

## 4. Coordination and Social Models

### Part 4: Coordination models (I): Social Models

### Electronic Institutions

Javier Vázquez-Salceda

MASD



Knowledge Engineering and Machine Learning Group  
UNIVERSITAT POLITÈCNICA DE CATALUNYA  
<https://kemig.upc.edu>

### Introduction to Electronic Institutions

---

- Institutions as Social Structures
- e-Institutions
- Approaches in Literature



Knowledge Engineering and Machine Learning Group  
UNIVERSITAT POLITÈCNICA DE CATALUNYA  
<https://kemig.upc.edu>

## Introduction

- Open multi-agent systems (MAS) have to cope with several issues
  - Heterogeneity among members
  - Communication
  - Participants' trust

} • *Coordination*  
• *Cooperation*

- Agent **Autonomy** VS **Control**

- Idea: multi-agent design can benefit from social abstractions
  - Study the problem from the societal and the individual points of view.

3

## Introduction

### Institutions as Social Structures

- **Social Structures** define a social level to enhance coordination by means of *interaction patterns*
- **Institutions** are a kind of social structure where a corpora of constraints shape the behaviour of the members of a group
- The definition of a (human) Institution uses to include
  - **Regulations** about the interactions
  - **Conventions**: 'institutional facts' vs 'brute facts'
    - e.g. 1 : '*murder*' (vs killing)
    - e.g. 2 : '*incest*' (vs sexual act)
    - e.g. 3 : '*ownership*' (vs physical possession)
    - e.g. 4 : '*marriage*' (vs living together)
  - **Procedures** and **protocols** for creating and determining institutional facts

4

## Introduction

### e-Institutions

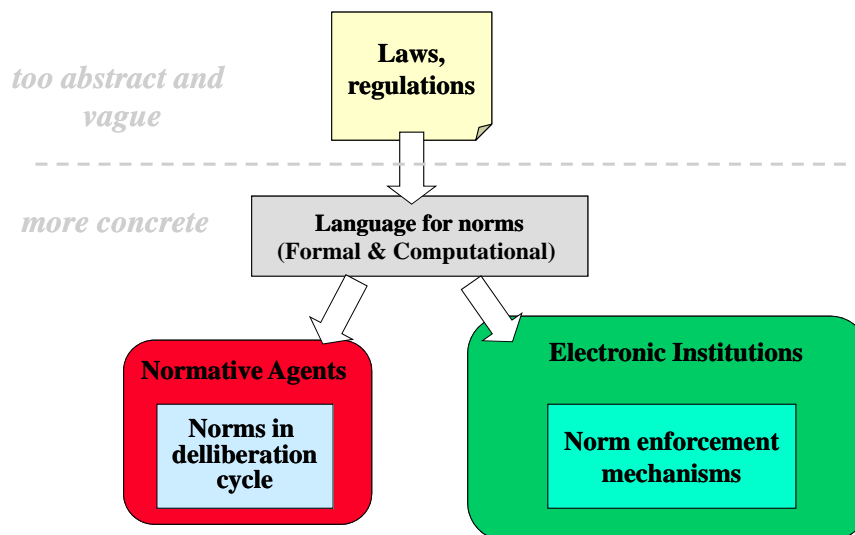
- An **e-Institution** is the computational model of an institution through
  - The specification of the institution's **norms** in (some) suitable formalism(s).
  - The specification of the institution's procedures and protocols
- In the context of MAS they:
  - reduce **uncertainty** of other agents' behaviour
  - reduce **misunderstanding** in interaction
  - allows agents to **foresee the outcome** of an interaction
  - simplify the **decision-making** (reduce the possible actions)

- Agent **behaviour guided by Norms**

5

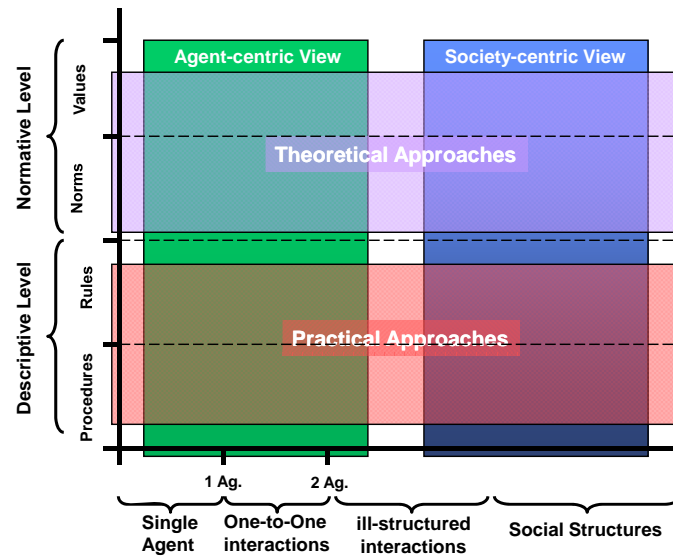
## Introduction

### Why a Language for Norms?



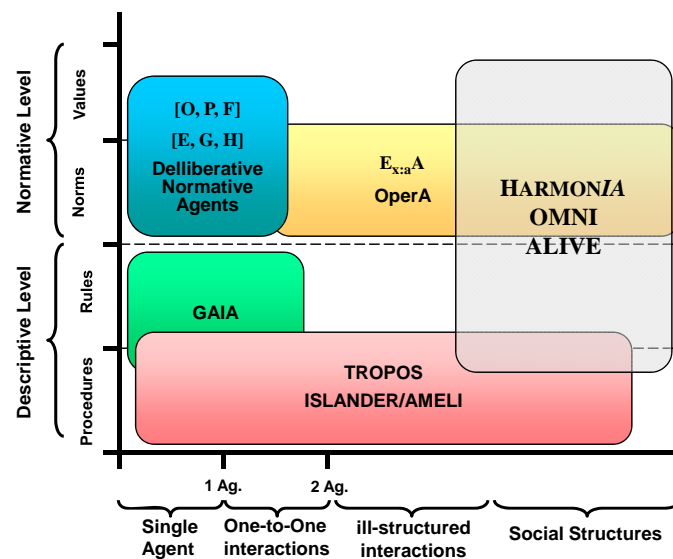
6

## State of the Art (I)



7

## State of the Art (II)



8

## Norms from the Agent perspective

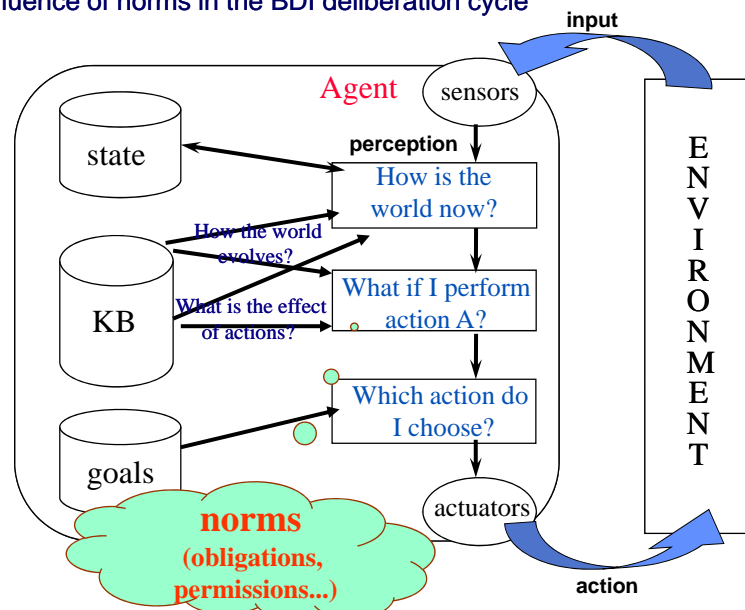
- Influence of norms in Agent behaviour
- Possible World semantics
- Norms in the agent interpreter



Knowledge Engineering and Machine Learning Group  
UNIVERSITAT POLITÈCNICA DE CATALUNYA  
<https://kemig.upc.edu>

## Norms and Agents (I)

Influence of norms in the BDI deliberation cycle



## Norms and Agents (II)

- How do norms influence the behaviour of the agent?
  - Agent has **no knowledge** about norms
  - Norms are **built-into** the agent's code
  - Norms are **built-into** the plans and protocols the agent uses
  - Norms are **explicit** elements in the agent's reasoning
    - Agent may or may not **adopt** the norms
    - Agent may or may not **follow** the norms
      - agent **follows** the norm **whenever possible**
      - agent **violates** the norm **sometimes**
      - agent **violates** the norm **always if possible**

11

## Norms and Agents (III)

- **Problems:**
  - Norms are more abstract than the procedures
  - Norms do not have operational semantics

### **Example:**

*Regulation: "It is forbidden to discriminate potential recipients of an organ based on their age (race, religion,...)"*

*Formal norm:  $F(\text{discriminate}(x,y,\text{age}))$*

*Procedure: does not contain action "discriminate"*

12

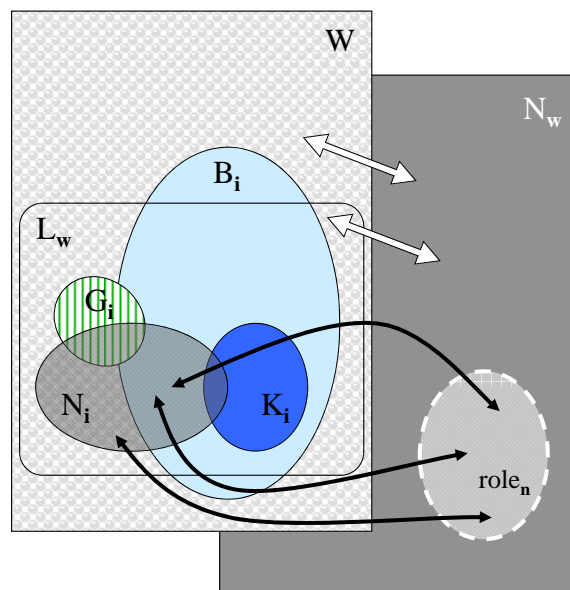
## Norms and Agents (IV)

- Most of the approaches talk about **norms**, but a close-up look shows that they are working at completely different **levels of abstraction**
- **Idea**: there are **several levels of abstraction** involved in a normative system

13

## Norms and Agents (V)

Possible Worlds Semantics



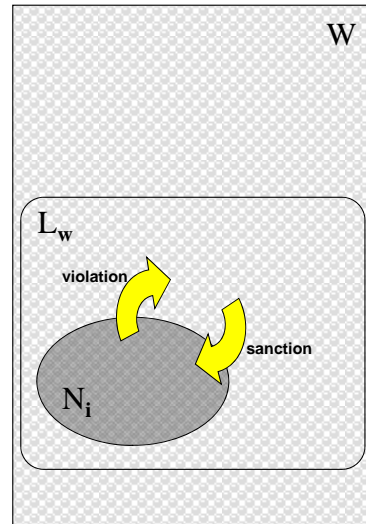
14

## Norms and Agents (V)

Legally accessible worlds

- The concept of legally accessible worlds allows to describe
  - wanted (legal) and unwanted (illegal) behaviour
  - acceptable (safe) and unacceptable (unsafe) states
- **Violations** when agents breaks one or more norms, entering in an illegal (unsafe) state.
- **Sanctions** are actions to make agents become legal (safe) again.
- Sanctions may include the actions to recover the system from a violation

Safety  
Soundness



15

## Norms and Agents (VI)

- **Problem:** in this model interpretation of norms is completely done by the agents.
  - How to ensure that two agents that play the same role have similar sound interpretations of the norms that apply to them?
- **Solution:** to fix part of the interpretation in a given context.

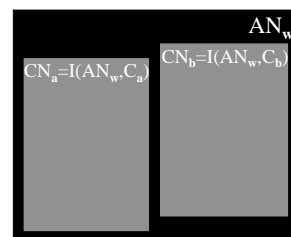
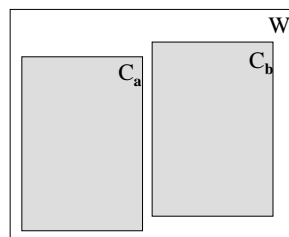
• Idea: Agents do not have a relation with the WHOLE world but a part → **context** of an agent.

16



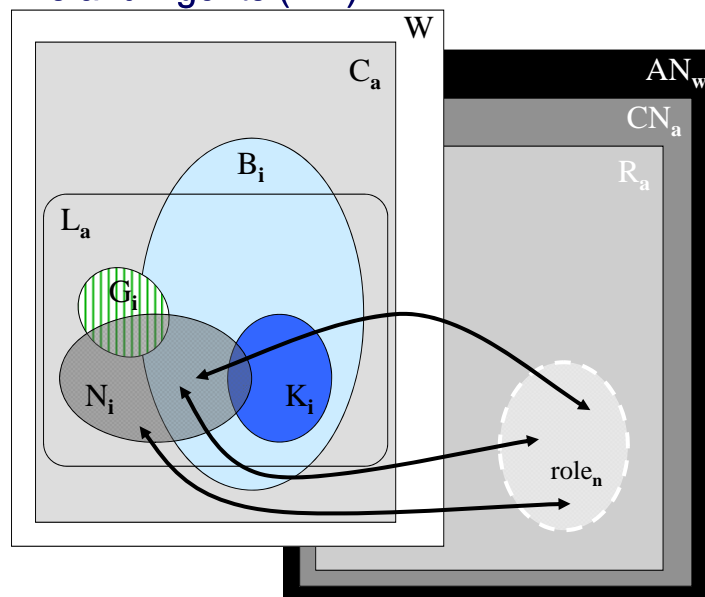
## Norms and Agents (VII)

- A **Context** is a set of worlds with a shared vocabulary and a normative framework
- Effects on the Normative Dimension
  - The generic norms applied to the world as a whole are called **Abstract Norms**.
  - **Concrete Norms** are interpretations of the abstract norms in a given context

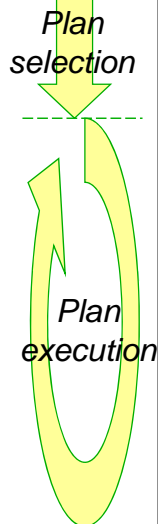


17

## Norms and Agents (VIII)



18



Agent Control Loop Version 7

```

1.
2.  $B := B_0$ ;
3.  $I := I_0$ ;
4. while true do
5.   get next percept  $\rho$ ;
6.    $B := brf(B, \rho)$ ;
7.    $D := options(B, I)$ ;
8.    $I := filter(B, D, I)$ ;
9.    $\pi := plan(B, I)$ ;
10.  while not (empty( $\pi$ )
        or succeeded( $I, B$ )
        or impossible( $I, B$ )) do
11.     $\alpha := hd(\pi)$ ;
12.    execute( $\alpha$ );
13.     $\pi := tail(\pi)$ ;
14.    get next percept  $\rho$ ;
15.     $B := brf(B, \rho)$ ;
16.    if reconsider( $I, B$ ) then
17.       $D := options(B, I)$ ;
18.       $I := filter(B, D, I)$ ;
19.    end-if
20.    if not sound( $\pi, I, B$ ) then
21.       $\pi := plan(B, I)$ 
22.    end-if
23.  end-while
24. end-while

```

## Norms and Agents (IX)

### Norm obligations

add actions to the set of options and may define some priorities or precedence

### Norm prohibitions

delete actions from the set of options

19

## Norms from the Institutional perspective

- SMART normative model
- AMELI/ISLANDER
- HARMONIA
- OMNI



Knowledge Engineering and Machine Learning Group  
UNIVERSITAT POLITÈCNICA DE CATALUNYA

<https://kemlg.upc.edu>

## SMART Normative Systems model

Fabiola Lopez y Lopez, Mike Luck and M. d'Inverno.

- It is based in the SMART agent specification framework presented by d'Inverno and Luck [8].
  - The framework defines concepts such as
    - *objects*,
    - *agents* (which are objects with goals)
    - *autonomous agents* (which are agents with motivations).
  - This framework is developed in the Z specification language [9].
- The SMART framework has been extended [2][7] to introduce, as part of the framework, representations of
  - *norms*
  - *normative agent*
  - *Normative MAS*
- The authors have also presented an architecture for autonomous social and normative agents in order to reason about norms.

21

## SMART Normative Systems model

Normative MAS

```

NormativeMAS
  AgentWorld
    nmasname : NMASName
    members : P AgentName
    generalnorms : P Norm
    enforcenorms : P Norm
    rewardnorms : P Norm
    legislationnorms : P Norm
    environment : EnvState

  members ⊆ idagents
  ∀ ag : members • (normativeAg ag).norms ∩ generalnorms ≠ ∅
  ∀ sn : generalnorms • sn.addressees ⊆ members
  ∀ en : enforcenorms • (∃ n : generalnorms • enforces (en, n))
  ∀ rn : rewardnorms • (∃ n : generalnorms • rewardnorm (rn, n))
  ∀ ln : legislationnorms • legislate (ln, environment)

```

22

## SMART Normative Systems model

### Norm definition

<i>Norm</i>
<i>normativegoals</i> : $\mathbb{P} \text{ Goal}$
<i>addressees</i> : $\mathbb{P} \text{ AgentName}$
<i>beneficiaries</i> : $\mathbb{P} \text{ AgentName}$
<i>context</i> : <i>EnvState</i>
<i>exceptions</i> : <i>EnvState</i>
<i>rewards</i> : $\mathbb{P} \text{ Goal}$
<i>punishments</i> : $\mathbb{P} \text{ Goal}$
<i>normativegoals</i> $\neq \emptyset$
<i>addressees</i> $\neq \emptyset$
<i>context</i> $\neq \emptyset$
<i>context</i> $\cap$ <i>exceptions</i> = $\emptyset$
<i>rewards</i> $\cap$ <i>punishments</i> = $\emptyset$

23

## SMART Normative Systems model

### Norm addressees, legislators, defenders and promoters

- Norms are related not only with the agents that should fulfil it or enforce it, but also with agents such as the one that issued the norm, the one(s) responsible of its enforcement (the *defenders*, the one that modified it or the one(s) that may be affected by a non-compliance of the norm.

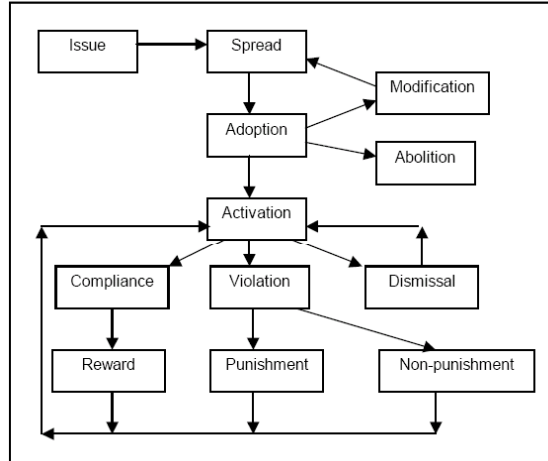
<i>AuthoritiesNMAS</i>
<i>NormativeMAS</i>
<i>legislators</i> : $\mathbb{P} \text{ AgentName}$
<i>defenders</i> : $\mathbb{P} \text{ AgentName}$
<i>promoters</i> : $\mathbb{P} \text{ AgentName}$
$\forall lg : \text{legislators} \bullet (\exists lnorm : \text{legislationnorms} \bullet lg \in lnorm.addressees)$
$\forall df : \text{defenders} \bullet (\exists enorm : \text{enforcenorms} \bullet df \in enorm.addressees)$
$\forall pm : \text{promoters} \bullet (\exists rnorm : \text{rewardnorms} \bullet pm \in rnorm.addressees)$

24

## SMART Normative Systems model

### Norm Lifecycle

- Norms are not modelled as static constraints but as objects that can have several states (such as *issued*, *active*, *modified*, *fulfilled* or *violated*),



25

## SMART Normative Systems model

### Norm Lifecycle

```

MASState
NormativeMAS
allinstances : P NormInstance
activenorms : P NormInstance
fulfillednorms : P NormInstance
unfulfillednorms : P NormInstance
rewardednorms : P NormInstance
punishednorms : P NormInstance

∀ in : allinstances • (∃ n : generalnorms • isnorminstance (in, n))
∀ na : activenorms • activenorm (na, environment)
∀ fn : fulfillednorms • fulfilled (fn, environment)
∀ ufn : unfulfillednorms • (¬ fulfilled (ufn, environment))
∀ rn : rewardednorms • (∃ rgn : rewardnorms •
  (rewardnorm (rgn, rn) ∧ fulfilled (rgn, environment)))
∀ pn : punishednorms • (∃ egn : enforcenorms •
  (enforces (egn, pn) ∧ fulfilled (egn, environment)))
  
```

26

## SMART Normative Systems model

### Norms and the concept of Power

- They also have analysed in [7] the different power relations that may arise in an agent society,
  - *institutional power*: social structures define norms that entitle agents to direct the behaviour of others (*institutional power*)
  - *personal power*: the power of an agent given by its capabilities to satisfy goals and the power of other agents to benefit or to hinder those goals.
- However,
  - no implementation of the architecture applying it to a real problem has been reported in literature,
  - there are no tools to support the development of a normative multi-agent system following their framework.

27

## AMELI

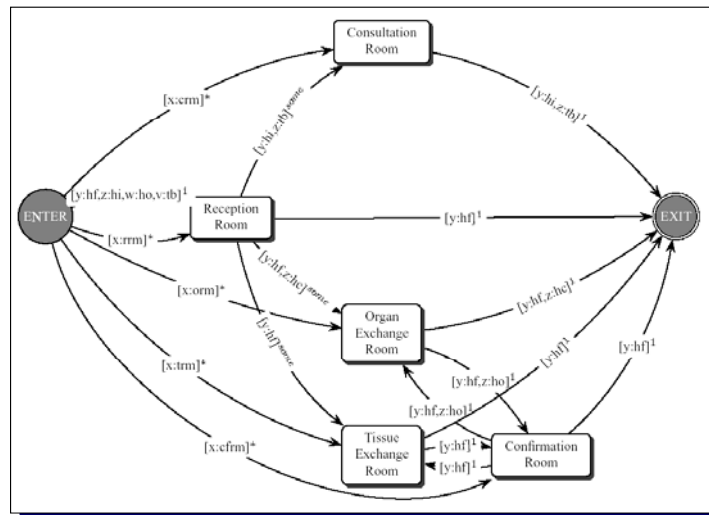
M. Esteve, J.A. Rodriguez-Aguilar and P. Noriega

- AMELI [3] is an institution middleware that is based in an electronic institution specification (ISLANDER).
- The ISLANDER framework [4] [3] is composed of:
  - A **Dialogical Framework**
    - Linguistic and social structure (**roles**) to give meaning to agent interactions,
  - A **Performative Structure**
    - **scenes** and relationships between scenes (navigation, precedence, causality)
  - **Rules**
    - Role-dependent conventions to establish social commitments
- Two hypotheses:
  - All agent actions are **messages**, observable by the e-institution
  - An Agent should **never** break the norms.

28

## AMELI

## ISLANDER: Performative Structure (I)

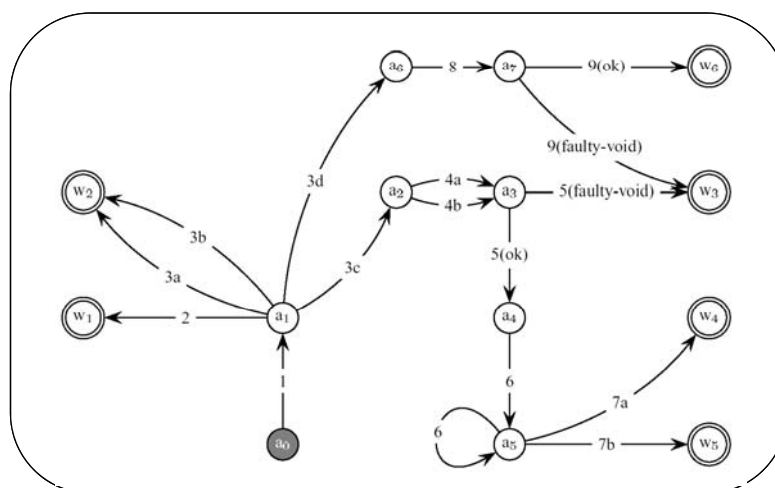


29

## AMELI

## ISLANDER: Performative structure (II)

- Scene conversational graphs: *Reception Room*



30

## AMELI

### Role of the AMELI middleware

- The AMELI middleware aims to:
  - Mediate and facilitate agent communication within conversations (scenes).
  - Coordinate and enforce:
    - to guarantee the correct evolution of each conversation (preventing errors made by the participating agents by filtering erroneous illocutions, thus protecting the institution).
    - To guarantee that agents' movements between scenes comply with the specification.
    - To control which obligations participating agents acquire and fulfil.
  - Manage information to facilitate participating agents the information they need to participate in the institution.

31

## AMELI

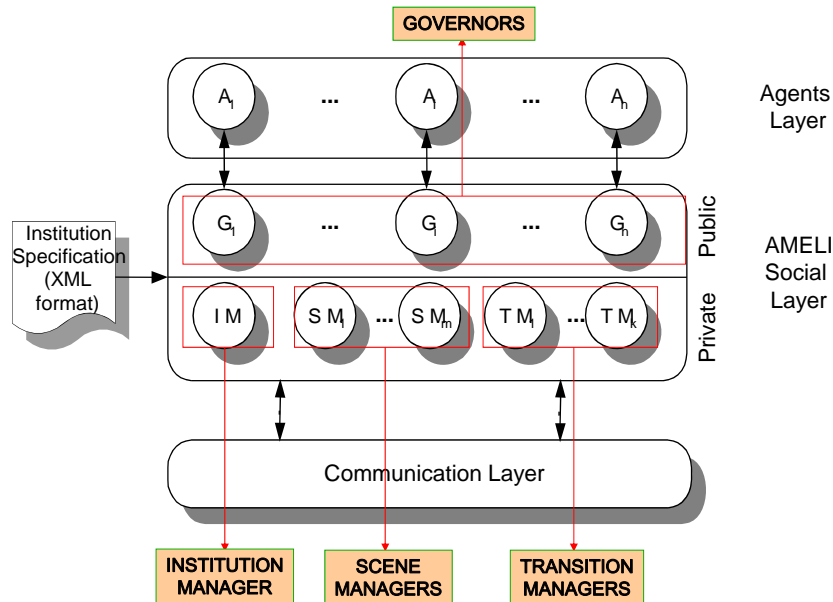
### Social Layer

- The current implementation of the social layer is composed of four types of agents:
  - An *institution manager* that starts the institution, authorises agents to enter, and controls the creation of scenes.
  - *Scene managers* responsible for governing scenes.
  - *Transition managers* control agents' movements between scenes.
  - *Governors* devoted to mediate the interaction of an agent with the rest of agents within the institution and to control agents' obligations.
    - Facilitates to the agent some information about the state of the institution.
    - Coordinates with other agents of the social layer for the correct execution of the institution.
    - Keeps track whether the agent pending acquires new obligations and fulfils some of its pending obligations.

32



## AMELI



33

## AMELI

### Limitations

- AMELI comes with a toolkit to help create institutional specifications, create the associated governors and create an e-Institution instance from an ISLANDER specification.
- But:
  - Weak notion of norms (norms only as **restrictions**, not as preference shapers).
  - Norms are never to be broken → **no autonomy**
  - Norms are only defined at the action-interaction level → **too low-level**
  - The only actions that can be controlled are messages
  - Message interactions are defined step by step → **no flexibility**.
  - Norm handling is done by the Governors, not the agents → agents **cannot reason** about the norms
  - No mechanism designed for the agents to introduce the norms in their reasoning → designer tends to **hard-code** them.

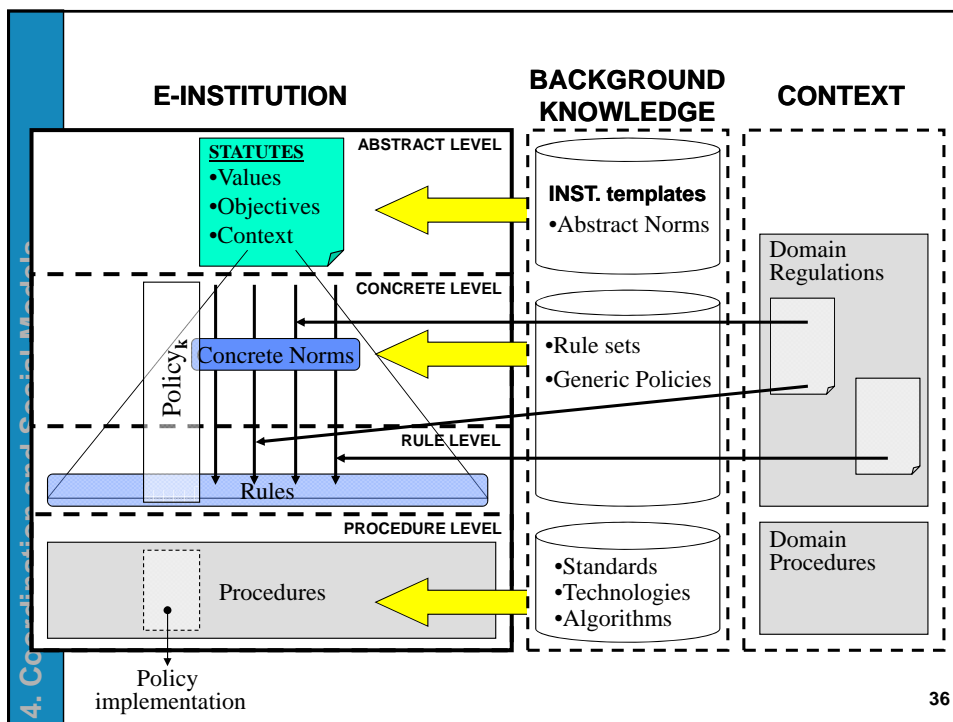
34

## HARMONIA

J. Vázquez-Salceda

- Approaches in literature were **too theoretical** (e.g. SMART) or **too practical** (e.g. AMELI)
  - There was a **gap** between the specification of abstract norms and the concrete implementation inside **e-Organizations**
- HARMONIA** [1] is a multi-level framework that proposes a formal connection between the different abstraction levels of a **Normative System**
  - It distinguishes between **normative** and **practical** levels
  - Analysis of **Context in Normative Systems**
    - context and norms, nested contexts, super-contexts' influence*
  - Establishes **connection** between formal specification and agent implementation
    - top-down*: design guidance
    - bottom-up*: track the origins of a protocol/plan
  - Norm enforcement** as detecting illegal worlds

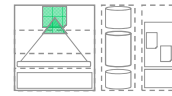
35



36

## HARMONIA

### Abstract Level



- The statutes define the
  - Values
  - Objectives
  - Context
 of the organization.

E.g.: *Organización Nacional de Trasplantes*:

The **main objective** of ONT is to **increase the number of organ donations** and the subsequent increase in available organs for transplants.  
 The ONT **operates according to the regulation of the national health system** and it strives to distribute the donated organs in the most **appropriate and correct** way according to the current technical knowledge and according to the **ethical principles of equality**.

37

## HARMONIA

### Abstract Level: Abstract Norms (I)



- **Values** are beliefs that we have about what is important and thus about how things should be.
  - “*Appropriate distribution*”
  - “*Distribution according to ethical principles of equality*”
  - “*Fairness of transaction*”
  - “*Respect privacy of persons*”
- **Values** can be considered as the most abstract level on which norms are expressed.
- The values of an organization can be defined by the set of **Abstract Norms** specified within the org. that contributes to that value

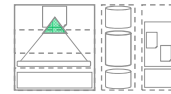
$$D(\text{equity}) ::= \{ F_{\text{ONT}}(\text{discriminate}(x, y, \text{age})), \\ O_{\text{ONT}}(\text{find\_best\_recipient}(\text{organ})), \\ \dots \}$$

$$D(\text{appropriate}(\text{distribution})) ::= O_{\text{ONT}}(\text{appropriate}(\text{distribution}))$$

38

**HARMONIA**

## Abstract Level: Abstract Norms (II)



- **Problem:** These norms are too **abstract**, as they use concepts that are not fully described in the organization's ontology.

- *"It is forbidden to **discriminate** based on age"*

$F_{\text{ONT}}(\text{discriminate}(x,y,\text{age})),$

- Norms can be **abstract** in the following ways:
  - They refer to an **abstract action**
  - They use **terms** that are **vague**
  - They abstract from **temporal aspects**
  - They abstract from **agents** and or **roles**
  - They refer to actions or situations that are **not** (directly) **controllable** and/or **verifiable** by the organization

39

**HARMONIA**

## Abstract Level: Abstract Norms (III)



- example 1: Abstract actions

*"a living donor should **consent** to the donation of an organ"*

$\left. \begin{array}{l} \text{sign}(\text{donor}, \text{contract}) \cup \text{carry}(\text{donor}, \text{will}) \cup \\ \text{tell}(\text{donor}, \text{family}) \end{array} \right\} \Rightarrow_{\text{ONT}} \text{Consent}(\text{donor})$

- example 2: Vague terms

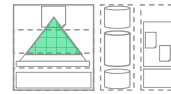
*"the **ONT** is obliged to ensure that the distribution of organs and tissues is **appropriate**"*

$\left. \begin{array}{l} O_{\text{ONT}}(\text{ensure\_quality}(\text{organ})) \wedge \\ O_{\text{ONT}}(\text{ensure\_compatibility}(\text{organ}, \text{recipient})) \end{array} \right\} \Rightarrow_{\text{ONT}} O_{\text{ONT}}(\text{appropriate}(\text{distribution}))$

40

**HARMONIA**

## Abstract/Concrete Level: Representing Norms



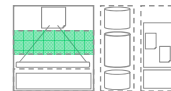
- Formal representation of norms needed
- Which logic?
  - Abstract and Concrete Norms permit, oblige or prohibit
  - Concrete Norms may be conditional
  - Concrete Norms may have temporal aspects
  - Concrete Norms are relativized to roles

→ variant of Deontic Logic

41

**HARMONIA**

## Concrete Level: Concrete Norms



- Concrete norms are the result of translating the abstract norms in the context of the organization into norms that make use of **terms** and **actions** that are defined in the organization's **ontology**.

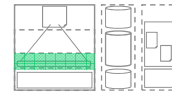
$$O_{\text{hosp}}(\text{consent}(\text{donor}(p,x)) < \text{done}(\text{transplant}(\text{hosp},x,p,q)))$$

- **Problem:** HOW is a concrete norm like this implemented in an e-Organization?

42

**HARMONIA**

## Rule Level (I)



- Translation from **Normative** dimension to a **Descriptive** one

- Idea: reduction from **Deontic Logic** to **Dynamic Logic**  
[J.-J. Meyer]

$$O_{\text{hosp}}(\text{consent}(\text{donor}(p,x)) < \text{do}(\text{transplant}(\text{hosp},x,p,q)))$$

$$\downarrow$$

$$[\text{transplant}(\text{hosp},x,p,q)]\text{done}(\text{consent}(\text{donor}))$$

$$O_{\text{buyer}}(\text{pay}(\text{goods},\text{seller},\text{price}) < \text{do}(\text{exit}(\text{buyer})))$$

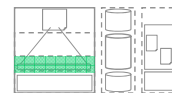
$$\downarrow$$

$$\text{not}(\text{done}(\text{pay}(\text{goods},\text{seller},\text{price}))) \rightarrow [\text{exit}(\text{buyer})]V(\text{fine}(\text{buyer}))$$

43

**HARMONIA**

## Rule Level (II)

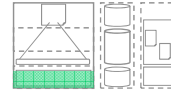


- Rules, Violations and Sanctions
  - Violation rules define violations
  - A **violation** is composed by
    - **pre-conditions**
    - **sanction**
    - **side effects**
  - **Pre-conditions** are used by Police Agents to detect violations.
  - **Sanctions** are used by Flexible Normative Agents to reason about the utility of breaking the related rule.
  - **Side effects** are used by internal agents to recover the system from the violation.

44

## HARMONIA

### Procedure Level



- Idea: the final implementation of the system
- Formally, translation from **Dynamic Logic** to a **Procedural Language**:

$$\{P\} a \{Q\} \equiv P \rightarrow [a]Q$$

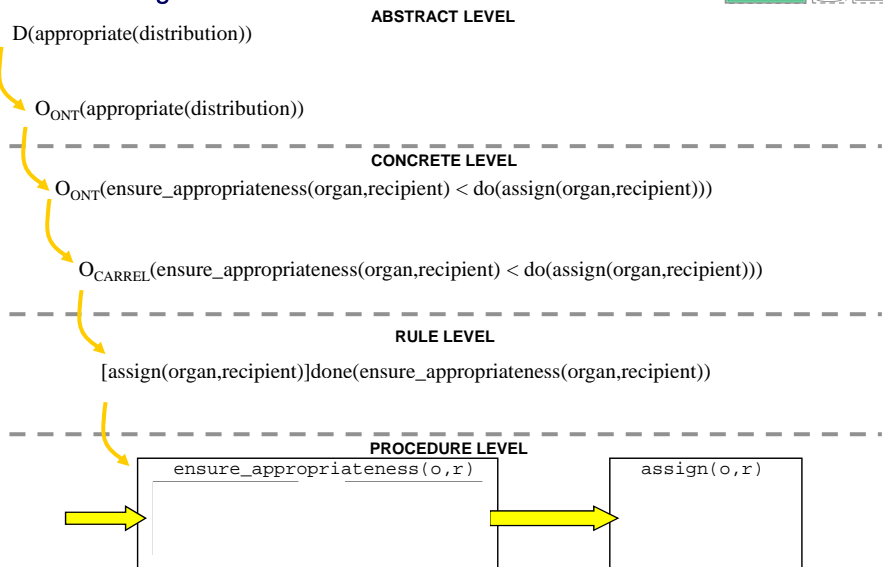
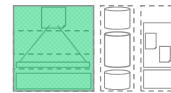
- Example:

$(\text{not}(\text{done}(\text{assign}(o, r))) \wedge \text{done}(\text{ensure\_appropriateness}(o, r)))$   
 $\rightarrow [\text{assign}(o, r)] \text{done}(\text{assign}(o, r))$   
 $\downarrow$   
 $\{\text{not}(\text{done}(\text{assign}(o, r))) \ \& \ \text{done}(\text{ensure\_appropriateness}(o, r))\}$   
**assign(o, r)**  
 $\{\text{done}(\text{assign}(o, r))\}$

45

## HARMONIA

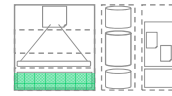
### Connecting with the Procedure Level



46

## HARMONIA

### Procedure Level: Implementing Norms in eInstitutions (I)

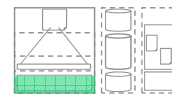


- Implementation of norms  $\neq$  Implementing a theorem prover from institutional perspective to check protocol compliance
- Implementation of a safe environment (**norm enforcement**)
- 2 options depending on control over agents
  - Defining constraints on unwanted behaviour
  - Defining violations and reacting to these violations
- our assumptions:
  - Norms can be sometimes violated by agents
  - The internal state of agents is neither observable nor controllable
    - actions cannot be imposed on an agent's intentions
    - agents as black boxes
    - only their observable behaviour and actions

47

## HARMONIA

### Procedure Level: Implementing Norms in eInstitutions (II)



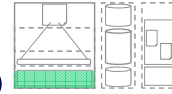
- Norm enforcement is not centralized but distributed in a set of agents, the Police Agents
  - They check if a given (observable) action was legal or illegal given the violation conditions defined for that context.
- The Agent Platform should assist the Police Agents, providing fast, very efficient aids for norm enforcement as additional platform services and mechanisms.
- A) *Detection of the occurrence of an action*
  - Police Agents may become overloaded checking ALL actions
  - **black list mechanism** (of actions to monitor) e.g., assign
  - **action alarm mechanism** (alarm to the Police Agent)
  - The Police Agent checks if conditions for a violation apply.

48



## HARMONIA

### Procedure Level: Implementing Norms in eInstitutions (III)

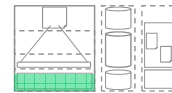


- *B) Detection of activation/deactivation of norms*
  - activation = when condition C is true
  - deactivation = when P holds, A is done or C is false
  - reaction time: time allowed between norm activation and reaction
  - Depending on the complexity to check C, the platform should implement the appropriate **fast-access data structures** and/or **processing mechanisms** to reduce Police Agents' computation burden
- *C) Deadline control*
  - a **clock trigger mechanism** to detect that a deadline has passed

49

## HARMONIA

### Procedure Level: Example of Norm violation definition



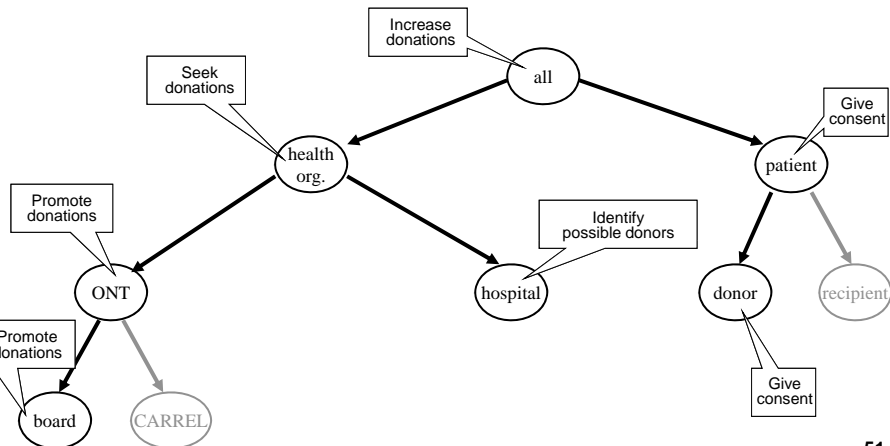
Norm	FORBIDDEN( <i>allocator</i> DO <i>assign(organ, recipient)</i> )
condition	IF NOT( <i>hospital</i> DONE <i>ensure_quality(organ)</i> )
Violation condition	NOT( <i>done(ensure_quality(organ))</i> ) AND <i>done(assign(organ, recipient))</i>
Detection mechanism	{ <i>detect_alarm(assign, 'starting')</i> ; <i>check(done(ensure_quality(organ)))</i> };
Sanction	<i>inform(board, "NOT(done(ensure_quality(organ)) AND done(assign(organ, recipient)))"</i> )
Repairs	{ <i>stop_assignment(organ)</i> ; <i>record("NOT(done(ensure_quality(organ)) AND done(assign(organ, recipient))", incident_log)</i> ; <i>detect_alarm(ensure_quality, 'done')</i> ; <i>check(done(ensure_quality(organ)))</i> ; <i>resume_assignment(organ)</i> };

50

**HARMONIA**

All levels: Roles (I)

- Role definition guided by goal distribution
  - Origin: Objectives in Statutes
- Distribution of responsibilities

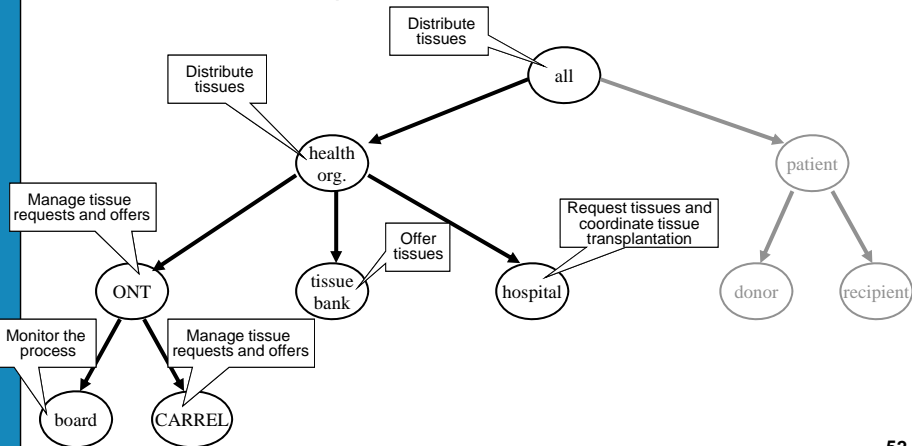


51

**HARMONIA**

All levels: Roles (I)

- Role definition guided by goal distribution
  - Origin: Objectives in Statutes
- Distribution of responsibilities

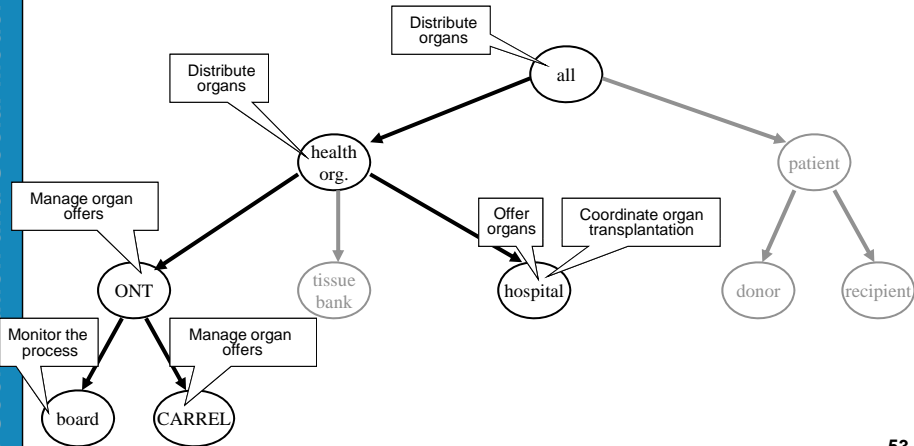


52

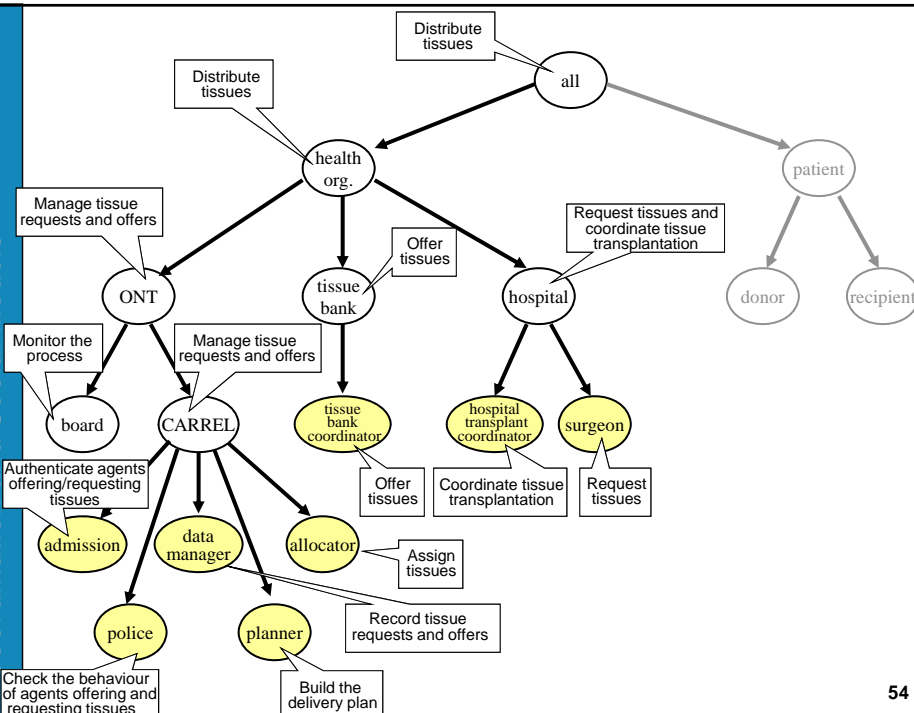
## HARMONIA

All levels: Roles (I)

- Role definition guided by goal distribution
  - Origin: Objectives in Statutes
- Distribution of responsibilities



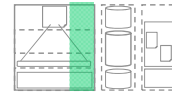
53



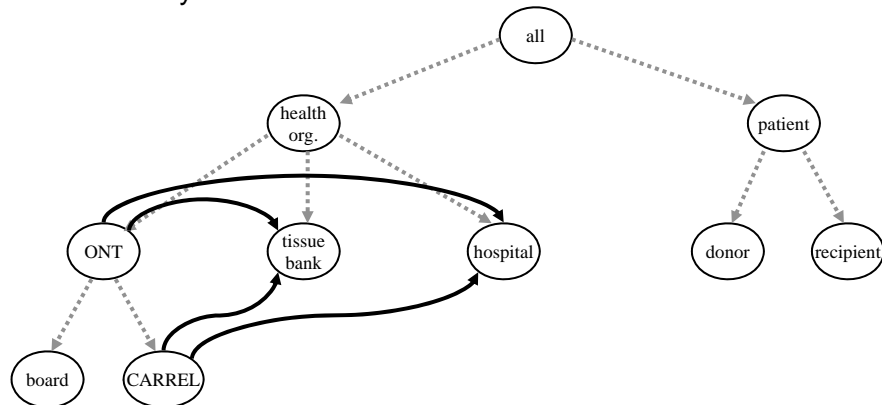
54

## HARMONIA

All levels: Roles (II)



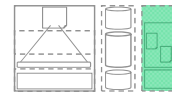
- Role hierarchy extended with **power relations** to model the distribution of responsibilities not defined in the hierarchy



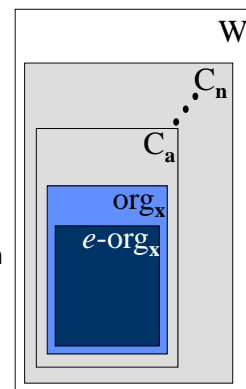
55

## HARMONIA

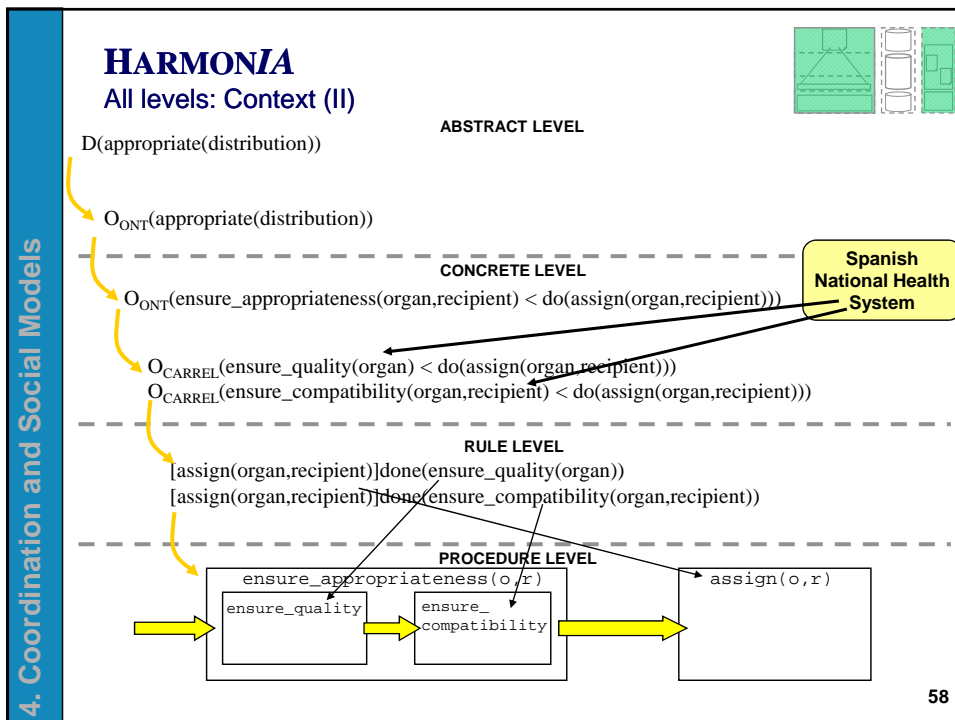
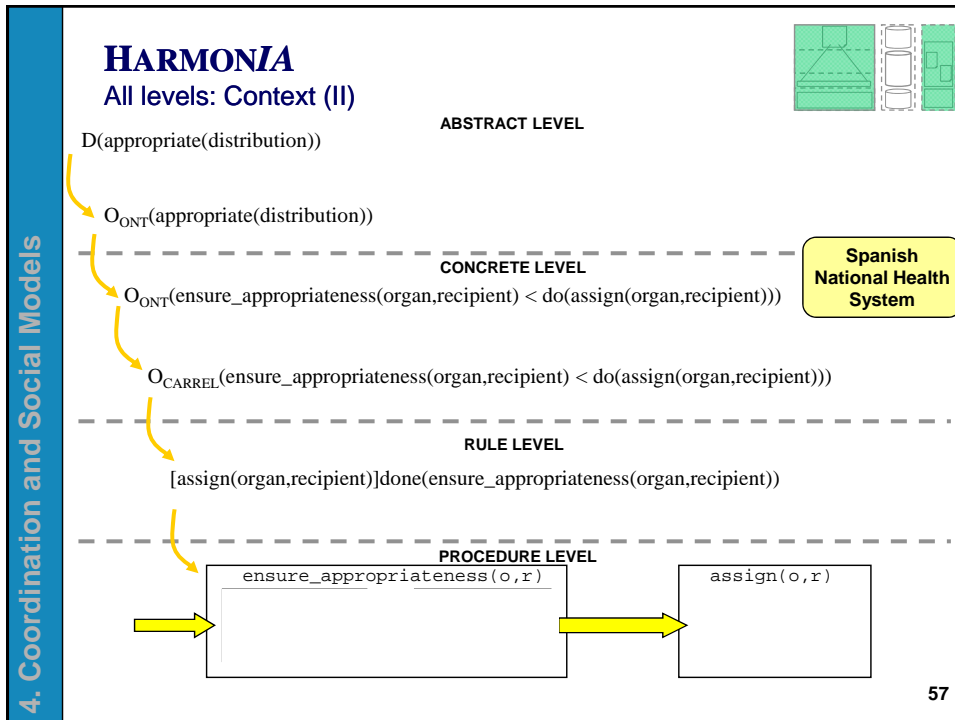
All levels: Context (I)



- **Statutes** make reference to a **surrounding context**
- Links with the idea of **nested contexts**
  - $e\text{-org}_x$  is a context defining a **vocabulary** and a **normative system**
  - there are super-contexts that have an **influence** in  $e\text{-org}_x$  definition
- formal view: **influence** as **interpretation** in the subcontext
  - counts-as operator  $\Rightarrow$  as a link between interpretations
- influence in several levels of abstraction
  - vocabulary (terms, predicates)
  - values, norms, rules and procedures



56

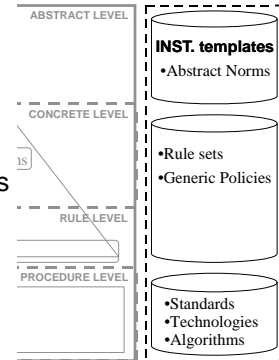


## HARMONIA

All levels: Background Knowledge

- The Background Knowledge is a repository containing templates that can be adapted to create new e-organizations

- At the abstract level, it provides a collection of abstract norms and the related ontology and abstract roles
- At the concrete and rule levels, it provides templates for some generic policies
  - e.g., the security policy
    - concrete norms, rules and ontology
- At the procedure level it provides a link with the standards, technologies and algorithms needed to implement the policies



- Idea for future: **Institutional templates** to be *parameterized, adapted or implemented* to build e-Institutions.

59

## OMNI

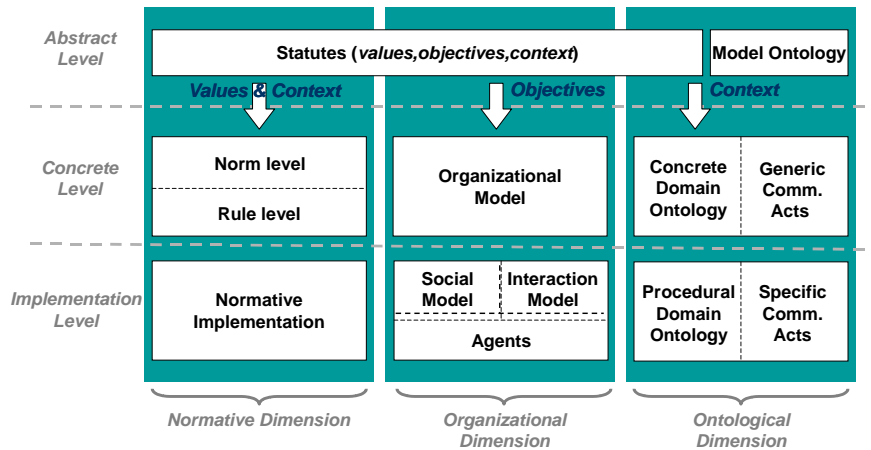
J. Vázquez-Salceda, V. Dignum and F. Dignum

- OMNI**: Organizational Model for Normative Institutions [5]
  - Integration of **HARMONIA**'s Normative concepts [1] with **OperA**'s Organizational concepts [6]
  - integrated framework for both
    - closed systems with fixed participants and interaction protocols,
    - open, flexible systems that allow and adapt to the participation of heterogeneous agents with different agendas.
- Layered Approach
  - Abstract Level: Requirement analysis
  - Concrete Level: Analysis and design process
  - Implementation Level: Design specification in a given multi-agent architecture
- Top-down
  - Methodological guidance for design
- Bottom-up
  - Trace of origin and motivation for rules and protocols

60

## OMNI

### Levels and dimensions



61

## OMNI

### Dimensions

- **Normative Dimension**
  - Norms and rules that guide agent behaviour
  - Includes a model of the environment regulations
  - Comes from **HARMONIA**
- **Organizational Dimension**
  - Captures the organizational structure and requirements
  - Comes from **OperA**, with norm language coming from **HARMONIA**
- **Ontological Dimension**
  - Content: concepts and relationships
  - Communication Language
  - Comes from **OperA**

62

## OMNI

### Abstract Level

- **Statutes**
  - **Main objective(s)** of the organization,
    - Guides organizational design
    - Input for the Organizational Model
  - **Values** that direct the fulfilling of this objective
    - Guides normative design
    - Input for the Norm level
  - **Context**
    - Guides the ontological design
    - Influences also the normative design
- **Generic Terms**
  - In-contextual concepts
- **Model Ontology**
  - concepts of the framework itself
  - E.g. norm, rule, role, group, violation, landmark...

63

## OMNI

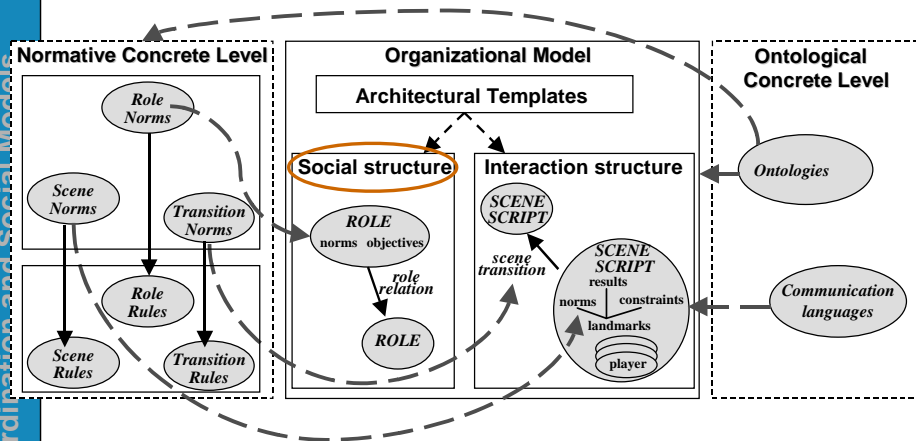
### Concrete Level

- Analysis and design process
  - Based on abstract values and objectives
- Refinement in three dimensions
  - **Organizational Model**
    - **Social Structure**
      - roles, groups, dependencies
    - **Interaction Structure**
      - scene scripts, connections, transitions
  - **Normative Structure**
    - role, scene and transition norms (deontic)
    - role, scene and transition rules (operational)
  - **Communication Structure**
    - Concrete ontological concepts, communicative acts, domain ontology

64



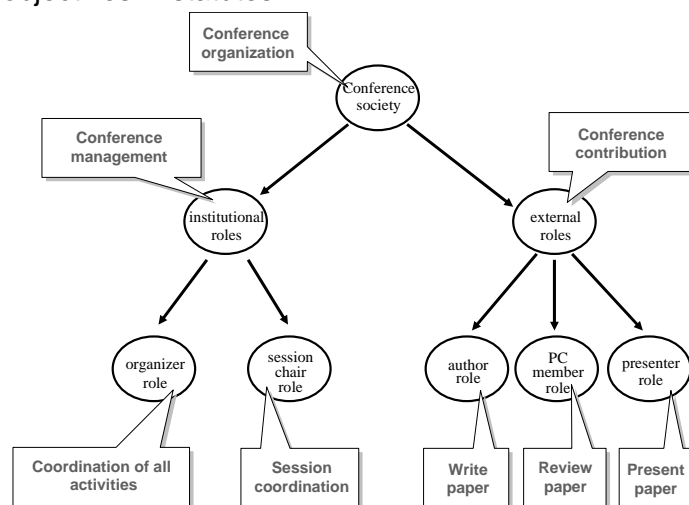
## OMNI Concrete Level



65

## OMNI Concrete Level: Social Structure

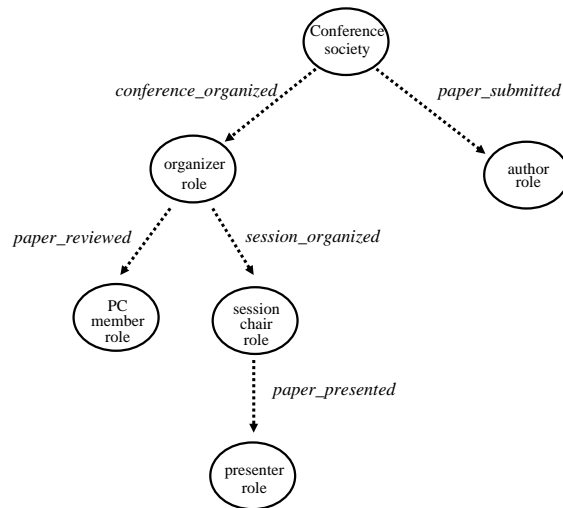
- Role Model based in goal decomposition coming from objectives in statutes



66

## OMNI

## Concrete Level: Social Structure. Role dependencies



67

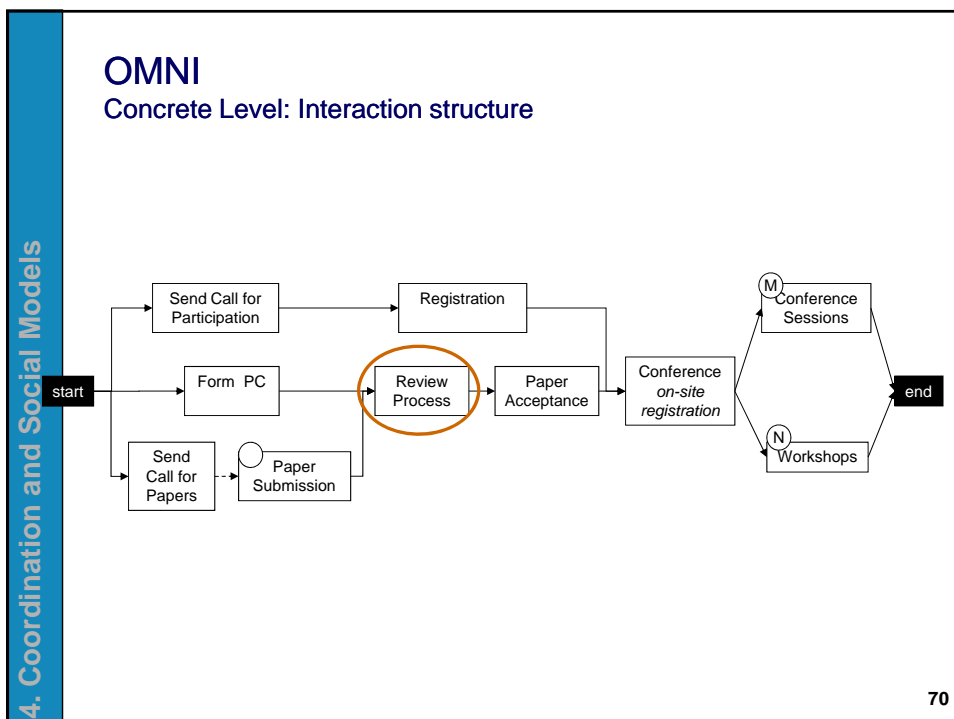
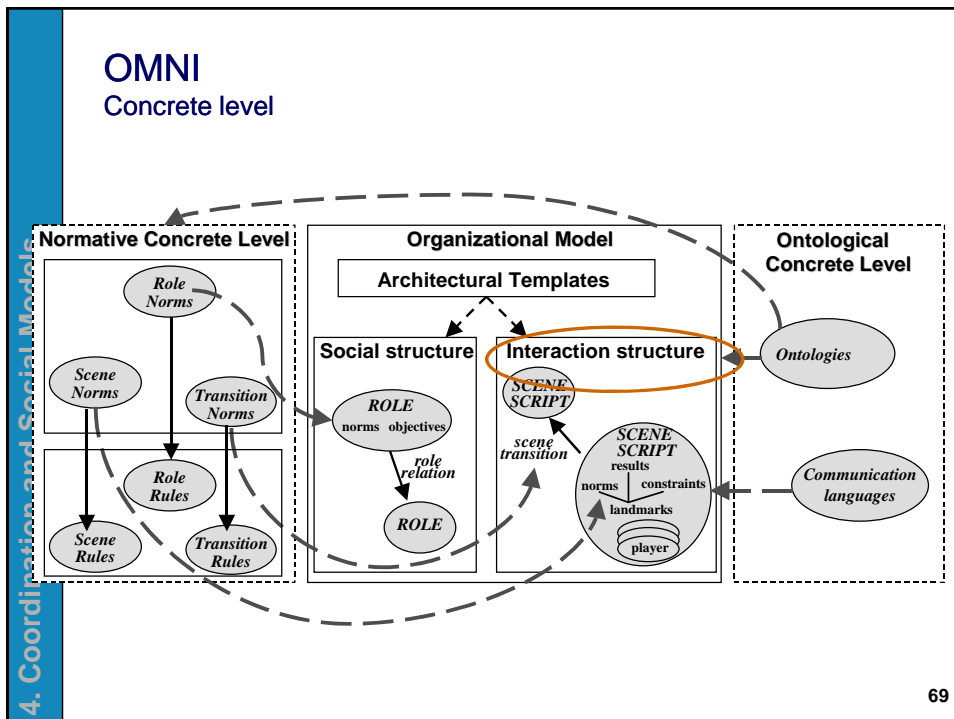
## OMNI

## Concrete Level: Social Structure. Role example

Role: PC Member	
Objectives	paper_reviewed(P, Rep)
Sub-objectives	{ read(P), report_written(P, Rep), review_received(Org, P, Rep) }
Rights	access-confman-program(me)
Norms	OBLIGED understand(English) IF DONE assigned (P, me, Deadline) THEN OBLIGED paper_reviewed(P, Rep) BEFORE Deadline IF DONE paper_assigned(P, me, _) AND direct_colleague(author(P)) THEN OBLIGED review_refused(P) BEFORE TOMORROW
Type	external

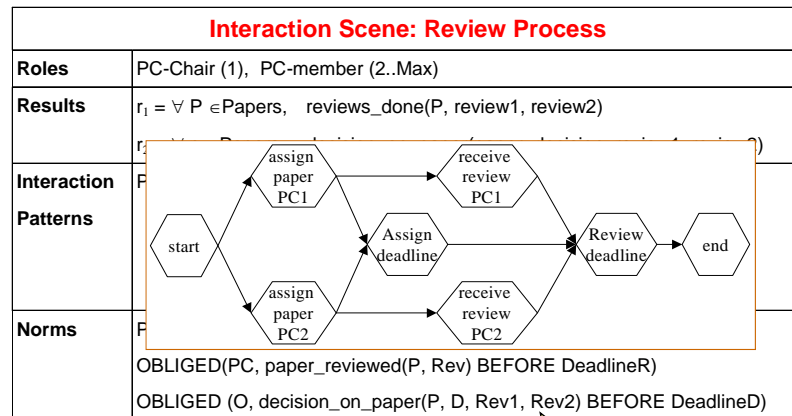
OperA Role descriptions +  
HARMONIA norm language

68



## OMNI

Concrete Level: Interaction Structure. Scene example



landmarks

OperA Scene descriptions +  
HARMONIA norm language

71

## OMNI

Implementation Level

- **Norm enforcement**
  - *Protocols* and *Rules* : enable agents to comply with organizational norms
- **Role enactment**
  - *Social Contracts* : commitments regulating the enactment of roles by individual agents.
  - *Interaction Contracts* : specific interactions such as agreed upon by agents
- **Ontologies**
  - *Specific Communication Acts* : actual communication languages actually as fixed in interaction contracts.
  - *Specific Communicative Acts* : implement the content ontologies.

72

## OMNI

Implementation Level: Social Model

- Role enacting agents (rea)
  - agents realizing expected behavior of role
- Social Contract
  - describes a specific agreement for a rea

social-contract(agent1, PC-member, {})

$\forall \text{scene: PC-member} \in \text{roles}(\text{scene}), \text{rea}(\text{agent1}, \text{PC-member}, \text{scene})$

social-contract(agent2, PC-member, {maximum to review papers is 3})

$\forall \text{scene: PC-member} \in \text{roles}(\text{scene}), \text{rea}(\text{agent2}, \text{PC-member}, \text{scene})$

73

## OMNI

Implementation Level: Interaction Model

- Scene Instantiation
  - protocols realizing landmarks
- Interaction contract
  - describes a specific performance of a scene

interaction-contract( {PC-Chair, pc1, pc2, pc3, pc4}, review-process, {}, P1)

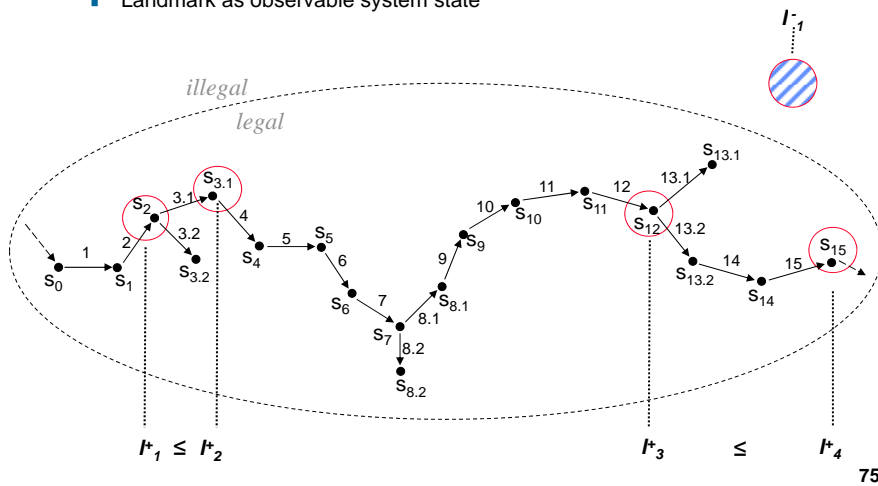
interaction-contract( { PC-Chair, pc3, pc5}, review-process,  
     { IF NOT reviews-done BEFORE DeadlineR THEN  
       PERMITTED(PC-Chair, paper-accepted(P)), P2)

74

## OMNI

### Implementation Level: Norm Enforcement

- Based in **HARMONIA**
- New idea: **OperA's** Landmarks to monitor norm execution
  - Landmark as observable system state



## References

- [1] J. Vázquez Salceda. "The Role of Norms and Electronic Institutions in Multiagent Systems", Birkhauser-Verlag, 2004
- [2] F. López y López. "Social Power and Norms: Impact on Agent Behaviour". PhD thesis, University of Southampton, 2003
- [3] M. Esteva. "Electronic Institutions: from specification to development". PhD thesis, Institut d'Investigació en Intel·ligència Artificial (IIIA), 2003.
- [4] J.A. Rodríguez-Aguilar. "On the Design and Construction of Agent-mediated Electronic Institutions". PhD thesis, Institut d'Investigació en Intel·ligència Artificial (IIIA), 2001.
- [5] J. Vázquez-Salceda, V. Dignum, F. Dignum. "Organizing Multiagent Systems". Journal of Autonomous Agents and Multi-Agent Systems, vol. 11 issue 3, pp. 307-360, 2005.
- [6] Virginia Dignum, "A Model for Organizational Interaction: Based on Agents, Founded in Logic". PhD dissertation, 2004

## References

- [7] F. López y López and M. Luck. “**Empowered Situations of Autonomous Agents**”. In F.J. Garijo, J.C. Riquelme, and M. Toro, editors, IBERAMIA 2002, volume 2527 of LNAI, pages 585–595, Berlin Heidelberg, 2002. Springer Verlag.
- [8] M. d’Inverno and M. Luck. “**Understanding Agent Systems**”. Springer Verlag, 2001.
- [9] J.M. Spivey. “**The Z Notation: A Reference Manual**”. Prentice Hall, London, 1992.

These slides are based mainly in [1], [5], [2], [3], and some material from F. Dignum, M. Esteva and M. Sergot