



ADELFE

(Toolkit to Develop Software with Emerging Functionalities)

Methodology & Meta-Model




Carole BERNON & Marie-Pierre GLEIZES
{bernon, gleizes}@irit.fr

SMAC (Cooperative MAS) Team
<http://www.irit.fr/SMAC/>
IRIT - Toulouse III - France





1



Contents

- Reasons that led us to ADELFE
- The methodology
 - ✓ Overview of the process
 - ✓ Specific points
 - ✓ Strengths and limits
- MAS meta-model adopted in ADELFE


2



Context

- Study of open and complex systems in evolutionary environments
- The system must realize the "right" function:
 - ✓ Theorem on functional adequacy
 - For any functionally adequate system in a given environment, there is a system having a cooperative internal medium which realises an equivalent function


3



Context

- Study of open and complex systems in evolutionary environments
- The system must realize the "right" function
 - ➔ cooperation at the component level
- The system must be able to adapt
 - ✓ Changing the organization of the system changes its function
 - ➔ self-organization


4



Context

- Study of open and complex systems in evolutionary environments
- The system must realize the "right" function
 - ➔ cooperation at the component level
- The system must be able to adapt
 - ➔ self-organization
- Algorithm a priori unknown
 - ✓ The function must emerge at the macro level
 - ✓ Not known at the micro (component) level
 - ✓ Local criterion
 - ➔ cooperation

5




Context

- Study of open and complex systems in evolutionary environments
- The system must realize the "right" function
 - ➔ cooperation at the component level
- The system must be able to adapt
 - ➔ self-organization
- Algorithm a priori unknown
 - ➔ emergent function

Self-organization led by cooperation
Adaptive Multi-Agent Systems (AMAS)


6



ADELFE

- Several applications implemented using this AMAS theory
- Need for a design methodology
- National RNTL-funded project
- From December 2000 to September 2003
- 4 partners:
 - ✓ ARTAL Technologies
 - ✓ TNI-Valiosys
 - ✓ IRIT
 - ✓ L3I

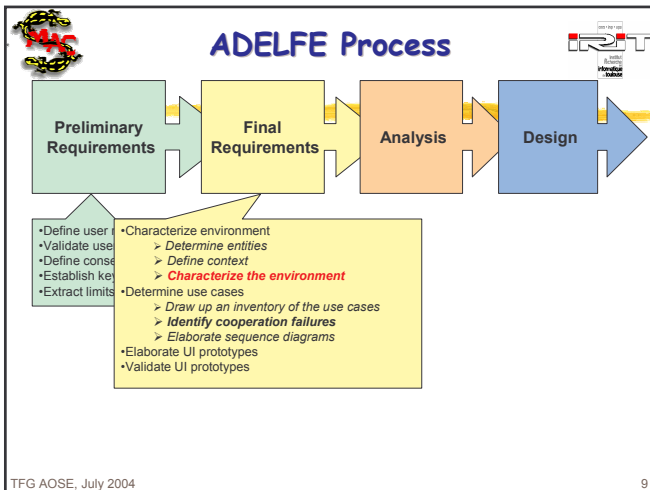

TFG AOSE, July 2004 7



A Methodology

- **Process**
 - ✓ Based on the RUP
- **Notations**
 - ✓ Uses UML and AUML notations
- **Tools**
 - ✓ Toolkit to follow the process
 - ✓ Customized OpenTool
 - ✓ Graphical tool for verifying AMAS adequacy
- **Available at:**
 - ✓ <http://www.irit.fr/ADELFE>

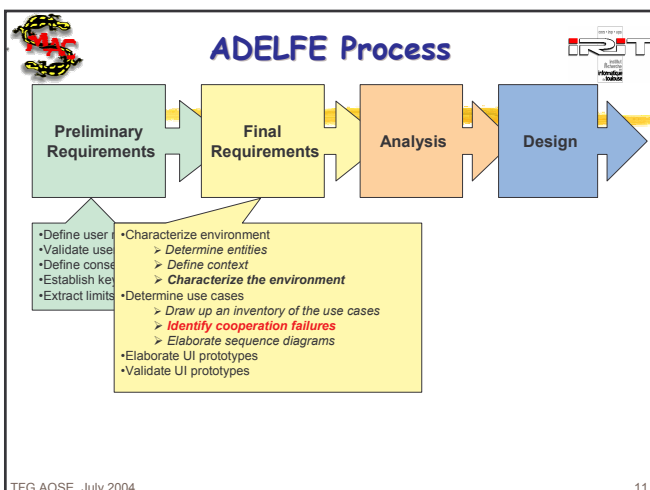
TFG AOSE, July 2004 8

Characterize the Environment

- Adaptation depends on interactions with it
- The designer has to qualify it:
 - ✓ Accessible
 - ✓ Continuous
 - ✓ Deterministic
 - ✓ Dynamic

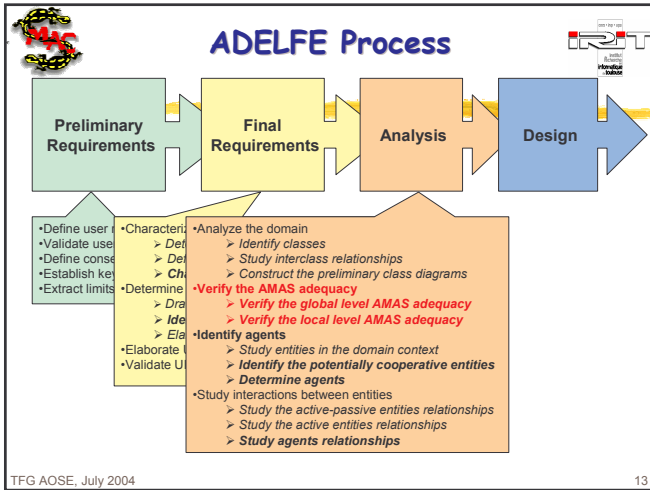
TFG AOSE, July 2004 10




Identify Cooperation Failures

- Situations not totally controlled by designers
 - ✓ « Wrong » interactions between entities and the system
 - Protocol not respected
 - Physical failure
 - Request not understood...
- Can lead to NCS
- Specific notation in use cases

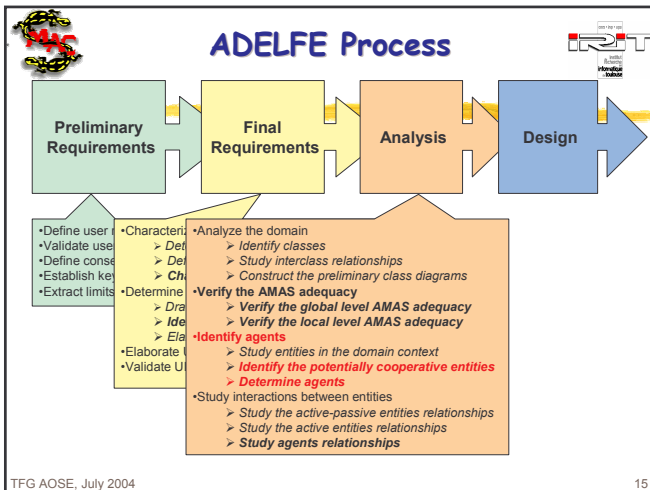
TFG AOSE, July 2004 12



Verify the AMAS Adequacy

- Is an AMAS needed at the global and/or at the local level?
 - ✓ No algorithm known
 - ✓ Evolution needed
 - ✓ Open, non-linear system
 - ✓ Distribution
 - ✓ Rationality and behaviour of the components
- Technology not suited → another methodology?
- Recursion → apply again

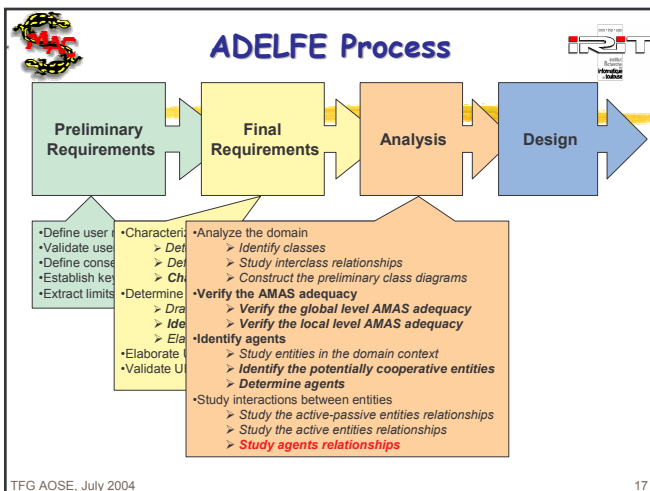
TFG AOSE, July 2004 14



Identify Agents

- Study criteria:
 - ✓ Autonomy
 - ✓ Local goal
 - ✓ Interactions
 - ✓ Partial view of its environment
 - ✓ Ability to negotiate
- Chosen among entities prone to cooperation failures
- Stereotyped with « cooperative agent » (class CooperativeAgent)

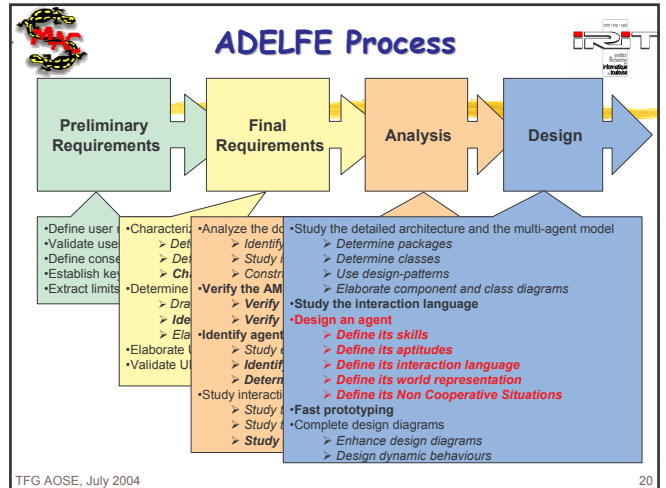
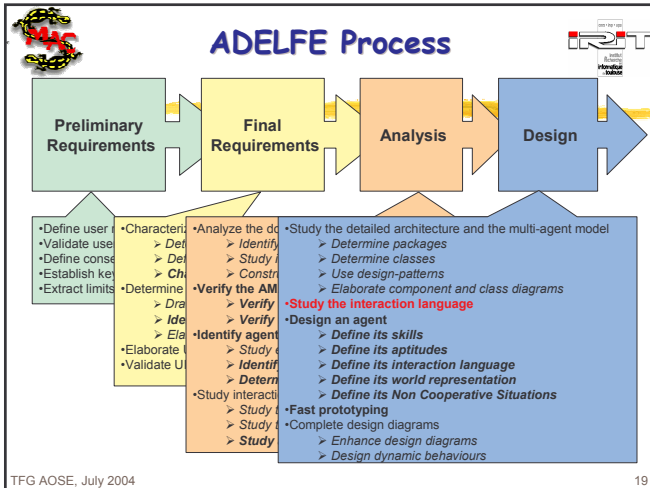
TFG AOSE, July 2004 16



Study Agents Relationships

- Interaction protocols expressed using AAML
- Extended to fit with ADELFE
 - ✓ Method attached to an OR or XOR node (to make a choice, aptitude)
 - ✓ Method attached to a reception point of a message (to detect a possible NCS)

TFG AOSE, July 2004 18



Design Agents

- « Classical » modules:
 - ✓ Interactions: assign interaction protocols to agents → generates a state machine per protocol
 - ✓ Skills: knowledge about a domain enabling an agent to execute actions
 - ✓ Representations: about itself, others, the environment
 - ✓ Aptitudes: capabilities to reason on knowledge or representations
- **Stereotypes:** <<interaction>>, <<skill>>, <<representation>> and <<aptitude>>
 - ✓ Rules to govern their use

TFG AOSE, July 2004 21

Design Agents (2)

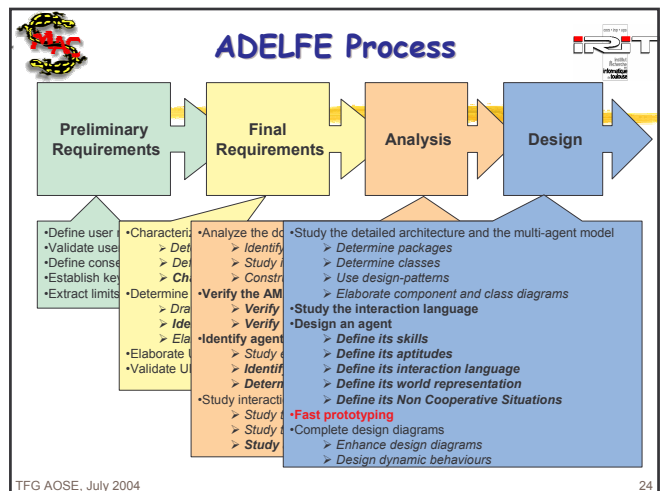
- **Cooperation module:**
 - ✓ Local cooperation rules (<<cooperation>>)
 - ✓ Enumerate all the NCS for an agent → model
 - Perception: Incomprehension, Ambiguity
 - Decision: Incompetence, Unproductiveness
 - Action: Concurrence, Conflict, Uselessness
 - ✓ For each NCS → table

TFG AOSE, July 2004 22

Design Agents (2)

Name	Total incompetence
State	Receipt of a request
Description	An agent faces a total incomprehension when it cannot associate any meaning to the message it received
Conditions	During the interpretation phase the agent compares the received request with its own representation and cannot extract any informative content from this message
Actions	The agent is cooperative and does not ignore the message, it transmits it to an agent it believes more relevant according to its representations on others

TFG AOSE, July 2004 23





Fast Prototyping



- Simulation functionality of OpenTool
- Running a state machine from a collaboration diagram
- Test behaviour of an agent
 - ✓ Deadlocks
 - ✓ Lacks
 - ✓ Inconsistency
 - ✓ ...



Strengths of ADELFE



- Can be used by a non specialist
 - ✓ Based on standards
 - ✓ Guide: interactive tool, AMAS adequacy tool
- Target systems
 - ✓ Takes into account dynamics
 - ✓ Coupled with a theory
- Does not suppose that agents are known
- Their behaviour may be tested



Limits of ADELFE



- Specialized
 - ✓ Linked with another methodology?
 - ✓ Fragments
- Some activities could be improved
 - ✓ Fast prototyping
- Some work definitions are lacking
 - ✓ But exist in the RUP



Why Meta-Modeling?



- Meta-model
 - ✓ Modelisation
 - ✓ Concepts definition
- Advantages
 - ✓ Unification eases model transformation
 - ✓ Design a system \equiv instantiate a meta-model



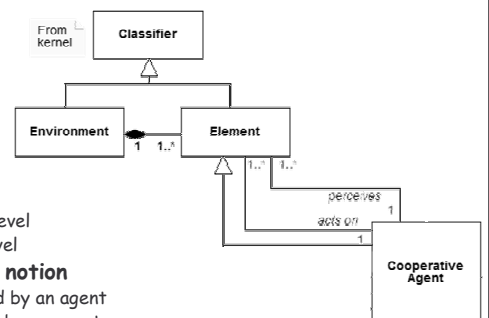
MAS Meta-Model



- Cooperative agent:
 - ✓ Ignores the global function
 - ✓ Local goal
 - ✓ Cooperative relations with others



The Environment



- Two levels
 - ✓ System level
 - ✓ Agent level
- Important notion
 - ✓ Perceived by an agent
 - ✓ Modified by an agent

Features of an Agent

- **An agent:**
 - ✓ Possesses knowledge about others, its environment, itself
 - ✓ Reasons about its knowledge and beliefs
 - ✓ Realises its partial function
 - ✓ Has intrinsic or physical properties
 - ✓ Communicates with other or its environment

TFG AOSE, July 2004 31

Adaptation

- **Representations and skills may evolve**
➔ **AMAS**

TFG AOSE, July 2004 32

Social Attitude of an Agent

- **Local cooperation rules**
- **Detect and remove NCS**

TFG AOSE, July 2004 33

MAS Meta-Model Adopted

- **Some notions are not useful**
 - ✓ Role: not a fixed organization
 - ✓ Goal: modelled by skills, aptitudes and representations
 - ✓ Ontology: adaptation to others (and their « language »)

TFG AOSE, July 2004 34

35