Applying Process Document Standardization to INGENIAS

DRAFT

Alma Gómez-Rodríguez Juan C. González-Moreno

June 18, 2010

Contents

Acknowledgements

This work has been supported by the project *Novos entornos colaborativos para o ensino* supported by Xunta de Galicia with grant 08SIN009305PR.

Introduction

The INGENIAS methodology covers the analysis and design of MAS and it is intended for general use; that is, with no restrictions on application domains. It has shown its capability and maturity as the supporting specification for the development of Multi-Agent Systems (MAS). The methodology is supported by the INGENIAS Development Kit (IDK), which contains a graphical editor for MAS specifications. Besides, the INGENIAS Agent Framework (IAF) [3] integrated in the IDK has been proposed for enabling a full model-driven development and transforming automatically specifications into code in the Java Agent DEvelopment Framework.

The software development process proposed by the methodology is based on Rational Unified Process (RUP)[6]. The methodology distributes the tasks of analysis and design in three consecutive phases: Inception, Elaboration and Construction, with several iterations (where iteration means a complete cycle of development, which includes the performance of some analysis, design, implementation and proofs tasks). The sequence of iterations leads to the procurement of the final system.

The process of development of INGENIAS methodology is often represented by its authors in a tabular form (see Figure 1). In the table, we can see that they consider three development phases: Inception, Elaboration and Construction, with two different types of workflow: Analysis and Design. The methodology pays few attention, compared to RUP, to Implementation and Test workflows, because it provides some tools which automatically generate code, in parallel with system's specification. Attending this facility, these workflows are considered not to be modeled as fundamental part of the process.

INGENIAS tries to follow and Model Driven Development(MDD) [1], so it is based on the definition of a set of meta-models that describe the elements that form a MAS from several viewpoints. The specification of a MAS is structured in five viewpoints: [5]

1. The definition, control and management of each agent mental state
2. The agent interactions
3. The MAS organization
4. The environment
5. The tasks and goals assigned to each agent

The development process is supported by a set of tools, which are generated from the meta-models specification by means of a meta-modeling processor (which is the core of the IDK). MAS modeling is facilitated by a graphical editor and verification tools. The methodology has been used in several examples from different domains, such as PC management, stock market, word-processor assistant, and specially the application to collaborative filtering information systems.

	Phases		
	Inception	Elaboration	Construction
Analysis	To generate use cases and identify actions of these use cases with the corresponding Interaction Model To outline the system architecture with an Organization Model To generate Environment Models which reflects Requirement elicitation	To refine use cases To generate Agent Models that detail the elements of the system architecture To continue with the Organization Models, identifying workflows and tasks To obtain Task and Goal Models to highlight control constraints (main goals, goal decomposition) To refine the Environment Model including new elements	To study the remaining use cases
Design	To generate a prototype using RAD tools such as ZEUS or AgentTool	To focus the Organization Model on workflow To refine Tasks and Goal Models reflecting the dependencies and needs identified in workflows and the relationships with system's goals To show how tasks are executed using Interaction Models To generate Agent Models which show required mental state patterns	To generate new Agent models or refining existing ones To study social relationships in order to refine the organization

Table 1: Lifecycle for INGENIAS Methodology

Figure 1: Lifecycle for Ingenias Methodology

Detailed references of the methodology from their authors can be found in [2, 4, 5].

The INGENIAS Process lifecycle

INGENIAS is based on RUP. RUP takes the system architecture as guideline for development, and, in the same line, INGENIAS take the Organizational Model as basis for the SMA definition and construction.

The INGENIAS Metamodel

INGENIAS is based on the concept of metamodel. A metamodel, according to INGENIAS, defines the primitives and the syntactic and semantic properties of a model. As previously stated INGENIAS provides five meta-models that constitute the five different views of the system. Each of these models is defined separately, so the general metamodel of the methodology is distributed among these five metamodels. This means that in the original proposal of INGENIAS there is not an unique metamodel, but five of them regarding the five models the methodology proposed to construct for developing a system.

Moreover, each of these metamodels is very detailed (fine grained). This is due to two main reasons: because it intends to be a precise definition of the specification language and its particular syntax and semantics and also because each metamodel introduced has all the modeling elements inherent to INGENIAS methodology and also other needed for the tool provided for development: the INGENIAS Development Kit (IDK).

In Figure 3, the model of the Agent Meta-model is shown. An agent is identified as an autonomous entity, with particular goals and a unique identity. So that, in the model all the properties related to agents are defined. Three fundamental elements are identified: the roles the agent must play, the tasks the agent must accomplish and its mental state. The relationships among them show how an agent can pursue its goals and how it achieves that goals executing a particular tasks. It is important to note that this model has some entities in common with other meta-models, in particular Agents and Roles.

Figure 4 the metamodel of the Organization Model is shown. The Organization Model shows the MAS architecture and defines the workflows of the system. From this workflow definitions new interactions, that will be incorporated to Interaction Model, can arise. This model also helps in identifying new relevant

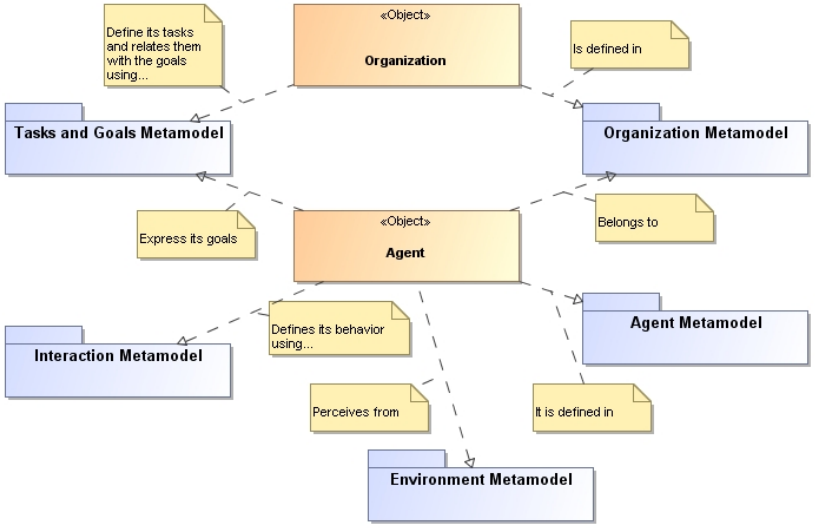

Figure 2: Global metamodel of INGENIAS Methodology

tasks or goals pursued globally (in contrast with goals pursued by individual agents).

Definition of MAS metamodel elements

In table 2 the basic elements taken from the metamodel are introduced. As the metamodels of INGENIAS are very detailed, only the most important concepts have been defined. For further details, the original documentation of the methodology must be revised [2, 4, 5]

Inception Phase

Metamodels are the key issue in MAS development, according to INGENIAS, because they are considered as the MAS specification language. But these models must integrate with the activities done to obtain them in a Software Engineering Process. This integration will be addressed in this section and the next ones.

Figure 5 shows a general description of this phase of development process. INGENIAS considers that the development initiates from the document describing the problem. This can be considered the initial input of the process. From this document, the Inception phase introduces the several activities, described in figure 5. Regarding the Analysis workflow at this level this activities must be done:

- Generate Use Cases

Concept	Definition	Cross References
Agent	An agent entity is an autonomous entity with identity, purposes and that performs activities to achieve its goals	Autonomous Entity
Application	An application is a wrapper to computational system entities. Computational represents a system having an interface and a concrete behavior	
Autonomous Entity	Root concept that represents an entity with identity and that pursues goals	Goal
Goal	According to the BDI model, a goal is a desired state that an agent wants to reach. In planning, a goal is represented by a world state. Here a goal is an entity by itself, however it can be related with a representation of the world state using satisfaction relationships with tasks. This relationships contains references to descriptions of mental states of agents, so they refer to the image of the world that agent have	Agent
Interaction	Interaction represents an exchange between two or more agents or roles. There can be only one initiator and at least one collaborator. An interaction also details the goal that pursues. This goal should be related with the goals of the participants.	Agent, Role & Goal
MentalState	A mental state represents the information an agent has in a certain moment. A MentalState is an aggregate of mental entities.	Agent
Organization	An organization is a set of agents, roles and resources that get together to achieve one or several goals. Inside an organization there are not other organizations, just groups. You can think of an organization as an enterprise. Internally it is composed by departments that may be restructured without affecting the external image of an enterprise.	Agent
Resource	Resource describes a resource according to TAEMS notation. Opposite to TAEMS, there is no distinction between consumable and non-consumable resources.	
Role	A role is a self-contained grouping of functionalities. When an agent plays a role we want to express that you have to execute tasks associated to a role and participate in the same interactions that role	Agent
Task	Tasks is the encapsulation of actions or non-distributable algorithms. Tasks can use Applications and resources. Tasks generate changes in the mental state of the agent that executes them. Changes consist of: (a) modifying, creating or destroying mental entities; or (b) changes in the perception of the world by acting over applications (applications act over the world producing events, that are perceived by the agent). Though tasks can be also assigned to roles, at the end, it will belong to an agent	Role
Workflow	A workflow is an abstraction to a process that has been automatized using activities and identifying responsibility relationships	

Table 2: Definition of MAS Metamodel Elements

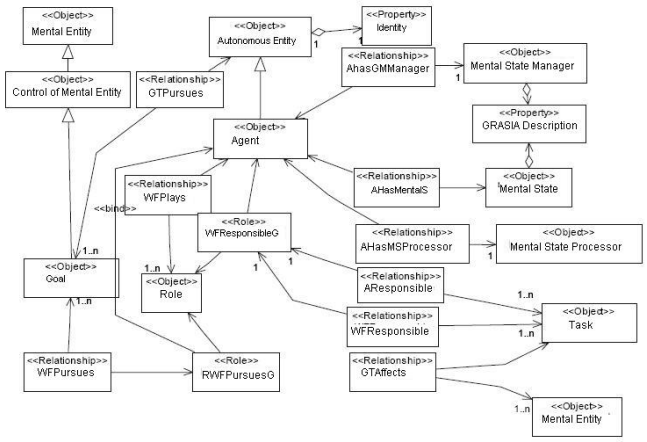

Figure 3: Agent Metamodel proposed by INGENIAS Methodology

- Generate the Environment Model
- Initiate the architecture using the Organization Model

In what respects to Design only the construction of a rapid prototype must be addressed.

All these activities and the tasks associated to each of them are shown in Figure 6. From this figure, we can identify the different tasks proposed by INGENIAS for Inception and the workproducts produced. Moreover, the roles responsible of each task as well as the kind of responsibility they assume are also shown.

Process roles

INGENIAS methodology makes no explicit reference to the roles implied in the development. Nevertheless, and considering the activities to be done and the level of abstraction of these activities two roles are thought to be implied in the process: the System Analyst and the Designer.

The System Analyst is responsible or performs the most part of the activities proposed in this phase. In particular, he will:

- Identify the Use Cases and construct and refine the Use Cases Diagram. From the initial description of the problem to solve, the analyst must obtain the use cases that will guide after the creation of the Interaction Model.
- Define the Environment Model, showing the interaction of the system with its environment. This will imply to: identify applications (in INGENIAS, all the software and hardware that interact with the system and can't be

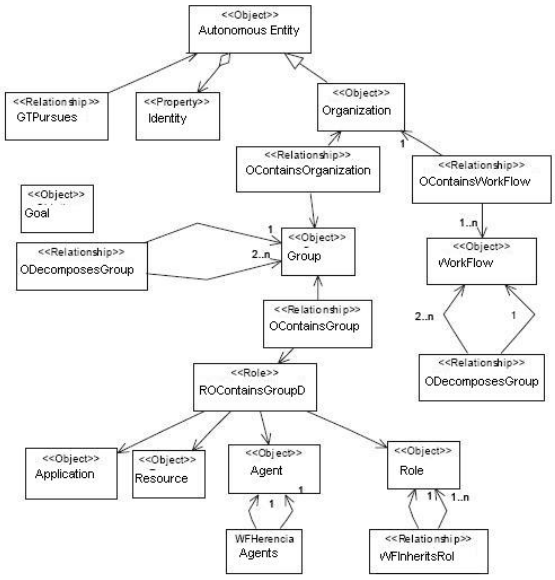

Figure 4: Organization Metamodel proposed by INGENIAS Methodology

Figure 5: Activities and Workflows of Inception phase proposed by INGENIAS Methodology

designed as agent will be considered an application); associate operations to particular applications and define agents perception on applications.

- Obtain the Architectural view of the System using the Organization Model. This means to generate a structural definition of the system by identify groups in the organization, generate group members and identify goals.

The second role identified in this phase has been the *Designer*. He must be responsible of generating the prototypes. According to INGENIAS literature, this will be done using a rapid application development tool such as ZEUS, Agent Tool or others.

Activity Details

This section details the activities previously outlined for Inception Phase.

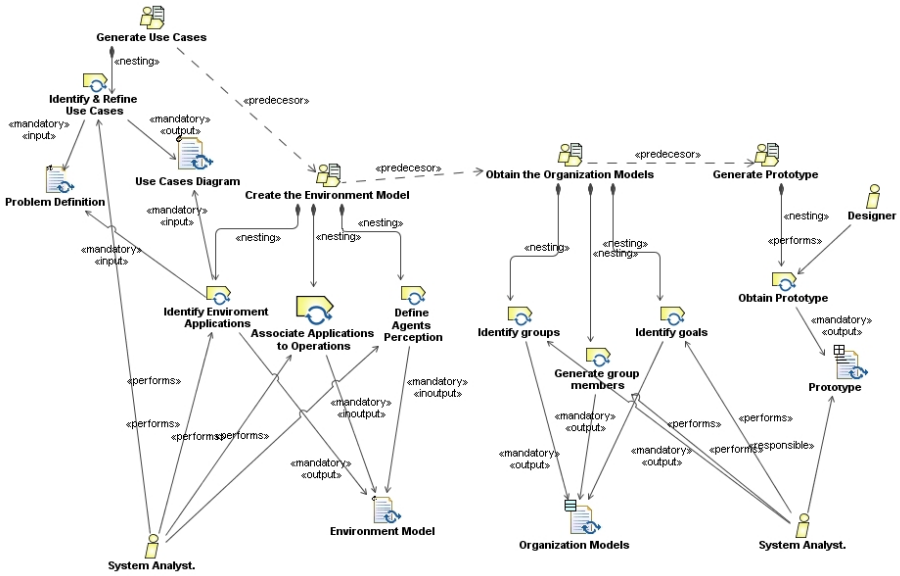

Figure 6: Detailed tasks of Inception activities

Generate Use Cases

The generation and refining of Use Cases has been identified as a unique task. The goal of this task will be to identify the intended functioning of the system. Knowing the functionalities the system must provide, will allow to identify interaction collaborators and initiators and also to discover the nature of such interactions that will affect the type of control applied to the agent: planning, cooperation, contract-net or competition.

Generate the Environment Model

The Environment Model tries to show the elements that constitute the environment of the system, and in consequence, that the agents have to perceive. The elements defined in this model are of three basic kinds: agents, resources and applications.

Figure 7 shows the task that must be accomplished for obtaining an Environment Model of the system to construct. These task are further explained in Table 3.

Initiate the architecture

One of the key activities in Inception Phase is to start the definition of system architecture. This is done by constructing the Organization Model, which reflects mainly the system's workflows.

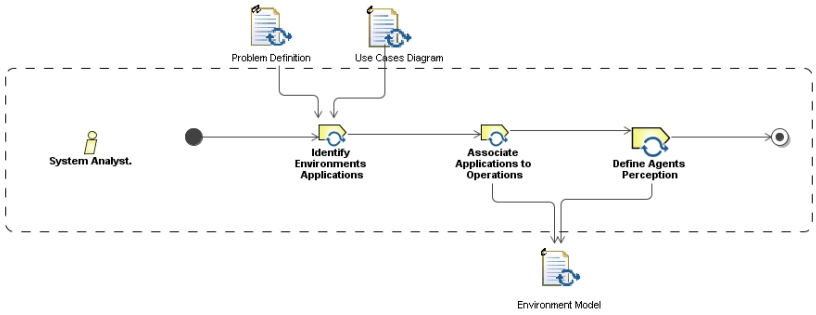

Figure 7: Obtaining the Environment Model in the Inception Phase of INGENIAS

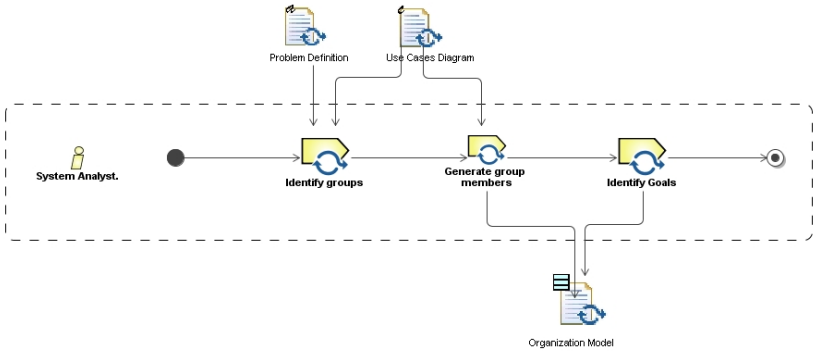

Figure 8: Obtaining the Organization model in the Inception Phase of INGENIAS

In Figure 8 the basic tasks related with the procurement of Organization Model in Inception activity are shown. These activities try to obtain an organizational view of the system, attending its structural, functional and social aspects. The detailed definition of tasks are addressed in Table 4.

Construction of a prototype

The generation of a prototype is a unique and simple task. As said previously the prototype will be generated using a RAD tool.

Work Products

The Inception phase produces as result four basic work-products: a Use Cases diagram, an Environment Model, one or more Organization Models and a prototype of the system to be built.

Activity	Task	Description	Roles Involved
Generate the Environment Model	Identify Environment Applications	All the software and hardware that interact with the system and that can not be designed following an agent oriented approach will be considered an application	System Analyst
Generate the Environment Model	Associate Applications and Operations	Operations are associated to the applications defined by requirements. These operations have a signature, a precondition and a post-condition. The identification of operations is an conventional engineering task.	System Analyst
Generate the Environment Model	Define Agents Perception	The main aim of this task is to define agents perception on environment applications, at this moment of process it is enough to relate agents and applications	System Analyst

Table 3: Task of Activity Generate the Environment Model of Inception Phase of INGENIAS

Activity	Task	Description	Roles Involved
Obtain the Organization Model	Identify groups	The groups in the system must be identified. In this way the participants in a particular work flow will be organized.	System Analyst
Obtain the Organization Model	Generate group members	Members (agents, roles, resources and applications) are assigned to groups creating the corresponding relationships. If needed, the groups can be decomposed in order to reduce complexity.	System Analyst
Obtain the Organization Model	Identify groups	The organization has a set of goals that must justify collaboration between agents. The goals identified in this task will after be assigned to individual agents or roles in the Task and Goals Model.	System Analyst

Table 4: Task of Activity Initiate Architecture of Inception Phase of INGENIAS

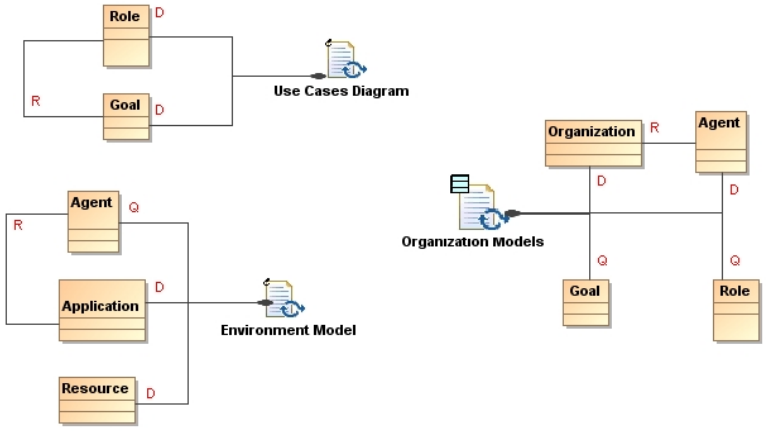

Figure 9: Structure of Inception Workproducts

The Inception phase produces as result four basic work-products: a Use Cases diagram, an Environment Model, one or more Organization Models and a prototype of the system to be built. The relationships among the models and the the metamodel elements are shown in Figure 9. Organization model, for instance, defines the organization metamodel element and the agents and uses the roles and goals previously defined. In this particular case, organization concept includes also the groups within the organization (see organization definition in table 2).

Elaboration Phase

These sections are currently under definition ...

Process roles

Activity Details

Work Products

Workproduct dependencies

Figure 10 introduces a global view of INGENIAS workproducts, as well as their dependencies. As shown in the Figure, Agent Model depends on Organization and Environment Models, while the Interaction Model shows dependencies from Agent and Task/Goal Models among others.

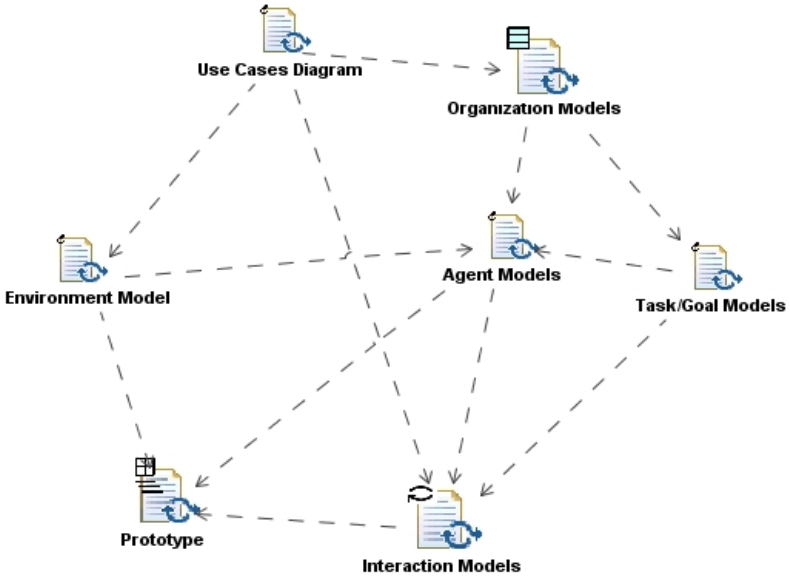

Figure 10: Dependences among Ingenias Workproducts

Bibliography

[1] C. Atkinson and T. Kuhne. Model-driven development: a metamodeling foundation. *Software, IEEE*, 20(5):36–41, 2003.

[2] Jorge Gómez-Sanz. *Modelado de Sistemas Multi-agente*. PhD thesis, Universidad Complutense de Madrid. Facultad de Informática, 2002.

[3] Jorge Gómez-Sanz. Ingenias Agent Framework. Development Guide V. 1.0. Technical report, Universidad Complutense de Madrid., 2008.

[4] Jorge J. Gómez-Sanz and Juan Pavón. Meta-modelling in agent oriented software engineering. In *Advances in Artificial Intelligence - IBERAMIA 2002, 8th Ibero-American Conference on AI, Seville, Spain, November 12-15, 2002, Proceedings*, volume 2527 of *Lecture Notes in Computer Science*, pages 606–615, 2002.

[5] Grupo de Investigación en Agentes Software: Ingeniería y Aplicaciones. INGENIAS Section. `http://grasia.fdi.ucm.es/main/?q=es/node/61`, 2009.

[6] Rational Software. Rational Unified Process: White Paper, 1998.