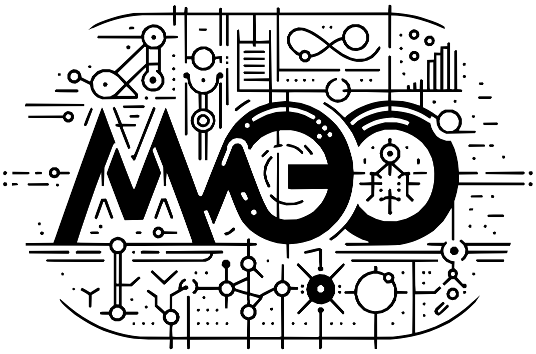
Bogdan Okreša Đurić

Developing a Framework for Agent Gamification Based on Ontologies

MAGO

Part 1: MAGO-Ag Ontology and Framework



UNIZG Faculty of Organization and Informatics
Artificial Intelligence Laboratory

This document contains some of the results and deliverables of the project ID **MOBODL-2023-08-5618** funded by the European Union and the Croatian Science Foundation.

Funded by
the European Union
NextGenerationEU

HrZZ
Croatian Science
Foundation

Authors: Bogdan Okreša Đurić.

Editor: Bogdan Okreša Đurić, dokresa [at] foi.unizg.hr.

Cover graphic generated by ChatGPT and DALL·E.

UNIZG FACULTY OF ORGANIZATION AND INFORMATICS
ARTIFICIAL INTELLIGENCE LABORATORY

AI.FOI.HR

Version: October 2024

In case you find any errors, do not hesitate to report them to dokresa [at] foi.unizg.hr.

Contents

I Phase 1: The Ontology 1

1 Specification 5

1.1 Specification Document 6

2 Knowledge Acquisition 9

2.1 Glossary of Terms 10

3 Conceptualisation 27

3.1 Applicable Concepts 27

3.2 Data Dictionary 30

4 Formalisation 37

5 Integration 39

6 Implementation 41

7 Evaluation 43

II Phase 2: The Framework 45

8 Framework Design, Description, and Implementation 47

8.1 Framework Design and Description 47

8.2 Framework Implementation 49

9 A Case Study: The RecipeWorld 57

9.1 The Ontology 58

III Appendices 65

Part I

Phase 1: The Ontology

Introduction

The development and engineering processes of the ontology described here follow the steps defined by the METHONTOLOGY ontology engineering methodology [1], [2]. This is the methodology of choice because it is very well defined, using clearly described steps to engineer the desired ontology. This and a dozen other ontology engineering methodologies were analysed more thoroughly in preparation for the author's doctoral thesis [2].

This chosen ontology engineering methodology is described as having a set number of steps, each described in detail by the authors of the methodology. In addition, steps related to the entire life cycle of an ontology are identified. An ontology life cycle is described by Iqbal, Murad, Mustapha *et al.* as '[...][a] set of stages through which the ontology moves during its life.' [3, p. 2997]

see figure 1

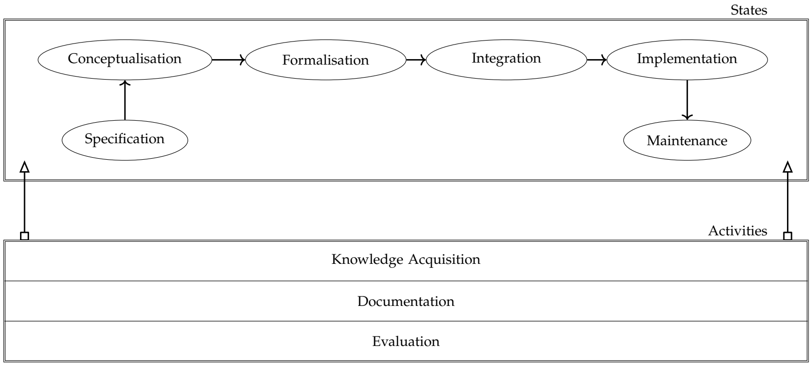


Figure 1: Basic steps of METHONTOLOGY ontology engineering methodology, reproduced from [2], adapted from [1]

The rest of this document is structured according to the defined METHONTOLOGY steps, representing the documentation process. First, the ontology specification document is described and given, followed by the description of knowledge acquisition as an ongoing process. Conceptualisation originates from the defined specification and is the input to formalisation. Once the ontology is formalised, it can be integrated with other ontologies of similar domains. Finally, the ontology is implemented and maintained. The ontology is evaluated in phase P1.2 of this part of this research. Each methodology step is briefly described at the beginning of the related chapter.

see chapter 1
see chapter 2
see chapter 3
see chapter 4
see chapter 5
see chapter 6
see chapter 7

1
Specification

> 'The goal of the specification phase is to produce an informal, semi-formal or formal ontology specification document written in natural language, using a set of intermediate representations or using competency questions, respectively.' [1, p. 2]

The specification state of engineering an ontology is the initial one. It contains the initial ontology description and the expectations for the finalised model. The result of this state is an initial ontology specification document that is not necessarily a formalised document or a document containing formal expressions. According to Fernández-López, Gómez-Pérez and Juristo, such a document should provide the answers to at least the following three questions [1]:

- What is the intended purpose of the ontology?
- How formal is the ontology expected to be implemented?
- What are the planned scope and granularity of the ontology?

The suggested approach to identify concepts that should be included in the scope of the ontology, that is, concepts that are planned to be modelled as a part of the current ontology, is a middle-out approach [1], [4]. This way, instead of using a bottom-up or a top-down approach, the author immediately identifies the key concepts and provides additional concepts by applying specialisation or generalisation as necessary and seen fit.

In the context of formality, Fernández-López, Gómez-Pérez and Juristo refer to Uschold and Gruninger when stating the four degrees of formality [4]:

highly informal is an ontology that is expressed using natural language with loosely constrained concept intention;

semi-informal ontology is still expressed using natural language but in a more structured and restricted way, using more constraints, which results in an increase in clarity and reduced ambiguity;

semi-formal degree of formality expects the ontology to be expressed using an '['[4]]artificial formally defined language;

rigorously formal ontology comprises concepts constrained by formal semantics and theorems, enriched by further proof of soundness and completeness.

To be comparable to the finalised ontology, or any stage of the ontology while it is being engineered, developed or implemented, when finished, the ontology specification document should adhere to the following [1]:

- the document should be concise, and the chosen concepts relevant to the topic and the planned purpose of the ontology, featuring no duplicate or unrelated concepts;
- the set of identified concepts should be partially complete when the chosen domain is considered, taking into account the selected level of granularity and the breadth of intention of each of the chosen concepts, since total completeness is next to impossible to achieve as new concepts can always be added to an existing specific-domain-related ontology;
- the document should be consistent in all its parts, including, but not limited to, a list of consistent concepts applicable to the chosen domain and scope of the ontology adhering to the selected level of formality and the general purpose of the ontology.

1.1 Specification Document

domain

MAGO-Ag ontology comprises concepts related to the domain of multiagent systems (MASs) of the general area of artificial intelligence (AI). A MAS is a system consisting of a set of agents located in an environment where they communicate. Fundamentally, every agent has sensors to perceive its environment and actuators to act upon it [5, p. 54]. In general, this environment is not static. In particular, the domain of the MAGO-Ag ontology are MASs, and more specifically, automatic instantiating of agents according to the data within the ontology, i.e. describing and instantiating a MAS.

purpose

The purpose of the MAGO-Ag ontology is to store data as pieces of knowledge that can be used to describe a MAS. The designed MAGO-Ag framework can then utilise such data to automatically instantiate modelled agents of the system described within the ontology. Specifically, the main purpose of the MAGO-Ag ontology is to provide concepts to enable the described process – modelling a MAS and instantiating the accompanying agents.

intended use

This ontology is intended to be used closely with the MAGO-Ag framework that is going to be developed throughout this project. Such a framework is expected to use this ontology as input, providing the necessary description of a MAS as an input. The user will be provided with an instance of the modelled system based on the input provided. Therefore, the ontology's intended purpose is to provide the concepts for describing, i.e. modelling, a MAS in a way ready to be translated into implementation. Furthermore, the modelled MAS is planned to adhere to the definition of the concept of a intelligent virtual environment (IVE).

scenario of use

A brief example of using the MAGO-Ag ontology with the accompanying framework is presented as follows. The system designer can model their system using the available concepts of the ontology, thus describing agents, their goals, roles, available behaviours, communication flows and communication media, artefacts of the environment, if any, etc. The resulting model can then be used as input to the MAGO-Ag framework and translated into a foundation for implementing the modelled system. The result is an implementation blueprint that the system developer is expected to enrich with actionable code customised to the specific implementation needs of the modelled system being implemented. Additional examples are provided below.

👁 see examples 1.1 and 1.2

Example 1.1.

◁ example 1.1. Using the MAGO-Ag ontology to model RecipeWorld

RecipeWorld is '[. . .] an agent-based model that simulates the emergence of a network out of a decentralized autonomous interaction;' [6] that can be evolved into a real-time implementation using, e.g. Python and SPADE [7] agents. The two types of agents within RecipeWorld can be described as service providers and service consumers. These types of agents, their goals, allowed interaction media, and behaviour are modelled using the MAGO-Ag ontology. The finalised model is then fed to the MAGO-Ag framework and translated into an implementation blueprint of the modelled system. Thus, the implementation process is made somewhat easier, especially concerning the semantic consistency of the implemented concepts.

SPADE Documentation

Example 1.2.

◁ example 1.2. A farming simulator combined with the MAGO-Ag ontology

The farming simulator referenced here is a system developed by a research team of Valencian Research Institute for Artificial Intelligence (VRAIN) of Universitat Politècnica de València (UPV). The primary purpose of the system is to implement a virtual model as a digital twin of a crop field and a custom number of tractors tasked with specific tasks related to various agricultural activities. Agents playing these digital twins and their environment are at the moment of writing this document described using a set of configuration files. Some of the parameters of these files can be converted to and modelled using the concepts planned to be part of the MAGO-Ag ontology. Thus, the system can be modelled using the concepts provided by the ontology, which ensures added semantic value to the modelled and implemented system, and generated into an implementation blueprint of the modelled system.

end users

Based on the above examples and the given scenario of use, the intended end users of the MAGO-Ag ontology are developers, especially developers of MASs. The level of competence in terms of programming languages and frameworks for implementing MASs is irrelevant since the modelled system's description can never be comprehensive enough to provide a one-on-one replica. Developers of systems classifiable as digital twins are also some of the intended users of this ontology.

degree of formality

In order to achieve the described, the MAGO-Ag ontology will be expressed using a high degree of formality, namely classified as rigorously formal by [4]. Such a formal structure, along with its many constraints, ensures that the ontology is clear, unambiguous, and easy to use. Furthermore, it is easier to use it to extend other ontologies or be extended by other ontologies. Furthermore, a highly formal expression of an ontology makes it easier to use it in conjunction with other digital systems.

intended users

The intended usersof the MAGO-Ag ontology are system designers and modellers who know how to work with an ontology and aim at modelling a system that can use the defined ontology as a basis for the implemented system, but also as a part of the implemented system. The ontology is expected to be an active part of the system, thus providing agents within the system with some basic knowledge about the system they are a part of.

scope

The ontology is planned to be related to the ontology developed by the authors as a part of previous research, published in [8]. MAGO-Ag ontology is planned as an extension of the ontology described in [8], featuring many of the same concepts, but enhanced with concepts that would make implementing the modelled system easier, i.e. featuring some of the concepts specific to system implementation, as an extension of modelling a given system. The scope of the MAGO-Ag ontology, therefore, encompasses concepts necessary for modelling large-scale multiagent systems (LSMASs), some concepts useful in modelling organisational aspects of a system of agents, and concepts that are useful for describing the implementation of such a system and translating the model into implementation blueprints. Amongst others, such concepts are included as: `agent`, `artefact`, `norm`, `behaviour`, `knowledge model`, `attribute`, `hasAttribute`, `providesBehaviour`...

level of granularity

The level of granularity stemming from the described is quite abstract. The ontology should include concepts that can be, for example, used to describe agents or artefacts in the system, but another layer of specification is expected to be added where specific types of agents should be described, and domain-specific artefacts defined. On top of this domain-specific layer is foreseen the individual-based layer. The MAGO-Ag ontology provides generalised concepts that should be specified by the system designer.

2
Knowledge Acquisition

> 'It is important to bear in mind that knowledge acquisition is an independent activity in the ontology development process. However, it coincides with other activities. [...] Most of the acquisition is done simultaneously with the requirements specification phase and decreases as the ontology development process progresses.' [1, p. 37]

The authors of METHONTOLOGY describe knowledge acquisition as a process that lasts throughout the ontology engineering process, yet it is not always of the same intensity. Early engineering process steps are richer in knowledge acquisition, classification and modelling. The main goal of this step is to identify sources of knowledge used as input for the remaining steps and to extract and acquire the knowledge necessary for successfully engineering the planned ontology.

This ontology's primary source of concepts, information, and knowledge is the MAMbO5 ontology presented in more detail in [8]. MAMbO5 results from an earlier collaboration instance of the sending and host institution, particularly this mobility's young researcher and the hosting research institute. The main goal of that ontology is to provide concepts related to modelling a multiagent system as an IVE, boosted with concepts used in describing the organisational features of a system of agents. An IVE in this context is a virtual system that can be seen as a model of a real system comprising agents, artefacts, and many other concepts related to the two. The agent and the artefact concepts are expected to be specialised for specific application areas when modelling a domain-specific scenario.

The purpose of MAGO-Ag ontology is to enable modelling a MAS in a way that is translatable into implementation, specifically in the SPADE-based implementation foundation of the modelled system. To do so, some concepts of the MAMbO5 ontology have to be modified, and some added, while the rest of the concepts can be left in the ontology for expressiveness and comprehensiveness. For knowledge acquisition in this part of the planned research and enhancing collaboration with the host institute, guided meetings have been performed, followed by structured brainstorming sessions and research plans, with the research team of Dr Carrascosa. Since the goal of this part of the planned research is aligned with a part of the research performed by Dr Carrascosa and his team, the rest of the planned research of this phase is conducted in cooperation with them.

Building on the MAMbO5 ontology, the MAGO-Ag ontology is planned to comply with the digital twin concept and the idea of containing concepts applicable to instantiating a MAS based on the model expressed using the developed ontology. Therefore, it must include some concepts related to the implementation domain, e.g., describing agents' internal variables or the knowledge models used. The following is an overview of the selection of the more interesting concepts, followed by a selection of the concepts that have to be added. Both the described tables include the concepts identified as such and are not necessarily exhaustive.

2.1 *Glossary of Terms*

A glossary of terms (GT) is a critical resource that defines and organizes key concepts, entities, and relationships within the modelled domain. This glossary forms the foundation for the ontology by ensuring that all participants in the development process share a common understanding of the domain's terminology.

The glossary is created to maintain conceptual clarity across the ontology. It avoids duplicate or overlapping concepts, ensuring every term is clearly defined. Additionally, GT serves as a stepping stone toward formalization, wherein these terms and relationships will later be translated into more structured, formal representations, such as in description logics or other ontology languages (e.g., OWL, RDF).

The GT typically includes:

- Concepts and definitions: A list of core terms representing the domain's important entities or phenomena. Each term is clearly defined to avoid ambiguity and to ensure consistency throughout the ontology.
- Relationships between concepts: The glossary may also describe how different concepts are interrelated. For example, it can specify hierarchies (e.g., subclasses), associations, or dependencies among the terms.
- Attributes and characteristics: Each concept might include specific attributes that describe it in greater detail. These attributes help formalize the ontology during later phases.

By ensuring that all relevant concepts are properly identified and explained, the GT helps to streamline the ontology's development process and ensures that the final product accurately reflects the domain's knowledge structure.

The following list of tables 2.1 – 2.45 encompasses definitions and descriptions of the most relevant terms that can be found in the

MAMBO5 ontology and the ontologies it was built upon. Other features, such as attributes and explicitly stated relationships between concepts, are not stated here since the concepts of the GT must be filtered before being added to the developed ontology. The content of the tables is directly derived from the author's doctoral thesis [2].

Table 2.1: *Acquisition* glossary entry

Concept name	Acquisition
Definition	An acquisition is the purchase of all or a portion of a corporate asset or target company[1].
Description	An acquisition is, in economical terms, described as, in layman's terms, one company buying another. This is usually done using stocks - the buyer buys most of the target company's ownership stakes to assume control of it[2]. Reasons for performing acquisitions are numerous, including to achieve economies of scale, greater market share, increased synergy, cost reductions, or new niche offerings.

Table 2.2: *Action* glossary entry

Concept name	Action (C)
Synonyms	Activity, Behaviour, Agent Action
Definition	An action is the building block of agents' activities.
Description	An action is esentially an agent's response to tasks. Whereby tasks are created to be met or reached, an action is the atomic concept for achieving tasks. In the context of this document, an action is the building block of a process, and agents' ability to act towards its environment in general. Every action can be used to fulfill at least one task.

Concept name	Agent (A)
Synonyms	Organisational Individual
Definition	A piece of software that can act upon its environment and perceive it.
Description	An agent in the context of this document is a piece of software that can interact with its environment, act upon it, and, in case of an intelligent agent, reason upon their accessible knowledge. Indeed, an agent is *anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.* [9] In the organisational context of this document, a software agent is essentially a model of a real-life person.

Table 2.3: *Agent* glossary entry

Concept name	Artefact
Definition	An artefact is an otherwise unclassified element of an organisation system.
Description	An artefact is, as of yet, a somewhat undefined concept, in the context of specifying its domain. Essentially, an artefact can be anything that is not classified using the other classes of this ontology. Furthermore, an artefact can be phisically representative (e.g. a chair), or an unphisical concept (e.g. knowledge). Artefacts therefore represent various concepts that the agents can interact with, or that affect the given environment or the given system, i.e. objects forming the environment.

Table 2.4: *Artefact* glossary entry

Concept name	Criteria of Organising
Definition	
Description	This concept comes from the OOVASIS ontology [10], [11] where it represents varius criteria of organising agents within an organisation. One of the criteria is ..., another ... . Therefore, this concept determines what are the grounds for creating the given organisation in the first place, and governs the decision flow in the context of deciding which organisational features (starting from architecture) are most suitable for the given criteria of organising.

Table 2.5: *Criteria of Organising* glossary entry

Table 2.6: *Design Factor* glossary entry

Concept name	Design Factor
Definition	A design factor is an internal or an external factor with significant influence on the design of an organisation.
Description	Everything that influences the design of an organisation on a non-neglectable level is considered a design factor. Design factors can be internal and external, relative to the given organisation. [11].

Table 2.7: *Design Method* glossary entry

Concept name	Design Method
Synonyms	Organisational Design Method
Definition	A design method is a common organisational design practice dealing with various aspects of organisational architecture.
Description	Every design method addresses a number of aspects of organisational architecture. A design method is esentially a common organisational design practice. [11]

Table 2.8: *Goal* glossary entry

Concept name	Goal (G)
Definition	A goal is a result towards which effort is directed - an end to be met.
Description	A goal is broadly defined as a result or achievement towards which effort is directed[3]. In the context of this document, a goal is a form of an objective. A goal is an end to be met or reached, and can consist of several sub-goals.

Table 2.9: *Heterarchical Organisational Structure* glossary entry

Concept name	Heterarchical Organisational Structure
Definition	Heterarchical organisational structure is an organisational structude without a single clearly defined pyramid-like structure.
Description	When there is no single clear pyramid-like line of control in an organisation, the given organisation can be described as having a heterarchical organisational structure. As opposed to hierarchical organisational structure, heterarchical organisational structure can be visualised as an oriented forest [12], or essentially using a network-based visualisation [11].

Concept name	Hierarchical Organisational Structure
Definition	Hierarchical organisational structure is an organisational structude with a single clearly defined pyramid-like structure.
Description	In contrast to the heterarchical organisational structure, hierarchical organisational structure can be identified by its basic pyramid-like form fostering hierarchical relations between organisation units. Such an organisational structure can be visualised using an oriented tree [12].

Table 2.10: *Hierarchical Organisational Structure* glossary entry

Concept name	Human Immersed Agent
Definition	Real-world agents that are represented in a IVE using their wearable tecchnology gadgets.
Description	Humans can be represented within a IVE and be available for interaction with the digital agents within the environment using digital aids, most prominently featured as wearable technology items, such as smartwatches and similar. Such agents are dubbed human immersed agents, since they are real-life people represented in the digital world using their attached piece of wearable discreet equipment.

Table 2.11: *Human Immersed Agent* glossary entry

Concept name	Hybrid Organisational Structure
Definition	Having mixed aspects of both heterarchical and hierarchical organisational structures, a hybrid organisational structure is a blend of the two.
Description	Having mixed aspects of both heterarchical and hierarchical organisational structures, a hybrid organisational structure is a blend of the two.

Table 2.12: *Hybrid Organisational Structure* glossary entry

Concept name	Inhabitant Agent
Definition	Every agent that is can be represented as phisically present in an IVE is considered an inhabitant agent.
Description	Agents that can be phisically represented within a IVE are called inhabitant agents. These agents can be of artificial or real-world nature. Usually various IVE artefacts exist within the IVE that represent various inhabitant agents [13]. It could be said that these agents have their habitats within their respective IVEs.

Table 2.13: *Inhabitant Agent* glossary entry

Concept name	Intelligent Virtual Environment (IVE)
Definition	An intelligent virtual environment is a virtual environment that simulates the real world, and is populated by autonomous intelligent entities. [14]
Description	Intelligent virtual environments are researched as an area on the intersection of two aspects pertaining to the concept of artificial intelligence, if only but marginally: intelligent tools and techniques that are embodied in autonomous agents (real-life and digital alike), and effective ways of representing them, along with various means of achieving different kinds of interaction amongst them [14], [15]. In other words, a IVE is a concepte that represents a virtual environment whose main goal is simulating a segment of the real world, populated by artificial autonomous entities (agents). [14]

Table 2.14: *Intelligent Virtual Environment* glossary entry

Concept name	IVE Law
Definition	A IVE law is a norm that is valid only within a specified physical space (a IVE workspace).
Description	A special kind of a norm, an IVE law is a norm that is constrained by its applicability to a specific physical space, i.e. a specific IVE workspace. Being applicable to only a restricted area means that every IVE law is valid only within the bounds of the given area (a IVE workspace), and never outside of that specified space. This kind of a norm is the key constraint of the concept of a situated organisational unit.

Table 2.15: *IVE Law* glossary entry

Concept name	IVE Workspace
Definition	
Description	Complimentary to the concept of a workspace, a IVE workspace represents a physical location, or a physically describable location.

Table 2.16: *IVE Workspace* glossary entry

Concept name	Knowledge Artefact (KnArt)
Definition	Knowledge artefact is a piece of knowledge of an agent or an organisation.
Description	A knowledge artefact is a piece of knowledge, or a set of knowledge terms available to agents within the system or within the IVE. Depending on the wanted level of abstraction, a knowledge artefact may represent a database containing various pieces of knowledge accessible by sets of agents, or individual pieces of knowledge. In the terms of rather undefined artefact class, knowledge artefacts are yet to be perfected in the context of knowledge representation and their suitability for representing knowledge of a IVE or a MAS.

Table 2.17: *Knowledge Artefact* glossary entry

Concept name	Manual
Definition	
Description	A manual defines the interface between individual agents and artefacts of a IVE. Including such a concept in the description of a IVE domain helps reduce unnecessary clutter in the context of setting ground-rules of how to use an artefact up front. The agents therefore immediately learn of the possibilities and applications of a given artefact without the need for exploring its possible uses.

Table 2.18: *Manual* glossary entry

Concept name	Merger
Definition	A merger is the process of organisational integration.
Description	In standard economical terms, a merger is a combination of more than one company by the transfer of the properties to one surviving company[4]. In the context of this document, merger can simply be regarded as an organisational integration.

Table 2.19: *Merger* glossary entry

Concept name	Norm
Definition	*Norms are informal rules that are socially enforced.* [16]
Description	Norms in general are not very different from the definition of a rule, their more generic counterpart. Used in a context of a population of a community, be it a natural or an artificial one, norms are expressions of desirable behaviour generally understood as rules indicating actions that are expected to be pursued. Norms are basically divided in three types: obligatory, prohibitive, and permissive. In the context of normative MASs though, there are three different terms associated with norms: conventions, social norms, and social laws [16], [17], and two categories [18]: conventions and essential norms.

Table 2.20: *Norm* glossary entry

Concept name	Normative System
Definition	*Systems in the behaviour of which norms play a role and which need normative concepts in order to be described or specified [. . .]* [19], [20]
Description	A normative system is a system built on norms and their enfonrcement upon the system, or system's definition of architecture based on the said norms. In the context of computer science, a normative system is described as a system whose behaviour is influenced by norms, and whose description or specification depencds on using normative concepts [19], [20].

Table 2.21: *Normative System* glossary entry

Concept name	Objective (O)
Definition	An objective is a high-level goal the be met, suitable for the context of strategic planning.
Description	An objective is more general than a goal, although their definitions are rather similar. Fulfilling several goals can lead an organisational unit towards fulfilling a set objective. Thus, an objective is more suitable in the context of strategic planning, while a goal is more suitably used in the context of short-term planning.

Table 2.22: *Objective* glossary entry

Table 2.23: *Observable Property* glossary entry

Concept name	Observable Property
Definition	An observable property is a peroperty of an artefact that can be observed by agents in the same IVE.
Description	This is a property of an artefact located in a IVE that is observable by other agents located within the same IVE. These are tighly connected to the concept of observable events, and can be influenced upon by an operation.

Table 2.24: *Organisation* glossary entry

Concept name	Organisation
Definition	An organisation is generally a group of agents structured according to a set criteria, with the basic goal of overcoming limitations of individual agency and achieving an organisation goal.
Description	An apt definition is given in [21] where an organisation is defined using several characteristics, including large-scale problem solving technology, composition of multiple agents, systems of goal-directed activities, etc. Furthermore, an essential benefit of organisations is identified in overcoming limitations of individual agency, especially cognitive, physical, temporal, and institutional.

Table 2.25: *Organisational Architecture* glossary entry

Concept name	Organisational Architecture
Definition	In the context of this document, organisational architecture is the superclass for all the organisation-related concepts that deal with more than one aspect of organisational architecture.
Description	All those concepts that deal with more than one aspect of organisational architecture, i.e. are not specialised as for example concepts that describe organisational structure only, are classified as belonging to the organisational architecture concept. [11] therefore identifies 15 such concepts.

Table 2.26: *Organisational Change* glossary entry

Concept name	Organisational Change
Synonyms	Organisational Dynamics
Definition	
Description	The concept of organisational change is closely tied to the intension of the concept of orgnaisational dynamics, since both concepts describe change to the established agent organisations. A change in the context of organisational change definition can be influenced by an organisational design method, yet unmistakingly it affects the organisational architecture of the given organisation. A change as defined here can adhere to one of the identified types of change (e.g. structural, cultural, strategic, etc.), can be attributed an impact of change, reason why the change started, and a key influence area (e.g. organisational memory) [11].

Table 2.27: *Organisational Culture* glossary entry

Concept name	Organisational Culture
Definition	*Organizational culture defines important intangible aspects of an organization including knowledge, social norms, reward systems, language and similar.* [10], [22]
Description	The concept of organisational culture encompasses all the intangible aspects of an organisation, such as knowledge, various types of norms, a system of rewards, languages used in the organisation, etc. Organisational culture is therefore a concept that is mostly based in the organisational units, i.e. in the individual agents forming the organisation, and is thus the most fuzzy concept of all the perspectives of an organisation. [11], [22] provide a quick overview of various conceptualisations of organisational architecture, where it is visible that organisational culture is an important part of an organisation.

Concept name	Organisational Environment
Definition	Organisational environment are all the external factors that have the capacity to influence an organisation.
Description	The concept of organisational environment encompasses all the concepts that represent factors external to an organisation that have a potential to influence the given organisation, such as external organisations or individuals, or external events. Main concerns when organisational environment is considered are directed towards identifying constraints imposed on the given organisation by the environment, and demands of the environment towards the given organisation. [22]

Table 2.28: *Organisational Environment* glossary entry

Concept name	Organisational Knowledge Network
Definition	Organisational knowledge network is a network created by interconnected pieces of organisational knowledge.
Description	A network connecting all the pieces of organisational knowledge is considered to build an organisational knowledge network that effectively collects and intertwines all the knowledge of an organisation, thus fostering knowledge sharing and reuse amongst the organisational units of the given organisation, i.e. ultimately individual agents.

Table 2.29: *Organisational Knowledge Network* glossary entry

Concept name	Organisational Structure
Definition	Organisational structure is a concept comprising various aspects and forms f structuring organisational units.
Description	Concepts used for describing various aspects and forms of structuring organisational units are categorised as belonging to the concept of organisational structure. Based on two different approaches, two criteria for classifying concepts of organisational structuring are used. The first depends on whether the given structure is the main structure or is it laid over the organisation, as a form of a superstructure. The second is based on the form of the structure, i.e. is it a hierarchical or heterarchical, or a mix of both.

Table 2.30: *Organisational Structure* glossary entry

Table 2.31: *Organisational Unit* glossary entry

Concept name	Organisational Unit (OU)
Definition	An organisational unit is the key elementary unit in the context of forming an organisation.
Description	An organisational unit is the elementary unit of an organisation that, under the influence of the other organisational concepts, forms an organisation. In the context of this document, and the area of LSMASs, an organisational unit is usually considered to represent an individual agent. Using the recursive definition though, an organisational unit that comprises multiple organisational units can be, under circumstances specified in [10], considered as an organisational unit. Using a more graphic explanation, a department organisational unit that comprises individual agents can be considered as individual organisational unit on a higher level of organisational hierarchy, where department organisational units form a higher-level organisational unit of a faculty.

Table 2.32: *Physical Artefact* glossary entry

Concept name	Physical Artefact
Synonyms	IVE Artefact
Definition	Physical artefacts are all the concepts that can be physically represented and included in a IVE.
Description	Every concept that describes objects that can be physically represented (e.g. a top hat), i.e. embodied and positioned on a topological map, and as such included in a IVE are classified as physical artefacts. Such elements have their role to play in the given IVE and usually contain a defined interface that governs the process of interaction of an agent with the given physical artefact.

Table 2.33: *Physical Property* glossary entry

Concept name	Physical Property
Definition	
Description	Physical properties are key elements of physical artefacts, i.e. artefacts that can be visualised in a physical space. Usually when an artefact is used, a physical event is generated, and a physical property is modified.

Table 2.34: *Plan* glossary entry

Concept name	Plan
Definition	A plan is a finite set of actions that leads to a specified goal.
Description	A plan is a finite set of actions that leads to a specified goal. An optimal plan cannot be made shorter if the same goal is retained in the process. The plan concept is especially useful when observing belief-desire-intention (BDI) agents, since it is driven by agents' desires and intentions.

Table 2.35: *Process* glossary entry

Concept name	Process (P)
Synonyms	Organisational Processes
Definition	A set of connected atomic actions.
Description	A process is in the context of this document defined as a set of atomic actions. Every process itself can be a part of another process, thus creating the recursive relation. A process can be performed in order for a goal to be met. It represents an activity or a procedure of an organisation [22].

Table 2.36: *Quest* glossary entry

Concept name	Quest (Q)
Definition	A quest is similar to a goal, but has a defined starting and ending situations.
Description	A quest is a similar to a goal, but it has a defined beginning and a defined end, i.e. a starting situation, and an ending situation[5]. In the context of massively multi-player on-line role-playing games (MMORPGs), a quest is what drives a story, and, in principle, motivates the player to continue playing the game. Furthermore, a quest is often given to the player by an in-game character. A quest usually has various stages, and represents a challenge for the given player, thus embarking them on an adventure.

Table 2.37: *Role* glossary entry

Concept name	Role (R)
Definition	A role is a set of norms with a common denominator.
Description	In the context of this document, a role is defined as a set of normative rules that are applicable to a particular part of the given organisation. Such normative rules are parts of the organisation's normative system, and can be grouped by specific criteria, thus forming roles. Roles are played by agents. When an agent plays a role, the role's constraints are applied to them, therefore constraining their possible actions, their perceivable goals, and their possibilities in general.

Table 2.38: *Rule* glossary entry

Concept name	Rule
Definition	Rules are elementary forms of constraints in normative systems, as they pose a basic aspect of defining standards.
Description	A rule is an atomic building block of a normative system. Rules are usually built in a general if-then form, meaning that two statements are connected with a causal link, thus regulating what happens (then part: consequent) if something else happens beforehand (if part: antecedent). Other forms of rules are possible as well, but are not used as often. For the most part, rules pose constraints on the given subject. Rules are commonly used for devising appropriate logical conditions for introducing modalities. [16]

Table 2.39: *Situated Organisational Unit* glossary entry

Concept name	Situated Organisational Unit
Definition	Every organisational unit that is tied to a location through a situated norm is considered a situated organisational unit.
Description	An organisational unit that is tied to a specific IVE, or a specific geographic or otherwise place, is a situated organisational unit. Furthermore, such an organisational unit has some situated norms that refer to it. The place that is essential to the situated relation of a situated organisational unit can be physical or digital, but can usually be represented visually, following the description of an inhabitant agent.

Table 2.40: *Strategic Alliance* glossary entry

Concept name	Strategic Alliance
Definition	Strategic alliance is a form of a long-lasting partnership of organisations of various forms, formed around a shared strategy, or a strategic goal.
Description	An alliance that is aimed at forming long-lasting partnerships consisting of organisations of various forms is dubbed a strategic alliance. A strategic alliance is formed around a strategy as a long-term objective that is shared amongst the strategic alliance members. Norms and regulations governing the expected behaviour within the strategic alliance are expected to be accepted by all the members, old and new alike.

Table 2.41: *Strategy* glossary entry

Concept name	Strategy
Synonyms	Organisational Strategy
Definition	*Strategy defines the long term objectives of an organization, action plans for their realization as well as tools on how to measure success.* [11], [22]
Description	A strategy is, in the context of planning and shared organisational values, a long-term objective that is specified mosotly as a vision. It may consist of a number of objectives, quests, and similar. Strategy is therefore tentative in the context of plans of achieving it, but is versatile in terms of temporal likeness to change. Since it represents a long-term planning concept, a strategy is the main driving force of strategic alliances as agent coalitions meant to provide long-term suport to its members.

Table 2.42: *Super Structure* glossary entry

Concept name	Super Structure
Definition	An inter-organisational structure formed above the conventional organisational structure.
Description	When organisations form structures comprising other organisations, a super-structure is formed. In the context of this document, a super-structure is thus described as an organisation of organisations, esentially spanning further than the usual reaches of a given average organisation. Such an inter-organisational structure is formed above the conventional organisational structure.

Table 2.43: *Task* glossary entry

Concept name	Task
Definition	A task is the building block of a quest.
Description	A task is the building block of a quest, i.e. its elementary part. A quest is built of atomic tasks that are easier to follow in execution phase, rather than the overview provided by the main definition of a quest. In MMORPGs a quest could demand an item to be retrieved, yet such a simple-sounding quest could consist of various tasks that have to be fulfilled in order for the main quest to be finished. The relation of quest and task concepts can be recursive[6].

Table 2.44: *Time Dependent Norm* glossary entry

Concept name	Time Dependent Norm
Definition	A norm that is dependent on the temporal aspect of the world is a time dependent norm.
Description	A time dependent norm is essentially a norm, but with an added temporal constraint. Particularly, a time dependent norm is constrained to a specific period in time, be it for its designated activity period, period during which the given norm is applicable, or simply the timeframe or a deadline when a change of the norm, or caused by the norm, is to be expected.

Table 2.45: *Workspace* glossary entry

Concept name	Workspace (W)
Definition	A workspace is the union of all the elements of a system, including agents, artefacts, etc.
Description	A workspace is the complete environment of a given system, including all the agents, artefacts, etc. What sets the concept of a workspace apart from the concept of an environment is the extent of the involved concepts, i.e. a workspace contains all the elements of an organisation and the whole system, while environment comprises only the elements that are external to the given organisation. It is worth noting that elements of the environment are an integral part of the whole system, since the life and activities of the given organisation are influenced by them.

3 Conceptualisation

The Conceptualisation phase in ontology development transforms the collected knowledge into a structured and meaningful model. During this phase, the key concepts, relationships, attributes, and rules of the domain are identified and organized to reflect the domain's structure. The goal is to create a clear and coherent conceptual model that captures the domain's essential elements without formalizing them into specific ontology languages. The conceptual model produced in this phase is not yet formalized but provides a foundational blueprint for the subsequent formalization and implementation phases of the ontology development process

3.1 Applicable Concepts

Overview of the select concepts contained in MAMbO5 ontology

Agent 'An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.' [5, p. 54]

Organisational unit An organisational unit is an `Agent` subclass that is a part of an organisation. Usually, a MAS can be considered an organisation comprising multiple agents. Strictly speaking, it would have to feature and use some other concepts related to defining, for example, an organisational structure. An organisational unit can be a part of another organisational unit and consist of organisational units. This feature, similar to the concept of holons and holarchy, allows for modelling organisations on different levels of abstraction or hierarchy. Each `OrganisationalUnit` instance is expected, at any given point in time, to be able to enact one of a set of `Role` individuals at its disposal.

Behaviour The concept of agent behaviour is described in the `ooooaflsmas` ontology as '[...] some kind of activity performed by some agent. It has to be acceptable by a normative system the agent belongs to.' In terms of SPADE implementation, behaviour is the most basic way of implementing the operations of an agent. Each agent can have multiple behaviours. Types of behaviours offered by default by SPADE can already be found in the `ooooaflsmas` ontology, i.e., one-shot, periodic, and finite state behaviour that acts according to the principles of finite automata. Each `Behaviour` individual is expected to provide its `Agent` individual with the ability to achieve a certain objective. Some innate `Behaviour` individuals will be available to specific `Agent` subclasses by default.

see figure 3.1

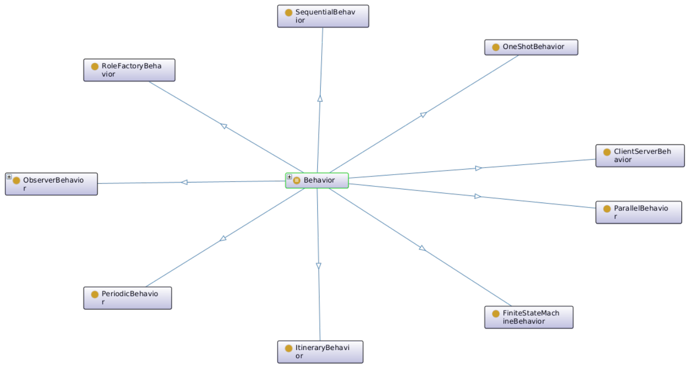


Figure 3.1: Subclasses of the `Behavior` concept in `ooooaflsmas`

The MAGO-Ag ontology is specialised for the context of developing Smart Python Agent Development Environment (SPADE) agents, wherefore it comprises only the behaviour concepts relevant to the behaviours explicitly implementable using SPADE.

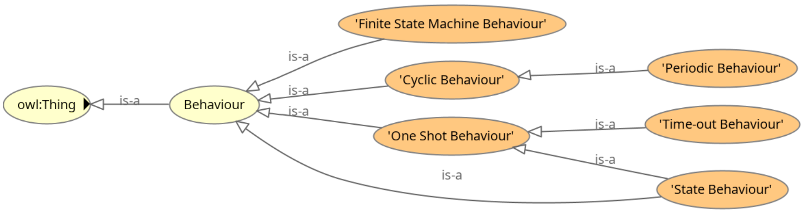


Figure 3.2: Subclasses of the class `Behaviour` in MAGO-Ag ontology

Role The concept of a role is described within the `ooooaflsmas` ontology as a 'prescribed or expected behavior associated with a particular position or status in a group or organization'. This concept is in the mentioned ontology designated as a direct subclass of the concept `Norm`, derived from the domain of organisational modelling and describing organisation systems. The `Norm` concept is defined therein as '(socially) accepted behavior in a defined group and [they] represent a blueprint for behaving in said group.' Based on the stated, the concept of `Role` is interesting because it enables defining a set of features that will be put at the agent's disposal playing the chosen role. In other words, roles can be used as a way of combining different features that can be enacted by an agent.

The `Role` concept was modelled in [2] as a concept which was related to (possibly) several instances of concepts describing behaviours and objectives, meaning that specific roles allow agents who enact them to attain a specific set of behaviours that enables and empowers them to achieve specific objectives, thus solving specific tasks. Different objectives demand the enactment of various roles from the pool of roles available (defined) in the system.

Within the MAGO-Ag ontology, the concept of a role is used to group features and individuals available to various agents as a 'package', i.e. a set of features that is available to various individual agents.

see figure 3.3

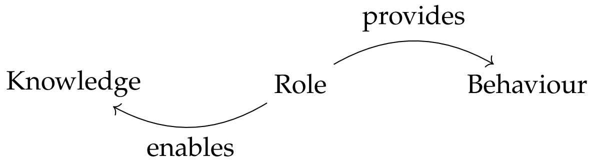


Figure 3.3: A `Role` individual can provide some `Behaviour` individuals and can enable access to some `Knowledge` individuals

Artefact An artefact is used in the `JaCalIVE` ontology as a comprehensive concept encompassing all interactive and non-interactive objects that are not suitable to be implemented as agents but should be present in the modelled system nonetheless. `MAMbO5` ontology recognises specific versions of artefacts as an `IVE_Artifact`, which is located within the IVE, and its even more specific version `Physical_Artifact`, which can be physically represented or is expected to be physically represented, and a complementary specification given as a `KnowledgeArtifact` that is an artefact that is abstract and describes various rules that can be found within the modelled system. Its initial authors describe the latter concept as encompassing 'a wide range of explicit knowledge,' including, but not limited to, knowledge models, such as machine learning models or neural networks. The `Norm` concept and its subclass `Role` are subclasses of the `KnowledgeArtifact` concept, as defined in `ooooaflsmas`.

see figure 3.4

The MAGO-Ag ontology introduces the concept of a `Software Artefact` as well, as a specific type of an `Artefact`. This concept describes artefacts that can be used by the system, usually by means of API endpoints, and can encompass different web services, including the XMPP server that is necessary for instantiating SPADE agents.

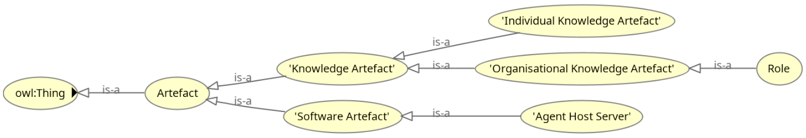


Figure 3.4: Class hierarchy on subclasses of the `Artifact` concept in MAGO-Ag ontology

Strategy is modelled within the MAGO-Ag ontology using several related concepts. In short, specific objectives can be achieved by conducting a set of actions, i.e. a plan, whereby individual actions are implemented using agent behaviours, while specific behaviours are provided by `Role` individuals. Modelling these concepts and the accompanying relationships within the ontology and translating those relationships to agent implementations eases the process of implementing agents that can play specific roles based on the goals they are faced with. Therefore, agents may be implemented as intelligent in terms of choosing the roles that can enable them to perform specific behaviours that will, in turn, help them achieve specific objectives they are faced with.

Workspace is a concept that represents a group of various elements, i.e. a union of all the elements of a system. In the context of MAGO-Ag, a workspace represents the immediate neighbourhood of a number of agents. Multiple `Workspace` individuals can, therefore, exist within a single system. The ultimate goal of the MAGO-Ag is to enable `Agent` instances to move between workspaces based on their neighbours, i.e. based on the intensity and frequency of their communication with other agents. For example, should agent Alice communicate more often with agent Charles, who is not in Alice's workspace, than with agent Bob, who is in Alice's workspace, then Alice might want to move to Charles' workspace, which might be more efficient in the long run.

3.2 *Data Dictionary*

The following are the descriptions and definitions of the key concepts and a selection of relationships that can be found in the MAGO-Ag ontology.

see tables 3.1 – 3.14

see tables 3.15 – 3.26

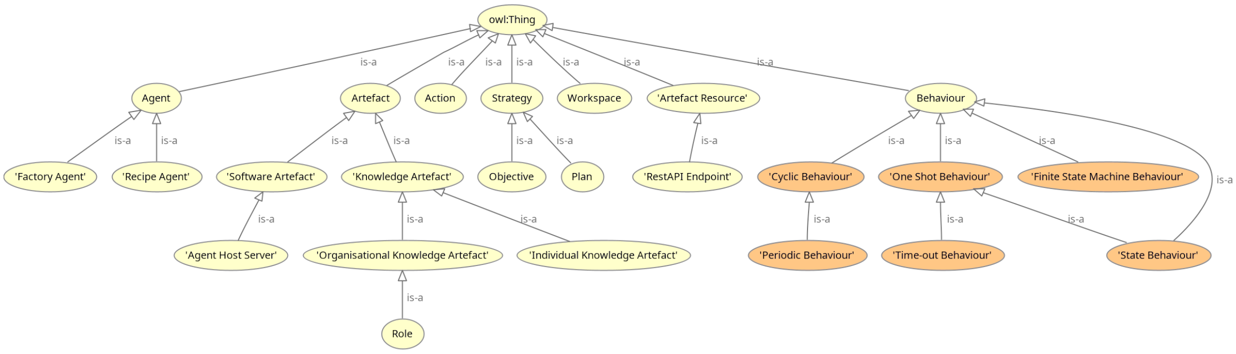


Figure 3.5: Concepts and their subconcepts in the MAGO-Ag ontology

Concept name	Action
Definition	An action is the building block of agents' activities.
Description	An action is esentially an agent's response to tasks. Whereby tasks are created to be met or reached, an action is the atomic concept for achieving tasks. In the context of this document, an action is the building block of a process, and agents' ability to act towards its environment in general. Every action can be used to fulfill at least one task.

Table 3.1: *Action* concept description

Concept name	Agent
Definition	A piece of software that can act upon its environment and perceive it.
Description	An agent in the context of this document is a piece of software that can interact with its environment, act upon it, and, in case of an intelligent agent, reason upon their accessible knowledge. Indeed, an agent is *anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.* [9] In the organisational context of this document, a software agent is essentially a model of a real-life person.

Table 3.2: *Agent* concept description

Concept name	Artefact
Definition	An artefact is an otherwise unclassified element of an organisation system.
Description	An artefact is, as of yet, a somewhat undefined concept, in the context of specifying its domain. Essentially, an artefact can be anything that is not classified using the other classes of this ontology. Furthermore, an artefact can be phisically representative (e.g. a chair), or an unphisical concept (e.g. knowledge). Artefacts therefore represent various concepts that the agents can interact with, or that affect the given environment or the given system, i.e. objects forming the environment.

Table 3.3: *Artefact* concept description

Concept name	Behaviour
Definition	A behaviour is an implemented action, or a part of one, that defines how an agent interacts with its environment or internal states.
Description	In the context of this document, a behaviour is a set of actions or operations that an agent can perform in response to its environment or internal states. Behaviours are essential components that define how an agent acts and reacts, allowing it to achieve its objectives and respond to changes. Moreover, a behaviour is, in this context, a way of implementing a specific action and is that which an agent can actually run in runtime.

Table 3.4: *Behaviour* concept description

Concept name	Knowledge Artefact
Definition	Knowledge artefact is a piece of knowledge of an agent or an organisation.
Description	A knowledge artefact is a piece of knowledge, or a set of knowledge terms available to agents within the system or within the IVE. Depending on the wanted level of abstraction, a knowledge artefact may represent a database containing various pieces of knowledge accessible by sets of agents, or individual pieces of knowledge. In the terms of rather undefined artefact class, knowledge artefacts are yet to be perfected in the context of knowledge representation and their suitability for representing knowledge of a IVE or a MAS. Nonetheless, the artefact usually considered to be a knowledge artefact is an ontology.

Table 3.5: *Knowledge Artefact* concept description

Concept name	Individual Knowledge Artefact
Definition	An individual knowledge artefact is knowledge specific to an individual agent.
Description	An individual knowledge artefact is a piece of knowledge or information that is specific to an individual agent. It represents the knowledge that an agent possesses individually, which may not be shared with other agents in the system.

Table 3.6: *Individual Knowledge Artefact* concept description

Concept name	Organisational Knowledge Artefact
Definition	An organisational knowledge artefact is knowledge shared among agents within an organisation.
Description	An organisational knowledge artefact is a piece of knowledge or information that is shared among agents within an organisation. It represents the collective knowledge accessible to agents, facilitating collaboration and consistent understanding across the organisation.

Table 3.7: *Organisational Knowledge Artefact* concept description

Concept name	Software Artefact
Definition	A software artefact is a software component or resource that agents can interact with.
Description	A software artefact is a type of artefact that refers to software components, modules, or systems within the organisation or outside of one. These can include software applications, libraries, or any digital resources that agents can interact with or use to perform actions.

Table 3.8: *Software Artefact* concept description

Concept name	Artefact Resource
Definition	An artefact resource is a resource associated with an artefact that agents can utilise.
Description	An artefact resource is a resource associated with an artefact that agents can utilise or access. It can be any supportive element that enhances the functionality of an artefact or provides additional capabilities to agents when interacting with the artefact.

Table 3.9: *Artefact Resource* concept description

Concept name	Objective
Definition	An objective is a high-level goal the be met, suitable for the context of strategic planning.
Description	An objective is more general than a goal, although their definitions are rather similar. Fulfilling several goals can lead an organisational unit towards fulfilling a set objective. Thus, an objective is more suitable in the context of strategic planning, while a goal is more suitably used in the context of short-term planning.

Table 3.10: *Objective* concept description

Concept name	Plan
Definition	A plan is a finite set of actions that leads to a specified goal.
Description	A plan is a finite set of actions that leads to a specified goal. An optimal plan cannot be made shorter if the same goal is retained in the process. The plan concept is especially useful when observing BDI agents, since it is driven by agents' desires and intentions.

Table 3.11: *Plan* concept description

Concept name	Role
Definition	A role is a set of norms with a common denominator.
Description	In the context of this document, a role is defined as a set of normative rules that are applicable to a particular part of the given organisation. Such normative rules are parts of the organisation's normative system, and can be grouped by specific criteria, thus forming roles. Roles are played by agents. When an agent plays a role, the role's constraints are applied to them, therefore constraining their possible actions, their perceivable goals, and their possibilities in general.

Table 3.12: *Role* concept description

Concept name	Strategy
Definition	*Strategy defines the long term objectives of an organization, action plans for their realization as well as tools on how to measure success.* [11], [22]
Description	A strategy is, in the context of planning and shared organisational values, a long-term objective that is specified mosotly as a vision. It may consist of a number of objectives, quests, and similar. Strategy is therefore tentative in the context of plans of achieving it, but is versatile in terms of temporal likeness to change. Since it represents a long-term planning concept, a strategy is the main driving force of strategic alliances as agent coalitions meant to provide long-term suport to its members.

Table 3.13: *Strategy* concept description

Concept name	Workspace
Definition	A workspace is the union of all the elements of a system, including agents, artefacts, etc.
Description	A workspace is the complete environment of a given system, including all the agents, artefacts, etc. What sets the concept of a workspace apart from the concept of an environment is the extent of the involved concepts, i.e. a workspace contains all the elements of an organisation and the whole system, while environment comprises only the elements that are external to the given organisation. It is worth noting that elements of the environment are an integral part of the whole system, since the life and activities of the given organisation are influenced by them.

Table 3.14: *Workspace* concept description

Concept name	can play role
Description	Associates an Agent with the Role(s) it can play within the organisation. Domain: Agent. Range: Role.

Table 3.15: *can play role* relationship description

Concept name	is part of role
Description	Indicates that a Role is a part of another Role, establishing hierarchical relationships between roles. Domain: Role. Range: Role.

Table 3.16: *is part of role* relationship description

Concept name	provides behaviour
Description	Associates a Role with the Behaviour(s) it provides, defining the actions or activities an agent can perform when playing that role. Domain: Role. Range: Behaviour.

Table 3.17: *provides behaviour* relationship description

Concept name	can access artefact
Description	Indicates that an Agent has access to a particular Artefact within the environment. Domain: Agent. Range: Artefact.

Table 3.18: *can access artefact* relationship description

Concept name	has URI
Description	Assigns a Uniform Resource Identifier (URI) to an Artefact for identification or access purposes. Domain: Artefact. Range: string (URI).

Table 3.19: *has URI* relationship description

Concept name	has name
Description	Assigns a human-readable name to an entity within the ontology. Domain: Any entity. Range: string.

Table 3.20: *has name* relationship description

Concept name	is before state
Description	Defines the transition from one State Behaviour to another in a finite state machine, indicating the sequence of states. Domain: State Behaviour. Range: State Behaviour.

Table 3.21: *is before state* relationship description

Concept name	has initial state
Description	Associates a Finite State Machine Behaviour with its initial State Behaviour(s). Domain: Finite State Machine Behaviour. Range: State Behaviour.

Table 3.22: *has initial state* relationship description

Concept name	has final state
Description	Associates a Finite State Machine Behaviour with its final State Behaviour(s). Domain: Finite State Machine Behaviour. Range: State Behaviour.

Table 3.23: *has final state* relationship description

Concept name	has action
Description	Associates a Process with the Action(s) that compose it. Domain: Process. Range: Action.

Table 3.24: *has action* relationship description

Concept name	has objective
Description	Associates an Action with the Objective(s) it aims to achieve. Domain: Action. Range: Objective.

Table 3.25: *has objective* relationship description

Concept name	has behaviour
Description	Associates an Action with the Behaviour(s) required to perform it. Domain: Action. Range: Behaviour.

Table 3.26: *has behaviour* relationship description

4 Formalisation

The formalization phase in ontology engineering focuses on applying formal semantics to the conceptual model of a domain. This phase aims to create an ontology that is precise, unambiguous, and capable of supporting complex reasoning. By using a formal language, the ontology's structure, relationships, and constraints become systematically interpretable by both humans and machines.

The process of formalising an ontology transforms a conceptual understanding into a functional and robust model, which can support various applications, ranging from data interoperability to intelligent system operations. Such a formal structure is essential for ensuring the ontology's adaptability, reusability, and ability to integrate across different domains and systems.

The main objectives of this phase are:

- Encoding concepts and relationships, i.e. transforming conceptual entities and their relationships into formal representations;
- Defining logical constraints by establishing rules, constraints, and axioms that govern how concepts interact, and their properties, to ensure logical consistency and support reasoning processes;
- Specifying hierarchies and classifications, i.e. organizing concepts into clearly defined hierarchies, such as classes and subclasses, that support reasoning processes;
- Ensuring compatibility with ontology languages by structuring the ontology in a language that supports formal semantics, allowing it to be used with reasoning tools and knowledge-based systems.

In ontology engineering, various serialization methods and formats are commonly used to encode ontologies in a structured and machine-readable manner. Each serialization format has distinct characteristics and purposes, offering different levels of expressiveness and compatibility with tools for ontology development and reasoning. Below are three of the most frequently used serialization methods.

RDF/XML is an XML-based serialization format for resource description framework (RDF) data, used extensively for ontologies that follow the RDF standard. It encodes RDF triples – subject, predicate, and object – in an extensible markup language (XML) format, making it compatible with XML tools and parsers. This method is often used for semantic web applications and data interchange across platforms, particularly when XML compatibility is a priority.

Turtle provides a more compact and human-readable serialization format for RDF. It uses a simplified, text-based syntax for representing RDF triples, making it easier to read and edit manually than RDF/XML. This method is popular for ontology development and editing, especially during the ontology design process, where readability is advantageous.

see listing 4.2

OWL/XML is an XML serialization for ontologies written in web ontology language (OWL). It is particularly designed to represent OWL constructs in a structured XML format, supporting all OWL semantics. This method is suitable for applications where interoperability with XML-based systems is essential and for storing complex ontologies where OWL-specific constructs are frequently used.

see listing 4.1

The complete Turtle syntax serialization of MAGO-Ag ontology is given in appendix MAGO-Ag Turtle Serialization.

see MAGO-Ag Turtle Serialization

```
<owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RCjIVQ3sTEfJR8m9yEY3Mhe">
    <owl:equivalentClass>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                <owl:Restriction>
                    <owl:onProperty
                    ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49"/>
                    <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</owl:hasValue>
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </owl:equivalentClass>
    <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
    <rdfs:label xml:lang="en-gb">Cyclic Behaviour</rdfs:label>
</owl:Class>
```

Listing 4.1: OWL/XML serialization of the concept labelled `Cyclic Behaviour`

```
:RCjIVQ3sTEfJR8m9yEY3Mhe rdf:type owl:Class ;
    owl:equivalentClass [
        owl:intersectionOf (
            :R7dQDUF81SOJC29Sctpy6aP
            [
              rdf:type owl:Restriction ;
              owl:onProperty :OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49 ;
              owl:hasValue "true"^^xsd:boolean
            ]
        ) ;
        rdf:type owl:Class
    ] ;
    rdfs:subClassOf :R7dQDUF81SOJC29Sctpy6aP ;
    rdfs:label "Cyclic Behaviour"@en-gb .
```

Listing 4.2: Turtle serialization of the concept labelled `Cyclic Behaviour`

5 Integration

The integration phase of this ontology engineering methodology focuses on harmonizing the developed ontology with other relevant ontologies or knowledge sources to create a unified knowledge ecosystem. This phase enriches the ontology's content, enhancing its semantic reach and improving its reusability across various applications.

Connecting with external ontologies supports data consistency, reduces redundancy, and ensures that the developed ontology can communicate effectively within broader systems, ranging from semantic web applications to intelligent multi-agent frameworks.

Integration is commonly performed using a combination of the following actions. Related equivalent concepts are identified between the current ontology and other ontologies to ensure consistency and avoid redundancy. Concepts and relationships from multiple ontologies are combined, often by importing parts of external ontologies or adding new entities that enhance the current model. Links are established between concepts in different ontologies, defining rules or correspondences to enable data interchange and semantic interpretation across systems. Adapting the ontology structure, used and existing naming conventions, and logical definitions ensures compatibility with the integrated ontologies or systems.

The MAGO-Ag ontology can be considered a filtered out and domain-specific subset of concepts from the Multiagent Model Based on Organisations for Intelligent Virtual Environments (MAMbO5) ontology. This is a natural continuation of the already established research cooperation between this research's host and sending institutions. The concepts of MAGO-Ag ontology bearing the same names as those of the MAMbO5 ontology may be considered to be the same, even though taking them for subconcepts should be the preferred approach since MAGO-Ag concepts are made to be adapted to implementation and are thus more domain-specific. Furthermore, newly-defined properties of the MAGO-Ag ontology, related to the concepts that can be found in MAMbO5 as well, present a valid argument in favour of regarding MAGO-Ag concepts as MAMbO5 subconcepts.

👁 see figure 5.1

For the sake of simplicity in the context of using the MAGO-Ag ontology with the MAGO-Ag framework, the ontology is not formally related to MAMbO5 ontology in its current form of implementation.

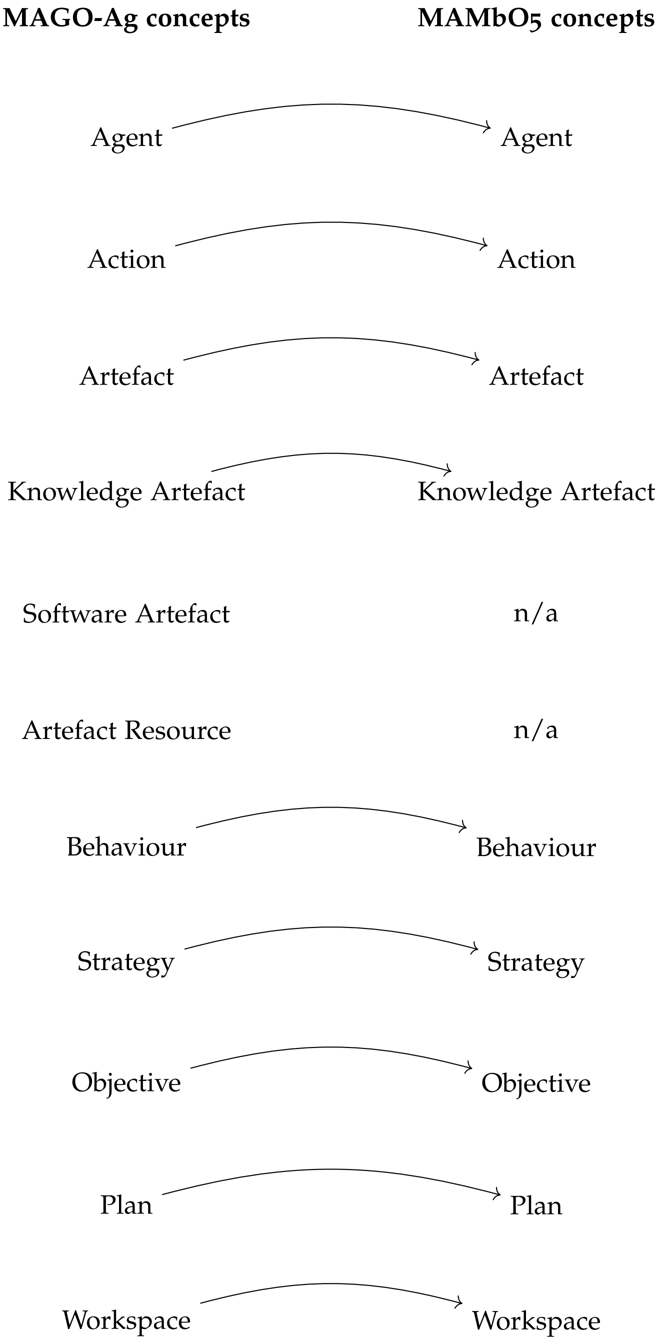


Figure 5.1: Mapping of the MAGO-Ag and the MAMbO5 concepts

6 Implementation

The implementation phase is a step in ontology engineering where the abstract model of the domain is transformed into a tangible, functional system component. This phase involves converting the formalized ontology into a format that is compatible with the chosen technological infrastructure, enabling it to interact with other systems, perform reasoning tasks, and respond to user queries. Successful implementation ensures the ontology accurately represents knowledge and operates smoothly within its intended environment. This process forms the foundation for intelligent systems, knowledge management tools, and data-driven applications.

The MAGO-Ag ontology was implemented using the Protégé tool for engineering ontologies. The implemented ontology is available on the GitHub repository of this research. The ontology is serialized using the OWL/XML method. Such a file, bearing the extension `.owx`, can be used by the programming language Python via the available libraries, such as `owlready2`.

↗ https://github.com/AILab-FOI/MAGO

The implemented ontology is consistent and provides additional knowledge when reasoned upon using one of the reasoners available in Protégé or `owlready2`.

The complete OWL/XML serialization of the implemented MAGO-Ag ontology is in appendix MAGO-Ag OWL/XML Serialization.

👁 see MAGO-Ag OWL/XML Serialization

7 Evaluation

The evaluation phase in ontology engineering focuses on verifying and validating the ontology's design, structure, and functionality. This phase ensures that the ontology accurately represents the intended domain and operates effectively within its application context. The evaluation phase helps identify potential improvements and confirms that the ontology is ready for deployment by systematically assessing factors such as accuracy, completeness, and operational efficiency. This quality control process establishes the ontology as a reliable and robust knowledge resource aligned with user requirements and technical expectations. Evaluation often involves both automated testing tools and manual reviews by domain experts.

Several activities are a part of the evaluation phase of the methodology used in this ontology engineering process. Quality assessment evaluates the ontology's internal consistency, logical coherence, and adherence to domain knowledge. It is tested against the intended scope regarding concepts and relationships to ensure the ontology is complete and accurate. The ontology's efficiency, scalability, and responsiveness are tested when integrated into the target environment. Validation against the specification is performed in order to verify that the ontology can address the features and requests defined during the specification phase, confirming its suitability for its intended applications.

The MAGO-Ag was evaluated by the domain experts from the host institution using the applicable interview and review approaches. Furthermore, the ontology was tested against the specification document. In addition to the stated, the ontology was, in part, presented in its finalised state at a professional conference [23].

👁 see chapter 1

Part II

Phase 2: The Framework

8
Framework Design, Description, and Implementation

The main objective of the MAGO-Ag framework is to provide a medium for converting a MAS model defined using the related MAGO-Ag ontology into a template for implementing the modelled system using SPADE library of Python. The rendered implementation template is not expected to include all the details that might be needed to run the finished system of agents. Still, it is planned to provide the initial implementation requirements of the modelled system. Including all the implementation details in the ontology might prove to be too cumbersome and taxing for the modelling process.

see figure 8.1

The framework is expected to translate the necessary elements of the ontology to classes, objects, and instances where applicable and provide the rest of the ontology knowledge to agents translated into applicable data types.

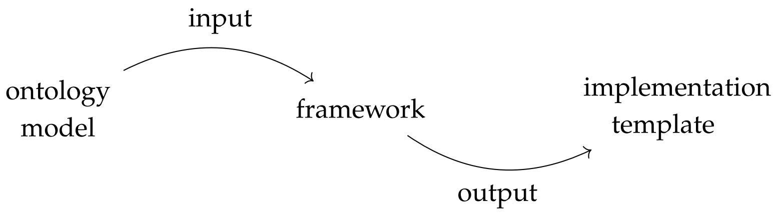


Figure 8.1: MAGO-Ag framework workflow

8.1 Framework Design and Description

The key requirements of the framework are, therefore, the following.

- `Agent` subconcepts must be translatable into classes extending the `Agent` class of SPADE. Every SPADE agent must connect to an extensible messaging and presence protocol (XMPP) server in order to be able to communicate with other agents. To connect to an XMPP server, the agent must have a `name` and the address of the XMPP server it is connecting to. Furthermore, individuals of the `Agent` must be translated into objects of the appropriate agent class defined in SPADE.

- Behaviour individuals can usually be found in the extension of one of the six subconcepts of the Behaviour concept. These individuals must be implemented by extending the appropriate class defined in SPADE. Since behaviour implementations highly depend on the intended use of the system and the agents therein, various details of the actual implementation of behaviour are not planned to be a part of the implementation template generated by this framework. Therefore, Behaviour individuals are expected to be translated only to the point of a defined behaviour class that can be instantiated by individual agents. One key observation is that agents, by default, know no behaviours. Instead, they learn about the available behaviours by playing, i.e. enacting, different Role individuals. Individuals of the Role concept are planned to be implemented in a way that is accessible by an agent, e.g. as a value of their internal attribute.

- By default, SPADE agents communicate using the XMPP protocol that requires a connection to an active XMPP server. Therefore, every agent must be connected to exactly one individual of the Agent Host Server concept. This concept must provide the host the Agent individual has to connect to, while the other part of the Jabber identifier (JID), the name, is provided by the Agent individual itself.

The MAGO-Ag ontology provides the basic concepts for the framework to translate. However, the framework must be able to work with additional subconcepts introduced to the Agent concept. This requirement stems from the need to allow the system modeller to create agent classes and their individual agents. Furthermore, the MAGO-Ag framework must work with individuals, even though treating individuals of Agent concept is expected to be different to how individuals of the Behaviour concept are treated; individual behaviours should be implemented as behaviour classes that will be instantiated by individual agents, while individual agents are instances of the applicable Agent class. Ultimately, extending the framework to include additional concepts that may be introduced to the related ontology in the future should not be extremely difficult.

Example 8.1.

◁ example 8.1. Subconcepts to the Agent concept

Subconcepts of the Agent concept can be Agent Factory and Agent Recipe in the domain where recipe agents consume a subset of the set of services provided by factory agents, which is, in turn, a subset of the system-wide set of possible services.

Finally, the framework should be implemented to provide the user with its functionality without requiring extensive programming or SPADE knowledge. In other words, the framework must be easy to run and provide the results straightforwardly.

8.2 *Framework Implementation*

The framework was developed in stages; each focused on one of the concepts that must be translated into the implementation template. Several Python classes are developed, to help the translation process.

Thing

Thing was the initial entity to be developed. The main idea of the `Thing` class was to create a set of methods and properties that will be common to all the classes used for translating the ontology to the appropriate implementation template. The `Thing` class is to be extended by the other classes participating in the translation process. Therefore, the class implementation includes the following key elements.

- Some basic values as class properties are stored that are planned to be available to and used by the other classes.
- Common methods for the following purposes are provided:
  - setting up the string template for the resulting implementation template;
  - rendering the implementation template via substituting the placeholder values in the implementation string template;
  - retrieving the implementation template if it is already rendered, whenever needed;
  - writing the rendered implementation template in a file on the local disk.

`set_implementation_template`
👁 see listing 8.1

`render_implementation`
👁 see listing 8.2

`get_implementation`
👁 see listing 8.3

`write_implementation_to_file`
👁 see listing 8.4

```python
def set_implementation_template(self, implementation_template: str):
    """Set the implementation template, following the string.Template syntax. This template is used to generate
    ↪  implementation of the object.

    Args:
        implementation_template (str): The implementation template to be filled in with appropriate values of objects
        ↪  of this class. Has to follow string.Template syntax.
    """
    self.implementation_template = Template(implementation_template)
    self.implementation = None
```

Listing 8.1: Implementation of the `set_implementation_template` method of the `Thing` class

```python
def render_implementation(self, substitutes: dict = None):
    """Fill in the provided implementation template with data. If no `substitutes` value is provided, attributes of
    ↪  the object are used (those must have the same names as the variables in the template string).

    Args:
        substitutes (dict, optional): A dictionary of the values to be used in the provided template. Defaults to
        ↪  None.

    Raises:
        ValueError: Error is raised if no template was set.
    """
    if not self.implementation_template:
        raise ValueError("No implementation template set.")

    substitutes = self.__dict__ if not substitutes else substitutes
    self.implementation = self.implementation_template.substitute(
        substitutes
    ).strip()
```

Listing 8.2: Implementation of the `render_implementation` method of the `Thing` class

```python
def get_implementation(self):
    """Return the rendered implementation.

    Returns:
        Returns the rendered implementation template or None if unavailable.
    """
    return self.implementation if self.implementation else None
```

Listing 8.3: Implementation of the `get_implementation` method of the `Thing` class

```python
def write_implementation_to_file(self, file_name: str = None):
    """Save the rendered implementation to a file.

    Args:
        file_name (str, optional): Name of the file to be written. If not provided, will be rendered based on the name of the class (`self.__class__.__name__`) and type of entity (`self.entity_type`). Defaults to None.
    """
    if not self.implementation:
        self.render_implementation()
    if not file_name:
        file_name = os.path.join(
            os.getcwd(),
            "Template",
            f"{self.__class__.__name__}_{self.entity_type}.py",
        )
    with open(file_name, "w") as file:
        file.write(self.implementation)

    logging.info(
        f"Implementation of {self.onto_individual if self.onto_individual else self.name} saved to {file_name}."
    )
```

Listing 8.4: Implementation of the `write_implementation_to_file` method of the `Thing` class

`Agent`

Agent was the following entity to be developed. The `Agent` class is developed to be the extension of the `Thing` class. In addition to the method for setting up the rendering string template, the `Agent` class features three other methods with the following functionalities.

`get_related_roles_and_behaviours`

see listing 8.5

- In order to provide the `Agent` individual with the knowledge of roles and the behaviours they enable, role and behaviour combinations are retrieved, constrained to the roles available to the individual agent.

`render_agent_instantiation`

see listing 8.6

- SPADE agents are instances of their respective agent classes. Therefore, the `Agent` class provides the method for rendering a part of the agent instantiation code.

`render_agent_import`

see listing 8.7

- Since agent individuals may pertain to custom agent subconcepts, it is necessary to allow their successful translation to implementation templates. This is why the `Agent` class provides the method for rendering the import statement for a particular agent.

```python
def get_related_roles_and_behaviours(self):
    result = {}

    result.update(
        {
            role.iri: {"name": role.has_name, "behaviours": {}}
            for role in self.onto_individual.can_play_role
        }
    )

    for role in self.onto_individual.can_play_role:
        result.get(role.iri).get("behaviours").update(
            {
                behaviour.iri: behaviour.has_name
                for behaviour in role.provides_behaviour
            }
        )

    return result if result else None
```

Listing 8.5: Implementation of the `get_related_roles_and_behaviours` method of the `Agent` class

```
    def render_agent_instantiation(self):
        self.related_roles_and_behaviours = self.get_related_roles_and_behaviours()

        agent_instantiation_template = """
agent = $agent_type("$name@$host_server", "$password")
agent.uri = "$uri"
agent.knowledge_artefact_uris = $knowledge_artefact_uris
agent.available_roles_and_behaviours = $related_roles_and_behaviours
agent.system_features = $system_features
agent_individuals.setdefault("$host_server", {}).update({"$name": agent})
"""
        self.set_implementation_template(agent_instantiation_template)

        self.render_implementation()
        return self.get_implementation()
```

Listing 8.6: Implementation of the `render_agent_instantiation` method of the `Agent` class

```
    def render_agent_import(self):
        self.set_implementation_template(
            """
from Agent_$agent_type import *
"""
        )
        self.render_implementation()
        return self.get_implementation()
```

Listing 8.7: Implementation of the `render_agent_import` method of the `Agent` class

Behaviour is the next entity to be developed. The `Behaviour` class is developed to be the extension of the `Thing` class as well.

`Behaviour`

Since all the `Behaviour` individuals are ultimately individuals of the `Behaviour` concept in the ontology, and their further classification is performed via the applied reasoning processes since their respective subconcepts are implemented as defined classes, their specific type (i.e. cyclic, periodic, one-shot, timeout, finite state machine, or state) is determined using a specific method of the `Behaviour` class based on their individual's data property values.

`determine_behaviour_type`
👁 see listing 8.8

`Finite State Machine Behaviour` is the most complex behaviour to translate. Even though the behaviour class is implemented in the same manner the other behaviour classes are implemented, the finite state machine (FSM) behaviour must have all the states and their transitions set up too. The process of setting up the initial state, the other states, and their respective transitions requires traversing through the individuals of the ontology. Therefore, an extra method was developed as a part of the behaviours of the `Finite State Machine Behaviour` concept that will be used to set up the behaviour in runtime.

`get_fsm_states`
👁 see listing 8.9

`render_fsm_implementation`
👁 see listing 8.10

```
def determine_behaviour_type(self):
    """Determines the type of behaviour based on the ontology individual and agent attributes.

    Returns:
        str: The name of the behaviour type.
    """
    if self.onto_individual.is_before_state or self.onto_individual.is_after_state:
        return "State"
    elif self.onto_individual.has_initial_state:
        return "FSMBehaviour"
    else:
        key = (bool(self.cycling), bool(self.period))
        behaviour_mapping = {
            (False, False): "OneShotBehaviour",
            (False, True): "TimeOutBehaviour",
            (True, False): "CyclicBehaviour",
            (True, True): "PeriodicBehaviour",
        }
        return behaviour_mapping.get(key, "UnknownBehaviour")
```

Listing 8.8: Implementation of the `determine_behaviour_type` method of the `Behaviour` class

Strategy concepts are the last entities to be developed. The `Plan` class is developed to be the extension of the `Thing` class again.

`Strategy`

```
def get_fsm_states(self):
    states = set()
    transitions = []
    visited = set()
    initial_state = None

    initial_state = self.onto_individual.has_initial_state

    stack = list(initial_state)
    while stack:
        current_state = stack.pop()
        if current_state in visited:
            continue
        visited.add(current_state)
        states.add(current_state)

        # Get next states
        next_states = current_state.is_before_state
        transitions.extend([(current_state, ns) for ns in next_states])
        stack.extend(next_states)

    return initial_state, states, transitions
```

Listing 8.9: Implementation of the `get_fsm_states` method of the `Behaviour` class

```
def render_fsm_implementation(self):
    initial_state, states, transitions = self.get_fsm_states()

    implementation = []

    state_names = {}
    state_names.update(
        {
            state: (
                clean_string(state.has_name)
                if state.has_name
                else clean_string(state.name)
            )
            for state in states
        }
    )

    print(state_names)

    for state in states:
        state_name = state_names.get(state)
        class_name = f"{state_name}()"
        is_initial = state in initial_state
        if is_initial:
            code_line = f"self.add_state(name='{state_name}', state={class_name}, initial=True)"
        else:
            code_line = f"self.add_state(name='{state_name}', state={class_name})"
        implementation.append(code_line)

    for source_state, dest_state in transitions:
        source_name = state_names.get(source_state)
        dest_name = state_names.get(dest_state)
        code_line = (
            f"self.add_transition(source='{source_name}', dest='{dest_name}')"
        )
        implementation.append(code_line)

    # Join the code lines into a single string
    return textwrap.indent(text="\n".join(implementation), prefix="        ")
```

Listing 8.10: Implementation of the `render_fsm_implementation` method of the `Behaviour` class

The goal of translating the strategy-related concepts into the implementation template is to provide agent instances with the knowledge from the ontology in a way that is more Python-friendly. Furthermore, by removing the necessity of having the ontology available to agent instances, the implementation template is rendered as a more independent system. This decision introduces some other constraints, though, such as no reasoning over the knowledge rendered as data. On the other hand, access to an ontology can still be given to an agent using the `Knowledge Artefact` concept.

The strategy-related concepts are translated using a single method that translates the related `Plan`, `Action`, `Behaviour`, and `Objective` individuals into a Python dictionary retaining their respective connections.

`get_plan_action_behaviour_objective`

👁 see listing 8.11

```python
def get_plan_action_behaviour_objective(self):
    plan_dict = {}

    plan_iri = self.onto_individual.iri

    # Get the plan name
    plan_name = (
        self.onto_individual.has_name
        if self.onto_individual.has_name
        else self.onto_individual.name
    )

    # Initialize the plan entry
    plan_entry = {"name": plan_name, "actions": {}}

    # Get the actions associated with the plan
    actions = self.onto_individual.requires_action
    for action in actions:
        action_iri = action.iri
        # Get the action name
        action_name = action.has_name if action.has_name else action.name

        # Initialize the action entry
        action_entry = {"name": action_name, "objectives": {}, "behaviours": {}}

        # Get the objectives associated with the action
        objectives = action.has_objective
        for objective in objectives:
            objective_iri = objective.iri
            # Get the objective name
            objective_name = (
                objective.has_name if objective.has_name else objective.name
            )

            # Add the objective to the action's objectives
            action_entry["objectives"][objective_iri] = objective_name

        # Get the behaviours associated with the action
        behaviours = action.has_behaviour
        for behaviour in behaviours:
            behaviour_iri = behaviour.iri
            # Get the behaviour name
            behaviour_name = (
                behaviour.has_name if behaviour.has_name else behaviour.name
            )

            # Add the behaviour to the action's behaviours
            action_entry["behaviours"][behaviour_iri] = behaviour_name

        # Add the action entry to the plan's actions
        plan_entry["actions"][action_iri] = action_entry

    # Add the plan entry to the main dictionary
    plan_dict[plan_iri] = plan_entry

    logging.info(f"Plan {plan_name} visited.")

    return plan_dict
```

Listing 8.11: Implementation of the `get_plan_action_behaviour_objective` method of the `Plan` class

Knowledge artefact individuals are provided to the connected agents via their unique resource identifiers (URIs). Each agent is provided with a property of data type dictionary, consisting of artefacts' names and URIs.

`get_related_knowledge_artefact_uris`

👁 see listing 8.12

Workspace is the last entity to be developed. The Python class for this concept is used to perform most of the necessary rendering and

```python
def get_related_knowledge_artefact_uris(onto_individual) -> dict:
    """Retrieves all the knowledge artefacts the individual can access.

    Args:
        onto_individual (Agent individual): The agent individual.

    Returns:
        dict: Dictionary with artefact names and URIs.
    """

    artefacts = set()
    artefacts.update(onto_individual.can_access_artefact)
    organisational_artefacts = [
        role.can_access_artefact
        for role in onto_individual.can_play_role
    ]
    artefacts.update(artefact for artefacts in organisational_artefacts for artefact in artefacts)

    artefact_names_uris = {
        artefact.has_name: artefact.has_uri
        for artefact in artefacts
        if "Knowledge" in str(artefact.is_a[0].label[0])
    }

    return artefact_names_uris
```

Listing 8.12: Implementation of the `get_related_knowledge_artefact_uris` function

writing to file operations of the related concepts and individuals. This class contains all the methods necessary for getting all the data from the ontology and rendering the implementation templates. While writing rendered implementation templates to files is mostly performed by calling the related methods of the `Thing` class, the following more interesting methods are used to retrieve data from the ontology containing the modelled MAS:

- `read_agents_from_ontology` implements how to read the ontology, retrieve the most important data related to agents, and prepare those data for rendering agent implementation templates;

see listing 8.13

- `render_behaviours_from_ontology` renders behaviour implementation templates based on their ontology data

see listing 8.14

- `read_plan_from_ontology` takes all the `Plan` individuals and renders their related data as a Python dictionary.

see listing 8.15

```python
def read_agents_from_ontology(self, onto: Ontology = None):
    if onto is None:
        onto = self.onto

    agent_classes = onto.search(subclass_of=onto.search(label="Agent"))
    for agent_class in agent_classes:
        agents = onto.search(type=agent_class)
        self.agents.setdefault(agent_class.label[0], []).extend(
            [
                Agent(
                    agent_type=agent.is_a[0].label[0],
                    host_server=agent.lives_on_host[0].label[0],
                    name=agent.has_name,
                    password="tajna",
                    onto_individual=agent,
                )
                for agent in agents
            ]
        )
```

Listing 8.13: Implementation of the `read_agents_from_ontology` method of the `Workspace` class

Translation script is the very last element that must be added to the mix, in order to make the framework easily runnable and usable. The main translation script starts by reading the ontology using the `owlready2` library, then instantiates the `Workspace` object,

see listing 8.16

```
def render_behaviours_from_ontology(self, onto: Ontology = None):
    if onto is None:
        onto = self.onto

    has_name = onto.search_one(label="has name")

    behaviours = onto.search(type=onto.search_one(label="Behaviour"))

    behaviours_mago = [
        Behaviour(
            cycling=behaviour.is_repeating,
            period=behaviour.has_period if behaviour.has_period else None,
            onto_individual=behaviour,
        )
        for behaviour in behaviours
    ]

    self.behaviours_rendered = "\n\n\n".join(
        [
            behaviour.render_behaviour_implementation()
            for behaviour in behaviours_mago
        ]
    )

    return self.behaviours_rendered
```

Listing 8.14: Implementation of the `render_behaviours_from_ontology` method of the `Workspace` class

```
def read_plan_from_ontology(self, onto: Ontology = None):
    if onto is None:
        onto = self.onto

    result = {}

    plan = onto.search_one(
        label="Plan"
    ).instances()

    for plan in plan:
        result.update(
            Plan(
                onto_individual=plan
            ).get_plan_action_behaviour_objective()
        )

    return result
```

Listing 8.15: Implementation of the `read_plan_from_ontology` method of the `Workspace` class

and by calling the `write_implementation_to_disk` of the instantiated `World` object finally runs the translation process. Thus, the ontology is consulted, the necessary string templates are rendered, and the implementation template is written to respective files.

```python
import os
import argparse
from owlready2 import World, Ontology, onto_path, set_render_func, sync_reasoner
from mago_workspace import Workspace


def render_using_label(entity):
    return entity.label.first() or entity.name


def render_using_iri(entity):
    return entity.iri


def main(ontology_name="MAGO-Ag.owx"):
    onto_path.append(os.getcwd())

    mago_world = World()
    onto: Ontology = mago_world.get_ontology(ontology_name).load(reload=True)

    set_render_func(render_using_label)

    sync_reasoner()

    template_folder = os.path.join(os.getcwd(), "Template")
    if not os.path.exists(template_folder):
        os.makedirs(template_folder)

    aMAGOWorld = Workspace(ontology=onto, name="World")
    aMAGOWorld.write_implementation_to_disk()


if __name__ == "__main__":
    parser = argparse.ArgumentParser(
        description="MAGO-Ag ontology translation framework"
    )
    parser.add_argument(
        "-o",
        "--ontology-name",
        type=str,
        default="MAGO-Ag.owx",
        help="Optional name of the ontology file to be used, string argument with default value 'MAGO-Ag.owx'",
    )

    args = parser.parse_args()
    main(args.ontology_name)
```

Listing 8.16: The main script of the framework

9
A Case Study: The RecipeWorld

The context of choice for this case study, which shows how the developed framework works, is the RecipeWorld, an agent based modelling (ABM) model developed by Fontana and Terna [6].

the RecipeWorld

In RecipeWorld, networks arise naturally from the interactions of autonomous agents, whereas each agent is engaged in a series of steps – just like following a recipe – towards a specific goal. It depicts a model wherein each agent operates independently, yet their actions contribute to creating a more extensive, interconnected network. This world is not pre-planned or engineered to form a network; instead, the network takes shape as a by-product of agents working through ordered sequences of tasks, much like individual workers assembling parts on a production line.

Each agent in RecipeWorld has a role, capabilities, and purpose, represented by 'orders' that contain technical instructions or objectives. To achieve these objectives, agents execute 'recipes,' which are sequences of steps. For example, a recipe agent might need to perform one action before another, or it might be required to complete several tasks simultaneously, similar to how ingredients are combined in specific ways when following a recipe in cooking or a sequence of steps in factory production. These steps, or events, activate agents, guiding their interactions with other agents and forming links between them. As recipe agents with their assigned tasks move from agent to agent, the network grows dynamically, mimicking the real-world flow of goods in production or information in communication.

Over time, the network emerges out of the repetitive yet purposeful actions of agents. These links can strengthen with each repeated interaction, highlighting frequently used connections, while weaker connections fade. As the network is established, it reflects the standard actions of its agents.

Example 9.1.

◃ example 9.1. Service providers and consumers

> Consider a network of factories (i.e. 'service providers') where each unit specialises in a particular task within a production sequence – we shall call this specialisation an offered 'service'. Recipe agents (i.e. 'service consumers') move from one factory to another based on their respective recipes and their required services. This com-

> motion enables connections to be formed based on the sequence of steps completed.

RecipeWorld’s organic network creation has broader implications. It offers a neat way to test ideas in a simulated environment, observing how policies and other features might affect interconnected systems, from production lines to social services. Moreover, RecipeWorld’s networks can help reverse-engineer observed real-world networks, inferring the underlying behaviours from which they emerged. Rather than setting out to control or shape the network from above, RecipeWorld allows networks to emerge naturally from the ground up, showing how order, complexity, and efficiency can spontaneously develop from individual agents’ simple, goal-oriented actions.

9.1 *The Ontology*

The example ontology comprises two particular types of agents: `Factory Agent` and `Recipe Agent` . These two subconcepts of the `Agent` concept reflect the service-consuming and service-providing agents of the RecipeWorld model described above. The three provided individuals per the above mentioned concept are enough to showcase how the agent individuals are translated to their respective implementation templates.

There are two `Plan` individuals in the example ontology. Each of those is related to a set of `Action` individuals that can be used to reach specified `Objective` individuals and are implemented using the connected `Behaviour` individuals. Furthermore, three `Role` individuals are defined, some of them providing specific `Behaviour` individuals. Individual `Agent` individuals are connected to specific `Role` individuals. Lastly, there are several `Behaviour` individuals in the ontology, some of them modelled as `State` individuals related to `Finite State Machine Behaviour` individual.

see figure 9.1

see figure 9.2

see figure 9.3

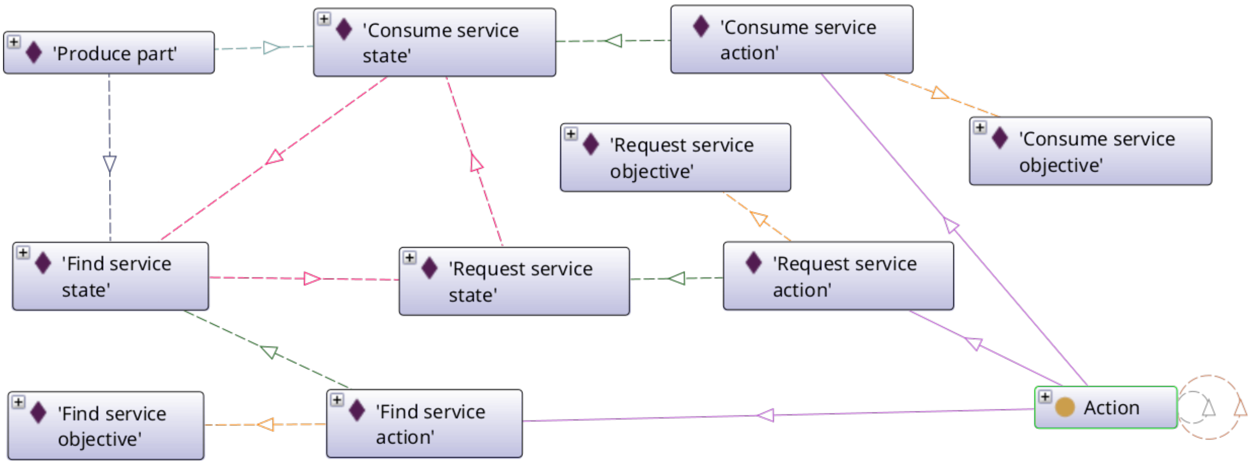


Figure 9.1: Selected individuals related to the `Produce part` individual of concept `Plan`

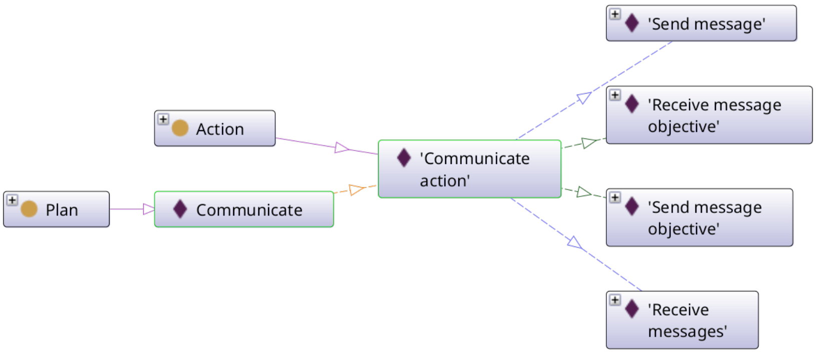


Figure 9.2: Selected individuals related to the `Communicate` individual of concept `Plan`

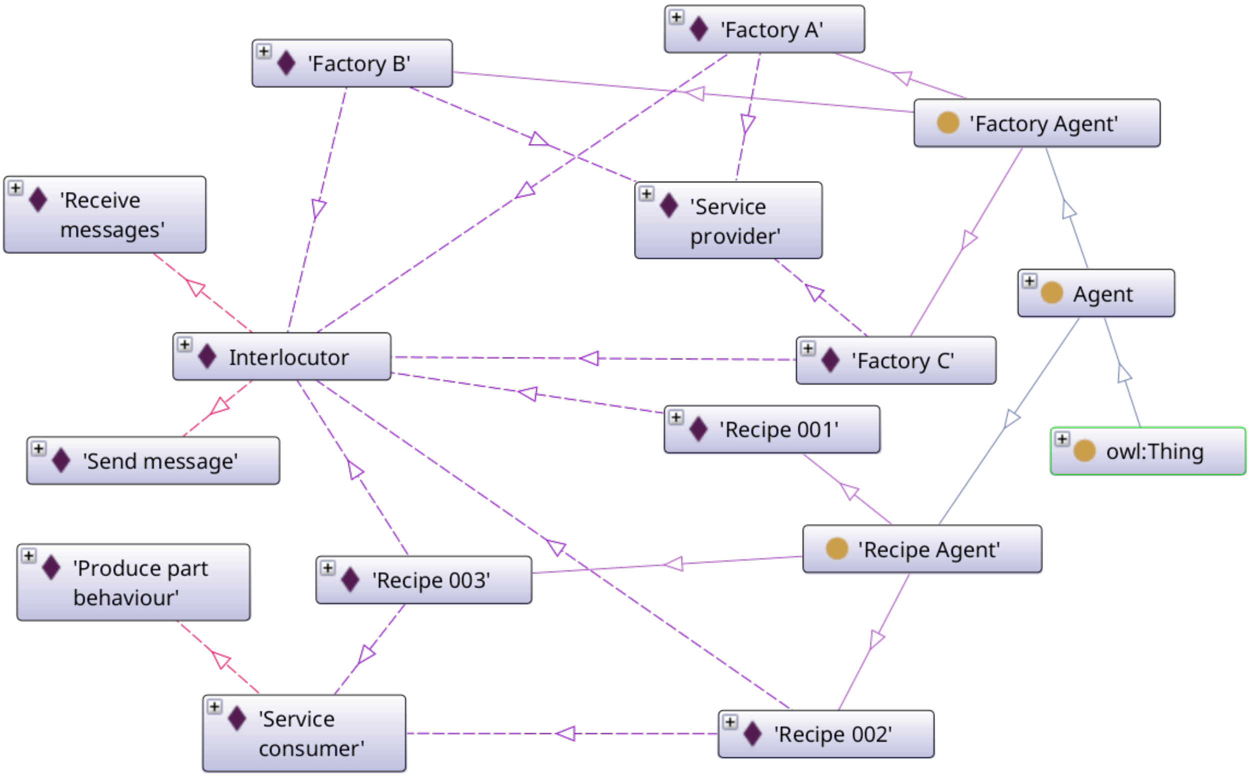


Figure 9.3: Agents of the example and their connected roles and behaviours

The modelled system is translated into the implementation template consisting of the following four files:

- `Agent_Factory_Agent.py` is the place where the `Factory Agent` concept is translated into a SPADE agent class. see listing 9.1
- `Agent_Recipe_Agent.py` is the place where the `Recipe Agent` concept is translated into a SPADE agent class. see listing 9.2
- `Behaviours.py` is where all the `Behaviour` individuals' implementation templates are located. see listing 9.3
- `Worksapce_workspace.py` contains agent instantiations and should be run if the modelled and now translated system is to be run. see listing 9.4

Listing 9.1: The `Agent_Factory_Agent.py` file

```python
from itertools import chain

from owlready2 import World
from spade.agent import Agent
from Behaviours import *


class Factory_Agent(Agent):

    def execute_sparql(self, world: World=None, query: str=None, parameters: list=None) -> list:
        """Execute a SPARQL query in the provided owlready2 World instance.

        Args:
            world (World): An owlready2 World instance containing the relevant data.
            query (str): The SPARQL query to be executed. Parameters are designated as `??`.
            parameters (list, optional): Parameters to be sequentially provided to the query. Defaults to None.

        Returns:
            list: The result of the query.
        """
        if world is None:
            world = self.world

        if query is None:
            ValueError("Query string must be provided.")

        prepared_query = world.prepare_sparql(sparql=query)
        column_names = [name.replace("?", "") for name in prepared_query.column_names]

        query_res = prepared_query.execute(params=parameters)
        query_res = [dict(zip(column_names, result)) for result in query_res]

        return query_res

    async def setup(self):
        print(f"{self.name}: New agent running.")

        self.knowledge_artefacts = {}

        for ka_name, ka_uri in self.knowledge_artefact_uris.items():
            world = World()
            self.knowledge_artefacts.setdefault(ka_name, {}).update({
                "world": world,
                "ontology": world.get_ontology(ka_uri).load(reload=True)
            })

        if self.available_roles_and_behaviours is not None:
            self.available_roles = list([entry.get("name") for entry in self.available_roles_and_behaviours.values()])
            self.available_behaviours = list(set(chain(*[entry.get("behaviours", {}).values()
              for entry in self.available_roles_and_behaviours.values()])))
        else:
            self.available_roles = None
            self.available_behaviours = None

        print(self.name, self.available_roles, self.available_behaviours, self.knowledge_artefacts)

        self.world = self.knowledge_artefacts.get("Main ontology").get("world")
        self.onto_individual = self.world.search_one(iri=self.uri)
```

Listing 9.2: The `Agent_Recipe_Agent.py` file

```python
from itertools import chain

from owlready2 import World
from spade.agent import Agent
from Behaviours import *


class Recipe_Agent(Agent):

    def execute_sparql(self, world: World=None, query: str=None, parameters: list=None) -> list:
        """Execute a SPARQL query in the provided owlready2 World instance.

        Args:
            world (World): An owlready2 World instance containing the relevant data.
            query (str): The SPARQL query to be executed. Parameters are designated as `??`.
            parameters (list, optional): Parameters to be sequentially provided to the query. Defaults to None.

        Returns:
            list: The result of the query.
        """
        if world is None:
            world = self.world

        if query is None:
            ValueError("Query string must be provided.")

        prepared_query = world.prepare_sparql(sparql=query)
        column_names = [name.replace("?", "") for name in prepared_query.column_names]

        query_res = prepared_query.execute(params=parameters)
        query_res = [dict(zip(column_names, result)) for result in query_res]

        return query_res

    async def setup(self):
        print(f"{self.name}: New agent running.")

        self.knowledge_artefacts = {}

        for ka_name, ka_uri in self.knowledge_artefact_uris.items():
            world = World()
            self.knowledge_artefacts.setdefault(ka_name, {}).update({
                "world": world,
                "ontology": world.get_ontology(ka_uri).load(reload=True)
            })

        if self.available_roles_and_behaviours is not None:
            self.available_roles = list([entry.get("name") for entry in self.available_roles_and_behaviours.values()])
            self.available_behaviours = list(set(chain(*[entry.get("behaviours", {}).values()
              for entry in self.available_roles_and_behaviours.values()])))
        else:
            self.available_roles = None
            self.available_behaviours = None

        print(self.name, self.available_roles, self.available_behaviours, self.knowledge_artefacts)

        self.world = self.knowledge_artefacts.get("Main ontology").get("world")
        self.onto_individual = self.world.search_one(iri=self.uri)
```

```
from spade.behaviour import *

class Register(CyclicBehaviour):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def run(self) -> None:
        print("Running the behaviour.")


class Request_service(State):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def run(self) -> None:
        print("Running the behaviour.")


class Receive_messages(CyclicBehaviour):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def run(self) -> None:
        print("Running the behaviour.")


class Send_message(CyclicBehaviour):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def run(self) -> None:
        print("Running the behaviour.")


class Consume_service(State):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def run(self) -> None:
        print("Running the behaviour.")


class Find_service(State):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def run(self) -> None:
        print("Running the behaviour.")


class Produce_part(FSMBehaviour):

    async def on_start(self) -> None:
        print("Starting behaviour.")

    async def on_end(self) -> None:
        print("Ending behaviour.")

    async def state_setup(self):
        self.add_state(name='Find_service', state=Find_service(), initial=True)
        self.add_state(name='Request_service', state=Request_service())
        self.add_state(name='Consume_service', state=Consume_service())
        self.add_transition(source='Find_service', dest='Request_service')
        self.add_transition(source='Request_service', dest='Consume_service')
        self.add_transition(source='Consume_service', dest='Find_service')
```

Listing 9.3: The `Behaviours.py` file

```
import spade
from Agent_Factory_Agent import *
from Agent_Recipe_Agent import *


async def main():
    agent_individuals = {}
    agent = Factory_Agent("Osijek@localhost", "tajna")
    agent.uri = "http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d910535c_5b5e_4af4_b95f_e6dab7023ddd"
    agent.knowledge_artefact_uris = {'Personal knowledge': 'https://ai.foi.hr/PersonalOntology.owx', 'The other
    ↪  ontology': 'https://ai.foi.hr/MAGO-Ag.owx'}
    agent.available_roles_and_behaviours =
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b': {'name': 'Service
    ↪  provider', 'behaviours': {}},
    ↪  'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c': {'name': 'Scout',
    ↪  'behaviours': {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55':
    ↪  'Receive messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
    ↪  'Send message'}}}
    agent.system_features = None
    agent_individuals.setdefault("localhost", {}).update({"Osijek": agent})
    agent = Factory_Agent("Rijeka@localhost", "tajna")
    agent.uri = "http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_f892df91_9c0d_4345_9dcd_e7aa86062631"
    agent.knowledge_artefact_uris = {'The other ontology': 'https://ai.foi.hr/MAGO-Ag.owx'}
    agent.available_roles_and_behaviours =
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b': {'name': 'Service
    ↪  provider', 'behaviours': {}},
    ↪  'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c': {'name': 'Scout',
    ↪  'behaviours': {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55':
    ↪  'Receive messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
    ↪  'Send message'}}}
    agent.system_features = None
    agent_individuals.setdefault("localhost", {}).update({"Rijeka": agent})
    agent = Factory_Agent("Zagreb@localhost", "tajna")
    agent.uri = "http://dragon.foi.hr/mago-a.owx#Broj%20jedan"
    agent.knowledge_artefact_uris = {'Personal knowledge': 'https://ai.foi.hr/PersonalOntology.owx', 'Main ontology':
    ↪  'https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx',
    ↪  'The other ontology': 'https://ai.foi.hr/MAGO-Ag.owx'}
    agent.available_roles_and_behaviours =
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b': {'name': 'Service
    ↪  provider', 'behaviours': {}},
    ↪  'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c': {'name': 'Scout',
    ↪  'behaviours': {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55':
    ↪  'Receive messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
    ↪  'Send message'}}}
    agent.system_features = {'money': 100, 'recipe': [2, 3, 9]}
    agent_individuals.setdefault("localhost", {}).update({"Zagreb": agent})
    agent = Recipe_Agent("Recipe_003@localhost", "tajna")
    agent.uri = "http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_12c619ee_d747_4d66_8165_25991df28f70"
    agent.knowledge_artefact_uris = {'Personal knowledge': 'https://ai.foi.hr/PersonalOntology.owx', 'Main ontology':
    ↪  'https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx',
    ↪  'The other ontology': 'https://ai.foi.hr/MAGO-Ag.owx'}
    agent.available_roles_and_behaviours =
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228': {'name':
    ↪  'Warrior', 'behaviours':
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_7bdd32fb_271c_4f2c_aafd_d20c74aa22b9': 'Produce part'}},
    ↪  'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c': {'name': 'Scout',
    ↪  'behaviours': {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55':
    ↪  'Receive messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
    ↪  'Send message'}}}
    agent.system_features = None
    agent_individuals.setdefault("localhost", {}).update({"Recipe_003": agent})
    agent = Recipe_Agent("Recipe_001@localhost", "tajna")
    agent.uri = "http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_9e8b35f2_d0ca_42f0_82dd_5dd72cf9e7c4"
    agent.knowledge_artefact_uris = {'The other ontology': 'https://ai.foi.hr/MAGO-Ag.owx'}
    agent.available_roles_and_behaviours =
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c': {'name': 'Scout',
    ↪  'behaviours': {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55':
    ↪  'Receive messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
    ↪  'Send message'}}}
    agent.system_features = None
    agent_individuals.setdefault("localhost", {}).update({"Recipe_001": agent})
    agent = Recipe_Agent("Recipe_002@localhost", "tajna")
    agent.uri = "http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt1"
    agent.knowledge_artefact_uris = {'Main ontology':
    ↪  'https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx',
    ↪  'The other ontology': 'https://ai.foi.hr/MAGO-Ag.owx'}
    agent.available_roles_and_behaviours =
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228': {'name':
    ↪  'Warrior', 'behaviours':
    ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_7bdd32fb_271c_4f2c_aafd_d20c74aa22b9': 'Produce part'}},
    ↪  'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c': {'name': 'Scout',
    ↪  'behaviours': {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55':
    ↪  'Receive messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
    ↪  'Send message'}}}
    agent.system_features = None
    agent_individuals.setdefault("localhost", {}).update({"Recipe_002": agent})
    for agent in [agent for host_dict in agent_individuals.values() for agent in host_dict.values()]:
        agent.plan_action_behaviour_objective =
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_4ce555df_0daa_451c_8cee_3869fb46f599': {'name':
        ↪  'Produce plan', 'actions':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_395d25cd_dd02_43da_9c3e_c558a17cf629': {'name':
        ↪  'Request service', 'objectives':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7': 'Request
        ↪  service'}, 'behaviours':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_44e0578c_c818_4c97_a1cf_081d31543933': 'Request
        ↪  service'}}, 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5':
        ↪  {'name': 'Find service', 'objectives':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac': 'Find
        ↪  service'}, 'behaviours':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246': 'Find
        ↪  service'}}, 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_af1dbcdc_a9f4_4c0d_87b1_ecb111fca6ea':
        ↪  {'name': 'Consume service', 'objectives':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d37cd96b_e3f8_41e6_860a_821b97903d66': 'Consume
        ↪  service'}, 'behaviours':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b': 'Consume
        ↪  service'}}}}, 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_df91a3a0_7a62_46fb_ac02_75dde4d82493':
        ↪  {'name': 'Manufacturing Plan', 'actions':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_42465976_d81c_4bd1_8ed9_6a94dbd2958f': {'name':
        ↪  'Communicate', 'objectives':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_1db58781_1969_4eb6_8b3c_24f81f921806':
        ↪  'OWLNamedIndividual_1db58781_1969_4eb6_8b3c_24f81f921806',
        ↪  'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ab6715e3_2bf2_4b4e_a00a_45a0ae4a8428':
        ↪  'OWLNamedIndividual_ab6715e3_2bf2_4b4e_a00a_45a0ae4a8428'}, 'behaviours':
        ↪  {'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55': 'Receive
        ↪  messages', 'http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_cedc48c8_e579_4079_abbb_a9ba8a420fcd':
        ↪  'Send message'}}}}}
        await agent.start()

spade.run(main())
```

Listing 9.4: The `Workspace_workspace.py` file

Part III

Appendices

MAGO-Ag Turtle Serialization

```
@prefix : <http://dragon.foi.hr/mago-a.owx#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix owlr: <http://www.lesfleursdunormal.fr/static/_downloads/owlready_ontology.owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base <http://dragon.foi.hr/mago-a.owx#> .

<http://dragon.foi.hr/mago-a.owx#> rdf:type owl:Ontology ;
                                   rdfs:comment "The ontology for MAGO research visit." .

#################################################################
#    Annotation properties
#################################################################

###  http://www.lesfleursdunormal.fr/static/_downloads/owlready_ontology.owl#python_name
owlr:python_name rdf:type owl:AnnotationProperty .


#################################################################
#    Object Properties
#################################################################

###  http://dragon.foi.hr/mago-a.owx#R7MvUIWpnOdfB3dxpRXTUAK
:R7MvUIWpnOdfB3dxpRXTUAK rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R7dQDUF81SOJC29Sctpy6aP ;
                         rdfs:range :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:label "implements action"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R7NTCPsMvJdxinMk2P28ppg
:R7NTCPsMvJdxinMk2P28ppg rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:label "has final objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R7o5TdA2HQOQxwVPrUJrLpq
:R7o5TdA2HQOQxwVPrUJrLpq rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:label "is before objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R84IuDzXJ7QCZi1bY6fzFxZ
:R84IuDzXJ7QCZi1bY6fzFxZ rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDm65h4GNQjimv0axMIRnMX ;
                         rdfs:range :R84KrWu3ZBbXCifW6B9GSmZ ;
                         owlr:python_name "requires_action" ;
                         rdfs:label "requires action"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R8ZWdA7HWDzIIwhQZrSwAIj
:R8ZWdA7HWDzIIwhQZrSwAIj rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         owl:inverseOf :RDGoFrovO0PhBCQGPkFeUuR ;
                         rdf:type owl:IrreflexiveProperty ;
                         rdfs:domain :RDgoxtmVWC61IdqtaNK9QBf ;
                         rdfs:range :RDgoxtmVWC61IdqtaNK9QBf ;
                         owlr:python_name "is_after_state" ;
                         rdfs:label "is after state"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R8ZXgNqWa8M3RnwigsFBxvc
:R8ZXgNqWa8M3RnwigsFBxvc rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R9iXau1ZAN6oZTRbdCgbmgd ;
                         rdfs:range :R9iXau1ZAN6oZTRbdCgbmgd ;
                         rdfs:label "features role"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R8nS5zlvvUfE9N1xZBYTpcG
:R8nS5zlvvUfE9N1xZBYTpcG rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                         owlr:python_name "has_objective" ;
                         rdfs:label "has objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R9QlFkpwG4P6QT3YaFGdAPM
:R9QlFkpwG4P6QT3YaFGdAPM rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:range :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:label "is after action"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R9Y2982rI3hZL4oBhv8Ohhv
:R9Y2982rI3hZL4oBhv8Ohhv rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDm65h4GNQjimv0axMIRnMX ;
                         rdfs:range :RDm65h4GNQjimv0axMIRnMX ;
                         rdfs:label "is part of process"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R9wYaBAe5LQpMMDI7zKHWhc
:R9wYaBAe5LQpMMDI7zKHWhc rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:range :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:label "is before action"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RB8ofKEO8zM8jcK0WetPpLI
:RB8ofKEO8zM8jcK0WetPpLI rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain [ rdf:type owl:Class ;
                                       owl:unionOf ( :R9iXau1ZAN6oZTRbdCgbmgd
                                                     :R9lwUrkbSftwsmlPcopLgzT
                                                   )
                                     ] ;
                         rdfs:range :RB4lINuYfn41lvT2d2GduUO ;
                         owlr:python_name "can_access_artefact" ;
                         rdfs:label "can access artefact"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RBOL8y2xs4VvKc6D0kikOup
:RBOL8y2xs4VvKc6D0kikOup rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R9iXau1ZAN6oZTRbdCgbmgd ;
                         rdfs:range :R9iXau1ZAN6oZTRbdCgbmgd ;
                         owlr:python_name "is_part_of_role" ;
                         rdfs:label "is part of role"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RBmpSay8yjDZmklOv6Mw1rc
:RBmpSay8yjDZmklOv6Mw1rc rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf :RCcC3SXJ5MvuHMJyzs8OpU6 ;
                         rdf:type owl:FunctionalProperty ;
                         rdfs:domain :R9lwUrkbSftwsmlPcopLgzT ;
                         rdfs:range :R9iXau1ZAN6oZTRbdCgbmgd ;
                         owlr:python_name "plays_role" ;
                         rdfs:label "plays role"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RBntfnmvB9JVuK1xvAe6Hm8
:RBntfnmvB9JVuK1xvAe6Hm8 rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:label "is after objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RC6rldf2VsWBaHtUobIELQL
:RC6rldf2VsWBaHtUobIELQL rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDMv7ayhi9xuRAzbfALLhKS ;
                         rdfs:range :RDgoxtmVWC61IdqtaNK9QBf ;
                         owlr:python_name "has_initial_state" ;
                         rdfs:label "has initial state"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCAv5UwFPy5OUQdjN61nXbw
:RCAv5UwFPy5OUQdjN61nXbw rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R7dQDUF81SOJC29Sctpy6aP ;
                         rdfs:range :R9iXau1ZAN6oZTRbdCgbmgd ;
                         owlr:python_name "is_provided_by_role" ;
                         rdfs:label "is provided by role"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCPuCz8AunmmX3PS6A0U0XL
:RCPuCz8AunmmX3PS6A0U0XL rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDMv7ayhi9xuRAzbfALLhKS ;
                         rdfs:range :RDgoxtmVWC61IdqtaNK9QBf ;
                         owlr:python_name "has_final_state" ;
                         rdfs:label "has final state"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCWrWcZCQ3lxj6jFw9zRJ6m
:RCWrWcZCQ3lxj6jFw9zRJ6m rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RB4lINuYfn41lvT2d2GduUO ;
                         rdfs:range :RChP69av2xqRwh6LALfwsKq ;
                         owlr:python_name "provides_resource" ;
                         rdfs:label "provides resource"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCcC3SXJ5MvuHMJyzs8OpU6
:RCcC3SXJ5MvuHMJyzs8OpU6 rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R9lwUrkbSftwsmlPcopLgzT ;
                         rdfs:range :R9iXau1ZAN6oZTRbdCgbmgd ;
                         owlr:python_name "can_play_role" ;
                         rdfs:label "can play role"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCkUIZx8f7TQwTrh3wmfyBD
:RCkUIZx8f7TQwTrh3wmfyBD rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:range :R7dQDUF81SOJC29Sctpy6aP ;
                         owlr:python_name "has_behaviour" ;
                         rdfs:label "is implemented using behaviour"@en-gb .


```

```
###  http://dragon.foi.hr/mago-a.owx#RCsnmcyqIe70G3lagH4y3bE
:RCsnmcyqIe70G3lagH4y3bE rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:label "is part of objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDGoFrovO0PhBCQGPkFeUuR
:RDGoFrovO0PhBCQGPkFeUuR rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdf:type owl:IrreflexiveProperty ;
                         rdfs:domain :RDgoxtmVWC61IdqtaNK9QBf ;
                         rdfs:range :RDgoxtmVWC61IdqtaNK9QBf ;
                         owlr:python_name "is_before_state" ;
                         rdfs:label "is before state"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDrm00K6AJVd5OJmoVCqPy6
:RDrm00K6AJVd5OJmoVCqPy6 rdf:type owl:ObjectProperty ;
                         rdfs:subPropertyOf owl:topObjectProperty ;
                         rdfs:domain :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                         rdfs:label "has initial objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RY56GiCpqJHON677qnE5sT
:RY56GiCpqJHON677qnE5sT rdf:type owl:ObjectProperty ;
                        rdfs:subPropertyOf owl:topObjectProperty ;
                        rdfs:domain :R9iXau1ZAN6oZTRbdCgbmgd ;
                        rdfs:range :R7dQDUF81SOJC29Sctpy6aP ;
                        owlr:python_name "provides_behaviour" ;
                        rdfs:label "provides behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#ReKEcwE01zu4hSOL1gSBu2
:ReKEcwE01zu4hSOL1gSBu2 rdf:type owl:ObjectProperty ;
                        rdfs:subPropertyOf owl:topObjectProperty ;
                        rdfs:domain :RDqoRvx1aDeStUw4zNRdR0o ;
                        rdfs:range :RDqoRvx1aDeStUw4zNRdR0o ;
                        rdfs:label "features objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RoDHPMWwB604EEv4qQbsGC
:RoDHPMWwB604EEv4qQbsGC rdf:type owl:ObjectProperty ;
                        rdfs:subPropertyOf owl:topObjectProperty ;
                        rdfs:domain :RDm65h4GNQjimv0axMIRnMX ;
                        rdfs:range :RDm65h4GNQjimv0axMIRnMX ;
                        rdfs:label "features process"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RpmDQcEqabFOwsZHv82QuP
:RpmDQcEqabFOwsZHv82QuP rdf:type owl:ObjectProperty ;
                        rdfs:subPropertyOf owl:topObjectProperty ;
                        rdfs:domain :R9lwUrkbSftwsmlPcopLgzT ;
                        rdfs:range :RCdOsGdA0yrGZdnzgQesQx7 ;
                        owlr:python_name "lives_on_host" ;
                        rdfs:label "lives on host"@en-gb .


#################################################################
#    Data properties
#################################################################

###  http://dragon.foi.hr/mago-a.owx#OWLDataProperty_13076287_43a5_47ba_80d4_aa0b2fcbf767
:OWLDataProperty_13076287_43a5_47ba_80d4_aa0b2fcbf767 rdf:type owl:DatatypeProperty ;
                                                      rdfs:range xsd:string ;
                                                      rdfs:label "is implemented as"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
:OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab rdf:type owl:DatatypeProperty ,
                                                               owl:FunctionalProperty ;
                                                      rdfs:domain [ rdf:type owl:Class ;
                                                                    owl:unionOf ( :RB4lINuYfn41lvT2d2GduUO
                                                                                  :RChP69av2xqRwh6LALfwsKq
                                                                                )
                                                                  ] ;
                                                      rdfs:range xsd:anyURI ;
                                                      owlr:python_name "has_uri" ;
                                                      rdfs:label "has URI"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
:OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49 rdf:type owl:DatatypeProperty ;
                                                      rdfs:domain :R7dQDUF81SOJC29Sctpy6aP ;
                                                      rdfs:range xsd:boolean ;
                                                      owlr:python_name "is_repeating" ;
                                                      rdfs:comment "This property defines whether the behaviour is
                                                      ↪  repeating or not, designating cyclic or one-shot behaviour
                                                      ↪  types." ;
                                                      rdfs:label "is repeating"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLDataProperty_8f71ec98_a3b3_4f5b_b39f_dab911dd0b9f
:OWLDataProperty_8f71ec98_a3b3_4f5b_b39f_dab911dd0b9f rdf:type owl:DatatypeProperty ;
                                                      rdfs:domain :OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00 ;
                                                      rdfs:range xsd:string ;
                                                      owlr:python_name "uses_input_template" ;
                                                      rdfs:comment "API Endpoint will accept input following the JSON
                                                      ↪  template presented here." ;
                                                      rdfs:label "uses input template"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLDataProperty_ee1f7846_6528_4e64_9e86_cc34af99f912
:OWLDataProperty_ee1f7846_6528_4e64_9e86_cc34af99f912 rdf:type owl:DatatypeProperty ;
                                                      rdfs:domain :OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00 ;
                                                      rdfs:range xsd:string ;
                                                      owlr:python_name "uses_output_template" ;
                                                      rdfs:comment "API Endpoint will provide ouptut following the
                                                      ↪  JSON template presented here." ;
                                                      rdfs:label "uses output template"@en-gb .

```

```

###  http://dragon.foi.hr/mago-a.owx#R8l3NnmqnvjfzJkS5rDNvFm
:R8l3NnmqnvjfzJkS5rDNvFm rdf:type owl:DatatypeProperty ;
                         rdfs:subPropertyOf owl:topDataProperty ;
                         rdf:type owl:FunctionalProperty ;
                         rdfs:domain :R7dQDUF81SOJC29Sctpy6aP ;
                         rdfs:range xsd:positiveInteger ;
                         owlr:python_name "has_period" ;
                         rdfs:comment "This property contains the information  about the length of the temporal
                         ↪  designation of a behaviour. A cyclic behaviour that has a period is a periodic behaviour,
                         ↪  and an acyclic behaviour that has a defined period is considered to be a time-out
                         ↪  behaviour, i.e. a one-shot behaviour that runs its main loop only after a specific period
                         ↪  of time passes."@en-gb ;
                         rdfs:label "has period"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RBffMlV8TQxoNtbLRneUYsb
:RBffMlV8TQxoNtbLRneUYsb rdf:type owl:DatatypeProperty ;
                         rdfs:subPropertyOf owl:topDataProperty ;
                         rdfs:range xsd:string ;
                         rdfs:comment "JSON description of the basic features of the modelled system, e.g. {\"number
                         ↪  of agents\": 10}"@en-gb ;
                         rdfs:label "has system features"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDxmweKrzFGij8P5HOOsXZK
:RDxmweKrzFGij8P5HOOsXZK rdf:type owl:DatatypeProperty ;
                         rdfs:subPropertyOf owl:topDataProperty ;
                         rdf:type owl:FunctionalProperty ;
                         rdfs:range xsd:string ;
                         owlr:python_name "has_name" ;
                         rdfs:label "has name"@en-gb .


#################################################################
#    Classes
#################################################################

###  http://dragon.foi.hr/mago-a.owx#OWLClass_256a9011_3e88_4389_acde_aa6320fe2953
:OWLClass_256a9011_3e88_4389_acde_aa6320fe2953 rdf:type owl:Class ;
                                               rdfs:subClassOf :R9lwUrkbSftwsmlPcopLgzT ;
                                               rdfs:label "Factory Agent"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00
:OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00 rdf:type owl:Class ;
                                               rdfs:subClassOf :RChP69av2xqRwh6LALfwsKq ;
                                               rdfs:label "RestAPI Endpoint"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLClass_30630ea0_e2d1_4057_8419_5ec603426309
:OWLClass_30630ea0_e2d1_4057_8419_5ec603426309 rdf:type owl:Class ;
                                               owl:equivalentClass [ owl:intersectionOf ( :R8EpacdsHKWIyPDRwsmkSta
                                                                                          [ rdf:type owl:Restriction ;
                                                                                            owl:onProperty
                                                                                            ↪  :R8l3NnmqnvjfzJkS5rDNvFm
                                                                                            ↪  ;
                                                                                            owl:someValuesFrom
                                                                                            ↪  xsd:positiveInteger
                                                                                          ]
                                                                                        ) ;
                                                                     rdf:type owl:Class
                                                                   ] ;
                                               rdfs:subClassOf :R8EpacdsHKWIyPDRwsmkSta ;
                                               rdfs:label "Time-out Behaviour"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce
:OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce rdf:type owl:Class ;
                                               rdfs:subClassOf :R9lwUrkbSftwsmlPcopLgzT ;
                                               rdfs:label "Recipe Agent"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP
:R7dQDUF81SOJC29Sctpy6aP rdf:type owl:Class ;
                         owl:disjointWith :R84KrWu3ZBbXCifW6B9GSmZ ;
                         rdfs:label "Behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ
:R84KrWu3ZBbXCifW6B9GSmZ rdf:type owl:Class ;
                         rdfs:label "Action"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R8EpacdsHKWIyPDRwsmkSta
:R8EpacdsHKWIyPDRwsmkSta rdf:type owl:Class ;
                         owl:equivalentClass [ owl:intersectionOf ( :R7dQDUF81SOJC29Sctpy6aP
                                                                    [ rdf:type owl:Restriction ;
                                                                      owl:onProperty
                                                                      ↪  :OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
                                                                      ↪  ;
                                                                      owl:hasValue "false"^^xsd:boolean
                                                                    ]
                                                                  ) ;
                                               rdf:type owl:Class
                                             ] ;
                         rdfs:subClassOf :R7dQDUF81SOJC29Sctpy6aP ;
                         rdfs:label "One Shot Behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd
:R9iXau1ZAN6oZTRbdCgbmgd rdf:type owl:Class ;
                         rdfs:subClassOf :RWOhadIHzH5yfBTDwuDICD ;
                         rdfs:label "Role"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT
:R9lwUrkbSftwsmlPcopLgzT rdf:type owl:Class ;
                         owlr:python_name "Agent" ;
                         rdfs:label "Agent"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO
```

```
:RB4lINuYfn41lvT2d2GduUO rdf:type owl:Class ;
                         rdfs:label "Artefact"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RBGp2C7WAHzeIMoeFp3K6lr
:RBGp2C7WAHzeIMoeFp3K6lr rdf:type owl:Class ;
                         rdfs:label "Strategy"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RBzqbNmJP5lfpPIgYCvLGDa
:RBzqbNmJP5lfpPIgYCvLGDa rdf:type owl:Class ;
                         rdfs:subClassOf :RB4lINuYfn41lvT2d2GduUO ;
                         rdfs:label "Software Artefact"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7
:RCdOsGdA0yrGZdnzgQesQx7 rdf:type owl:Class ;
                         rdfs:subClassOf :RBzqbNmJP5lfpPIgYCvLGDa ;
                         rdfs:label "Agent Host Server"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq
:RChP69av2xqRwh6LALfwsKq rdf:type owl:Class ;
                         rdfs:label "Artefact Resource"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RCjIVQ3sTEfJR8m9yEY3Mhe
:RCjIVQ3sTEfJR8m9yEY3Mhe rdf:type owl:Class ;
                         owl:equivalentClass [ owl:intersectionOf ( :R7dQDUF81SOJC29Sctpy6aP
                                                                    [ rdf:type owl:Restriction ;
                                                                      owl:onProperty
                                                                      ↪   :OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
                                                                      ↪   ;
                                                                      owl:hasValue "true"^^xsd:boolean
                                                                    ]
                                                                  ) ;
                                               rdf:type owl:Class
                                             ] ;
                         rdfs:subClassOf :R7dQDUF81SOJC29Sctpy6aP ;
                         rdfs:label "Cyclic Behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDMv7ayhi9xuRAzbfALLhKS
:RDMv7ayhi9xuRAzbfALLhKS rdf:type owl:Class ;
                         owl:equivalentClass [ owl:intersectionOf ( :R7dQDUF81SOJC29Sctpy6aP
                                                                    [ rdf:type owl:Restriction ;
                                                                      owl:onProperty :RC6rldf2VsWBaHtUobIELQL ;
                                                                      owl:someValuesFrom :RDgoxtmVWC61IdqtaNK9QBf
                                                                    ]
                                                                    [ rdf:type owl:Restriction ;
                                                                      owl:onProperty :RCPuCz8AunmmX3PS6A0U0XL ;
                                                                      owl:someValuesFrom :RDgoxtmVWC61IdqtaNK9QBf
                                                                    ]
                                                                  ) ;
                                               rdf:type owl:Class
                                             ] ;
                         rdfs:subClassOf :R7dQDUF81SOJC29Sctpy6aP ;
                         rdfs:label "Finite State Machine Behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf
:RDgoxtmVWC61IdqtaNK9QBf rdf:type owl:Class ;
                         owl:equivalentClass [ owl:intersectionOf ( :R7dQDUF81SOJC29Sctpy6aP
                                                                    [ rdf:type owl:Class ;
                                                                      owl:unionOf ( [ rdf:type owl:Restriction ;
                                                                                      owl:onProperty
                                                                                      ↪   :R8ZWdA7HWDzIIwhQZrSwAIj ;
                                                                                      owl:someValuesFrom
                                                                                      ↪   :R7dQDUF81SOJC29Sctpy6aP
                                                                                    ]
                                                                                    [ rdf:type owl:Restriction ;
                                                                                      owl:onProperty
                                                                                      ↪   :RDGoFrovO0PhBCQGPkFeUuR ;
                                                                                      owl:someValuesFrom
                                                                                      ↪   :R7dQDUF81SOJC29Sctpy6aP
                                                                                    ]
                                                                                  )
                                                                    ]
                                                                  ) ;
                                               rdf:type owl:Class
                                             ] ;
                         rdfs:subClassOf :R8EpacdsHKWIyPDRwsmkSta ;
                         rdfs:label "State Behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDl2yHd2gEWyEDMtUcL67d
:RDl2yHd2gEWyEDMtUcL67d rdf:type owl:Class ;
                        rdfs:subClassOf :Rm3YabirGXstrtBtkrthoH ;
                        rdfs:label "Individual Knowledge Artefact"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX
:RDm65h4GNQjimv0axMIRnMX rdf:type owl:Class ;
                         rdfs:subClassOf :RBGp2C7WAHzeIMoeFp3K6lr ;
                         rdfs:label "Plan"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o
:RDqoRvx1aDeStUw4zNRdR0o rdf:type owl:Class ;
                         rdfs:subClassOf :RBGp2C7WAHzeIMoeFp3K6lr ;
                         rdfs:label "Objective"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#RWOhadIHzH5yfBTDwuDICD
:RWOhadIHzH5yfBTDwuDICD rdf:type owl:Class ;
                        rdfs:subClassOf :Rm3YabirGXstrtBtkrthoH ;
                        rdfs:label "Organisational Knowledge Artefact"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#Rce2iHbgKH3gy3TygYasFi
:Rce2iHbgKH3gy3TygYasFi rdf:type owl:Class ;
                        rdfs:label "Workspace"@en-gb .
```

```

###  http://dragon.foi.hr/mago-a.owx#Rip7CRo5Sv8C6NFZrhvEDH
:Rip7CRo5Sv8C6NFZrhvEDH rdf:type owl:Class ;
                        owl:equivalentClass [ owl:intersectionOf ( :RCjIVQ3sTEfJR8m9yEY3Mhe
                                                                   [ rdf:type owl:Restriction ;
                                                                     owl:onProperty :R8l3NnmqnvjfzJkS5rDNvFm ;
                                                                     owl:someValuesFrom xsd:positiveInteger
                                                                   ]
                                                                 ) ;
                                              rdf:type owl:Class
                                            ] ;
                        rdfs:label "Periodic Behaviour"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#Rm3YabirGXstrtBtkrthoH
:Rm3YabirGXstrtBtkrthoH rdf:type owl:Class ;
                        rdfs:subClassOf :RB4lINuYfn41lvT2d2GduUO ;
                        rdfs:label "Knowledge Artefact"@en-gb .


#################################################################
#    Individuals
#################################################################

###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_0d1a0882_c9ac_494c_9872_f4f90e94b7fd
:OWLNamedIndividual_0d1a0882_c9ac_494c_9872_f4f90e94b7fd rdf:type owl:NamedIndividual ,
                                                                  :R7dQDUF81SOJC29Sctpy6aP ;
                                                         :OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
                                                         ↪   "false"^^xsd:boolean ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Register" ;
                                                         rdfs:label "Register"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_12c619ee_d747_4d66_8165_25991df28f70
:OWLNamedIndividual_12c619ee_d747_4d66_8165_25991df28f70 rdf:type owl:NamedIndividual ,
                                                                  :OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce ;
                                                         :RB8ofKEO8zM8jcK0WetPpLI
                                                         ↪   :OWLNamedIndividual_f1ff2f43_7dbb_483e_8f22_1a5102c9a8cc
                                                         ↪   ;
                                                         :RCcC3SXJ5MvuHMJyzs8OpU6
                                                         ↪   :OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228
                                                         ↪   ,

                                                                                 ↪   :OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c
                                                                                 ↪   ;
                                                         :RpmDQcEqabFOwsZHv82QuP :rcdosgda0yrgzdnzgqesqx76 ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Pizza Naepolitana" ;
                                                         rdfs:label "Drugi recept"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2
:OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2 rdf:type owl:NamedIndividual ,
                                                                  :RWOhadIHzH5yfBTDwuDICD ;
                                                         :OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
                                                         ↪   "https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx"^^xsd:anyURI
                                                         ↪   ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "The other ontology"^^xsd:anyURI ;
                                                         rdfs:label "Secondary ontology"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b
:OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b rdf:type owl:NamedIndividual ,
                                                                  :R9iXau1ZAN6oZTRbdCgbmgd ;
                                                         :RY56GiCpqJHON677qnE5sT
                                                         ↪   :OWLNamedIndividual_0d1a0882_c9ac_494c_9872_f4f90e94b7fd
                                                         ↪   ,

                                                                                ↪   :OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55
                                                                                ↪   ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Wizard" ;
                                                         rdfs:label "Wizard"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_44e0578c_c818_4c97_a1cf_081d31543933
:OWLNamedIndividual_44e0578c_c818_4c97_a1cf_081d31543933 rdf:type owl:NamedIndividual ,
                                                                  :R7dQDUF81SOJC29Sctpy6aP ;
                                                         :R8ZWdA7HWDzIIwhQZrSwAIj
                                                         ↪   :OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246
                                                         ↪   ;
                                                         :RDGoFrovO0PhBCQGPkFeUuR
                                                         ↪   :OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b
                                                         ↪   ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Rotate" ;
                                                         rdfs:label "Rotate"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_4ce555df_0daa_451c_8cee_3869fb46f599
:OWLNamedIndividual_4ce555df_0daa_451c_8cee_3869fb46f599 rdf:type owl:NamedIndividual ,
                                                                  :RDm65h4GNQjimv0axMIRnMX ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Delivery Plan"^^xsd:anyURI ;
                                                         rdfs:label "Delivery Plan"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac
:OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac rdf:type owl:NamedIndividual ,
                                                                  :RDqoRvx1aDeStUw4zNRdR0o ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Increase Efficiency"^^xsd:anyURI ;
                                                         rdfs:label "Increase Efficiency"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7
:OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7 rdf:type owl:NamedIndividual ,
                                                                  :RDqoRvx1aDeStUw4zNRdR0o ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Reduce Costs"^^xsd:anyURI ;
                                                         rdfs:label "Reduce Costs"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246
:OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246 rdf:type owl:NamedIndividual ,
                                                                  :R7dQDUF81SOJC29Sctpy6aP ;
                                                         :OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
                                                         ↪   "true"^^xsd:boolean ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Observe environment" ;
                                                         rdfs:label "Observe environment"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_7bdd32fb_271c_4f2c_aafd_d20c74aa22b9
:OWLNamedIndividual_7bdd32fb_271c_4f2c_aafd_d20c74aa22b9 rdf:type owl:NamedIndividual ,
                                                                  :R7dQDUF81SOJC29Sctpy6aP ;
                                                         :RC6rldf2VsWBaHtUobIELQL
                                                         ↪  :OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246
                                                         ↪  ;
                                                         :RCPuCz8AunmmX3PS6A0U0XL
                                                         ↪  :OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Navigate" ;
                                                         rdfs:label "Navigate"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228
:OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228 rdf:type owl:NamedIndividual ,
                                                                  :R9iXau1ZAN6oZTRbdCgbmgd ;
                                                         :RB8ofKEO8zM8jcK0WetPpLI
                                                         ↪  :OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2
                                                         ↪  ,

                                                                                  ↪  :OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098
                                                                                  ↪  ;
                                                         :RY56GiCpqJHON677qnE5sT
                                                         ↪  :OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246
                                                         ↪  ,

                                                                                 ↪  :OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55
                                                                                 ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Warrior" ;
                                                         rdfs:label "Warrior"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5
:OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5 rdf:type owl:NamedIndividual ,
                                                                  :R84KrWu3ZBbXCifW6B9GSmZ ;
                                                         :R8nS5zlvvUfE9N1xZBYTpcG
                                                         ↪  :OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac
                                                         ↪  ,

                                                                                  ↪  :OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7
                                                                                  ↪  ;
                                                         :RCkUIZx8f7TQwTrh3wmfyBD
                                                         ↪  :OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Assemble Components"^^xsd:anyURI ;
                                                         rdfs:label "Assemble Components"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55
:OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55 rdf:type owl:NamedIndividual ,
                                                                  :R7dQDUF81SOJC29Sctpy6aP ;
                                                         :OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
                                                         ↪  "true"^^xsd:boolean ;
                                                         :R8l3NnmqnvjfzJkS5rDNvFm "5"^^xsd:positiveInteger ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Check messages" ;
                                                         rdfs:label "Check messages"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098
:OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098 rdf:type owl:NamedIndividual ,
                                                                  :RWOhadIHzH5yfBTDwuDICD ;
                                                         :OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
                                                         ↪  "https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx"^^xsd:anyURI
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Main ontology"^^xsd:anyURI ;
                                                         rdfs:label "Main ontology"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c
:OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c rdf:type owl:NamedIndividual ,
                                                                  :R9iXau1ZAN6oZTRbdCgbmgd ;
                                                         :RB8ofKEO8zM8jcK0WetPpLI
                                                         ↪  :OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Scout"^^xsd:anyURI ;
                                                         rdfs:label "Scout"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b
:OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b rdf:type owl:NamedIndividual ,
                                                                  :R7dQDUF81SOJC29Sctpy6aP ;
                                                         :RDGoFrovO0PhBCQGPkFeUuR
                                                         ↪  :OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Move forwards" ;
                                                         rdfs:label "Move forwards"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_df91a3a0_7a62_46fb_ac02_75dde4d82493
:OWLNamedIndividual_df91a3a0_7a62_46fb_ac02_75dde4d82493 rdf:type owl:NamedIndividual ,
                                                                  :RDm65h4GNQjimv0axMIRnMX ;
                                                         :R84IuDzXJ7QCZi1bY6fzFxZ
                                                         ↪  :OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Manufacturing Plan"^^xsd:anyURI ;
                                                         rdfs:label "Manufacturing Plan"@en .


###  http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_f1ff2f43_7dbb_483e_8f22_1a5102c9a8cc
:OWLNamedIndividual_f1ff2f43_7dbb_483e_8f22_1a5102c9a8cc rdf:type owl:NamedIndividual ,
                                                                  :RDl2yHd2gEWyEDMtUcL67d ;
                                                         :OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
                                                         ↪  "https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx"^^xsd:anyURI
                                                         ↪  ;
                                                         :RDxmweKrzFGij8P5HOOsXZK "Personal knowledge" ;
                                                         rdfs:label "Personal ontology"@en .


###  http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt1
:r9lwurkbsftwsmlpcoplgzt1 rdf:type owl:NamedIndividual ,
                                   :OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce ;
                          :RCcC3SXJ5MvuHMJyzs8OpU6 :OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228 ;
                          :RpmDQcEqabFOwsZHv82QuP :rcdosgda0yrgzdnzgqesqx76 ;
                          :RDxmweKrzFGij8P5HOOsXZK "SevenOfNine" ;
                          rdfs:label "Broj dva"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt2
:r9lwurkbsftwsmlpcoplgzt2 rdf:type owl:NamedIndividual ,
                                   :R9lwUrkbSftwsmlPcopLgzT ;
                          :RB8ofKEO8zM8jcK0WetPpLI :OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098 ,
                                                   :rcdosgda0yrgzdnzgqesqx76 ;
                          :RDxmweKrzFGij8P5HOOsXZK "Jimmy" ;
                          rdfs:label "Agent six"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt3
:r9lwurkbsftwsmlpcoplgzt3 rdf:type owl:NamedIndividual ,
                                   :R9lwUrkbSftwsmlPcopLgzT ;
                          :RCcC3SXJ5MvuHMJyzs8OpU6 :OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b ,
                                                   :OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228 ;
                          :RDxmweKrzFGij8P5HOOsXZK "Jeanie" ;
                          rdfs:label "Agent seven"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt4
:r9lwurkbsftwsmlpcoplgzt4 rdf:type owl:NamedIndividual ,
                                   :R9lwUrkbSftwsmlPcopLgzT ;
                          :RDxmweKrzFGij8P5HOOsXZK "Janice" ;
                          rdfs:label "Agent eight"@en-gb .


###  http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx71
:rcdosgda0yrgzdnzgqesqx71 rdf:type owl:NamedIndividual ,
                                   :RCdOsGdA0yrGZdnzgQesQx7 ;
                          rdfs:label "drugi_host.com" .


###  http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76
:rcdosgda0yrgzdnzgqesqx76 rdf:type owl:NamedIndividual ,
                                   :RCdOsGdA0yrGZdnzgQesQx7 ;
                          :OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab "http://localhost"^^xsd:anyURI ;
                          rdfs:label "localhost" .


###  http://dragon.foi.hr/mago-a.owx#Broj%20jedan
<http://dragon.foi.hr/mago-a.owx#Broj%20jedan> rdf:type owl:NamedIndividual ,
                                                        :OWLClass_256a9011_3e88_4389_acde_aa6320fe2953 ;
                                               :RB8ofKEO8zM8jcK0WetPpLI
                                               ↪   :OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098 ;
                                               :RpmDQcEqabFOwsZHv82QuP :rcdosgda0yrgzdnzgqesqx76 ;
                                               :RDxmweKrzFGij8P5HOOsXZK "Broj jedan" ;
                                               rdfs:label "Prvi agent"@en-gb .


###  Generated by the OWL API (version 4.5.29.2024-05-13T12:11:03Z) https://github.com/owlcs/owlapi
```

Listing 9.5: Turtle serialization of the MAGO-Ag ontology

MAGO-Ag OWL/XML Serialization

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://dragon.foi.hr/mago-a.owx#"
     xml:base="http://dragon.foi.hr/mago-a.owx"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:owlr="http://www.lesfleursdunormal.fr/static/_downloads/owlready_ontology.owl#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <owl:Ontology rdf:about="http://dragon.foi.hr/mago-a.owx#">
        <rdfs:comment>The ontology for MAGO research visit.</rdfs:comment>
    </owl:Ontology>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Annotation properties
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://www.lesfleursdunormal.fr/static/_downloads/owlready_ontology.owl#python_name -->

    <owl:AnnotationProperty rdf:about="http://www.lesfleursdunormal.fr/static/_downloads/owlready_ontology.owl#python_name"/>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Object Properties
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_691196cf_4ee3_4e4f_b8f0_7ee4aecc248c -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_691196cf_4ee3_4e4f_b8f0_7ee4aecc248c">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R8nS5zlvvUfE9N1xZBYTpcG"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:label xml:lang="en-gb">is achieved by action</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_78dd3dec_c844_450a_8bd0_412f69614517 -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_78dd3dec_c844_450a_8bd0_412f69614517">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RCWrWcZCQ3lxj6jFw9zRJ6m"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO"/>
        <rdfs:label xml:lang="en-gb">is provided by</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_ca58846c_df5f_4584_9375_8e686fdbb3fe -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_ca58846c_df5f_4584_9375_8e686fdbb3fe">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R84IuDzXJ7QCZi1bY6fzFxZ"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <rdfs:label xml:lang="en-gb">is required in plan</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_d04b28e8_062e_4b0d_82c6_a6b9f6b31226 -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#OWLObjectProperty_d04b28e8_062e_4b0d_82c6_a6b9f6b31226">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RpmDQcEqabFOwsZHv82QuP"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <rdfs:label xml:lang="en-gb">hosts</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R7MvUIWpnOdfB3dxpRXTUAK -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R7MvUIWpnOdfB3dxpRXTUAK">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RCkUIZx8f7TQwTrh3wmfyBD"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:label xml:lang="en-gb">implements action</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R7NTCPsMvJdxinMk2P28ppg -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R7NTCPsMvJdxinMk2P28ppg">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:label xml:lang="en-gb">has final objective</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R7o5TdA2HQOQxwVPrUJrLpq -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R7o5TdA2HQOQxwVPrUJrLpq">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RBntfnmvB9JVuK1xvAe6Hm8"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:label xml:lang="en-gb">is before objective</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R84IuDzXJ7QCZi1bY6fzFxZ -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R84IuDzXJ7QCZi1bY6fzFxZ">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <owlr:python_name>requires_action</owlr:python_name>
        <rdfs:label xml:lang="en-gb">requires action</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R8ZWdA7HWDzIIwhQZrSwAIj -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R8ZWdA7HWDzIIwhQZrSwAIj">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RDGoFrovO0PhBCQGPkFeUuR"/>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#IrreflexiveProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
        <owlr:python_name>is_after_state</owlr:python_name>
        <rdfs:label xml:lang="en-gb">is after state</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R8ZXgNqWa8M3RnwigsFBxvc -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R8ZXgNqWa8M3RnwigsFBxvc">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <rdfs:label xml:lang="en-gb">features role</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R8nS5zlvvUfE9N1xZBYTpcG -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R8nS5zlvvUfE9N1xZBYTpcG">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <owlr:python_name>has_objective</owlr:python_name>
        <rdfs:label xml:lang="en-gb">has objective</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R9QlFkpwG4P6QT3YaFGdAPM -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R9QlFkpwG4P6QT3YaFGdAPM">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R9wYaBAe5LQpMMDI7zKHWhc"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:label xml:lang="en-gb">is after action</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#R9Y2982rI3hZL4oBhv8Ohhv -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R9Y2982rI3hZL4oBhv8Ohhv">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <rdfs:label xml:lang="en-gb">is part of process</rdfs:label>
    </owl:ObjectProperty>
```

```
<!-- http://dragon.foi.hr/mago-a.owx#R9wYaBAe5LQpMMDI7zKHWhc -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R9wYaBAe5LQpMMDI7zKHWhc">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
    <rdfs:label xml:lang="en-gb">is before action</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RB8ofKEO8zM8jcK0WetPpLI -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RB8ofKEO8zM8jcK0WetPpLI">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
                <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO"/>
    <owlr:python_name>can_access_artefact</owlr:python_name>
    <rdfs:label xml:lang="en-gb">can access artefact</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RBOL8y2xs4VvKc6D0kikOup -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RBOL8y2xs4VvKc6D0kikOup">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
    <owlr:python_name>is_part_of_role</owlr:python_name>
    <rdfs:label xml:lang="en-gb">is part of role</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RBmpSay8yjDZmklOv6Mw1rc -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RBmpSay8yjDZmklOv6Mw1rc">
    <rdfs:subPropertyOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RCcC3SXJ5MvuHMJyzs8OpU6"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
    <owlr:python_name>plays_role</owlr:python_name>
    <rdfs:label xml:lang="en-gb">plays role</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RBntfnmvB9JVuK1xvAe6Hm8 -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RBntfnmvB9JVuK1xvAe6Hm8">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
    <rdfs:label xml:lang="en-gb">is after objective</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RC6rldf2VsWBaHtUobIELQL -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RC6rldf2VsWBaHtUobIELQL">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDMv7ayhi9xuRAzbfALLhKS"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
    <owlr:python_name>has_initial_state</owlr:python_name>
    <rdfs:label xml:lang="en-gb">has initial state</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RCAv5UwFPy5OUQdjN61nXbw -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RCAv5UwFPy5OUQdjN61nXbw">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <owl:inverseOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RY56GiCpqJHON677qnE5sT"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
    <owlr:python_name>is_provided_by_role</owlr:python_name>
    <rdfs:label xml:lang="en-gb">is provided by role</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RCPuCz8AunmmX3PS6A0U0XL -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RCPuCz8AunmmX3PS6A0U0XL">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDMv7ayhi9xuRAzbfALLhKS"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
    <owlr:python_name>has_final_state</owlr:python_name>
    <rdfs:label xml:lang="en-gb">has final state</rdfs:label>
</owl:ObjectProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RCWrWcZCQ3lxj6jFw9zRJ6m -->

<owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RCWrWcZCQ3lxj6jFw9zRJ6m">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO"/>
    <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq"/>
    <owlr:python_name>provides_resource</owlr:python_name>
    <rdfs:label xml:lang="en-gb">provides resource</rdfs:label>
</owl:ObjectProperty>
```

```
    <!-- http://dragon.foi.hr/mago-a.owx#RCcC3SXJ5MvuHMJyzs8OpU6 -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RCcC3SXJ5MvuHMJyzs8OpU6">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <owlr:python_name>can_play_role</owlr:python_name>
        <rdfs:label xml:lang="en-gb">can play role</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RCkUIZx8f7TQwTrh3wmfyBD -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RCkUIZx8f7TQwTrh3wmfyBD">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <owlr:python_name>has_behaviour</owlr:python_name>
        <rdfs:label xml:lang="en-gb">is implemented using behaviour</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RCsnmcyqIe70G3lagH4y3bE -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RCsnmcyqIe70G3lagH4y3bE">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:label xml:lang="en-gb">is part of objective</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RDGoFrovO0PhBCQGPkFeUuR -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RDGoFrovO0PhBCQGPkFeUuR">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#IrreflexiveProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
        <owlr:python_name>is_before_state</owlr:python_name>
        <rdfs:label xml:lang="en-gb">is before state</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RDrm00K6AJVd5OJmoVCqPy6 -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RDrm00K6AJVd5OJmoVCqPy6">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:label xml:lang="en-gb">has initial objective</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RY56GiCpqJHON677qnE5sT -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RY56GiCpqJHON677qnE5sT">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <owlr:python_name>provides_behaviour</owlr:python_name>
        <rdfs:label xml:lang="en-gb">provides behaviour</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#ReKEcwE01zu4hSOL1gSBu2 -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#ReKEcwE01zu4hSOL1gSBu2">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <rdfs:label xml:lang="en-gb">features objective</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RoDHPMWwB604EEv4qQbsGC -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RoDHPMWwB604EEv4qQbsGC">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <rdfs:label xml:lang="en-gb">features process</rdfs:label>
    </owl:ObjectProperty>



    <!-- http://dragon.foi.hr/mago-a.owx#RpmDQcEqabFOwsZHv82QuP -->

    <owl:ObjectProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RpmDQcEqabFOwsZHv82QuP">
        <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <rdfs:range rdf:resource="http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7"/>
        <owlr:python_name>lives_on_host</owlr:python_name>
        <rdfs:label xml:lang="en-gb">lives on host</rdfs:label>
    </owl:ObjectProperty>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Data properties
    //
    ///////////////////////////////////////////////////////////////////////////////////////
```

```
 -->




<!-- http://dragon.foi.hr/mago-a.owx#OWLDataProperty_13076287_43a5_47ba_80d4_aa0b2fcbf767 -->

<owl:DatatypeProperty
↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_13076287_43a5_47ba_80d4_aa0b2fcbf767">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:label xml:lang="en-gb">is implemented as</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab -->

<owl:DatatypeProperty
↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO"/>
                <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq"/>
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#anyURI"/>
    <owlr:python_name>has_uri</owlr:python_name>
    <rdfs:label xml:lang="en-gb">has URI</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49 -->

<owl:DatatypeProperty
↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49">
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <owlr:python_name>is_repeating</owlr:python_name>
    <rdfs:comment>This property defines whether the behaviour is repeating or not, designating cyclic or one-shot
    ↪  behaviour types.</rdfs:comment>
    <rdfs:label xml:lang="en-gb">is repeating</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#OWLDataProperty_8f71ec98_a3b3_4f5b_b39f_dab911dd0b9f -->

<owl:DatatypeProperty
↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_8f71ec98_a3b3_4f5b_b39f_dab911dd0b9f">
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owlr:python_name>uses_input_template</owlr:python_name>
    <rdfs:comment>API Endpoint will accept input following the JSON template presented here.</rdfs:comment>
    <rdfs:label xml:lang="en-gb">uses input template</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#OWLDataProperty_ee1f7846_6528_4e64_9e86_cc34af99f912 -->

<owl:DatatypeProperty
↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_ee1f7846_6528_4e64_9e86_cc34af99f912">
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owlr:python_name>uses_output_template</owlr:python_name>
    <rdfs:comment>API Endpoint will provide ouptut following the JSON template presented here.</rdfs:comment>
    <rdfs:label xml:lang="en-gb">uses output template</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#R8l3NnmqnvjfzJkS5rDNvFm -->

<owl:DatatypeProperty rdf:about="http://dragon.foi.hr/mago-a.owx#R8l3NnmqnvjfzJkS5rDNvFm">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#positiveInteger"/>
    <owlr:python_name>has_period</owlr:python_name>
    <rdfs:comment xml:lang="en-gb">This property contains the information  about the length of the temporal
    ↪  designation of a behaviour. A cyclic behaviour that has a period is a periodic behaviour, and an acyclic
    ↪  behaviour that has a defined period is considered to be a time-out behaviour, i.e. a one-shot behaviour
    ↪  that runs its main loop only after a specific period of time passes.</rdfs:comment>
    <rdfs:label xml:lang="en-gb">has period</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RBffMlV8TQxoNtbLRneUYsb -->

<owl:DatatypeProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RBffMlV8TQxoNtbLRneUYsb">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owlr:python_name>has_system_features</owlr:python_name>
    <rdfs:comment xml:lang="en-gb">JSON description of the basic features of the modelled system, e.g.
    ↪  {&quot;number of agents&quot;: 10}</rdfs:comment>
    <rdfs:label xml:lang="en-gb">has system features</rdfs:label>
</owl:DatatypeProperty>



<!-- http://dragon.foi.hr/mago-a.owx#RDxmweKrzFGij8P5HOOsXZK -->

<owl:DatatypeProperty rdf:about="http://dragon.foi.hr/mago-a.owx#RDxmweKrzFGij8P5HOOsXZK">
    <rdfs:subPropertyOf rdf:resource="http://www.w3.org/2002/07/owl#topDataProperty"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <owlr:python_name>has_name</owlr:python_name>
```

```
        <rdfs:label xml:lang="en-gb">has name</rdfs:label>
    </owl:DatatypeProperty>
    


    <!-- 
    ////////////////////////////////////////////////////////////////////////////////////////
    //
    // Classes
    //
    ////////////////////////////////////////////////////////////////////////////////////////
     -->


    <!-- http://dragon.foi.hr/mago-a.owx#OWLClass_256a9011_3e88_4389_acde_aa6320fe2953 -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#OWLClass_256a9011_3e88_4389_acde_aa6320fe2953">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <rdfs:label xml:lang="en-gb">Factory Agent</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00 -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#OWLClass_27e9637c_fe4f_4875_8d16_087da4a1cb00">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq"/>
        <rdfs:label xml:lang="en-gb">RestAPI Endpoint</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#OWLClass_30630ea0_e2d1_4057_8419_5ec603426309 -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#OWLClass_30630ea0_e2d1_4057_8419_5ec603426309">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R8EpacdsHKWIyPDRwsmkSta"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="http://dragon.foi.hr/mago-a.owx#R8l3NnmqnvjfzJkS5rDNvFm"/>
                        <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#positiveInteger"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R8EpacdsHKWIyPDRwsmkSta"/>
        <rdfs:label xml:lang="en-gb">Time-out Behaviour</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <rdfs:label xml:lang="en-gb">Recipe Agent</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP">
        <owl:disjointWith rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <rdfs:label xml:lang="en-gb">Behaviour</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ">
        <rdfs:label xml:lang="en-gb">Action</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#R8EpacdsHKWIyPDRwsmkSta -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#R8EpacdsHKWIyPDRwsmkSta">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                    <owl:Restriction>
                        <owl:onProperty
                        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49"/>
                        <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">false</owl:hasValue>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <rdfs:label xml:lang="en-gb">One Shot Behaviour</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RWOhadIHzH5yfBTDwuDICD"/>
        <rdfs:label xml:lang="en-gb">Role</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT">
```

```
        <owlr:python_name>Agent</owlr:python_name>
        <rdfs:label xml:lang="en-gb">Agent</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO">
        <rdfs:label xml:lang="en-gb">Artefact</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RBGp2C7WAHzeIMoeFp3K6lr -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RBGp2C7WAHzeIMoeFp3K6lr">
        <rdfs:label xml:lang="en-gb">Strategy</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RBzqbNmJP5lfpPIgYCvLGDa -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RBzqbNmJP5lfpPIgYCvLGDa">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO"/>
        <rdfs:label xml:lang="en-gb">Software Artefact</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7 -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RBzqbNmJP5lfpPIgYCvLGDa"/>
        <rdfs:label xml:lang="en-gb">Agent Host Server</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RChP69av2xqRwh6LALfwsKq">
        <rdfs:label xml:lang="en-gb">Artefact Resource</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RCjIVQ3sTEfJR8m9yEY3Mhe -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RCjIVQ3sTEfJR8m9yEY3Mhe">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                    <owl:Restriction>
                        <owl:onProperty
                        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49"/>
                        <owl:hasValue rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</owl:hasValue>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <rdfs:label xml:lang="en-gb">Cyclic Behaviour</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RDMv7ayhi9xuRAzbfALLhKS -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RDMv7ayhi9xuRAzbfALLhKS">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="http://dragon.foi.hr/mago-a.owx#RC6rldf2VsWBaHtUobIELQL"/>
                        <owl:someValuesFrom rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
                    </owl:Restriction>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="http://dragon.foi.hr/mago-a.owx#RCPuCz8AunmmX3PS6A0U0XL"/>
                        <owl:someValuesFrom rdf:resource="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <rdfs:label xml:lang="en-gb">Finite State Machine Behaviour</rdfs:label>
    </owl:Class>



    <!-- http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RDgoxtmVWC61IdqtaNK9QBf">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                    <owl:Class>
                        <owl:unionOf rdf:parseType="Collection">
                            <owl:Restriction>
                                <owl:onProperty
                                ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#R8ZWdA7HWDzIIwhQZrSwAIj"/>
                                <owl:someValuesFrom
                                ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                            </owl:Restriction>
                            <owl:Restriction>
                                <owl:onProperty
                                ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#RDGoFrovO0PhBCQGPkFeUuR"/>
                                <owl:someValuesFrom
                                ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
                            </owl:Restriction>
                        </owl:unionOf>
                    </owl:Class>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#R8EpacdsHKWIyPDRwsmkSta"/>
        <rdfs:label xml:lang="en-gb">State Behaviour</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#RDl2yHd2gEWyEDMtUcL67d -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RDl2yHd2gEWyEDMtUcL67d">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#Rm3YabirGXstrtBtkrthoH"/>
        <rdfs:label xml:lang="en-gb">Individual Knowledge Artefact</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RBGp2C7WAHzeIMoeFp3K6lr"/>
        <rdfs:label xml:lang="en-gb">Plan</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RBGp2C7WAHzeIMoeFp3K6lr"/>
        <rdfs:label xml:lang="en-gb">Objective</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#RWOhadIHzH5yfBTDwuDICD -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#RWOhadIHzH5yfBTDwuDICD">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#Rm3YabirGXstrtBtkrthoH"/>
        <rdfs:label xml:lang="en-gb">Organisational Knowledge Artefact</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#Rce2iHbgKH3gy3TygYasFi -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#Rce2iHbgKH3gy3TygYasFi">
        <rdfs:label xml:lang="en-gb">System</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#Rip7CRo5Sv8C6NFZrhvEDH -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#Rip7CRo5Sv8C6NFZrhvEDH">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="http://dragon.foi.hr/mago-a.owx#RCjIVQ3sTEfJR8m9yEY3Mhe"/>
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="http://dragon.foi.hr/mago-a.owx#R8l3NnmqnvjfzJkS5rDNvFm"/>
                        <owl:someValuesFrom rdf:resource="http://www.w3.org/2001/XMLSchema#positiveInteger"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:label xml:lang="en-gb">Periodic Behaviour</rdfs:label>
    </owl:Class>
    


    <!-- http://dragon.foi.hr/mago-a.owx#Rm3YabirGXstrtBtkrthoH -->

    <owl:Class rdf:about="http://dragon.foi.hr/mago-a.owx#Rm3YabirGXstrtBtkrthoH">
        <rdfs:subClassOf rdf:resource="http://dragon.foi.hr/mago-a.owx#RB4lINuYfn41lvT2d2GduUO"/>
        <rdfs:label xml:lang="en-gb">Knowledge Artefact</rdfs:label>
    </owl:Class>
    


    <!-- 
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Individuals
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->


    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_0d1a0882_c9ac_494c_9872_f4f90e94b7fd -->

    <owl:NamedIndividual
    ↪   rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_0d1a0882_c9ac_494c_9872_f4f90e94b7fd">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
        ↪   rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">false</OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49>
        <RDxmweKrzFGij8P5HOOsXZK>Register</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Register</rdfs:label>
    </owl:NamedIndividual>
    


    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_12c619ee_d747_4d66_8165_25991df28f70 -->

    <owl:NamedIndividual
    ↪   rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_12c619ee_d747_4d66_8165_25991df28f70">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce"/>
        <RB8ofKEO8zM8jcK0WetPpLI
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_f1ff2f43_7dbb_483e_8f22_1a5102c9a8cc"/>
        <RCcC3SXJ5MvuHMJyzs8OpU6
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228"/>
        <RCcC3SXJ5MvuHMJyzs8OpU6
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c"/>
        <RpmDQcEqabFOwsZHv82QuP rdf:resource="http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76"/>
        <RDxmweKrzFGij8P5HOOsXZK>Pizza Naepolitana</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Drugi recept</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RWOhadIHzH5yfBTDwuDICD"/>
        <OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx</OWLDataProperty_15a4602e_47e2_4459
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">The other
        ↪  ontology</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Secondary ontology</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <RY56GiCpqJHON677qnE5sT
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_0d1a0882_c9ac_494c_9872_f4f90e94b7fd"/>
        <RY56GiCpqJHON677qnE5sT
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55"/>
        <RDxmweKrzFGij8P5HOOsXZK>Wizard</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Wizard</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_44e0578c_c818_4c97_a1cf_081d31543933 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_44e0578c_c818_4c97_a1cf_081d31543933">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <R8ZWdA7HWDzIIwhQZrSwAIj
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246"/>
        <RDGoFrovO0PhBCQGPkFeUuR
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b"/>
        <RDxmweKrzFGij8P5HOOsXZK>Rotate</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Rotate</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_4ce555df_0daa_451c_8cee_3869fb46f599 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_4ce555df_0daa_451c_8cee_3869fb46f599">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Delivery
        ↪  Plan</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Delivery Plan</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Increase
        ↪  Efficiency</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Increase Efficiency</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RDqoRvx1aDeStUw4zNRdR0o"/>
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Reduce
        ↪  Costs</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Reduce Costs</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49>
        <RDxmweKrzFGij8P5HOOsXZK>Observe environment</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Observe environment</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_7bdd32fb_271c_4f2c_aafd_d20c74aa22b9 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_7bdd32fb_271c_4f2c_aafd_d20c74aa22b9">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <RC6rldf2VsWBaHtUobIELQL
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246"/>
        <RCPuCz8AunmmX3PS6A0U0XL
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b"/>
        <RDxmweKrzFGij8P5HOOsXZK>Navigate</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Navigate</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <RB8ofKEO8zM8jcK0WetPpLI
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2"/>
        <RB8ofKEO8zM8jcK0WetPpLI
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098"/>
        <RY56GiCpqJHON677qnE5sT
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246"/>
        <RY56GiCpqJHON677qnE5sT
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55"/>
        <RDxmweKrzFGij8P5HOOsXZK>Warrior</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Warrior</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R84KrWu3ZBbXCifW6B9GSmZ"/>
        <R8nS5zlvvUfE9N1xZBYTpcG
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_5ea5abf1_cf9f_4fe0_8842_25b90ea7beac"/>
        <R8nS5zlvvUfE9N1xZBYTpcG
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_644f18d0_b719_4ce3_9f18_5ed00cfaf3e7"/>
        <RCkUIZx8f7TQwTrh3wmfyBD
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246"/>
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Assemble
        ↪  Components</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Assemble Components</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_c17807e1_7a97_4b2c_b0ef_354af3ac8e55">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</OWLDataProperty_1dd27442_1507_4890_8c6b_89ff9a5a4f49>
        <R8l3NnmqnvjfzJkS5rDNvFm
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#positiveInteger">5</R8l3NnmqnvjfzJkS5rDNvFm>
        <RDxmweKrzFGij8P5HOOsXZK>Check messages</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Check messages</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RWOhadIHzH5yfBTDwuDICD"/>
        <OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx</OWLDataProperty_15a4602e_47e2_4459
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Main
        ↪  ontology</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Main ontology</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_caa11e73_a20b_4e88_8068_63ae9cfe2e4c">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R9iXau1ZAN6oZTRbdCgbmgd"/>
        <RB8ofKEO8zM8jcK0WetPpLI
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_2a83ad76_03cf_474c_ace1_5667d7689ad2"/>
        <RDxmweKrzFGij8P5HOOsXZK
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Scout</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Scout</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_d422ba16_c564_422e_bca8_9e793add1c0b">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R7dQDUF81SOJC29Sctpy6aP"/>
        <RDGoFrovO0PhBCQGPkFeUuR
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_77064793_551f_4645_8bc4_7d69c3908246"/>
        <RDxmweKrzFGij8P5HOOsXZK>Move forwards</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Move forwards</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_df91a3a0_7a62_46fb_ac02_75dde4d82493 -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_df91a3a0_7a62_46fb_ac02_75dde4d82493">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RDm65h4GNQjimv0axMIRnMX"/>
        <R84IuDzXJ7QCZi1bY6fzFxZ
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_a5438295_391f_4d36_b6f9_b163ade1b2e5"/>
        <RDxmweKrzFGij8P5HOOsXZK rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">Manufacturing
        ↪  Plan</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Manufacturing Plan</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_f1ff2f43_7dbb_483e_8f22_1a5102c9a8cc -->

    <owl:NamedIndividual
    ↪  rdf:about="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_f1ff2f43_7dbb_483e_8f22_1a5102c9a8cc">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RDl2yHd2gEWyEDMtUcL67d"/>
        <OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">https://raw.githubusercontent.com/AILab-FOI/MAGO/main/Deliverables/Phase%201/Implementation/MAGO-Ag.owx</OWLDataProperty_15a4602e_47e2_4459
        <RDxmweKrzFGij8P5HOOsXZK>Personal knowledge</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en">Personal ontology</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt1 -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt1">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLClass_4e210354_6993_4cf7_96ae_3c18254ef7ce"/>
        <RCcC3SXJ5MvuHMJyzs8OpU6
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228"/>
        <RpmDQcEqabFOwsZHv82QuP rdf:resource="http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76"/>
        <RDxmweKrzFGij8P5HOOsXZK>SevenOfNine</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Broj dva</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt2 -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt2">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <RB8ofKEO8zM8jcK0WetPpLI
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098"/>
        <RB8ofKEO8zM8jcK0WetPpLI rdf:resource="http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76"/>
        <RDxmweKrzFGij8P5HOOsXZK>Jimmy</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Agent six</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt3 -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt3">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <RCcC3SXJ5MvuHMJyzs8OpU6
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_3a6da667_46f7_421b_84b9_afdae9406a3b"/>
        <RCcC3SXJ5MvuHMJyzs8OpU6
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_83d43aa5_243d_4472_8111_16f642b55228"/>
        <RDxmweKrzFGij8P5HOOsXZK>Jeanie</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Agent seven</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt4 -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#r9lwurkbsftwsmlpcoplgzt4">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#R9lwUrkbSftwsmlPcopLgzT"/>
        <RDxmweKrzFGij8P5HOOsXZK>Janice</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Agent eight</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx71 -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx71">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7"/>
        <rdfs:label>drugi_host.com</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76 -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#RCdOsGdA0yrGZdnzgQesQx7"/>
        <OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab
        ↪  rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">http://localhost</OWLDataProperty_15a4602e_47e2_4459_be8c_532c6e1062ab>
        <rdfs:label>localhost</rdfs:label>
    </owl:NamedIndividual>



    <!-- http://dragon.foi.hr/mago-a.owx#Broj%20jedan -->

    <owl:NamedIndividual rdf:about="http://dragon.foi.hr/mago-a.owx#Broj%20jedan">
        <rdf:type rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLClass_256a9011_3e88_4389_acde_aa6320fe2953"/>
        <RB8ofKEO8zM8jcK0WetPpLI
        ↪  rdf:resource="http://dragon.foi.hr/mago-a.owx#OWLNamedIndividual_ca5f1a1d_3af7_45da_a3d1_7ee2d7bbc098"/>
        <RpmDQcEqabFOwsZHv82QuP rdf:resource="http://dragon.foi.hr/mago-a.owx#rcdosgda0yrgzdnzgqesqx76"/>
        <RDxmweKrzFGij8P5HOOsXZK>Broj jedan</RDxmweKrzFGij8P5HOOsXZK>
        <rdfs:label xml:lang="en-gb">Prvi agent</rdfs:label>
    </owl:NamedIndividual>
</rdf:RDF>



<!-- Generated by the OWL API (version 4.5.29.2024-05-13T12:11:03Z) https://github.com/owlcs/owlapi -->
```

Listing 9.6: OWL/XML serialization of the MAGO-Ag ontology

MAGO-Ag Framework Implementation Files

```python
from owlready2 import World, Ontology, NamedIndividual
from string import Template

import os
from itertools import chain
import textwrap
from json import loads

from aux import *

import logging

logging.basicConfig(
    level=logging.INFO,
    format="%(asctime)s - %(levelname)s - %(message)s",
    filename="log.log",
)


class Thing:
    """Contains the methods that are common to all the classes that are a part of the translation process. These methods are used for setting the implementation template, rendering implementation based on the set template (no set prior template raises an error), getting the implementation, and writing the implementation to a file."""

    def __init__(
        self, entity_type: str, uri: str = None, name: str = None, onto_individual=None
    ) -> None:
        self.entity_type = clean_string(entity_type)
        self.uri = uri
        self.implementation_template = None
        self.implementation = None

        self.world = World()
        self.onto: Ontology = self.world.get_ontology("MAGO-Ag.owx").load(reload=True)
        self.onto_properties = {}

        self.onto_individual: NamedIndividual = onto_individual

        if name is None and onto_individual is not None:
            self.name = clean_string(self.onto_individual.has_name)
        else:
            self.name = clean_string(name)

        if uri is None and onto_individual is not None:
            self.uri = self.onto_individual.iri
        else:
            self.uri = uri

        if onto_individual:
            self.onto_properties.clear()
            for prop in onto_individual.get_properties():
                # print(prop.label)
                self.onto_properties.setdefault(
                    prop.label[0] if prop.label else prop.name,
                    getattr(onto_individual, prop.python_name),
                )

            # print(self.onto_properties)

            for name, value in self.onto_properties.items():
                setattr(self, name, value)

        logging.info(
            f"Individual {self.onto_individual if self.onto_individual else self.name} of type {self.entity_type} created."
        )

    def set_implementation_template(self, implementation_template: str):
        """Set the implementation template, following the string.Template syntax. This template is used to generate implementation of the object.

        Args:
            implementation_template (str): The implementation template to be filled in with appropriate values of objects of this class. Has to follow string.Template syntax.
        """
        self.implementation_template = Template(implementation_template)
        self.implementation = None

    def render_implementation(self, substitutes: dict = None):
        """Fill in the provided implementation template with data. If no `substitutes` value is provided, attributes of the object are used (those must have the same names as the variables in the template string).

        Args:
            substitutes (dict, optional): A dictionary of the values to be used in the provided template. Defaults to None.

        Raises:
            ValueError: Error is raised if no template was set.
        """
        if not self.implementation_template:
            raise ValueError("No implementation template set.")

        substitutes = self.__dict__ if not substitutes else substitutes
        self.implementation = self.implementation_template.substitute(
            substitutes
        ).strip()

    def get_implementation(self):
        """Return the rendered implementation.

        Returns:
            Returns the rendered implementation template or None if unavailable.
        """
        return self.implementation if self.implementation else None

    def write_implementation_to_file(self, file_name: str = None):
        """Save the rendered implementation to a file.

        Args:
            file_name (str, optional): Name of the file to be written. If not provided, will be rendered based on the
            ↪  name of the class (`self.__class__.__name__`) and type of entity (`self.entity_type`). Defaults to
            ↪  None.
        """
        if not self.implementation:
            self.render_implementation()
        if not file_name:
            file_name = os.path.join(
                os.getcwd(),
                "Template",
                f"{self.__class__.__name__}_{self.entity_type}.py",
            )
        with open(file_name, "w") as file:
            file.write(self.implementation)

        logging.info(
            f"Implementation of {self.onto_individual if self.onto_individual else self.name} saved to {file_name}."
        )

    def execute_sparql_query(self, query: dict, world: World) -> list:
        query_res = world.sparql(sparql=query.get("query"), params=query.get("params"))

        return list(query_res)
```

Listing 9.7: The `mago_thing.py` file

```
from mago_thing import *


class Agent(Thing):
    """A class containing all the data describing a MAGO agent."""

    def __init__(
        self,
        agent_type: str,
        host_server: str = None,
        password: str = None,
        # knowledge_artefact_uris: dict[str, str] = None,
        *args,
        **kwargs,
    ):
        super().__init__(entity_type=agent_type, *args, **kwargs)
        self.agent_type = clean_string(agent_type)
        self.host_server = host_server
        self.password = password
        self.behaviours = []
        self.query_roles = """"""
        self.knowledge_artefact_uris = get_related_knowledge_artefact_uris(
            self.onto_individual
        )
        self.system_features = loads(self.onto_individual.has_system_features) if
        ↪  self.onto_individual.has_system_features else None

    def render_agent_implementation(self):
        self.set_implementation_template(
            """
from itertools import chain

from owlready2 import World
from spade.agent import Agent
from Behaviours import *


class $agent_type(Agent):

    def execute_sparql(self, world: World=None, query: str=None, parameters: list=None) -> list:
        \"\"\"Execute a SPARQL query in the provided owlready2 World instance.

        Args:
            world (World): An owlready2 World instance containing the relevant data.
            query (str): The SPARQL query to be executed. Parameters are designated as `??`.
            parameters (list, optional): Parameters to be sequentially provided to the query. Defaults to None.

        Returns:
            list: The result of the query.
        \"\"\"
        if world is None:
            world = self.world

        if query is None:
            ValueError("Query string must be provided.")

        prepared_query = world.prepare_sparql(sparql=query)
        column_names = [name.replace("?", "") for name in prepared_query.column_names]

        query_res = prepared_query.execute(params=parameters)
        query_res = [dict(zip(column_names, result)) for result in query_res]

        return query_res

    async def setup(self):
        print(f"{self.name}: New agent running.")

        self.knowledge_artefacts = {}

        for ka_name, ka_uri in self.knowledge_artefact_uris.items():
            world = World()
            self.knowledge_artefacts.setdefault(ka_name, {}).update({
                "world": world,
                "ontology": world.get_ontology(ka_uri).load(reload=True)
            })

        if self.available_roles_and_behaviours is not None:
            self.available_roles = list([entry.get("name") for entry in self.available_roles_and_behaviours.values()])
            self.available_behaviours = list(set(chain(*[entry.get("behaviours", {}).values()
              for entry in self.available_roles_and_behaviours.values()])))
        else:
            self.available_roles = None
            self.available_behaviours = None

        print(self.name, self.available_roles, self.available_behaviours, self.knowledge_artefacts)

        self.world = self.knowledge_artefacts.get("Main ontology").get("world")
        self.onto_individual = self.world.search_one(iri=self.uri)
"""
        )
        self.render_implementation()
        return self.get_implementation()

    def get_related_roles_and_behaviours(self):
        result = {}

        result.update(
            {
                role.iri: {"name": role.has_name, "behaviours": {}}
                for role in self.onto_individual.can_play_role
            }
        )

        for role in self.onto_individual.can_play_role:
            result.get(role.iri).get("behaviours").update(
                {
                    behaviour.iri: behaviour.has_name
                    for behaviour in role.provides_behaviour
                }
            )

        return result if result else None

    def render_agent_instantiation(self):
        self.related_roles_and_behaviours = self.get_related_roles_and_behaviours()

        agent_instantiation_template = """
agent = $agent_type("$name@$host_server", "$password")
agent.uri = "$uri"
agent.knowledge_artefact_uris = $knowledge_artefact_uris
agent.available_roles_and_behaviours = $related_roles_and_behaviours
agent.system_features = $system_features
agent_individuals.setdefault("$host_server", {}).update({"$name": agent})
"""
        self.set_implementation_template(agent_instantiation_template)

        self.render_implementation()
        return self.get_implementation()

    def render_agent_import(self):
        self.set_implementation_template(
            """
from Agent_$agent_type import *
"""
        )
        self.render_implementation()
        return self.get_implementation()
```

Listing 9.8: The `mago_agent.py` file

```
from mago_thing import *


class Behaviour(Thing):
    """A class containing all the data describing a MAGO agent behaviour. Contains all the methods and attributes
   ↪ common to all the behaviour types."""

    def __init__(self, cycling: bool = False, period: int = None, *args, **kwargs):
        super(Behaviour, self).__init__(entity_type="behaviour", *args, **kwargs)

        if self.onto_individual is None:
            raise ValueError("Ontology individual must be supplied.")

        self.cycling = self.onto_individual.is_repeating
        self.period = self.onto_individual.has_period
        self.behaviour_type: str = self.determine_behaviour_type()

    def determine_behaviour_type(self):
        """Determines the type of behaviour based on the ontology individual and agent attributes.

        Returns:
            str: The name of the behaviour type.
        """
        if self.onto_individual.is_before_state or self.onto_individual.is_after_state:
            return "State"
        elif self.onto_individual.has_initial_state:
            return "FSMBehaviour"
        else:
            key = (bool(self.cycling), bool(self.period))
            behaviour_mapping = {
```

```python
                (False, False): "OneShotBehaviour",
                (False, True): "TimeOutBehaviour",
                (True, False): "CyclicBehaviour",
                (True, True): "PeriodicBehaviour",
            }
            return behaviour_mapping.get(key, "UnknownBehaviour")

    def get_fsm_states(self):
        states = set()
        transitions = []
        visited = set()
        initial_state = None

        initial_state = self.onto_individual.has_initial_state

        stack = list(initial_state)
        while stack:
            current_state = stack.pop()
            if current_state in visited:
                continue
            visited.add(current_state)
            states.add(current_state)

            # Get next states
            next_states = current_state.is_before_state
            transitions.extend([(current_state, ns) for ns in next_states])
            stack.extend(next_states)

        return initial_state, states, transitions

    def render_fsm_implementation(self):
        initial_state, states, transitions = self.get_fsm_states()

        implementation = []

        state_names = {}
        state_names.update(
            {
                state: (
                    clean_string(state.has_name)
                    if state.has_name
                    else clean_string(state.name)
                )
                for state in states
            }
        )

        print(state_names)

        for state in states:
            state_name = state_names.get(state)
            class_name = f"{state_name}()"
            is_initial = state in initial_state
            if is_initial:
                code_line = f"self.add_state(name='{state_name}', state={class_name}, initial=True)"
            else:
                code_line = f"self.add_state(name='{state_name}', state={class_name})"
            implementation.append(code_line)

        for source_state, dest_state in transitions:
            source_name = state_names.get(source_state)
            dest_name = state_names.get(dest_state)
            code_line = (
                f"self.add_transition(source='{source_name}', dest='{dest_name}')"
            )
            implementation.append(code_line)

        # Join the code lines into a single string
        return textwrap.indent(text="\n".join(implementation), prefix="        ")

    def prepare_behaviour_implementation_template(self):
        template = [
            """
class $name(${behaviour_type}):"""
        ]

        template.append(
            """
    async def on_start(self) -> None:
        print("Starting behaviour.")"""
        )

        template.append(
            """
    async def on_end(self) -> None:
        print("Ending behaviour.")"""
        )

        if "FSM" not in self.behaviour_type:
            template.append(
                """
    async def run(self) -> None:
        print("Running the behaviour.")"""
            )

        if "FSM" in self.behaviour_type:
            template.append(
                f"""
    async def state_setup(self):
{self.render_fsm_implementation()}
                """
            )

        return "\n".join(template)

    def render_behaviour_implementation(self):
        implementation_template = self.prepare_behaviour_implementation_template()
        self.set_implementation_template(implementation_template)
        self.render_implementation()
        implementation = self.get_implementation()
        logging.info(f"Behaviour {self.name} implementation rendered.")
        return implementation
```

Listing 9.9: The `mago_behaviour.py` file

```python
from mago_thing import *


class Plan(Thing):
    def __init__(self, *args, **kwargs):
        super().__init__(entity_type="plan", *args, **kwargs)

    def get_plan_action_behaviour_objective(self):
        plan_dict = {}

        plan_iri = self.onto_individual.iri

        # Get the plan name
        plan_name = (
            self.onto_individual.has_name
            if self.onto_individual.has_name
            else self.onto_individual.name
        )

        # Initialize the plan entry
        plan_entry = {"name": plan_name, "actions": {}}

        # Get the actions associated with the plan
        actions = self.onto_individual.requires_action
        for action in actions:
            action_iri = action.iri
            # Get the action name
            action_name = action.has_name if action.has_name else action.name

            # Initialize the action entry
            action_entry = {"name": action_name, "objectives": {}, "behaviours": {}}

            # Get the objectives associated with the action
            objectives = action.has_objective
            for objective in objectives:
                objective_iri = objective.iri
                # Get the objective name
                objective_name = (
                    objective.has_name if objective.has_name else objective.name
                )

                # Add the objective to the action's objectives
                action_entry["objectives"][objective_iri] = objective_name

            # Get the behaviours associated with the action
            behaviours = action.has_behaviour
            for behaviour in behaviours:
                behaviour_iri = behaviour.iri
                # Get the behaviour name
                behaviour_name = (
                    behaviour.has_name if behaviour.has_name else behaviour.name
                )

                # Add the behaviour to the action's behaviours
                action_entry["behaviours"][behaviour_iri] = behaviour_name

            # Add the action entry to the plan's actions
            plan_entry["actions"][action_iri] = action_entry

        # Add the plan entry to the main dictionary
        plan_dict[plan_iri] = plan_entry

        logging.info(f"Plan {plan_name} visited.")

        return plan_dict
```

Listing 9.10: The `mago_plan.py` file

```python
# from owlready2 import Ontology
from mago_thing import *
from mago_agent import Agent
from mago_behaviour import Behaviour
from mago_plan import Plan


class Workspace(Thing):
    """The world containing general system data and agent instantiation."""

    def __init__(self, ontology: Ontology = None, *args, **kwargs):
        super().__init__(entity_type="workspace", *args, **kwargs)
        self.agents: dict[str, list[Agent]] = {}
        self.agent_import_sources = None
        self.agent_instantiation = None
        self.behaviours_rendered = None
        self.onto = ontology

    def add_agent_to_list(self, agent: Agent):
        self.agents.setdefault(agent.is_a[0].label[0], []).extend(agent)

    def get_list_of_agents(self):
        return self.agents

    def read_agents_from_ontology(self, onto: Ontology = None):
        if onto is None:
            onto = self.onto

        agent_classes = onto.search(subclass_of=onto.search(label="Agent"))
        for agent_class in agent_classes:
            agents = onto.search(type=agent_class)
            self.agents.setdefault(agent_class.label[0], []).extend(
                [
                    Agent(
                        agent_type=agent.is_a[0].label[0],
                        host_server=agent.lives_on_host[0].label[0],
                        name=agent.has_name,
                        password="tajna",
                        onto_individual=agent,
                    )
                    for agent in agents
                ]
            )

    def render_behaviours_from_ontology(self, onto: Ontology = None):
        if onto is None:
            onto = self.onto

        has_name = onto.search_one(label="has name")

        behaviours = onto.search(type=onto.search_one(label="Behaviour"))

        behaviours_mago = [
            Behaviour(
                cycling=behaviour.is_repeating,
                period=behaviour.has_period if behaviour.has_period else None,
                onto_individual=behaviour,
            )
            for behaviour in behaviours
        ]

        self.behaviours_rendered = "\n\n\n".join(
            [
                behaviour.render_behaviour_implementation()
                for behaviour in behaviours_mago
            ]
        )

        return self.behaviours_rendered

    def read_plan_from_ontology(self, onto: Ontology = None):
        if onto is None:
            onto = self.onto

        result = {}

        plan = onto.search_one(
            label="Plan"
        ).instances()

        for plan in plan:
            result.update(
                Plan(
                    onto_individual=plan
                ).get_plan_action_behaviour_objective()
            )

        return result

    def write_behaviour_implementations_to_file(self):
        if not self.behaviours_rendered:
            self.render_behaviours_from_ontology()

        file_name = os.path.join(os.getcwd(), "Template", "Behaviours.py")
        with open(file_name, "w") as file:
            file.write("from spade.behaviour import *\n\n")
            file.write(self.behaviours_rendered)

    def render_agent_import_sources(self):
        if not self.agents:
            self.read_agents_from_ontology()

        self.agent_import_sources = "\n".join(
            [agents[0].render_agent_import() for agents in self.agents.values()]
        )

    def render_agent_instantiation(self):
        if not self.agents:
            self.read_agents_from_ontology()

        self.agent_instantiation = textwrap.indent(
            "\n".join(
                [
                    agent.render_agent_instantiation()
                    for agent in [
                        agent
                        for agent_class in self.agents.values()
                        for agent in agent_class
                    ]
                ]
            ),
            "    ",
        )

    def write_agent_implementations_to_files(self):
        for agents in self.agents.values():
            agents[0].render_agent_implementation()
            agents[0].write_implementation_to_file()

    def render_world_implementation(self):
        self.set_implementation_template(
            """
import spade
$agent_import_sources


async def main():
    agent_individuals = {}
$agent_instantiation
    for agent in [agent for host_dict in agent_individuals.values() for agent in host_dict.values()]:
        agent.plan_action_behaviour_objective = $plan_action_behaviour_objective
        await agent.start()

spade.run(main())
"""
        )

        self.plan_action_behaviour_objective = self.read_plan_from_ontology()

        self.render_agent_import_sources()
        self.render_agent_instantiation()

        self.render_implementation()
        return self.get_implementation()

    def write_implementation_to_disk(self):
        if not self.implementation:
            self.render_world_implementation()

        self.write_behaviour_implementations_to_file()

        self.write_agent_implementations_to_files()

        self.write_implementation_to_file()

        logging.info(f"{self.name} system successfully written to disk.")
```

Listing 9.11: The `mago_workspace.py` file

```
import os
import argparse
from owlready2 import World, Ontology, onto_path, set_render_func, sync_reasoner
from mago_workspace import Workspace


def render_using_label(entity):
    return entity.label.first() or entity.name


def render_using_iri(entity):
    return entity.iri


def main(ontology_name="MAGO-Ag.owx"):
    onto_path.append(os.getcwd())

    mago_world = World()
    onto: Ontology = mago_world.get_ontology(ontology_name).load(reload=True)

    set_render_func(render_using_label)

    sync_reasoner()

    template_folder = os.path.join(os.getcwd(), "Template")
    if not os.path.exists(template_folder):
        os.makedirs(template_folder)

    aMAGOWorld = Workspace(ontology=onto, name="World")
    aMAGOWorld.write_implementation_to_disk()


if __name__ == "__main__":
    parser = argparse.ArgumentParser(
        description="MAGO-Ag ontology translation framework"
    )
    parser.add_argument(
        "-o",
        "--ontology-name",
        type=str,
        default="MAGO-Ag.owx",
        help="Optional name of the ontology file to be used, string argument with default value 'MAGO-Ag.owx'",
    )

    args = parser.parse_args()
    main(args.ontology_name)
```

Listing 9.12: The main script of the framework

Bibliography

[1] M. Fernández-López, A. Gómez-Pérez and N. Juristo, 'METHONTOLOGY: From Ontological Art Towards Ontological Engineering,' in *Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series*, Stanford US-CA: Stanford University, 1997, pp. 33–40.

[2] B. Okreša Đurić, 'Organizational Modeling of Large-Scale Multi-Agent Systems with Application to Computer Games,' Doctoral thesis, University of Zagreb, Varaždin HR, 2018, 236 pp.

[3] R. Iqbal, M. A. A. Murad, A. Mustapha and N. M. Sharef, 'An Analysis of Ontology Engineering Methodologies: A Literature Review,' *Research Journal of Applied Sciences, Engineering and Technology*, vol. 6, no. 16, pp. 2993–3000, 2013, ISSN: 20407459.

[4] M. Uschold and M. Gruninger, 'Ontologies: Principles, Methods and Applications,' *The Knowledge Engineering Review*, vol. 11, no. 2, pp. 93–136, 1996. DOI: 10.1017/S0269888900007797.

[5] S. J. Russell and P. Norvig, Eds., *Artificial Intelligence: A Modern Approach* (Pearson Series in Artificial Intelligence), 4th ed. Harlow, UK: Pearson Education Limited, 2022, 1166 pp., ISBN: 978-1-292-40113-3.

[6] M. Fontana and P. Terna, 'From Agent-based models to network analysis (and return): The policy-making perspective,' Department of Economics and Statistics "Cognetti de Martiis", University of Turin, Torino, IT, 201507, 2015, pp. 1–19.

[7] M. E. Gregori, J. P. Cámara and G. A. Bada, 'A jabber-based multi-agent system platform,' in *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems - AAMAS '06*, New York US-NY: ACM Press, 2006, p. 1282, ISBN: 1-59593-303-4. DOI: 10.1145/1160633.1160866.

[8] B. Okreša Đurić, J. Rincon, C. Carrascosa, M. Schatten and V. Julian, 'MAMbO5: A new Ontology Approach for Modelling and Managing Intelligent Virtual Environments Based on Multi-Agent Systems,' *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 9, pp. 3629–3641, 2019, ISSN: 1868-5145. DOI: 10.1007/s12652-018-1089-4.

[9] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach* (Prentice Hall Series in Artificial Intelligence), 3rd ed., S. Russell and P. Norvig, red. New Jersey, USA: Prentice Hall, 2010, 1132 pp., ISBN: 978-0-13-604259-4.

[10] M. Schatten, 'Organizational Architectures for Large-Scale Multi-Agent Systems' Development: An Initial Ontology,' *Advances in Intelligent Systems and Computing*, vol. 290, S. Omatu, H. Bersini, J. M. Corchado, S. Rodŕiguez, P. Pawlewski and E. Bucciarelli, Eds., pp. 261–268, 2014. DOI: 10.1007/978-3-319-07593-8_31.

[11] M. Schatten, P. Grd, M. Konecki and R. Kudelić, 'Towards a Formal Conceptualization of Organizational Design Techniques for Large Scale Multi Agent Systems,' *Procedia Technology*, vol. 15, pp. 576–585, 2014, ISSN: 22120173. DOI: 10.1016/j.protcy.2014.09.018.

[12] E. Argente, J. Palanca, G. Aranda *et al.*, 'Supporting Agent Organizations,' in *Multi-Agent Systems and Applications V*, H.-D. Burkhard, G. Lindemann, R. Verbrugge and L. Z. Varga, Eds., ser. Lecture Notes in Artificial Intelligence 4696, Leipzig, DE: Springer, 2007, pp. 236–245, ISBN: 978-3-540-75254-7. DOI: 10.1007/978-3-540-75254-7_24.

[13] J. A. Rincon, E. Garcia, V. Julian and C. Carrascosa, 'Developing adaptive agents situated in intelligent virtual environments,' in *Hybrid Artificial Intelligence Systems*, ser. Lecture Notes in Computer Science 8480, vol. 8480 LNAI, Cham, CH: Springer, 2014, pp. 98–109, ISBN: 9783319076164. DOI: 10.1007/978-3-319-07617-1_9.

[14] J. A. Rincon, C. Carrascosa and E. Garcia, 'Developing intelligent virtual environments using MAM5 meta-model,' in *Advances in Practical Applications of Heterogeneous Multi-Agent Systems. The PAAMS Collection*, Y. Demazeau, F. Zambonelli, J. M. Corchado and J. Bajo, Eds., ser. Lecture Notes in Computer Science 8473, Cham, CH: Springer, 2014, pp. 379–382, ISBN: 9783319075501. DOI: 10.1007/978-3-319-07551-8_43.

[15] M. Luck and R. Aylett, 'Applying Artificial Intelligence to Virtual Reality: Intelligent Virtual Environments,' *Applied Artificial Intelligence*, vol. 14, no. 1, pp. 3–32, 2000, ISSN: 0883-9514, 1087-6545. DOI: 10.1080/088395100117142.

[16] M. A. Mahmoud, M. S. Ahmad, M. Z. Mohd Yusoff and A. Mustapha, 'A Review of Norms and Normative Multiagent Systems,' *The Scientific World Journal*, vol. 2014, pp. 1–23, 2014, ISSN: 2356-6140. DOI: 10.1155/2014/684587. pmid: 25110739.

[17] D. Villatoro, 'Self-organization in Decentralized Agent Societies Through Social Norms,' in *The 10th International Conference on Autonomous Agents and Multiagent Systems*, vol. 3, Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2011, pp. 1373–1374, ISBN: 978-0-9826571-7-1.

[18] J. S. Coleman, *Foundations of Social Theory*. Harvard University Press, 1998, 993 pp., ISBN: 978-0-674-31226-5.

[19] G. Boella, L. van der Torre and H. Verhagen, 'Introduction to Normative Multiagent Systems,' *Computational & Mathematical Organization Theory*, vol. 12, no. 2-3, pp. 71–79, 2006, ISSN: 1862-4405. DOI: 10.1007/s10588-006-9537-7.

[20] J.-J. C. Meyer and R. J. Wieringa, Eds., *Deontic Logic in Computer Science: Normative System Specification*. Chichester, UK: John Wiley and Sons Ltd., 1993, ISBN: 0-471-93743-6.

[21] K. M. Carley and L. Gasser, 'Computational Organization Theory,' in *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Ed., Cambridge, MA, USA: MIT Press, 1999, ch. Computatio, pp. 299–330, ISBN: 0-262-23203-0.

[22] M. Schatten, J. Ševa and I. Tomičić, 'A roadmap for scalable agent organizations in the Internet of Everything,' *Journal of Systems and Software*, vol. 115, pp. 31–41, 2016, ISSN: 01641212. DOI: 10.1016/j.jss.2016.01.022.

[23] B. Okreša Đurić, 'From Ontology to Action: Streamlining Multiagent System Development with SPADE,' Invited Lecture, presented at the AI2FUTURE (Zagreb, HR), 18/10/2024.

Notes

1. http://www.investinganswers.com/financial-dictionary/stock-valuation/acquisition-2224
2. For more information visit https://www.investopedia.com/terms/a/acquisition.asp
3. http://www.dictionary.com/browse/goal
4. For more information visit http://www.dictionary.com/browse/merger
5. https://medium.com/the-mission/why-you-should-change-your-goals-into-quests-2467bbef9867
6. define further, i.e. a main quest in a game can consist of several tasks, which can be quests themselves