COMPUTING PLATFORM ARCHITECTURE FOR MANAGEMENT OF RENEWABLE RESOURCES USING MULTI-AGENTS SYSTEM BASED ON THE CASE BASED REASONING APPROACH

[1]MOHAMED KOUISSI, [2]EL MOKHTAR EN-NAIMI, [3]NIHAD EL GHOUCH, [4]ABDELHAMID ZOUHAIR

[1,2,3,4]University of Abdelmalek Essaâdi (UAE), Faculty of Sciences and Technologies of Tangier, Computer Science Department, LIST laboratory, Morocco

E-mail: [1]mohamed.kouissi@gmail.com, [2]en-naimi@uae.ac.ma, [3]nihad_elghouch@hotmail.com, [4]zouhair07@gmail.com

ABSTRACT

Nowadays, the use of renewable resources has seen a remarkable increase due to economic growth, urbanization, technological developments and the difficulties related to energy availability such as gas, coal, oil, etc. This leads to high energy consumption and overexploitation of these resources. Several studies and research have been carried out to find solutions for the management of these resources, most of these studies do not propose intelligent and dynamic management in real-time of renewable resource consumption. In this article, we propose an architecture of a multi-agent system combined with Case Based Reasoning approach. This system aims to implement a cloud platform to create and simulate renewable resource management models, through a reasoning cycle gathering a set of agents collaborating with each other, to help decision makers anticipate and predict an environmental risk based on models already resolved. We present a first implementation of the architecture of our Multi-Agent System using Jade platform (Java Agent DEvelopment) and the web service integration gateway (WSIG) for a remote invocation of our Agents to expose services provided by them as web services.

Keywords: *Case Based Reasoning (CBR), Multi Agents Systems (MAS), Simulation, Renewable Resources, Web Services Integration Gateway (WSIG), Java Agent DEvelopment Framework (JADE)*

1. INTRODUCTION

Recently, the demand for energy use has tripled and the energy consumption has been rising due to population growth and improved living standards [1], which creates huge problem of energy availability and sustainable development. Also, with the depletion of non-renewable resources, critical situations that affect production and sustainable development at regional and world level come up. The renewable resources seem to be a serious alternative as a reliable source of energy. Consequently, the overexploitation of these resources will lead to its irreversible disappearance. Indeed, worldwide governments are working to increase the share of renewable energy sources, moving toward to meet our energy needs, reduce the energy consumption, and accordingly reduce environmental pollution as presented in [2].

The main challenge is to decrease the energy consumption to preserve natural resources around the world and protect the ecosystem from further deterioration and destruction, by proposing intelligent and dynamic management of energy in order to offer real-time support and monitoring for future use.

However, most researches don’t propose shared and distributed platform to model and simulate models in a smart way using the artificial intelligent paradigm and taking into account the interactions between different actors regarding renewable resources field based on past experiences.

In order to deal with this issue, we propose a smart computing platform architecture for intelligent management of renewable resources. This architecture presents a new hybrid approach Multi-Agent System (MAS) [3] and Case Based Reasoning (CBR), to design and develop a decision aid system.

Our approach is based on past experiences which allows to follow and manage renewable resources [4] in real time to predict and anticipate future problems.

The rest of this paper is organized as follows: In section two, we present some related works. In the third section, we present the state of the art of which we introduce the main concepts related to our approach. Section four is devoted to present our proposed approach and architecture. In section five, we present the design and modeling of our approach. The sixth section is dedicated to the implementation of our approach using the Web Services Integration Gateway (WSIG) and Java Agent DEvelopment Framework (JADE). In the last section, we present the conclusion and the perspectives of our future work.

2. RELATED WORK

Many studies and researches have been contributed to propose solutions for the management of common renewable resources. The author of [5] presents a new approach of a decision aid system using the multi-agent paradigm. He has proposed a platform to simulate interactions between actors for water management. This approach doesn't integrate the Case Based Reasoning paradigm and can't be applied to support other types of renewable resources. The author of [6] has proposed a model to assist on implementing green practice for the purpose of sustainability attainment in institutions and universities. This approach uses the Artificial intelligence and Case Based Reasoning techniques to support institutions acquire Geen practice in universities to decrease the energy consumption and anticipate environmental issues.

Both of the two approaches use multi-agent paradigm or CBR approach to implement the system, therefore none of them use a distributed system which the Agents are located in remote servers to communicate with each other in order to achieve their goals. Also, they don't propose an adaptive solution for modeling and developing decisional aid system to manage and support different type of renewable resources, and also to simulate and reuse models that have been created by experts or decision makers.

3. STATE OF THE ART

3.1 Case Based Reasoning

The Case based Reasoning (CBR) is a problem-solving paradigm that solves a problem based on similar past problems already solved by referring to [7]. The CBR approach has been used to solve creative tasks like recipe generation, poems, music etc. CBR contains a set of cases stored in a case base that constitute the data source of knowledge which is pointed out by [8]. The Case Based Reasoning takes its origins from Schank's memory [9], it was especially interested by relying past experiences in problem-solving and learning processes. Solving a problem using the Case Based Reasoning approach can be done through a typical cycle with a set of steps: Retrieve, Reuse, Revise and Retain which revolve around a domain knowledge base application [10]. The aim of the Retrieve step is to filter and setup the specification of the problem by completing its description related to its domain of knowledge. This gives a more detailed description of the target problem. In the reuse step, the retrieved case is then used to find a set of cases in the cases base that is considered similar to the target case. The search of similar source cases is based on the similarity measure. The solutions of the selected source cases are reused to obtain solutions to the target case. Next, in the revision step, the proposed solution will be treated, analyzed, corrected, refused or accepted by the user. The retain step allows to incorporate what is useful to store and save in the cases base. It gives us the ability to synthesize new knowledges in the cases base. This step also serves to optimize the cases base so that the new cases are reused later. These steps are more detailed in [11, 12].

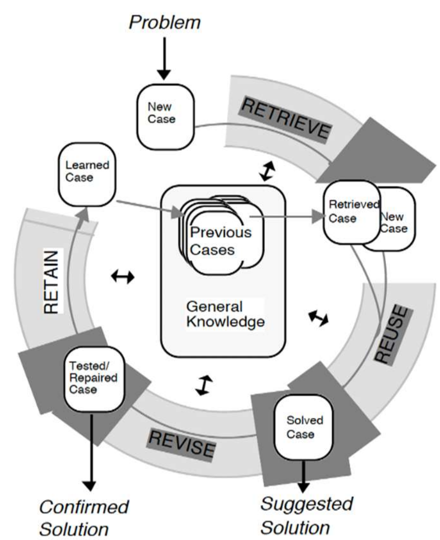


Figure 1. CBR cycle [7]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.2 Multi-Agents System (MAS)

A Multi-Agents System is defined as a set of agents, operating in an environment, communicating each other and with the environment. Agents may behave towards each other as collaborators, competitors or strangers [13, 14].

A MAS contains a set of agents that interact with each other to achieve desired goals [15] autonomously and asynchronously with communication protocols and they are able to act and learn from their environment as presented in [16]. By referring to [17], It is composed of:

- ✓ An environment.
- ✓ A set of objects in the environment, which can be created, perceived and modified by agents.
- ✓ A set of agents and the relationships that link them together in the environment.
- ✓ Operations and behaviors for each agent.

3.3 Why the Jade Platform

JADE (Java Agent DEvelopment)[18], fully implemented in Java [19]. It is an open source platform for agent based applications. To implement our approach, we opted for Jade platform, which offers a lot of advantages:

- ✓ JADE is developed in Java, it runs in all operating systems and it is open source.
- ✓ JADE agents communicate based on the Agent Communication Language.
- ✓ JADE uses a middle-ware that complies with the Foundation for Intelligent Physical Agent specifications (FIPA) [20].
- ✓ Jade uses the Artificial Intelligence paradigm.

In addition to that, JADE platform contains the Web Services Integration Gateway (WSIG), which is an add-on component offering an interface between the JSP client and agents using HTTP. See more details in 3.4

3.4 Jade Web Services Integration Gateway

The web service integration gateway (WSIG) [21] is a support for invocation of Jade agent services from web service clients. The main goal of WSIG is to expose services provided by agents in the JADE DF (Directory Facilitator) as web services, it gives developers the ability to make specific functionality requirement then may have.

The WSIG provides support for Web services stack, using the WebService Description Language (WSDL) [22] for service descriptions, Simple Object Access Protocol (SOAP) [23] for message transport and a Universal Description Discovery and Integration (UDDI) [24] repository for publishing Web services using Models. WSIG is composed of two main elements: WSIG Servlet and WSIG Agent.

The WSIG Servlet is responsible for Managing incoming HTTP/SOAP requests, also, preparing the corresponding agent action and passing it to the WSIG Agent. Once the agent action has been served, it will be converted into a SOAP message to be sent back to the client as a response HTTP/SOAP. The WSIG Agent is the mediator or the gateway between the web application client and the Agents. It is responsible for:

- ✓ Forwarding agent actions received from the WSIG Servlet to the corresponding agents actually able to serve them and getting back responses.
- ✓ Receiving notifications concerning agent registration / deregistration by subscribing to the JADE Directory Facilitator.
- ✓ Creating the WebService Description Language related to each agent service registered with the Directory Facilitator.

A conceptual overview of the WSIG architecture is depicted in Figure 2.

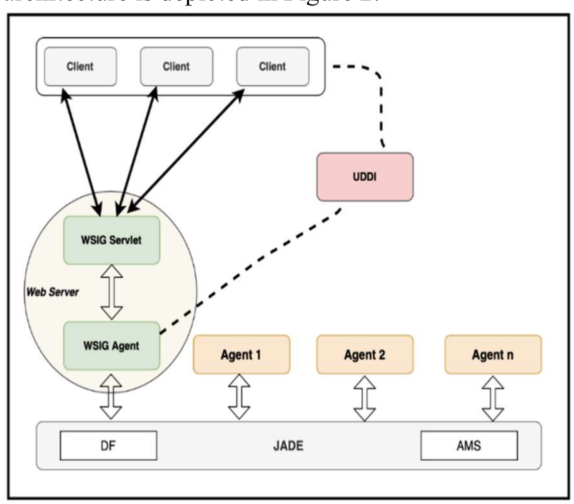


Figure 2. WSIG architecture

Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. OUR PROPOSED APPROACH AND ITS ARCHITECTURE

4.1 Presentation of Our Approach

In this part, we present the multi-Agents system proposed approach for the resolution of renewable resources management problem. A significant part of any decision systems is finding the right models of real system for viewing the problem.

The aim of our approach is to create a multi-agents system cloud platform for managing common renewable resources and which is based on case-based reasoning paradigm. It's allows decision makers to create, simulate and reuse the right multi-agent models from already solved ones.

Our approach offers many positive points as the following:

- Smart and dynamic management system to ensure an efficiently use of resources.
- Manage cloud platform for renewable resources using artificial intelligence.
- Create and simulate models.
- A generic platform to support any type of renewable resources.
- Benefit from the advantage of agents as decision making, autonomy, collaboration, communication.
- Benefit from saved past experiences to anticipate an environmental risk.

4.2 Presentation of Our Architecture

To help decision makers in the aid of the right decision to identify and anticipate environmental issues and to benefit from the advantages of agents, we propose an architecture of multi-agents system based on Case Based Reasoning (Figure 3).

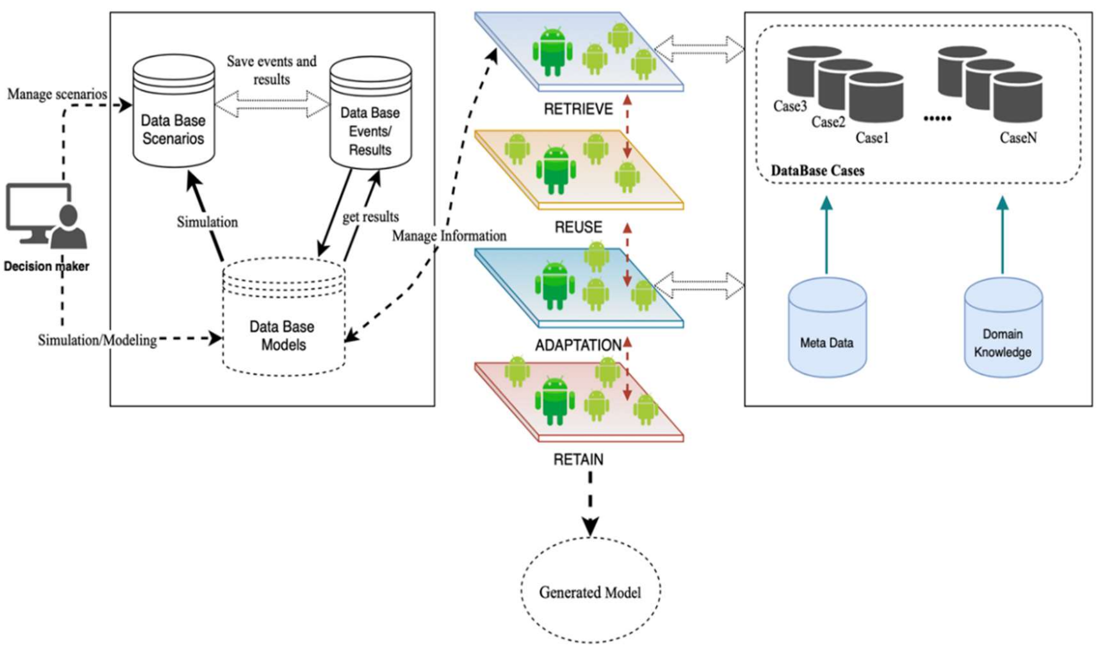


Figure 3. Our Proposed architecture

The general architecture of our system is composed of three parts:

The first part is dedicated to the creation and simulation of the models that will be created from a real system. The system allows the simulation of each model created according to a set of scenarios defined by the decision maker and which are stored in a database. Each generated model will be matched with a case which is identified by a source problem and its solution, their meta data, ontology and finally their domain of knowledge [25, 26, 27]. The results obtained after the simulation will be checked and compared with the data collected in the field.

The second part is the main component of the system that contains the core engine and the environment of the agents. It is composed of four Agent layers (Retrieve, Reuse, Adaptation and Retain): The Retrieve layer agent serves for managing information collected from the simulation and which describing the current state, as well as constructing the specification of the source problem to be solved to find similar cases to the source case. In the reuse layer, the agents are responsible for comparing and finding all the cases judged similar to the source case in the base case. This process is continuous and triggered whenever new data or modifications are made to the target case. The Adaptation layer agents aim to evaluate, reuse and adapt all cases deemed similar by the reuse layer to anticipate and predict solutions of the target case. The agents of the Retain layer are responsible for seeking for resolved cases that are considered similar to the source case and retaining them to enrich current data base cases. Finally, the proposal of the appropriate models associated with the already resolved cases.

The third part is responsible for saving all new cases during a simulation in the database with their metadata, ontology [28] and also their domain of knowledge. The ontology concepts will be developed to measure and match similar cases together.

5. DESIGN AND MODELING

To describe our system, we use the Agent Unified Modeling Language (AUML) detailed in [29] which is based on UML, a graphical modeling language for specifying, visualizing and constructing the artefacts of software systems. Multi-Agent system designers use AUML to design agents, represent their behaviors and their interactions with each other [30].

5.1 Use Case Diagram

The following figure 4 presents the use case diagram of our system detailing all the scenarios performed by the various agents.

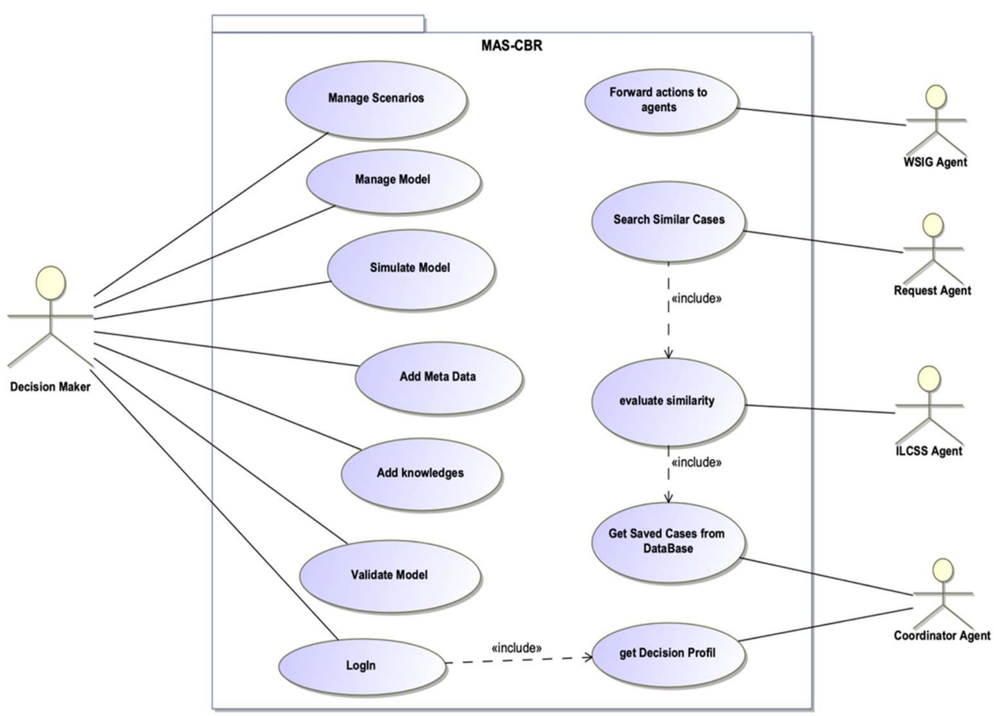


Figure 4. Use Case diagram for our MAS

Journal of Theoretical and Applied Information Technology
31st January 2021. Vol.99. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5.2 AUML Class Diagram

Our AUML class diagram has been modeled [31] with the MagicDraw UML tool [32, 33]. It is composed of two models. In the first model, we focus on presenting agents and the interactions modeling between them in the environment. The second model is the main component of the system, it's responsible for simulating interactions between agents using different cases stored in the database.

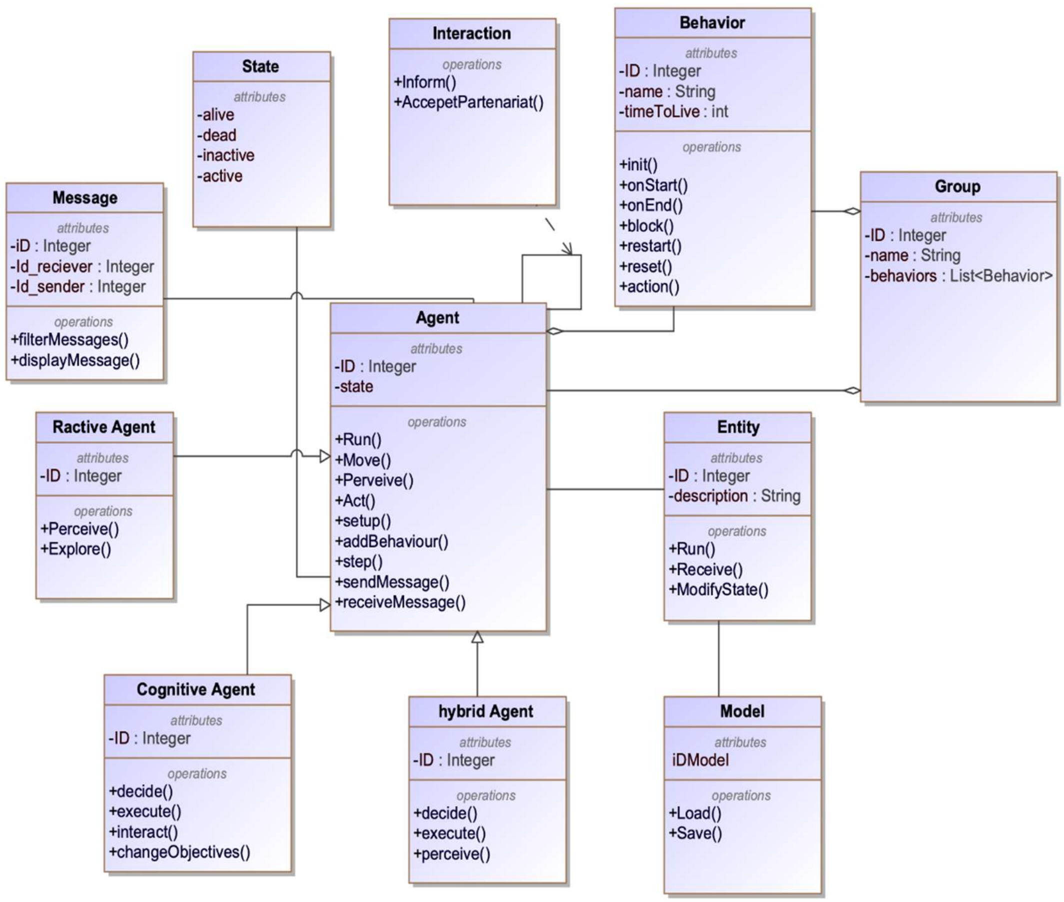


Figure 5. The first model of our MAS class diagram

Below the description of some entities of our class diagram:

- **Agent Class**: The agent class allows the creation of an Agent with defined properties or characteristics. In this class, we have three elements, Attributes, Roles and Perception. Attributes represent all data that an agent have to possess, Roles represent Agent responsibilities, and finally the perception of an agent with its environment and with other agents.
- **Entity Class**: The entity class constitute the spatial cell that contains agent or a set of agents. It contains all information about its environment. This entity can start agents, perceive information and also update their status.
- **Cognitive agent**: This agent can decide to perform actions to make decisions based on past cases.

- **Reactive agent**: This agent extends the Agent class with all its characteristics and properties.
- **Hybrid Agent Class**: This agent extends the Agent class, it illustrates a hybrid agent behavior that is a compromise between cognitive and reactive agents.
- **Message**: This class contains exchanged messages and information between all agents.
- **Behavior**: It's represents tasks and roles that can be executed by the agent.
- **State**: This class represents the agent status, an agent can be alive, dead, active, inactive.
- **Group**: A group is a pack of agent. Each agent may have many behaviors which is located to a specific group.
- **Model**: This class is used to create and store simulated models.

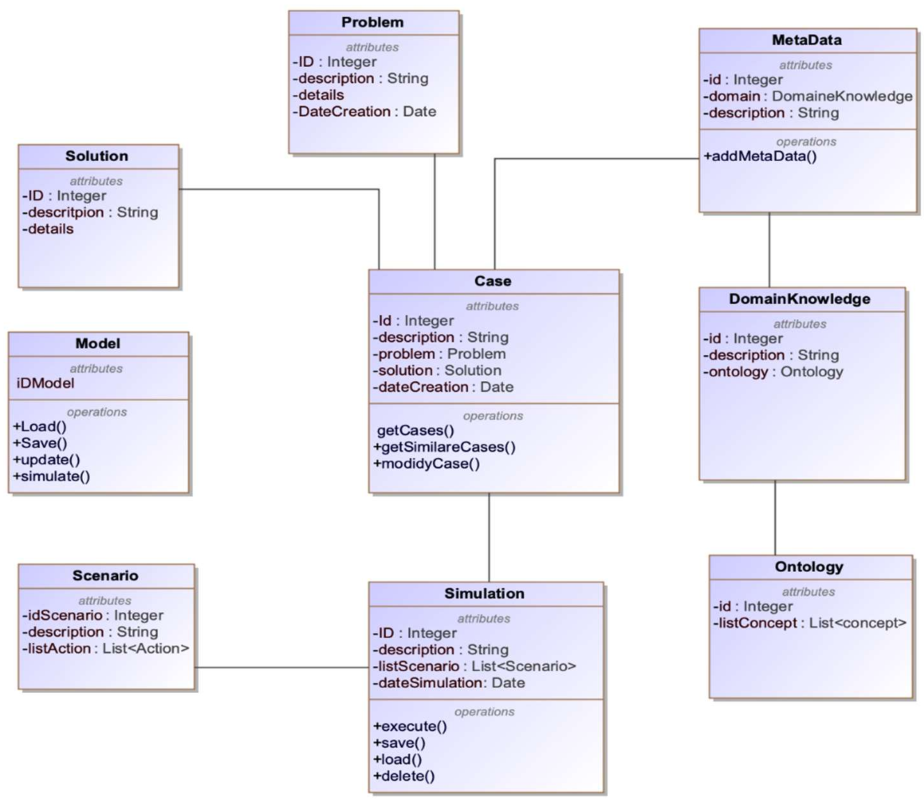


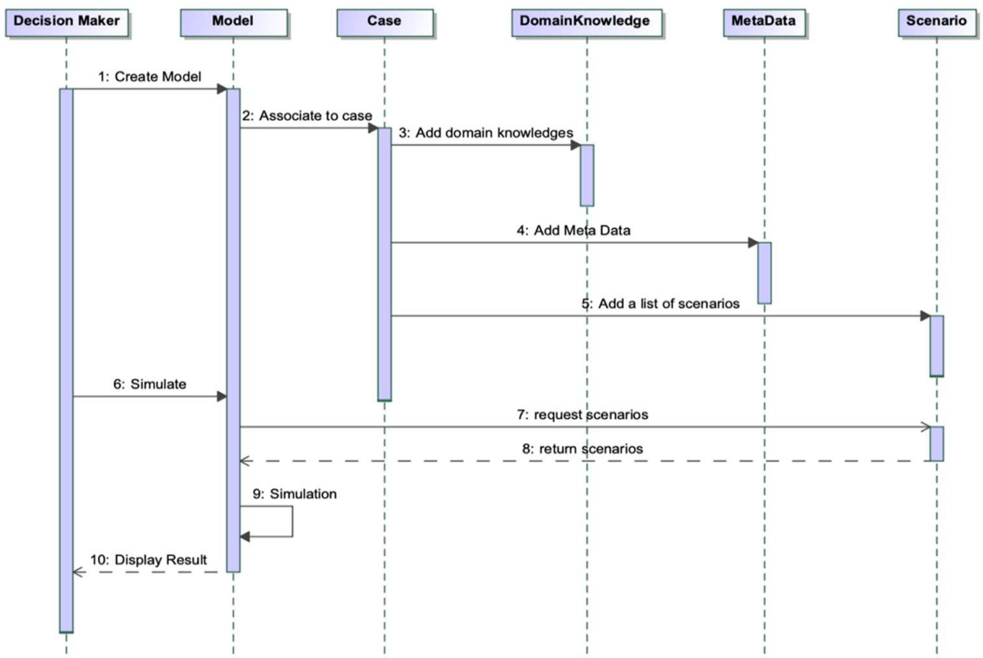
Figure 6. The second model of our MAS class diagram

- **Simulation Class**: The role of this class is to simulate models that have been created by decision makers according a set of scenarios.
- **Case Class**: This class represents a set of stored cases. Each case is linked to a source problem and the solution associated with it.
- **Domain Knowledge Class**: This class serves to store domains knowledge and ontologies related to each case.
- **Meta Data Class**: It is responsible for storing data description of each case according to their domain of knowledge.

- **Ontology**: it's hold the properties of each saved case according to their concept.

5.3 Sequence Diagrams

The sequence diagram shows the dynamic



exchanged messages between agents and system.

Figure 7. Sequence Diagram of the Scenari "Simulate New Model"

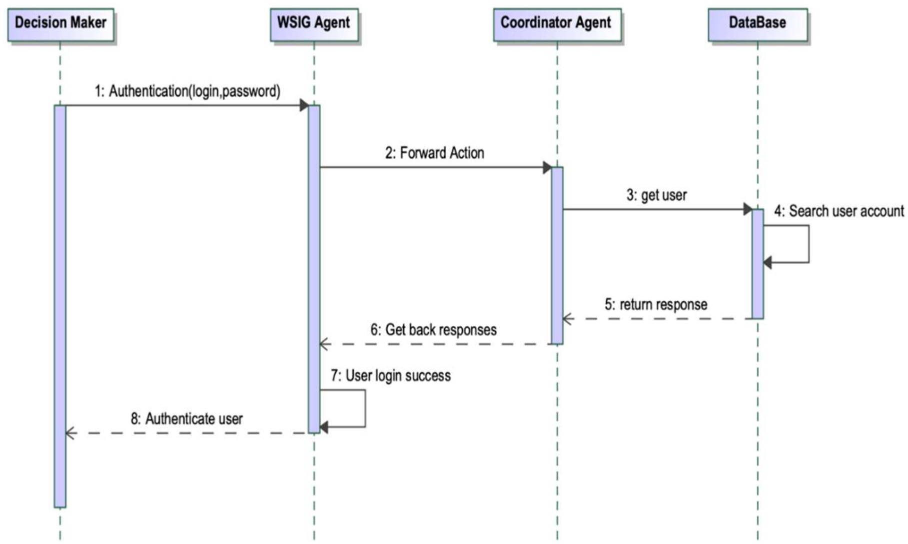


Figure 8. Sequence Diagram of the Scenario "Authenticate Decision Maker"

This figure 8 presents the scenario of the authentication of the decision maker into the system:

- **WSIG Agent:** It Forward the received actions to the corresponding agents able to serve them and getting back responses. The wsig agent is the gateway between the web application client and all Agents in the system.
- **Coordinator agent:** The coordinator agent is responsible for verification of decision maker account in the data base. It is also responsible for managing and synchronizing different requests between agents for a better performance of the system.

6. IMPLEMENTATION OF OUR APPROACH USING WSIG AND JADE PLATFORM

6.1 Technical Architecture

Our proposed technical architecture is composed of web application, remote server, remote data base and Jade platform. The architecture is depicted in Figure 9.

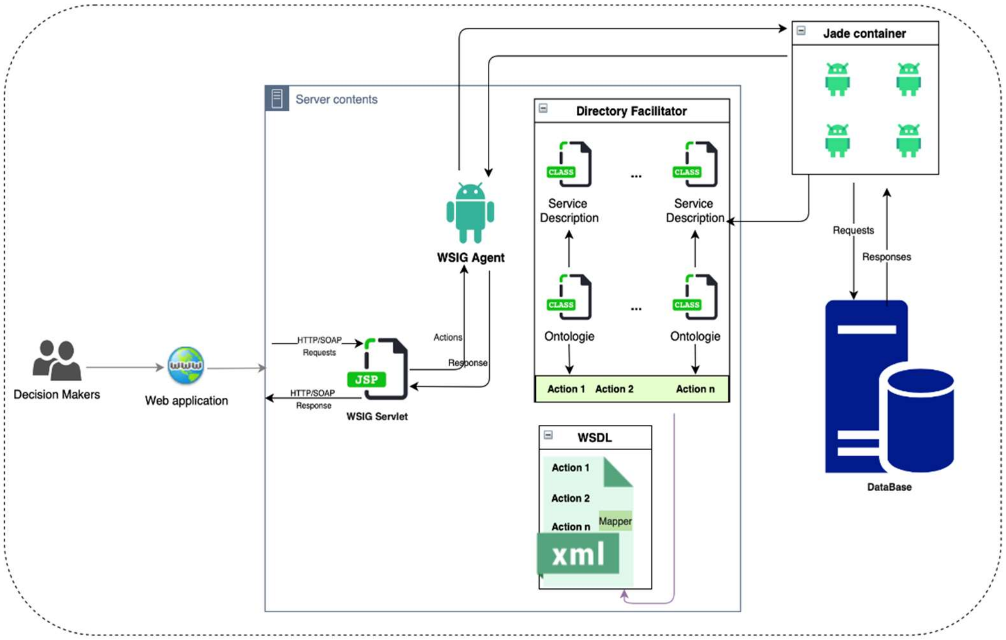


Figure 9. Technical Architecture

The Decision Makers can login to the platform via the web application which is hosted in a remote server. This server contains both the WSIG Servlet and WSIG Agent. The first is responsible for Managing incoming HTTP/SOAP requests, preparing the corresponding agent action and passing it to the WSIG Agent. The WSIG Agent will then forward agent actions received from the WSIG Servlet to the corresponding agents available in Jade container and getting back responses. Each agent hosted in the Jade container is associated with a service description which describes the service provided by the corresponding agent. The service description specifies a set of ontologies and actions in order to access the desired service to execute actions.

6.2 Implementation of the Use Case "Authenticate Decision Maker"

This use case is implemented using Jade platform and the web service integration gateway (WSIG). Firstly, we start Jade platform which is the main container of our agents. Then, we run the WSIG agent which is the mediator between the application client and our Agents, it's also responsible of forwarding agent actions. After that, we deploy the coordinator agent, which is responsible for verifying and fetching decision maker account from the data base. Below, some figures (10-15) that illustrate our result.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

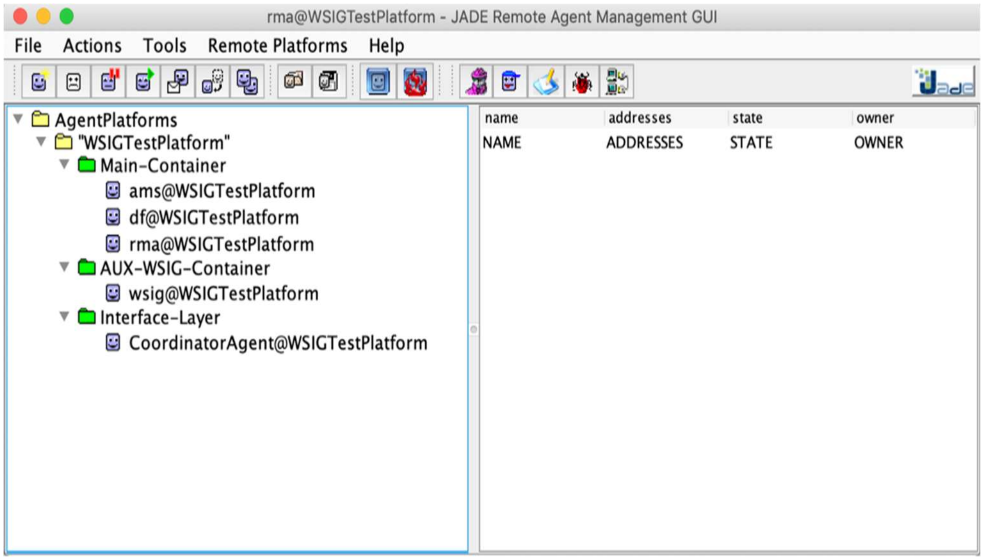


Figure 10. Start Jade Platform with WSIG and the Coordinator agent.

Figure 11. Page view to connect the decision maker

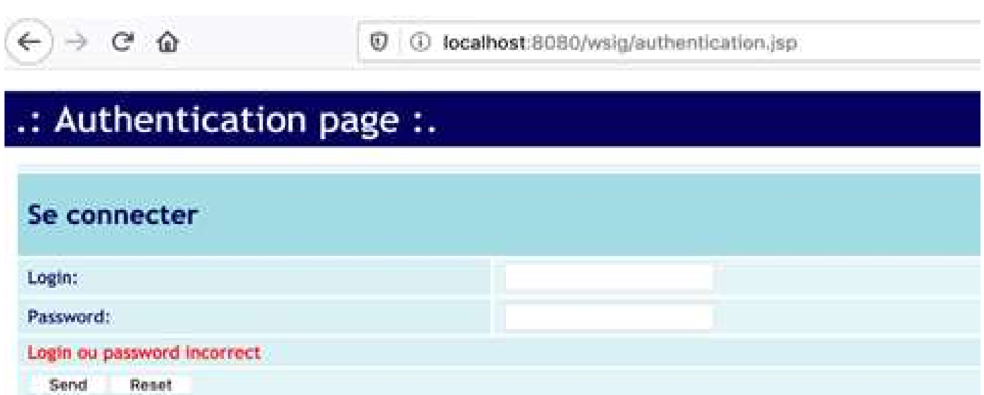


Figure 12. Connect with wrong login or password

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 13. Correct login and password

The figure below shows the communication between the WSIG Agent and the Coordinator Agent to connect the decision maker according to his login and password into the platform.

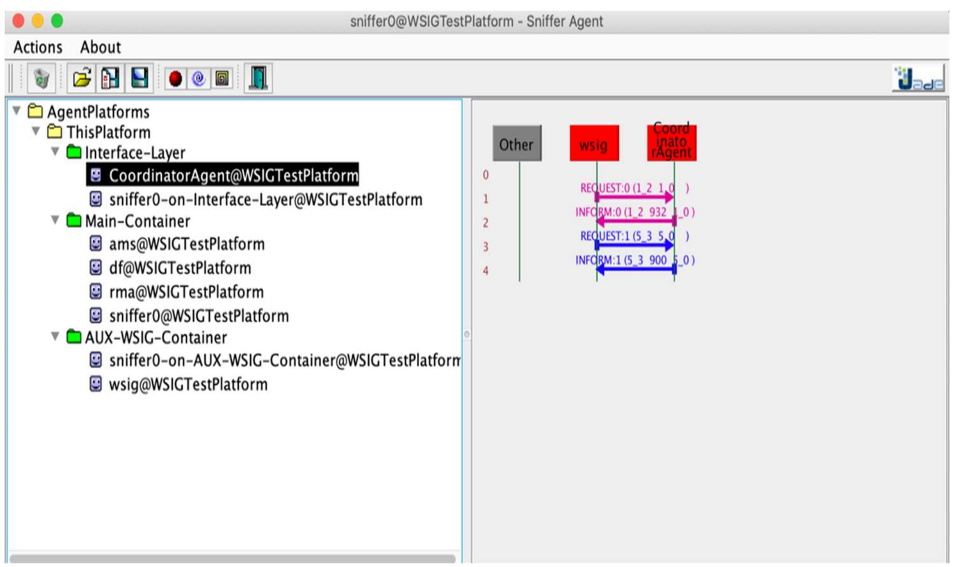


Figure 14. Communication between agents to authenticate the decision maker

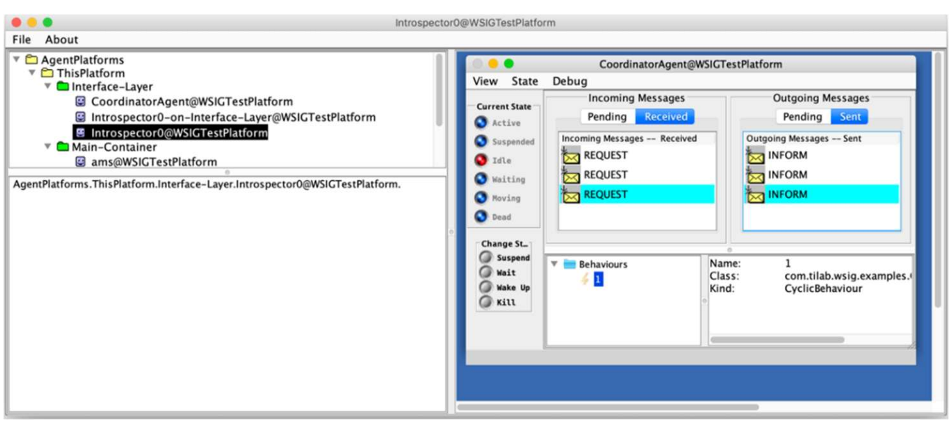


Figure 15. Transmitted and received messages for Coordinator Agent

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

7. CONCLUSION, LIMITATIONS AND PERSPECTIVES

In this paper, we have proposed an architecture of a new hybrid approach of MAS and CBR to manage in real time common renewable resources. The integration of agent technology in the CBR cycle will allow decision-makers to follow in real time the consumption of renewable resources to make the right decision to predict environmental issues, taking into account the advantages offered by the multi agent system paradigm. It allows also to create and simulate models of renewable resources. We started implementing this architecture under Jade platform and the web service integration gateway (WSIG) to expose services provided by our agents as web services. This coupled architecture with multi agent systems, the web service integration gateway and the case based reasoning will be useful to create a cloud platform for managing common renewable resources.

However, some limitations should be noted. First, our Multi-agent CBR system require several numbers of cases to efficiently learn and act. Also, our system requires a lot of dataset of different renewable resources to test our models, which are not all accessible. Another limitation is that the validation of models has to be done by domain experts, so if no domain experts are available there will be no validation of models.

Our future work concerns the implementation of other parts of our architecture, we would like also to test our model in real experiment.

REFERENCES:

[1] Riyad A. M, Ahmed M. S, and Khan R. L, "An adaptive distributed intrusion detection system architecture using multi agents". *International Journal of Electrical & Computer Engineering* (2088-8708), 9.

[2] Ninet Mohamed Ahmed, Hanaa Mohamed Farghally, and Faten Hosney Fahmy, "Optimal Sizing and Economical Analysis of PV-Wind Hybrid Power System for Water Irrigation using Genetic Algorithm", *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 3, pp. 1797-1814.

[3] Jacques Ferber, "Multi-Agent systems: general view", *Technical and computer science*, vol.16 n°8, pp. 979-1012.

[4] BOUSQUET François, "Modélisation d'accompagnement. Simulations multi-agents et gestion des ressources naturelles et renouvelables". *Thèse de doctorat. UCLB*, 2001

[5] Dominique URBANI, "Elaboration of a hybrid approach MAS-GIS for the definition of a decision aid system, application to water management", 2006.

[6] Bokolo ANTHONY JNR, Mazlina ABDUL MAJID, and Awanis ROMLI, "Hybrid Multi-Agents and Case Based Reasoning for Aiding Green Practice in Institutions of Higher Learning.", *Technical Gazette* 26.1 (2019): 13-21.

[7] Aamodt, Agnar, and Enric Plaza. "Case-based reasoning: Foundational issues, methodological variations, and system approaches.", *AI communications* 7.1 (1994): 39-59.

[8] Jnr, Bokolo Anthony, Mazlina Abdul Majid, and Awanis Romli. "Application of intelligent agents and case based reasoning techniques for green software development.", *TTEM* 12.1 (2017): 30-43.

[9] Schank, Roger C. "Dynamic memory: A theory of reminding and learning in computers and people", cambridge university press, 1983.

[10] Bouchon-Meunier, Bernadette, Maria Rifqi, and Sylvie Bothorel. "Towards general measures of comparison of objects.", *Fuzzy sets and systems* 84.2 (1996): 143-153.

[11] Nihad, El Ghouch, et al, "Hybrid approach of the fuzzy C-means and the K-nearest neighbors methods during the retrieve phase of dynamic case based reasoning for personalized follow-up of learners in real time.", *International Journal of Electrical & Computer Engineering* (2088-8708) 9 (2019).

[12] ZOUHAIR Abdelhamid, "Multi agent dynamic case based reasoning: Application to an Intelligent Tutor System", *PhD thesis, in Cotutelle between the Faculty of Sciences and Technologies of Tangier, Morocco and the University of Le Havre, France*, (2014).

[13] Chaib-Draa, Brahim, Imed Jarras, and Bernard Moulin, "Systèmes multi-agents: principes généraux et applications.", *Edition Hermès* 242 (2001): 1030-1044.

[14] WANG, Meijia, LI, Qingshan, LIN, Yishuai, et al, "A Personalized Metasearch Engine Based on Multi-agent System (P)", *The International Arab Journal of Information Technology*, vol. 16, no. 6, In: SEKE. 2018. p. 707-706.

[15] REHMAN, Shafiq Ur, NADEEM, Aamer, and SINDHU, "Towards Automated Testing of Multi-Agent Systems Using Prometheus Design Models.", *The International Arab*

Journal of Information Technology. 2016, vol. 16, no. 1

[16] Vafadar, Shiva, and Ahmad Barfourosh, "Towards Intelligence Engineering in Agent-Based Systems*." International Arab Journal of Information Technology (IAJIT)* 12.1 (2015).

[17] Weyns, Danny, et al. "Environments for multiagent systems state-of-the-art and research challenges.", *International Workshop on Environments for Multi-Agent Systems*. Springer, Berlin, Heidelberg, 2004.

[18] Jade Framework: http://jade.tilab.com

[19] Marir, Toufik, et al. "QM4MAS: a quality model for multi-agent systems.", *International Journal of Computer Applications in Technology* 54.4 (2016): 297-310.

[20] Odell J, "FIPA IEEE Foundation for Intelligent Physical Agents", (2007)

[21] WSIG https://jade.tilab.com/doc/tutorials/WSIG_Guide.pdf

[22] Chinnici, Roberto, et al. "Web services description language (wsdl) version 2.0*.", W3C Working Draft* 24 (2003).

[23] Gudgin, M., et al. "SOAP version 1.2 part 1: Messaging framework, 2nd edn. W3C recommendation, W3C, retrieved September 24, 2012." (2007).

[24] Graham, Steve, et al. *Building Web services with Java: making sense of XML, SOAP, WSDL, and UDDI*. SAMS publishing, 2004.

[25] Laallam, Fatima Zohra, Mohammed Lamine Kherfi, and Sidi Mohamed Benslimane. "A survey on the complementarity between database and ontologies: principles and research areas*.", International journal of computer applications in technology* 49.2 (2014): 166-187.

[26] Prabowo, Harjanto, et al. "Utilizing Mobile-based Deep Learning Model for Managing Video in Knowledge Management System*.", International Journal of Interactive Mobile Technologies (iJIM)* 12.6 (2018): 62-73.

[27] Saba, Djamel, et al. "Development of an ontology-based generic optimisation tool for the design of hybrid energy systems*.", International Journal of Computer Applications in Technology* 55.3 (2017): 232-243.

[28] Lyazidi, Achraf, and Salma Mouline, "A metamodel-driven definition and implementation of ONDAR: a home automation ontology.", *International Journal of Computer Applications in Technology* 58.3 (2018): 198-210.

[29] Huget M.P, et al. "Interaction Diagrams. Working Docuemnts. AUML Official Website" (2003), http://www.auml.org/

[30] B. Bauer, J. P. Müller, and J. Odell, "An extension of UML by protocols for multiagent interaction", *In International Conference on MultiAgent Systems* (2000) *(ICMAS'00)*, pages 207–214, Boston, Massachussetts.

[31] WAISHIANG, Cheah, HONG, Puah Chin, HALIN, Alfian Abdul, et al., "Agent-Oriented Requirement Engineering for Mobile Application Development", *International Journal of Interactive Mobile Technologies* (iJIM) (2017), vol. 11, no 6, p. 32-48.

[32] Bouquet, Fabrice, et al. "Description Formalisms in Agent Models.", *Agent-based Spatial Simulation with Netlogo. Elsevier*, 2015. 29-73.

[33] MagicDraw https://www.nomagic.com/services/training.