIC REASONING



ARTIFACT EPISTEMIC FUNCTION

ORA4MAS

Power Back To Agents

- Organisations are based on Coordination Artifacts
 - Encapsulating organisational functions
- Decentralised management
- Openness

ORG ARTIFACTS

- Organisational state(s) and rules, conventions and norms are embedded in organisational artifacts
- encapsulate **organisational functionalities**
--services-- for agents
 - functional and purposive approach
 - agents purposes are to exploit org artifacts in order to participate in organisational activities

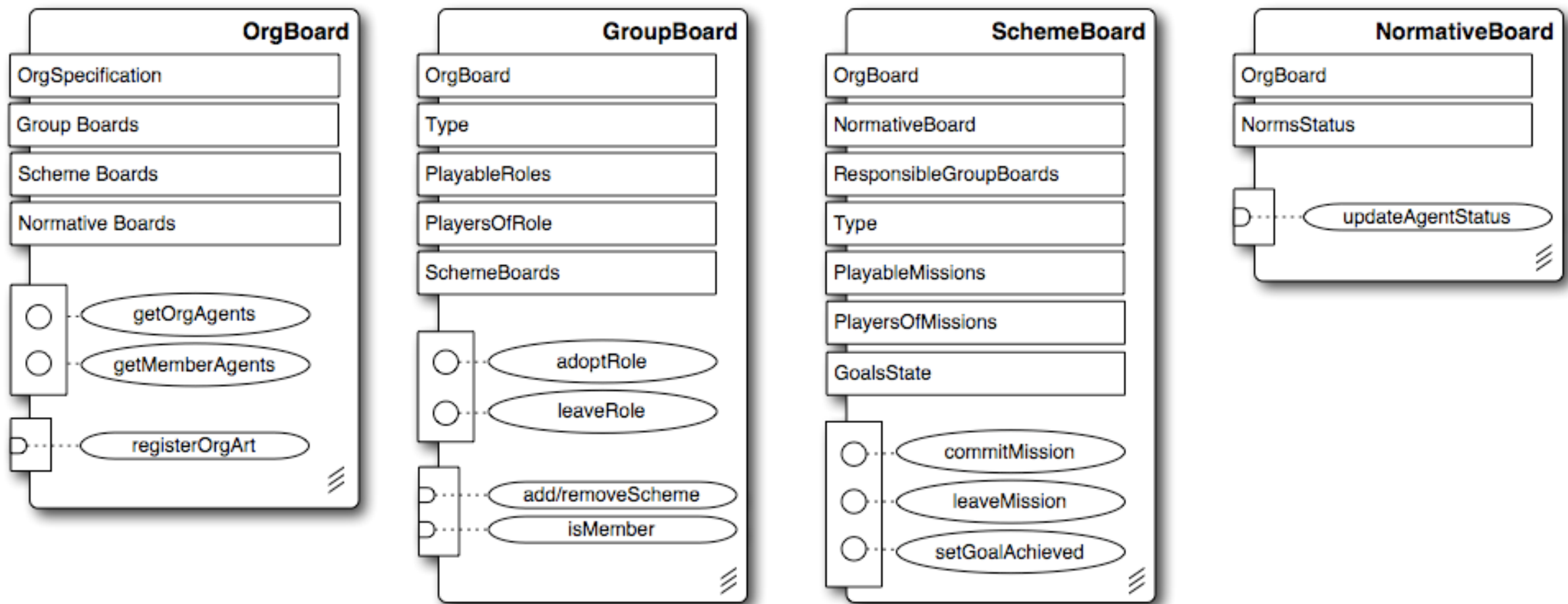
ORG (STAFF) AGENTS

- Organisational dynamics are elicited by (proactive) organisational agents
- Responsible of **management of activities** inside the organisation
 - observing, monitoring, reasoning and deciding about conventions, norm violation, sanctioning

Organisational Specification (OS)

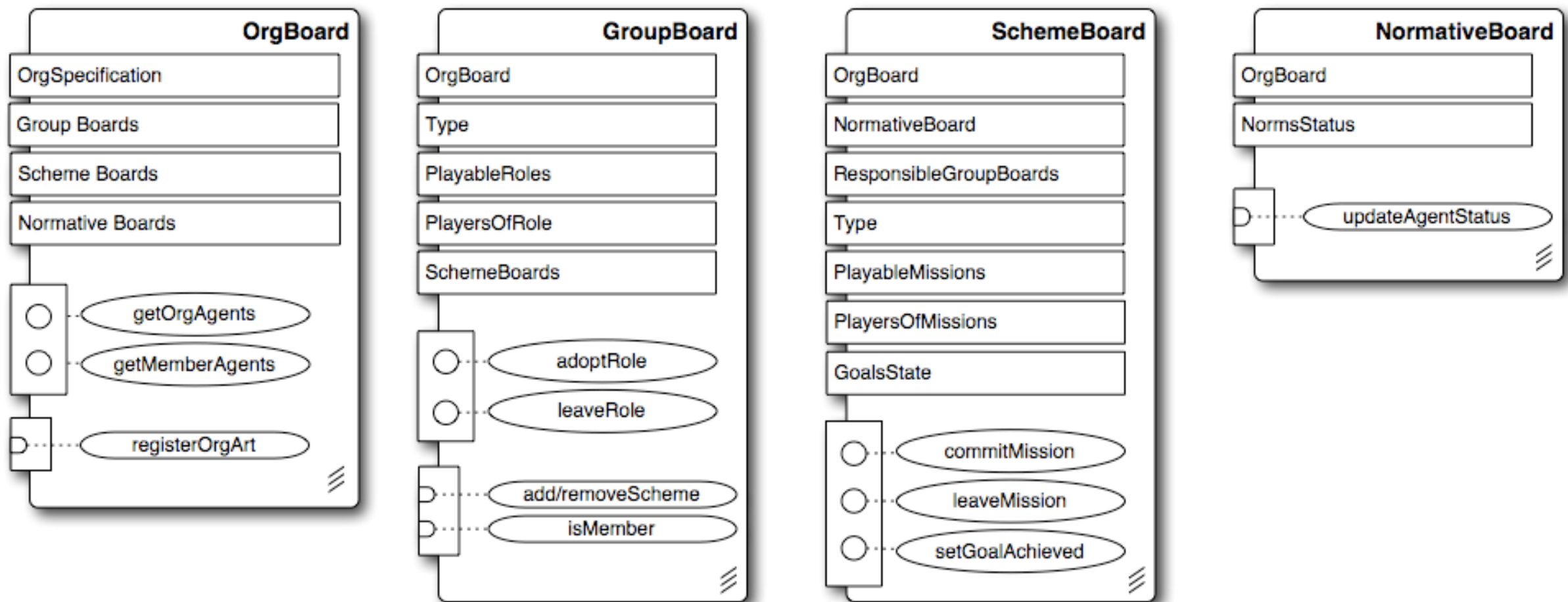
- *Moise+* is actually used as OS in ORA4MAS
- organisational model specifying 3 pivotal dimensions of an organisation
 1. structural specification (roles and groups)
 2. functional specification (goals, missions, schemes)
 3. deontic specification (norms, obligations and permissions)

ORA4MAS



- Few operations triggerable by agents and observable properties allow to manage the overall organisation
- The organisation control agent activities by coordinating them thorough events and signals

ORA4MAS

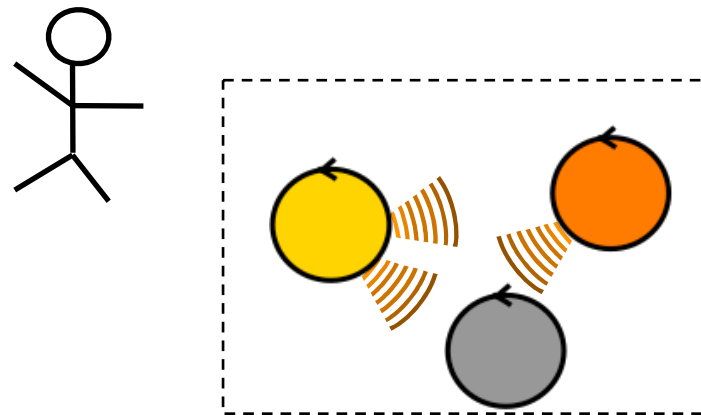


- To bring about organisation, create, modify and use it, agents need to know the semantic of ORA4MAS operations/properties
- **Artifact Relevant Facts:**
 - events, effects, observable states

PART 2

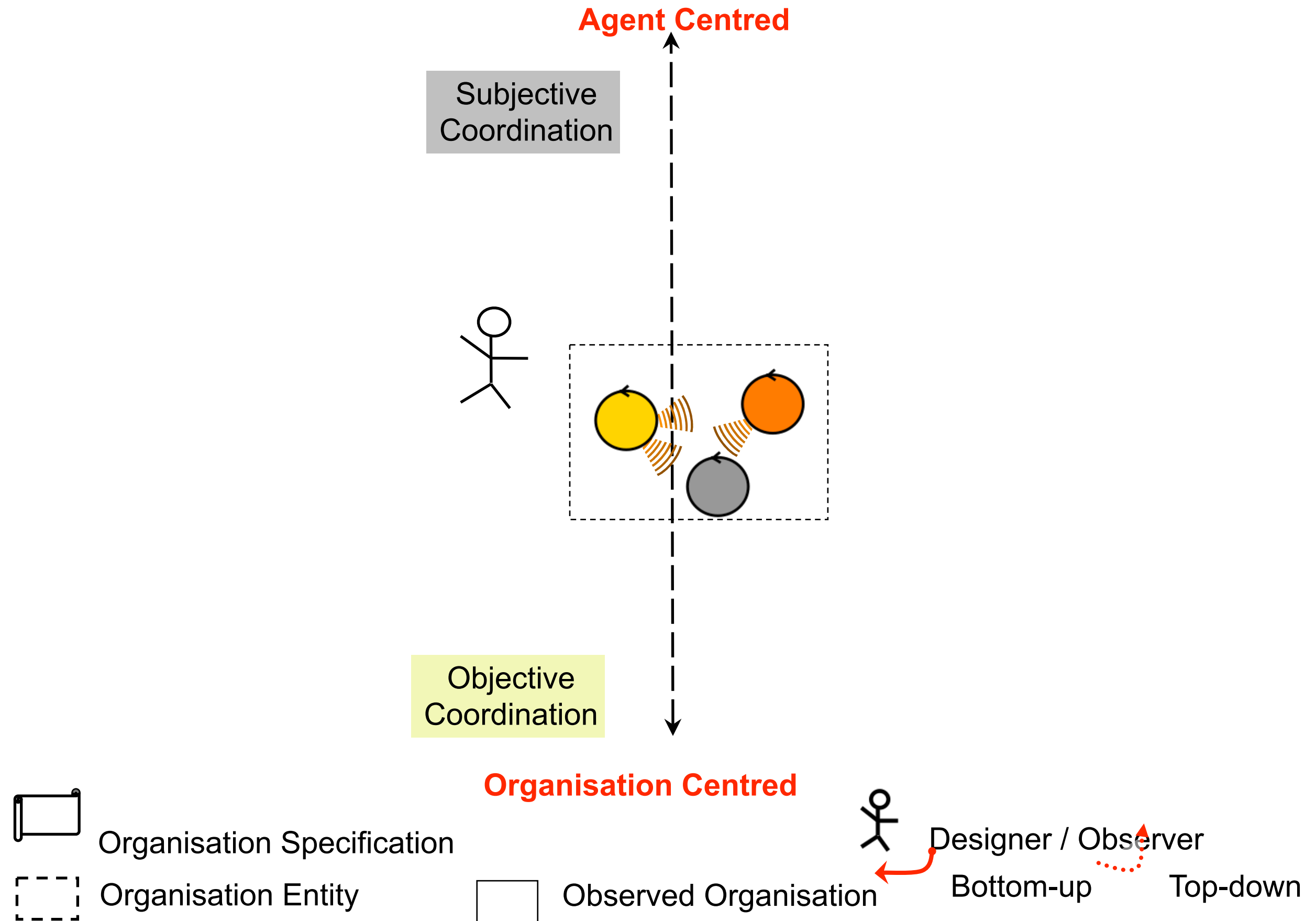
Organisation, Agents, Awareness

Organisations in MAS (global picture)

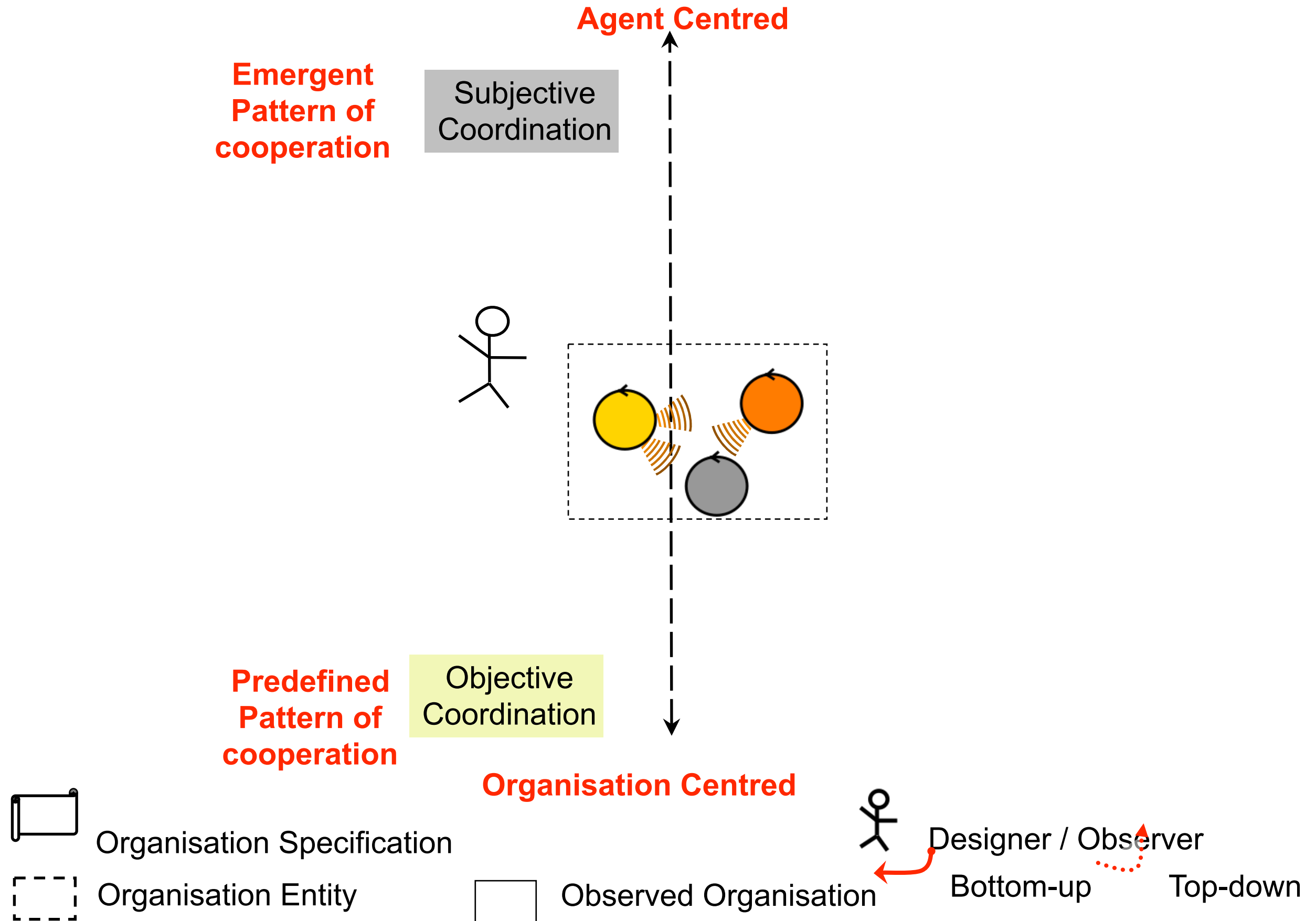


Designer / Observer

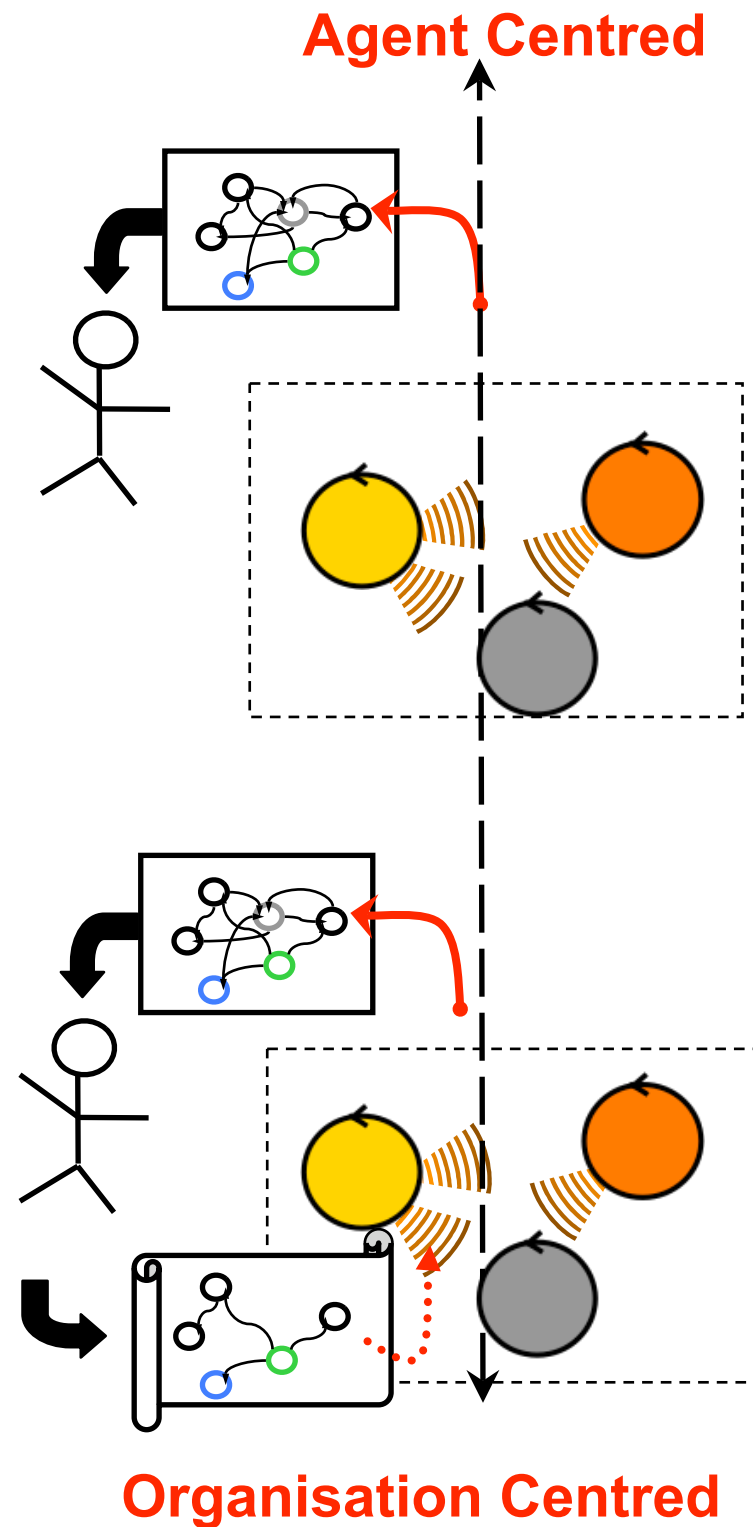
Organisations in MAS (global picture)

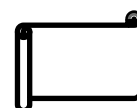


Organisations in MAS (global picture)

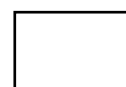


Organisations in MAS (global picture)



 Organisation Specification

 Organisation Entity

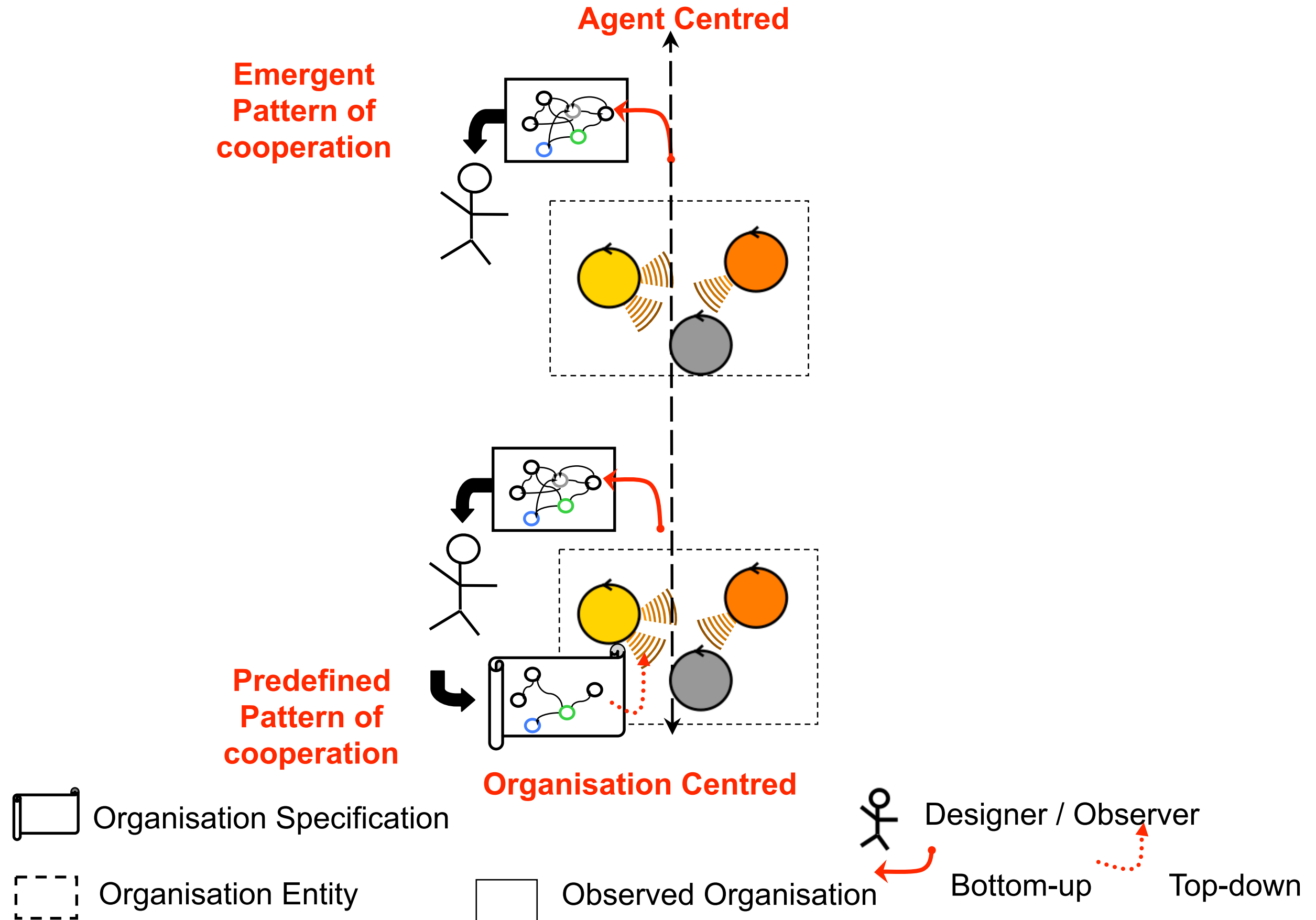
 Observed Organisation

 Designer / Observer

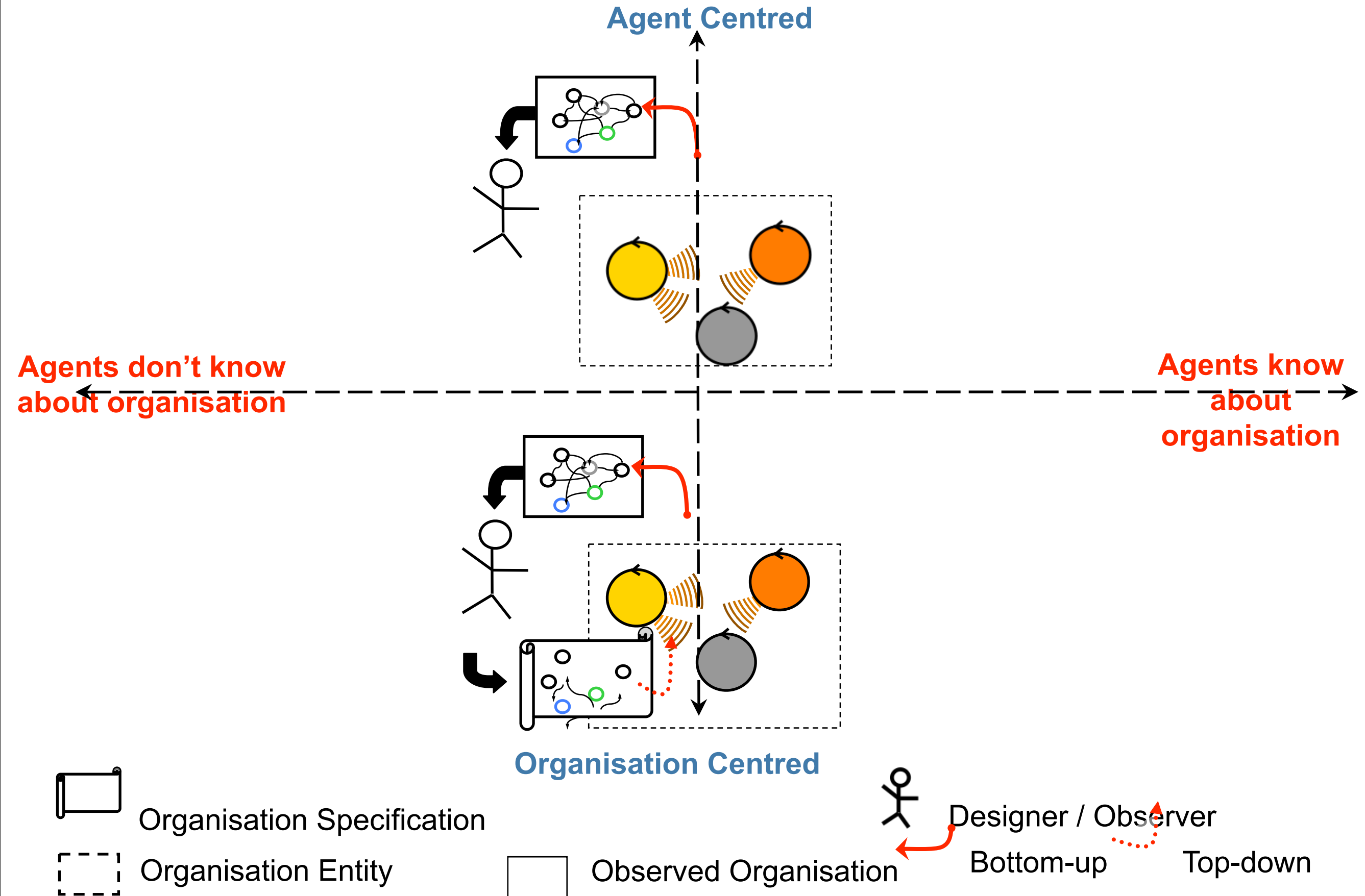
 Bottom-up

 Top-down

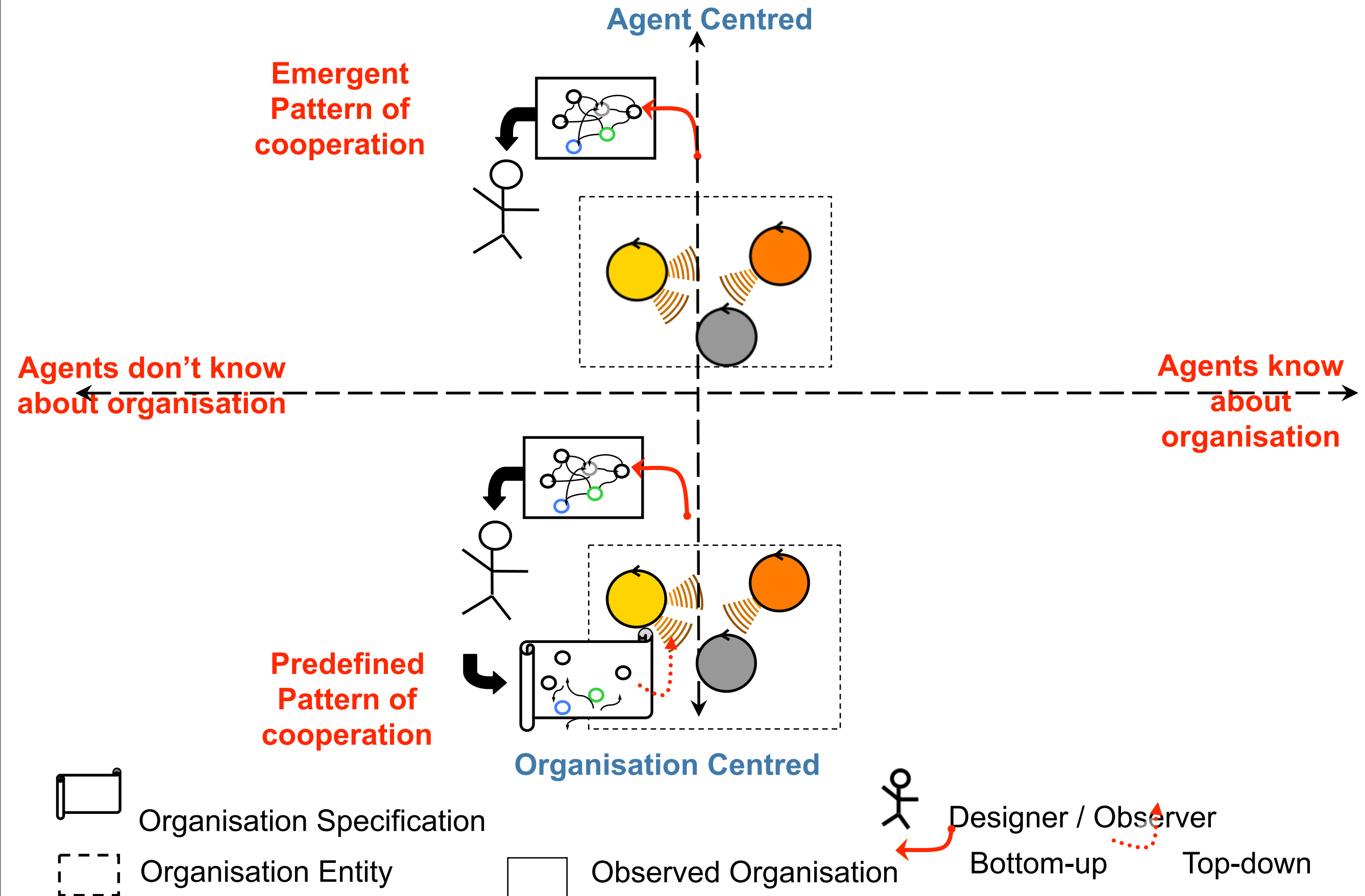
Organisations in MAS (global picture)



Organisations in MAS (global picture)

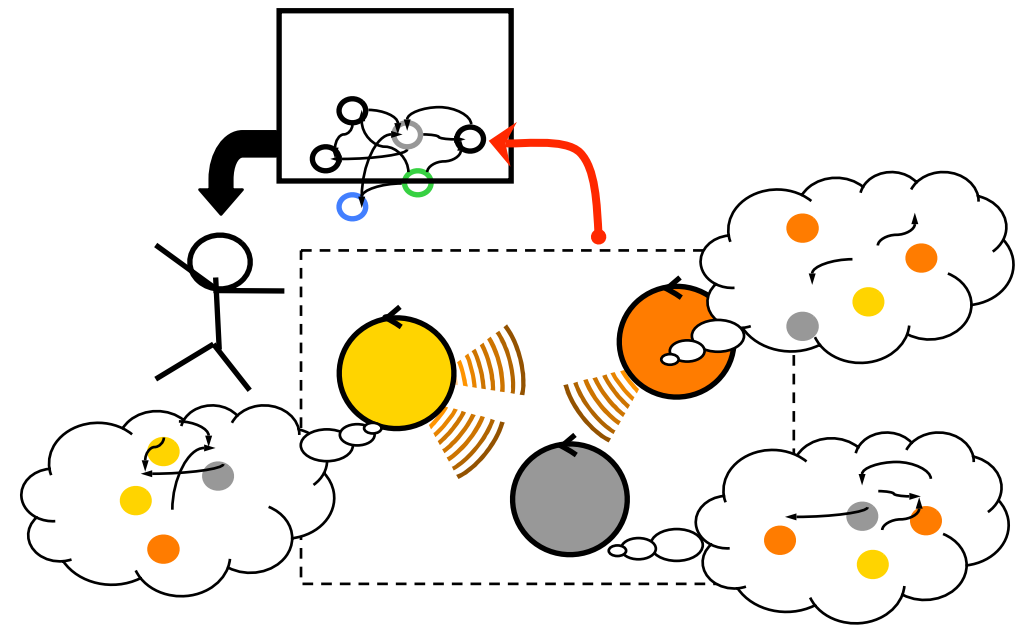
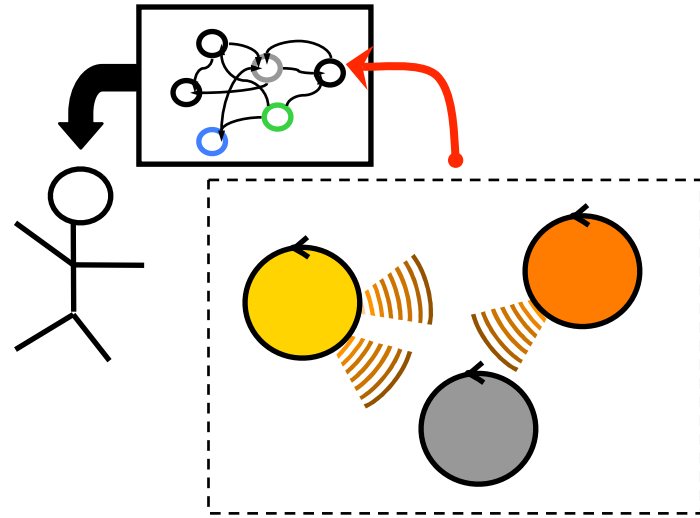


Organisations in MAS (global picture)



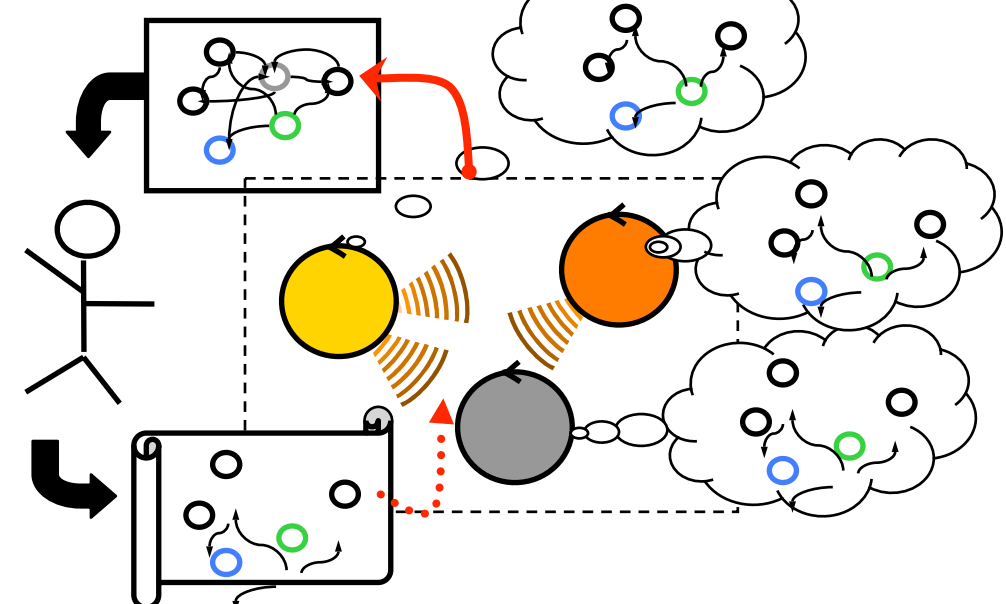
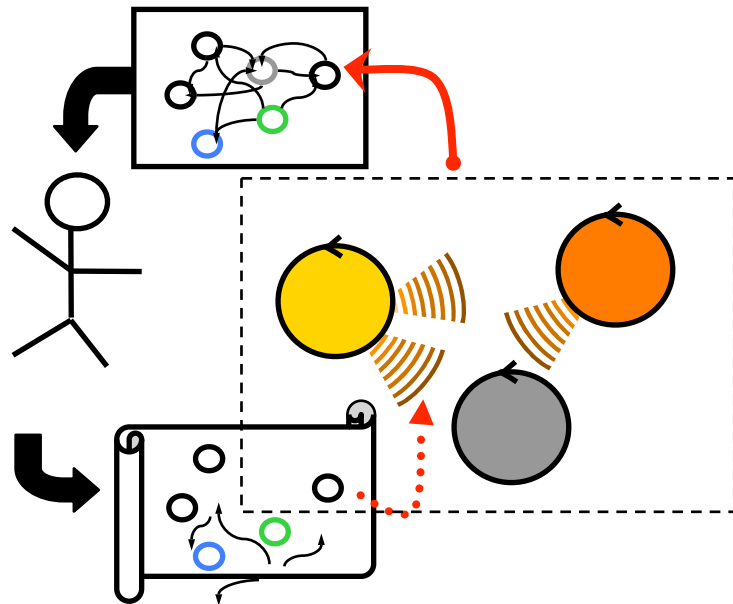
Organisations in MAS (global picture)

Agent Centred

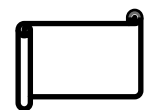


Agents don't know
about organisation

Agents know
about
organisation



Organisation Centred



Organisation Specification



Local Representation



Organisation Entity



Observed Organisation



Designer / Observer

Bottom-up

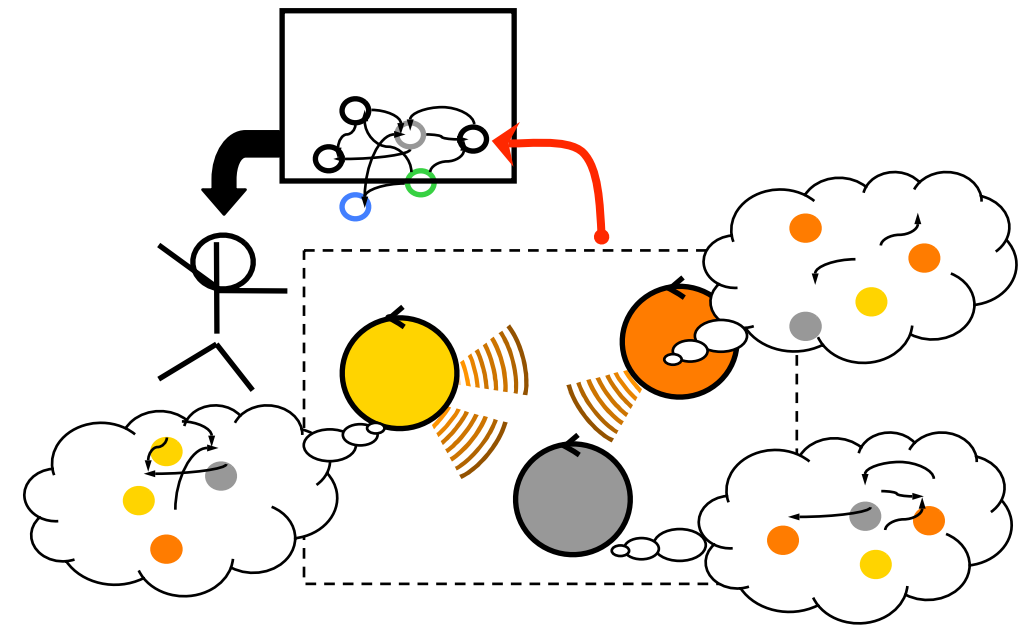
Top-down

Organisations in MAS (global picture)

Agent Centred

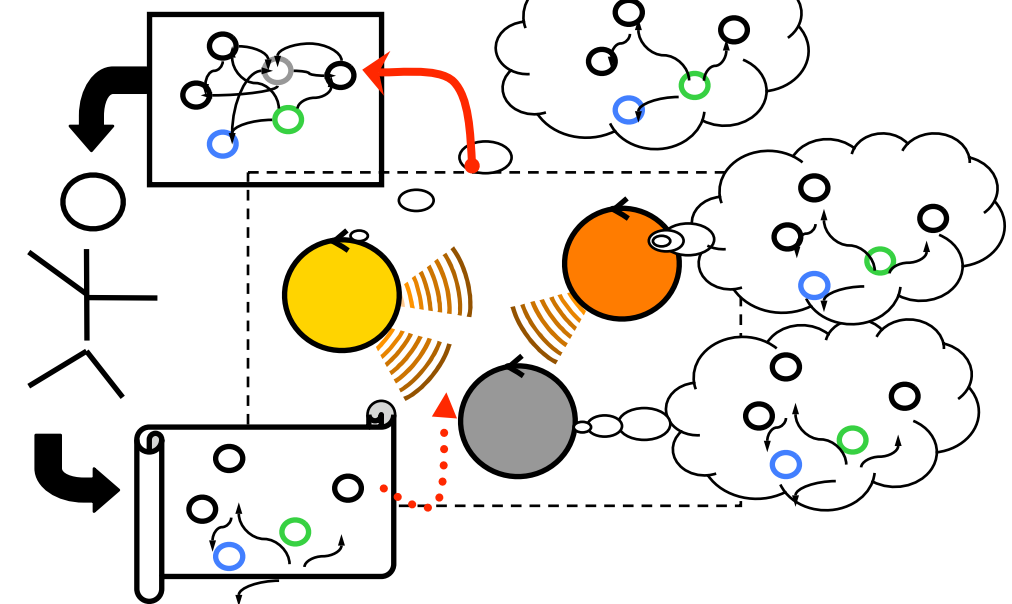
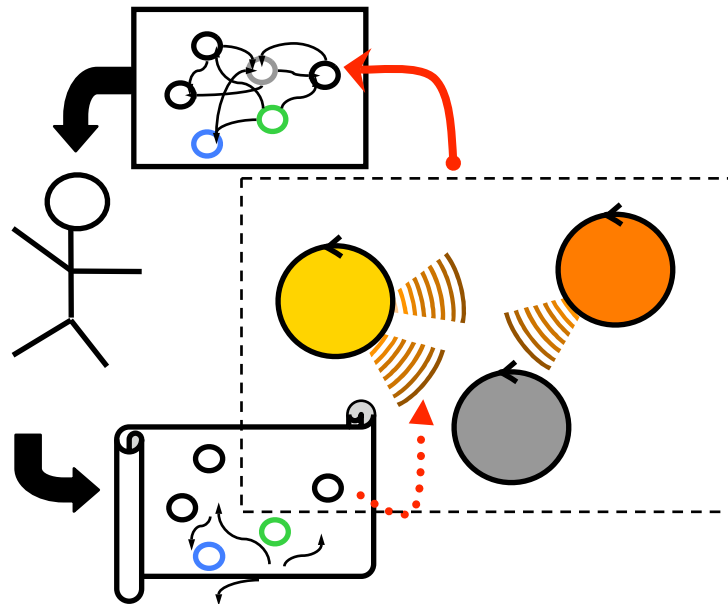
Swarms, AMAS, SASO
Self-organisations ...

Organisation is observed.
Implicitly programmed
in Agents, Interactions,
Environment.

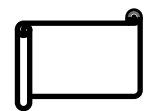


Agents don't know
about organisation

Agents know
about
organisation



Organisation Centred



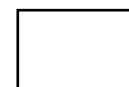
Organisation Specification



Local Representation



Organisation Entity



Observed Organisation



Designer / Observer

Bottom-up

Top-down

Organisations in MAS (global picture)

Agent Centred

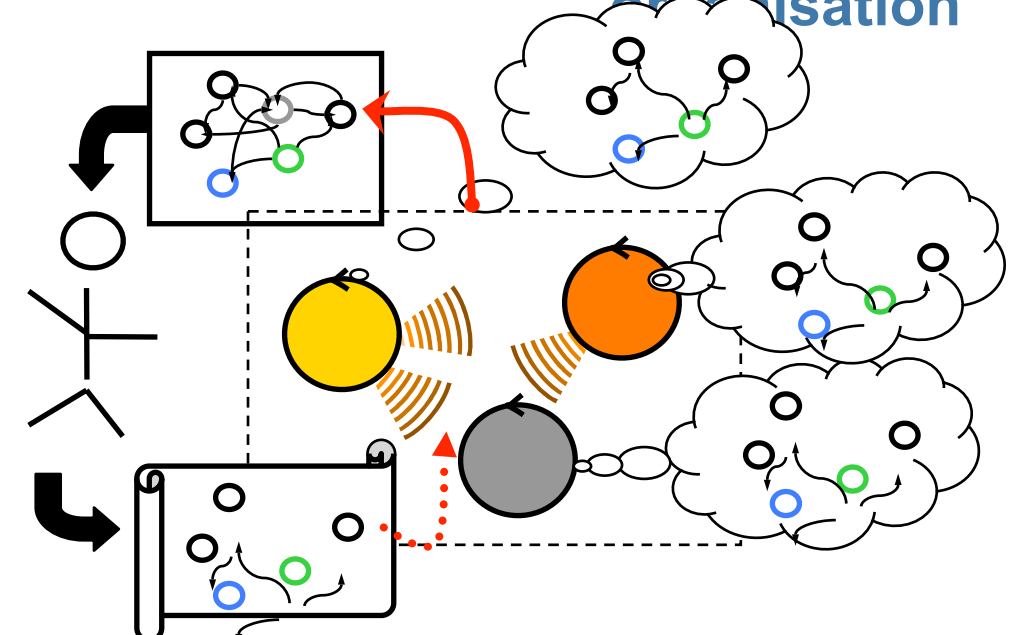
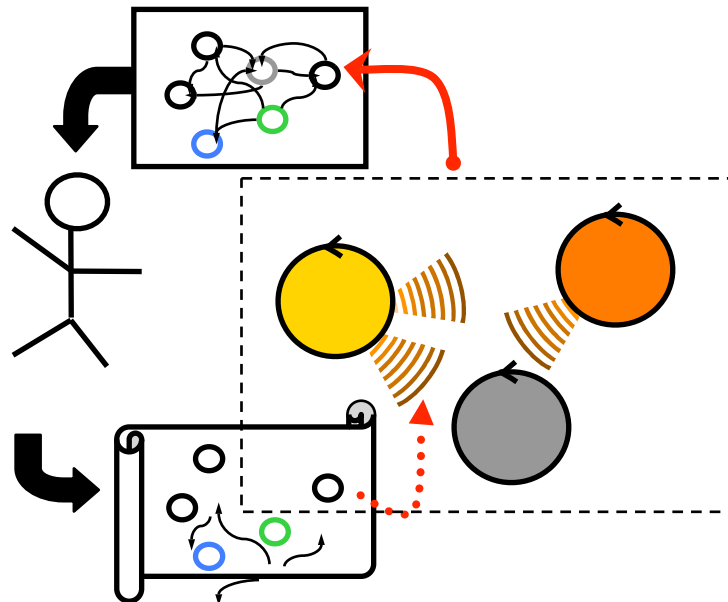
Swarms, AMAS, SASO
Self-organisations ...

Organisation is observed.
Implicitly programmed
in Agents, Interactions,
Environment.

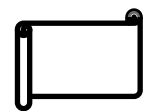
Social Reasoning
Coalition formation
Contract Net Protocol ...
Organisation is observed.
Coalition formation
mechanisms programmed
in Agents.

Agents don't know
about organisation

Agents know
about
organisation



Organisation Centred



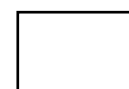
Organisation Specification



Local Representation



Organisation Entity



Observed Organisation

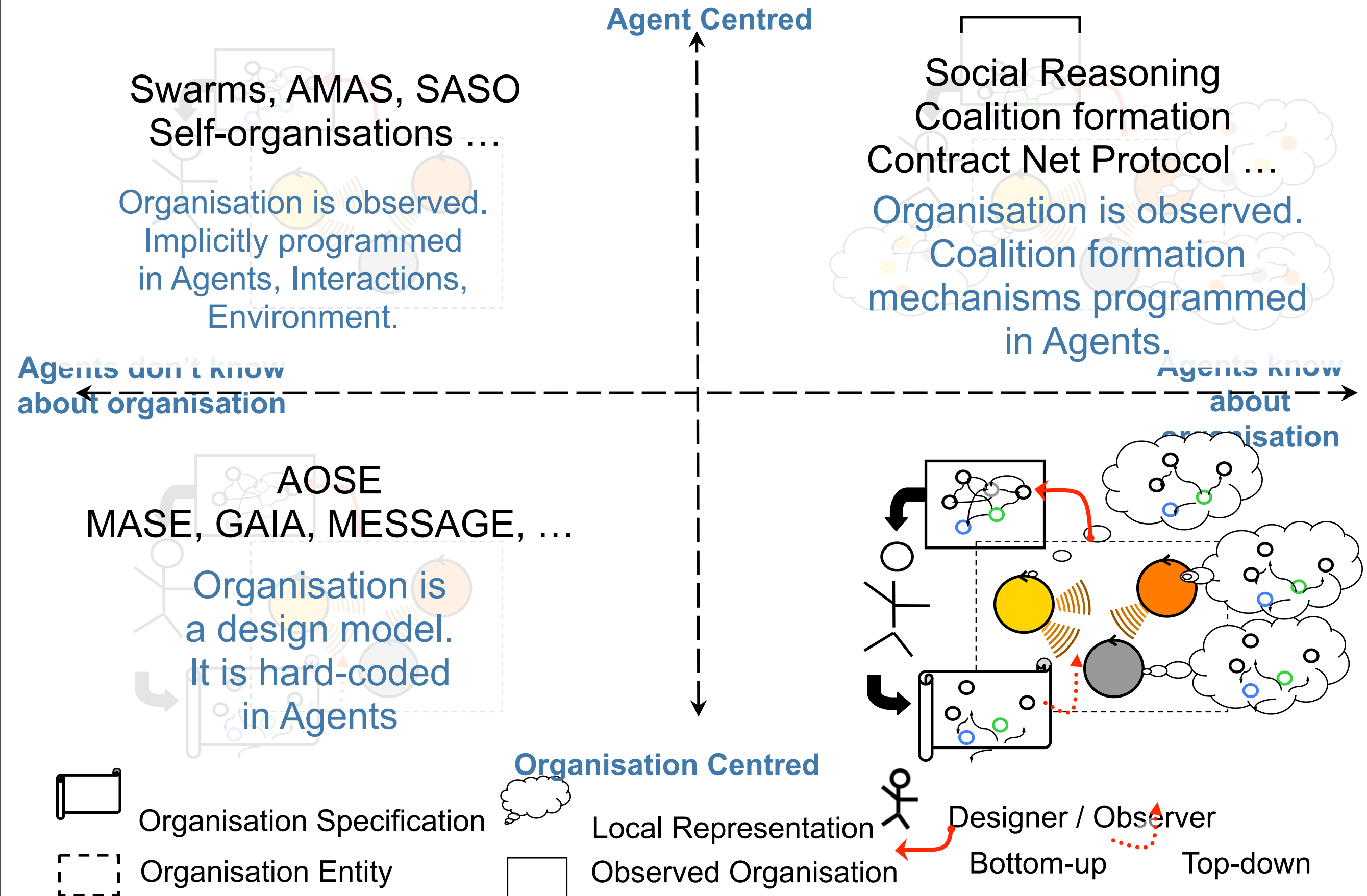


Designer / Observer

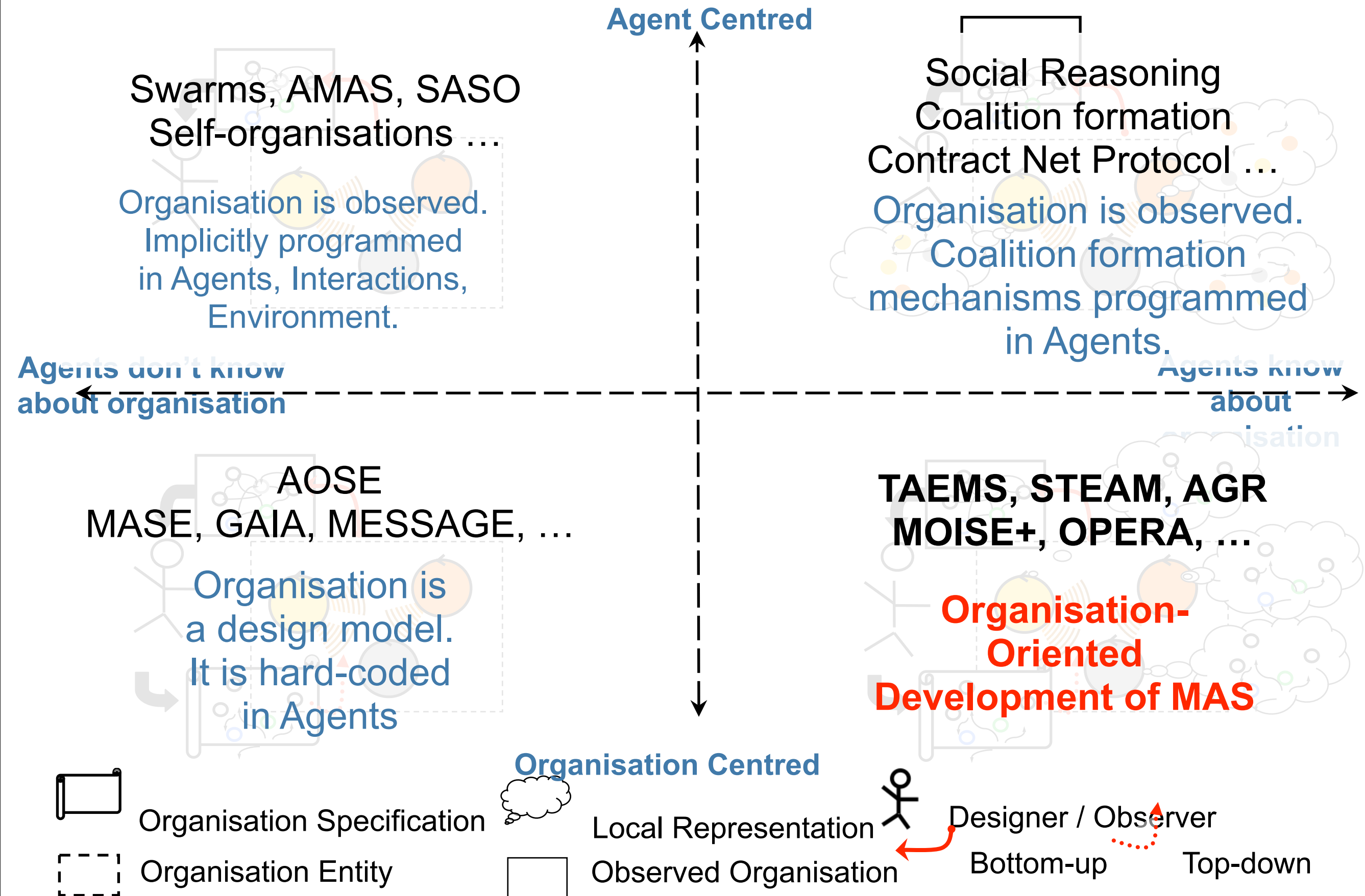
Bottom-up

Top-down

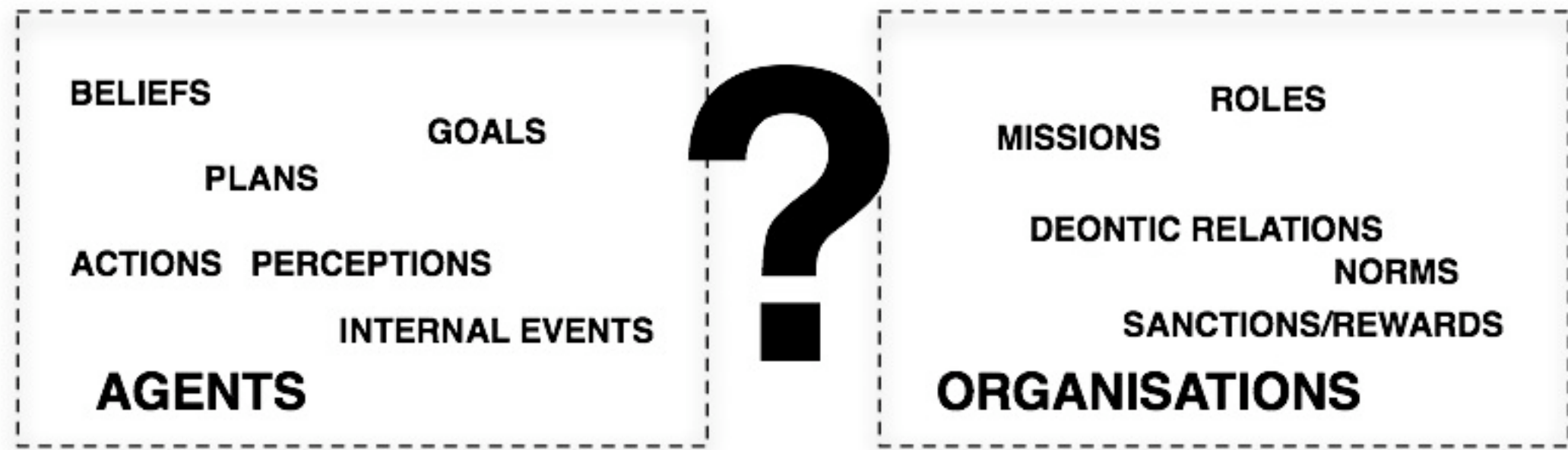
Organisations in MAS (global picture)



Organisations in MAS (global picture)



BRIDGE THE GAP



- Agents and Organisations are shaped on different concepts and notions
 - Agents are designed/programmed in terms of goals, beliefs, plans
 - Orgs are designed/programmed in terms of roles, norms, missions...

PART 3

Embodied Organisational Artifacts (EOA)

HUMAN ENVIRONMENTS



Human environments are instrumented with embodied resources (and staff agents) aimed at assisting activities.

HUMAN ENVIRONMENTS



A visitor entering the hospital has no way to direct communicate with "the hospital", nor has a complete knowledge of the organisation behind

HUMAN ENVIRONMENTS



Clients (patients) **ignores** the mechanisms and the structures at the basis of the organisation

HUMAN ENVIRONMENTS



Nonetheless, visitors find those things which they are interested to, and **find means to achieve their goals**

UNAWARE INTERACTIONS ??

SITUATING ORGANISATIONS

- Situate the Organisation in order to ease agent's computational load
 - No need *to know* organisational structures
 - situated org automatically update its states since agents
- Maintain Organisational Specification and artifact based infrastructure exploitable by agents that are aware of the organisational functioning
- Deploy environmental artifacts with the aim to situate the organisation in agent's world
 - agents that are not aware can indirectly interact with the organisation
 - no more need to bring about organisational concepts

INFRASTRUCTURES FOR CONTROLLING AGENTS ACTIVITIES

In an instrumented environment, conformity of agents behavior can be obtained (at least) in 2 ways:

1. REGIMENTATION as *practical impossibility* : agents can't perform undesirable activities
 - agents can't violate (i.e. barriers)
2. ENFORCEMENT through utility values : incentives (sanctions) to promote (prevent) (un)desired behavior
 - agents are free to obey or violate (i.e. traffic lights)

Embodied Organisational Artifacts (EOA) implement both

EMBODIED ORGANISATIONAL ARTIFACTS

EMBODIED ORGANISATIONAL ARTIFACTS

Special Environmental Artifacts which are directly deployed and controlled by the organization:

EMBODIED ORGANISATIONAL ARTIFACTS

Special Environmental Artifacts which are directly deployed and controlled by the organization:

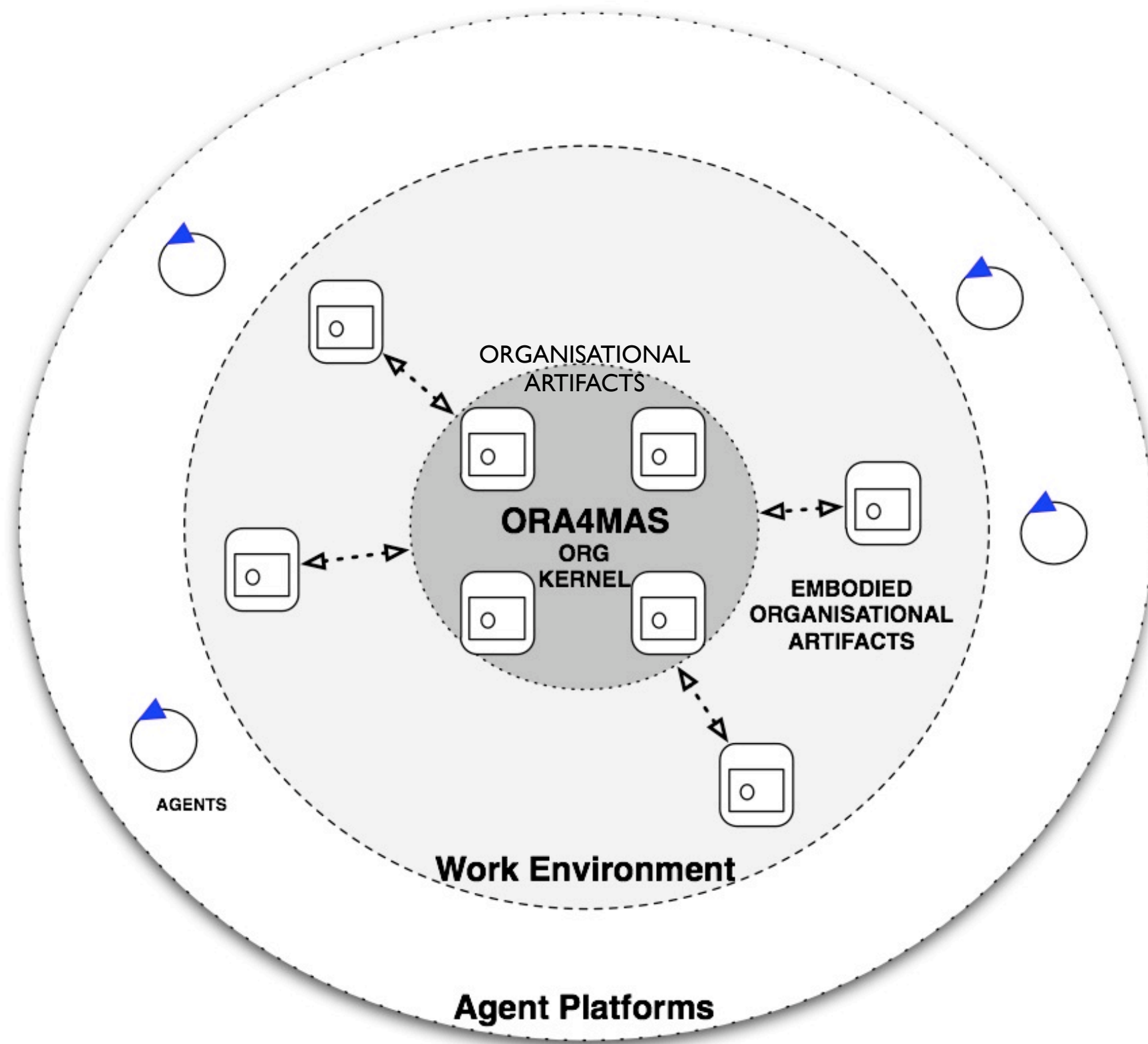
- In order to **rule** and **enable** agent activities
 - **rule** means coordinate and prevent deviation from undesirable states
 - **enable** means allow and provide means to achieve goals

EMBODIED ORGANISATIONAL ARTIFACTS

Special Environmental Artifacts which are directly deployed and controlled by the organization:

- In order to **rule** and **enable** agent activities
 - **rule** means coordinate and prevent deviation from undesirable states
 - **enable** means allow and provide means to achieve goals
- Examples: alarms, automatic doors, ticket-machines

Mediating Role



CHALLENGES

- Establish a functional (programmable) relation between environmental facilities and organisation
 - bidirectional
- Agents intentional actions to give rise to functional, unaware, collective phenomena
 - controlled by an organisation
- Reconciling emergence with cognition, intentional behavior with social function, planning agents with unaware cooperation/coordination

(*) Cristiano Castelfranchi, Engineering Social Order. ESAW 2000

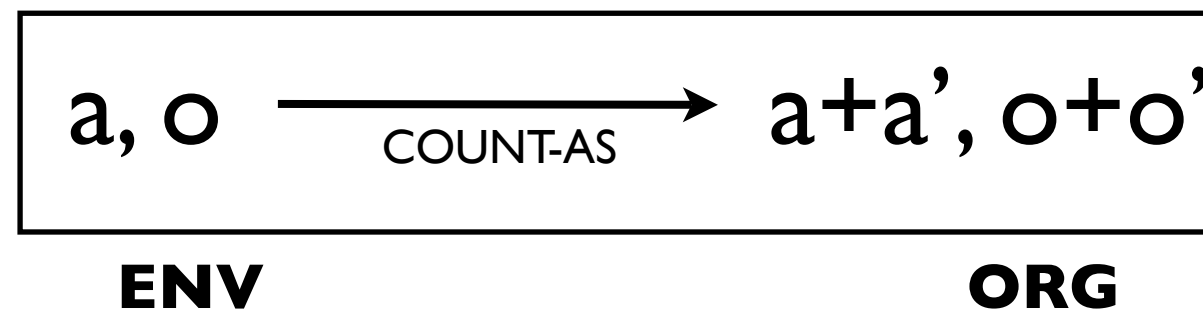
COUNT-AS

Count-as is a powerful "abstraction tool" to manage the relation between EOA and OA:

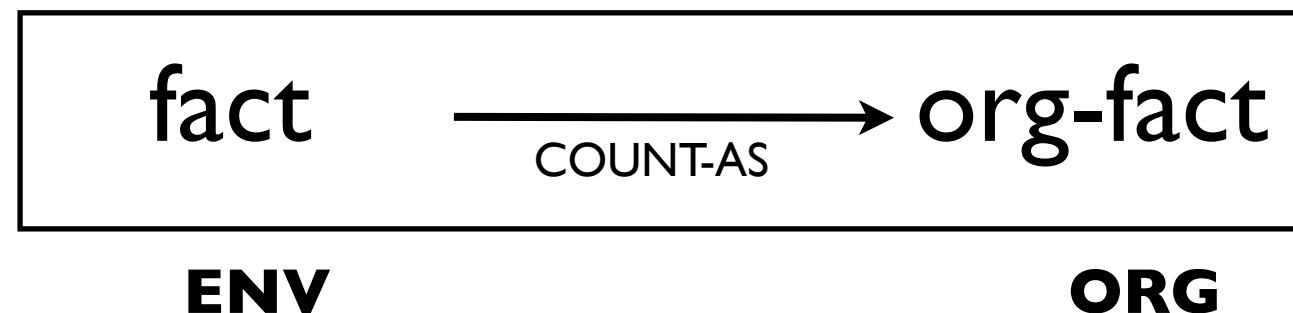
- “count-as” is a special relation between
 - the effects of an action performed by an active entity in the environment (say an agent) and
 - those special effects of that action which can be addressed on the institutional dimension of the system.
- Allow the organisation to automatically recognize world facts (events, states) and thus to update

THE *COUNT-AS* EFFECT

- A normal action of an agent *count-as* a conventional or an institutional action once it is contextualised in a particular institutional situation (Searle 69, Tuomela 99)

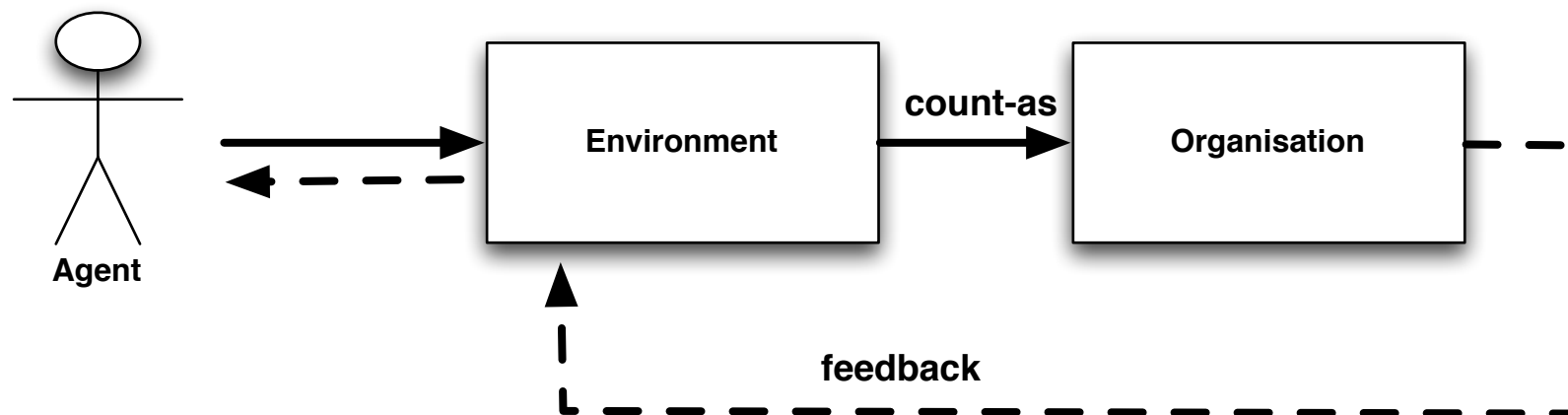
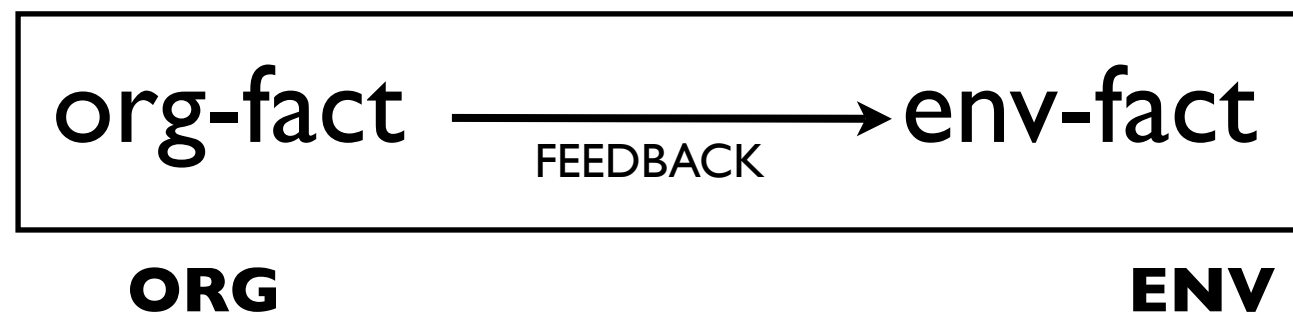


- The relation is from the concrete environment (brute facts, env-facts) to the organisation (i-facts, org-facts)

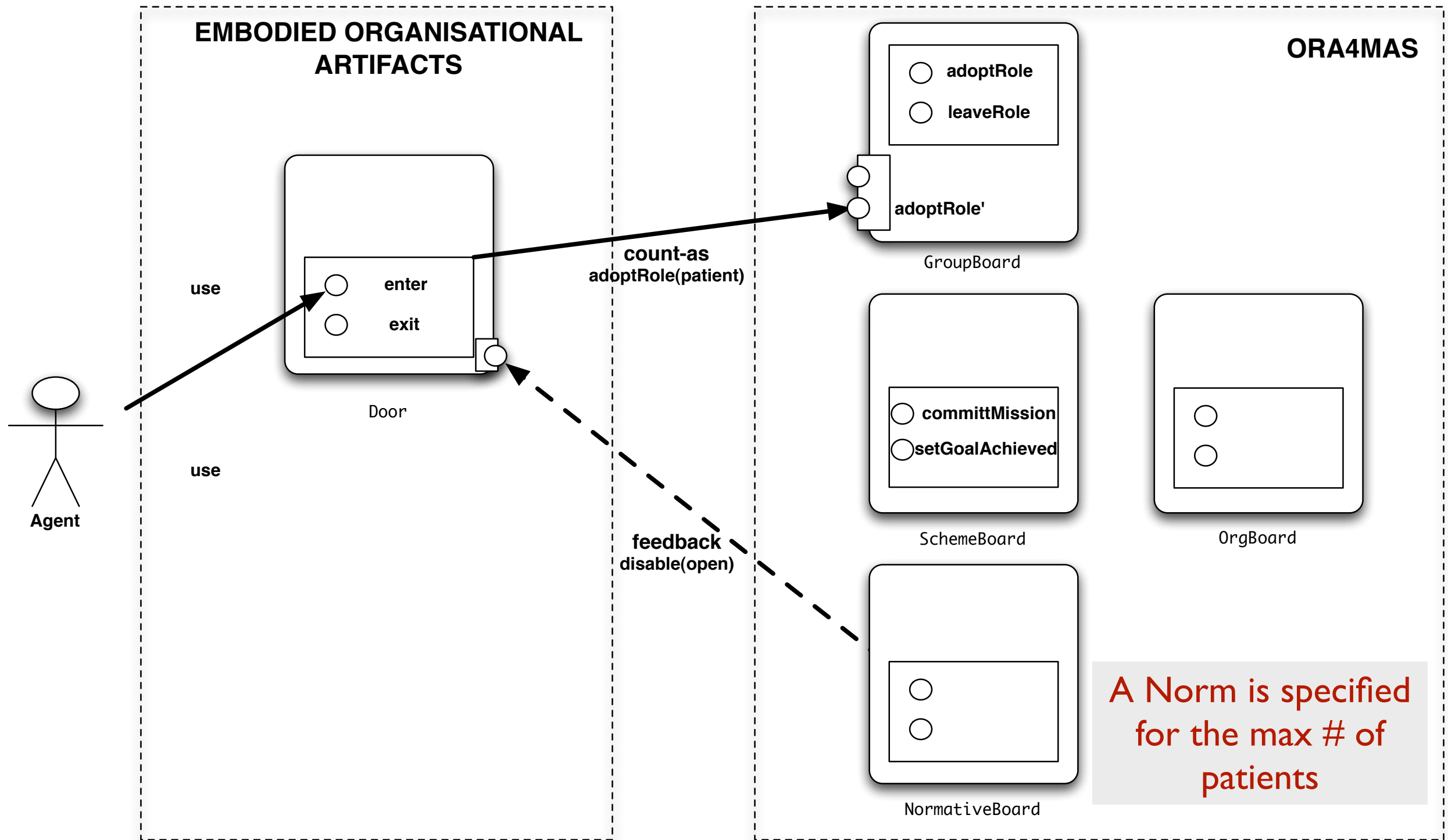


FEEDBACK

A dual relation is envisaged to manage relations from organisation to environments



LINK RELATIONS



PART 4

A Language for Programming Embodied Orgs

ORG RULES

A language to define relations between EOA and OA is provided

- Used to program ORG dynamics at an high level of abstraction:
 - Syntax (EBNF)
 - Dynamics (example)

Programming ORG MAS (I)

$ORG - MAS$	$::= \langle \overline{Agent}, \overline{Workspace}, \overline{ORG - RULE}, OS \rangle$	A&A Organisational System
$Agent$	$::= \langle agentName, agentProg \rangle$	
$Workspace$	$::= \langle \sigma, \overline{Artifact}, \overline{AgentBody} \rangle$	
$Artifact$	$::= \langle OA \rangle \langle EOA \rangle$	
EOA	$::= \langle artifactName, artifactProg \rangle$	Embodied Organisational Artifact
OA	$::= \langle artifactName, artifactProg, OS' \rangle$	Organisational Artifact
$agentName$	$::= \langle \gamma \rangle$	Agent id
$agentProg$	$::= \langle step, mState, perState, nextf, actf, \overline{\sigma} \rangle$	agent program
$step$	$::= Perc UpdSt ChAct ExecAct$	cycle step
$perState$	$::= \langle \overline{a - fact} \rangle$	Artifact related Percept state
$action$	$::= joinWsp(\sigma) quitWsp(\sigma) makeWsp(\sigma)$ $ use(\alpha, oname, opars, sid_{\perp}, timeout) sense(sid, filter, timeout)$ $ focus(\alpha, sid_{\perp} perp) stopFocus(\alpha) observeProp(\alpha, pname)$	workspace actions use/sense actions observation actions
$artifactName$	$::= \langle \alpha, \tau \rangle.$	Artifact
$artifactProg$	$::= \langle uic, \overline{a - obs - prop}, instate \rangle$	artifact program
uic	$::= \langle [guard] op - name \rangle$	usage interface control
$instate$	$::= \langle wstate, \overline{var}, \overline{o} \rangle$	execution model
o	$::= \langle \omega, op - name, op - state, op - cxt \rangle$	pending operation

Programming ORG MAS (II)

$a - fact$	$::= \langle a - obs - prop \rangle \langle a - ev \rangle$	Artifact Relevant Fact (Percept State)
$a - obs - prop$	$::= \langle artifactName, pname \mapsto pvalue \rangle$	Artifact Observable property
$a - event$	$::= \langle a - prop - ev \rangle \langle a - op - ev \rangle$	Artifact Event
$a - prop - ev$	$::= \langle artifactName, evtValue : evtType \rangle$	Artifact Property(-related) event
$a - op - ev$	$::= \langle artifactName, opName, resValue : resType \rangle$	Artifact Operation(-related) event
$ORG - RULE$	$::= \langle \overline{effect} \rangle \langle \overline{count - as} \rangle \langle \overline{feedback} \rangle$	Organisational Rules
$effect$	$::= \{ \langle \overline{fact} \rangle \} \langle opName \rangle \{ \langle \overline{fact} \rangle \}$	Operation Outcomes
$count - as$	$::= \langle \overline{fact} \rangle \longrightarrow \langle \overline{org - fact} \rangle$	Count as Rules
$feedback$	$::= \langle \overline{org - fact} \rangle \longrightarrow \langle \overline{env - fact} \rangle$	Regimentation and Sanctioning Rules
$fact$	$::= \langle \overline{env - fact} \rangle \langle \overline{org - fact} \rangle$	Relevant Fact
$env - fact$	$::= \langle a - fact \rangle$	Environment Relevant Fact
$org - fact$	$::= \langle a - fact \rangle$	Organisation Relevant Fact

A SIMPLE (not yet complete) EXAMPLE

According to the organisation, a visitor entering in a hospital reservation desk has to:

1. Announce himself
2. Pay the visit at the automatic machine
3. go_out

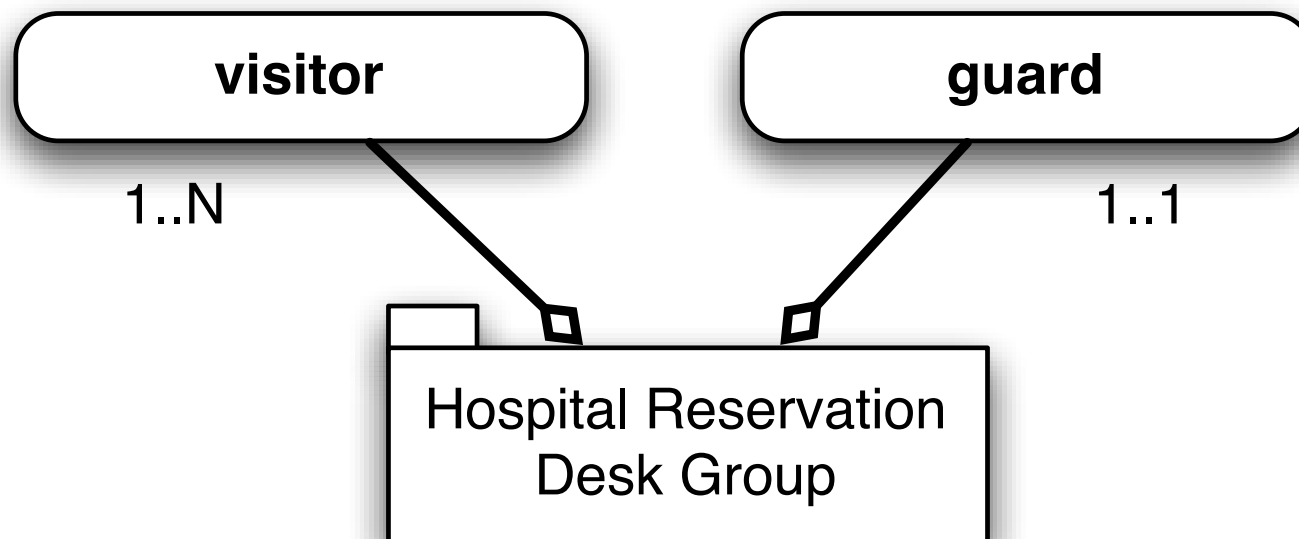
The room has a maximum capacity of visitor at the same time (# of attending patients). A staff agent (guard) controls wether patients go out without paying:

1. To call police (and provide fines)

Phase 1:

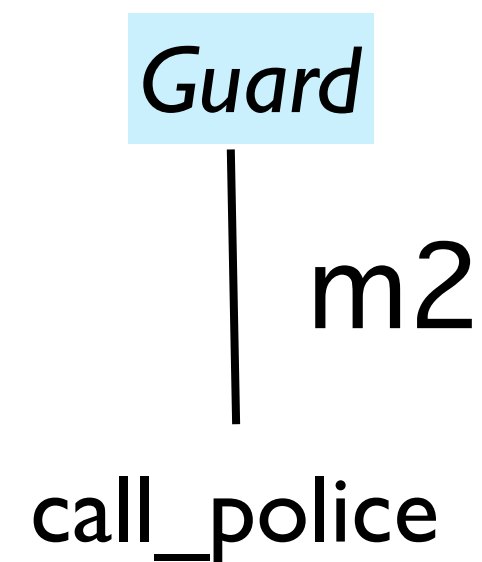
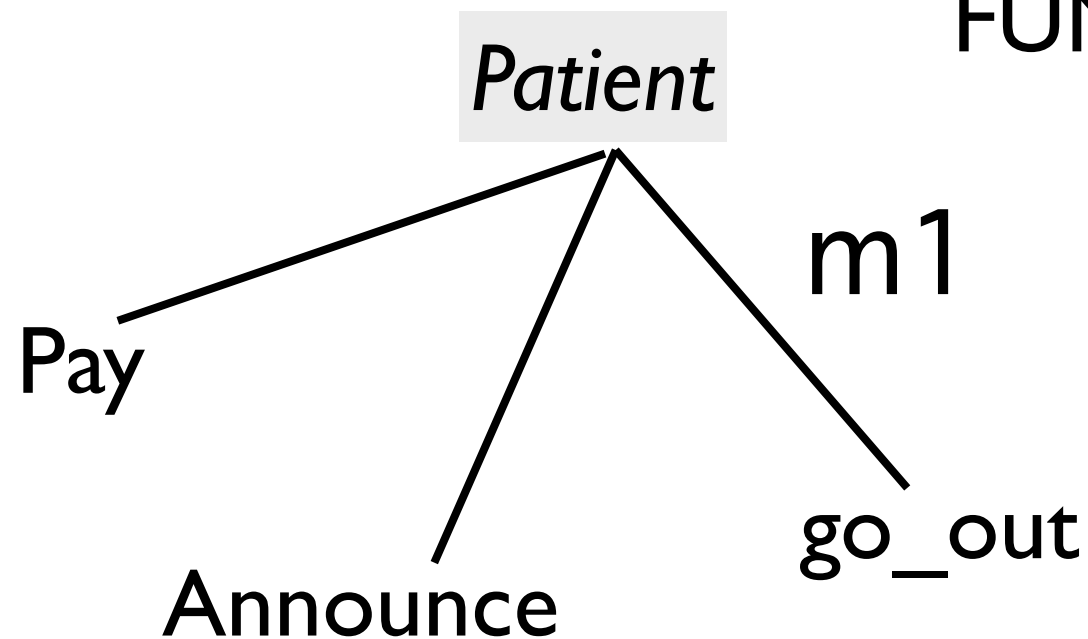
Define Organisation Specification

ORGANISATIONAL SPECIFICATION

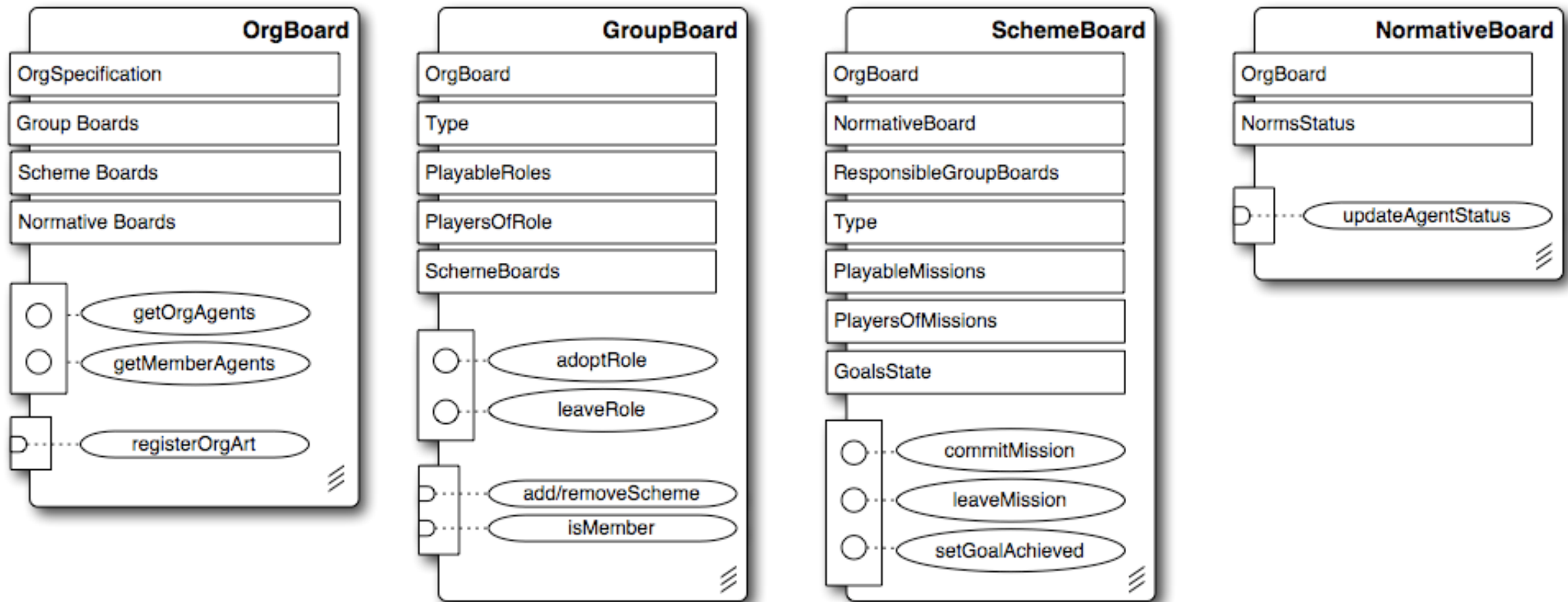


STRUCTURAL SPECIFICATION

FUNCTIONAL SPECIFICATION



OA (ORA4MAS)



Phase 2:

Deploy Embodied Organisational Artifacts (EOA)
to instrument the Environment

EOA (I)

EOA_1

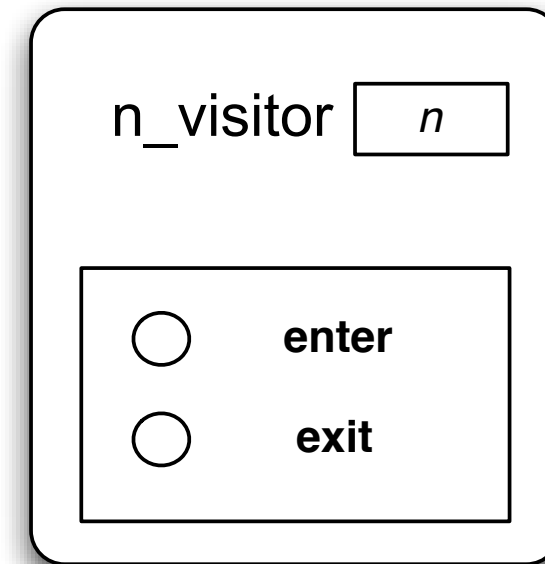
ArtifactName: **Room**, emse.g2i.artifact.room

uic (operations): enter, exit

a-obs-prop: n_visitor

a-op-event: enter, (op_triggered);
enter, (op_complete);
exit, (op_triggered);
exit, (op_complete);

a-prop-event: prop-changed, n_visitor



Room

EOA_2

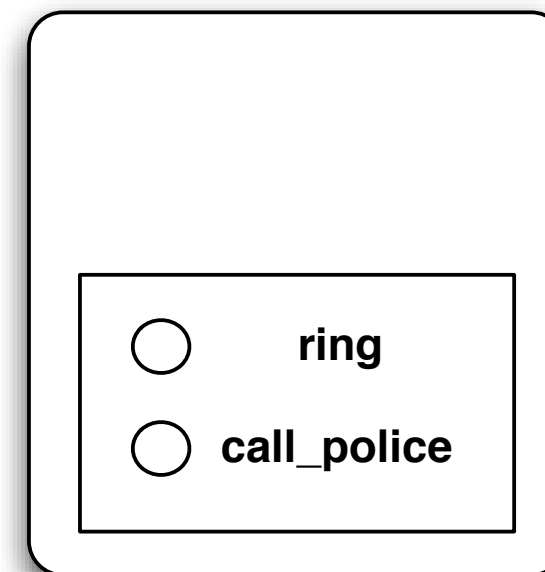
ArtifactName: **Alarm**, emse.g2i.artifact.alarm

uic (operations): ring, call_police

a-obs-prop: -

a-op-event: ring, (op_triggered);
ring, (signal,440Hz)
ring, (op_complete);
call_police, (op_triggered);
call_police, (signal,120Hz)
call_police, (op_complete);

a-prop-event: -



Alarm

EOA (II)

EOA_3

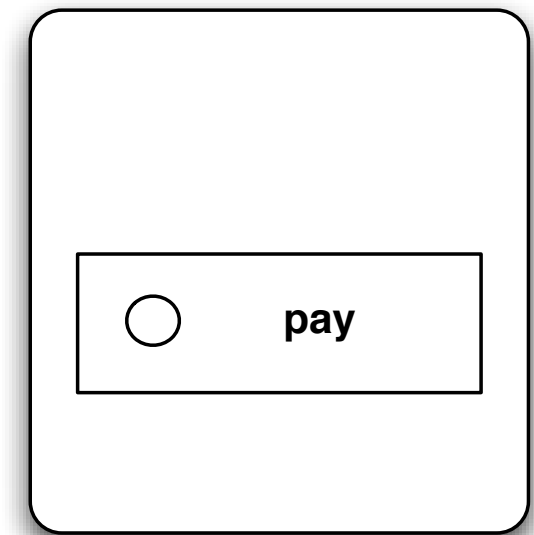
ArtifactName: **Cash_Desk**, emse.g2i.artifact.cashDesk

uic (operations): pay

a-obs-prop: -

a-op-event: pay, (op_triggered);
pay, (signal, receipt)
pay, (op_complete);

a-prop-event: -



Cash Desk

Phase 3:

Define COUNT-AS relations to notify
Organisational Artifacts (OA) about agent's
activities

COUNT-AS RULES

Room: a-op-ev (enter(\$a), op_triggered) $\longrightarrow$ GroupBoard: adoptRole(\$a, "Patient");

GroupBoard: adoptRole(\$a, "Patient"); $\longrightarrow$ SchemeBoard: committMission(\$a, m1)

Cash_Desk: a-op-ev (signal, receipt) $\longrightarrow$ SchemeBoard: setGoalAchieved(\$a, pay);

Alarm: a-op-ev (signal, 120Hz) $\longrightarrow$ SchemeBoard: setGoalAchieved(\$a, call_police)

Room: a-op-ev (exit(\$a), op_complete) $\longrightarrow$ GroupBoard: leaveRole(\$a, "Patient");

Room: a-op-ev (exit(\$a), op_complete) $\longrightarrow$ SchemeBoard: setGoalAchieved(\$a, go_out);

Phase 4:

Define FEEDBACK relations to control
Environment (EOAs)

FEEDBACK RULES

Regimentation is done by enabling/disabling uics

SchemeBoard:commitMission(\$a, m1)	→	Room: disable(\$a, exit)
SchemeBoard:setGoalAchieved(\$a, pay)	→	Room: enable(\$a, exit)

Enforcement is done by committing staff agents

NormativeBoard: a-prop-event prop-changed,N1	→	[Guard is scrutinizing NormativeBoard]
		SchemeBoard:commitMission(\$g, m2)

CONCLUSIONS

- All the model is based on events managed at the system level
 - by CARTAGO work environments and agents platforms
 - while related approaches (Dastani et al.) are based on states
- Give back agents their "autonomy": agents are not (only) coordinated as reactive entities by the signals coming from schemoboard;
 - they proactively use EOA - which are controlled by org
 - no need to know and reason on organisational notions
 - agent behavior is veichulated at the organisational level by EOAs

J. Campos, M. Lòopez-Sànchez, J. A. Rodrìguez-Aguilar, and M. Esteva. Formalising situatedness and adaptation in electronic institutions. In J. F. Hübner and O. Boissier, editors, Workshop on Coordination, Organizations, Institutions and Norms in Agent Systems (COIN-08), pages 103–117, 2008.

C. Castelfranchi. Engineering Social Order. In Engineering Societies in Agents World, volume 1972 of Lecture Notes Computer Science (LNCS), pages 1–18. Springer, 2000.

M. Dastani, D. Grossi, J.-J. Meyer, and N. Tinnemeier. Normative Multi-Agent Programs and Their Logics. In Workshop on Knowledge Representation for Agents and Multi-Agent Systems (KRAMAS-08), 2008.

C. Hewitt. Perfect Disruption: The Paradigm Shift from Mental Agents to ORGs. IEEE Internet Computing, 13:90–93, 2009.

J. F. Hübner, J. S. Sichman, and O. Boissier. MOISE+: Towards a Structural, Functional, and Deontic Model for MAS Organization. In C. Castelfranchi and W. L. Johnson, editors, Proc. of International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02), pages 501–502. ACM Press, 2002.

Kitio, O. Boissier, J. F. Hübner, and A. Ricci. Organisational artifacts and agents for open multi-agent organisations: “giving the power back to the agents”. In Coordination, Organizations, Institutions, and Norms in Agent Systems III, volume 4870 of LNCS, pages 171–186. Springer, 2008.

F. Y. Okuyama, R. H. Bordini, and A. C. da Rocha Costa. A distributed normative infrastructure for situated multi-agent organisations. In AAMAS (3), pages 1501–1504, 2008.

A. Ricci, M. Piunti, L. D. Acay, R. Bordini, J. Hubner, and M. Dastani. Integrating Artifact-Based Environments with Heterogeneous Agent-Programming Platforms. In AAMAS’08, Proceedings, 2008.

A. Ricci, M. Piunti, M. Viroli, and A. Omicini. Environment programming in CAR tAgO. In R. H. Bordini, M. Dastani, J. Dix, and A. El Fallah-Seghrouchni, editors, Multi-Agent Programming: Languages, Platforms and Applications, Vol. 2. Springer, 2009.

A. Ricci and M. Viroli. A Formal Model for Artifact-Based Environments in MAS Programming. In Proc of 7th Workshop on Programming Multi-Agent Systems (PROMAS-09), 2009.

L. Tummolini and C. Castelfranchi. The cognitive and behavioral mediation of institutions: Towards an account of institutional actions. Cognitive Systems Research, pages 307–323, 2006.

THANK YOU !!