Journal of Universal Computer Science, vol. 20, no. 13 (2014), 1766-1790
submitted: 4/1/14, accepted: 6/9/14, appeared: 28/11/14 © J.UCS

Semantic Based Support for Planning Information Delivery in Human-agent Collaborative Teams

Natasha Lino
(The University of Edinburgh, Edinburgh, United Kingdom
and Federal University of Paraíba, João Pessoa, Brazil
natasha@ci.ufpb.br)

Clauirton Siebra
(Federal University of Paraíba, João Pessoa, Brazil
clauirton@ci.ufpb.br)

Austin Tate
(Artificial Intelligence Application Institute, School of Informatics
The University of Edinburgh, Edinburgh, United Kingdom
a.tate@ed.ac.uk)

Abstract: Collaborative teams are organizations where joint members work together to solve mutual goals. Mixed-initiative planning systems are useful tools in such situations, because they can support several common activities performed in these organizations. However, as collaborative members are involved in different decision making planning levels, they consequently require different information types and forms of receiving planning information. Unfortunately, collaborative planning delivery is a subject that has not been given much attention by researchers, so that users cannot make the most of such systems since they do not have appropriate support for interaction with them. This work presents a general framework for planning information delivery, which is divided into two main parts: a knowledge representation aspect based on an ontological set and a reasoning mechanism for multimodality visualization. This framework is built on a mixed-initiative planning basis, which considers the additional requirements that the human presence brings to the development of collaborative support systems.

Keywords: Intelligent Planning, Ontology Design, Multiagent Systems
Categories: H.5.2, **:**M.4

1 Introduction

The principal feature of a collaborative team is the existence of a global goal, which motivates the activities of all its joint members. However, normally such members are not directly involved in the resolution of this goal, but in sub-tasks associated with it. Considering the diversity of such sub-tasks, it is natural that collaborative members carry out different planning and plan execution activities at different decision levels. The use of planning assistant agents [Kim, 04; Wickler, 06] is an appropriate option to support collaborative members in this decision structure. Agents can extend the human abilities and be customised for different planning activities performed along different decision-making levels. As these activities are different, collaborative

members require different types of planning information and methods to receive such information. Unfortunately, planning delivery is an aspect that is still not widely explored in the planning literature [Ghallab, 04]. Although many efforts have been made towards improving and developing new techniques and approaches for planning, they are centred in core planning problems, such as the efficiency of search algorithms, and few works particularly address the problem of visualisation.

With the transition from planners working in isolation in the past to the more recent mixed-initiative approaches [Ai-Chang, 04], it is evident that there is a need for new forms of interaction between human and software planners. In such systems, new requirements emerge [Penalver, 13] since the agents that are collaborating in the process have different backgrounds, play different roles and have different capabilities, responsibilities, etc. From a planning activity perspective, visualization can play two crucial roles: to support collaboration among participant agents in the case of a collaborative task and to allow proper interfacing between the software and human planners. However, the lack of more generic and elaborated approaches compromises a broader application and use of such systems. Furthermore, it also compromises their use in real world problem domains and situations where assistant planning services could be applied and supported by more sophisticated visualization approaches.

This paper proposes a general framework for planning delivery that aims at supporting an appropriate delivery mechanism regarding the requirements we are considering. The essence of this framework is based on the semantic modelling of the problem under the perspective of visualisation in planning systems. The framework is divided into two main parts: a knowledge representation aspect and a reasoning mechanism. In the knowledge representation aspect, the ontology set enables the organization and modelling of complex problem domains from the visualization perspective. The reasoning mechanism gives support for reasoning about the visualisation problem using the knowledge bases available for describing realistic collaborative environments. This framework is built on a mixed-initiative planning basis, which considers the additional requirements that the human presence brings to the development of collaborative planning agents. However these requirements are specified from the planning process perspective and they originally did not consider the information delivery aspect.

The remainder of this paper is organized as follows: Section 2 summarises the main works in planning visualization, stressing their principal features and limitations. Section 3 details our framework for planning information delivery in two parts. First we discuss the semantic modelling approach, which consists in an integrated ontology set for describing planning information from a visualisation perspective. Second, we give attention to the reasoning mechanism, which uses knowledge about the domain, described via the ontology set, to infer modalities of visualisation to a plan or parts of it. Section 4 exemplifies the use of this framework in an application domain, based on a disaster relief operation, where several agents are carrying out different tasks in a collaborative environment. Finally, Section 5 concludes this work, highlighting the contributions and research directions.

2 Visualization in Planning Systems

According to Kautz and Selman [Kautz, 98a], there are three types of planning knowledge, which must be presented by a planning system: knowledge about the domain, knowledge about good plans and explicit search-control knowledge. Later on and supported by their experiences in planning for military and oil spill domains, the work of Wilkins and des Jardins [Wilkins, 01] extended this list about planning knowledge mentioning that knowledge-based planners should also deal with: knowledge about interaction with users; knowledge about user's preferences and knowledge about plan repair during execution.

Based on these discussions of knowledge enrichment and broader use of knowledge based planning, we argue that this vision should be even more augmented to cover other aspects. Our call is that knowledge enhancement could also consider other aspects related to planning, such as planning information visualisation aspects. We claim that knowledge models, developed from the AI planning information visualisation perspective, are able to provide semantic support and reasoning to cover some of the existing gaps in the area and open it to a broad diversity of other services. Some of the existing gaps and problems that can be identified in the area of planning information visualisation are briefly introduced below:

- Absence of solutions: many existing and awarded planning systems do not even have an approach for information visualisation, such as the Graphplan [Blum, 97] and Blackbox [Kautz, 98b] planners;

- Lack of flexibility: the current solutions for visualisation in planning systems, in general, adopt only one solution for presenting information when, in some cases, it is not appropriate for every situation. The PRODIGY system [Veloso, 95], for example, adopts only a GUI (Graphical User Interface) approach, while the TRAINS [Allen, 01a] and TRIPS [Allen, 01b] planners mainly use a natural language based solution, together with restrict map based solutions. Nevertheless, these solutions do not suit all different cases in real world domains of planning;

- Design for a specific aspect of the planning process: visualisation approaches used in AI planning systems sometimes do not give support to the entire planning process (including domain modelling, generation, collaboration, replanning and execution) but, frequently, only to part of the process. There is a need to find general approaches to support planning information visualisation that will permit an uniform and integrated use of such approach for the development of solutions to every aspect of the planning process;

- Visualisation directly associated with the planning approach: information visualisation in some planning systems is closely attached to the planning approach and related aspects, such as the domain of application, the paradigm or search algorithm for planning, the plan representation method, the plan product, integration to scheduling, etc. For instance, it is common in integrated planners and schedulers to show temporal information, due to the nature of information that such systems manipulate. This fact limits the

broad use and scope for interaction with other systems. Furthermore, services that they can potentially provide are limited by the visualisation approach.

The issues discussed above make evident that there is a need of more global mechanisms that will provide general solutions for planning information visualisation. In this way, they open many research opportunities. Some examples are:

- Development of more general frameworks: general frameworks will give support to different planning paradigms regarding information visualisation. This would permit a broader flexibility and increase usability and portability;
- Use and integration of different modalities for information visualisation (multimodal approach): the integration and use of different modalities of information visualisation (such as textual, graphical, natural language, virtual reality, etc.) will permit an appropriate use of each modality in different situations. For example, in situations where users are executing some task that does not allow them to pay attention to the screen (visual based mechanisms for information visualisation), sound could be used as an alternative approach;
- Address issues regarding collaboration and different types of users involved in the process: some situations and scenarios require collaboration between users to solve problems in a mixed-initiative style of planning. This leads to the question of different types of users (or human agents) taking part in the process. Human agents may have different backgrounds, capabilities, authorities and preferences when working in a collaborative planning environment;
- Mobile computing for realistic collaborative environments: information visualisation aimed at mobile devices can play an important role. In realistic environments, human agents may need mobility to perform their tasks in the process. Thus, the idea of delivering information to mobile devices can support the planning process in many ways, from generation to execution of plans.

All these points discussed above were considered in our approach and they will be detailed in the next sections.

Despite the advances in AI Planning in the last decades, plan visualisation still is an area in AI planning that is scarcely investigated. A few works address questions regarding information visualisation for planning in a more effective way, e.g. [Gerevini, 08], [Gerevini, 11] and [Daley, 2005]. Garevini and Saetti work [Gerevini, 08][Gerevini, 11] presents a planning environment that supports plan visualization and mixed-initiative plan generation, in which the user can interact with the planner. In [Daley., 2005] the problem of planning debugging is addressed via visualisation. It is proposed a browse-based system for debugging constraint-based planning and scheduling systems where the visualization components consist of specialized views to display different forms of data (e.g. constraints, activities, resources, and causal links). However these approaches are closed related to the planning system, they are designed to a specific aspect of the planning process and lack in flexibility. Our own work on the O-Plan "PlanWorld" viewers [Tate, 95] and the I-X/I-Plan plug-in

viewers for elements of the <I-N-C-A> ontology [Tate, 14a] were intended to provide a flexible approach to support different planning user roles and multiple styles and modalities of plan presentation.

The idea of modelling components of systems based on ontologies is increasing in the literature. Research groups are exploring this concept from different perspectives. An approach of presenting a device model as an ontology to allow mobile communication appears in [Chen, 04] and an ontological framework for semantic description of devices is shown in [Bandara, 04]. A software engineering work about ontology-based device modelling for embedded systems development process that allows flexibility is discussed in [Thamboulidis, 07a] and [Thamboulidis, 07b].

Regarding a more general perspective analysis of related works, groups are building and applying ontologies for knowledge management as in [Mahmoudi, 07] or with the goal of sharing conceptual engineering knowledge [Mizoguchi, 00] as examples.

3 The Information Delivery Approach

This section introduces the framework proposed for semantic support for information delivery in collaborative domains. Using semantic modelling techniques (ontologies), several knowledge models complement each other to structure a planning delivery knowledge model. Based on that model, a reasoning mechanism outputs delivery methods, tailored for each situation. Section 3.1 details the semantic modelling, while Section 3.2 discusses the reasoning mechanism.

3.1 Semantic Modelling

The semantic modelling concerns the following sub-ontologies: Multi-Modality Visualisation Ontology, Planning Information Ontology, Devices Ontology, Agents Ontology (Organisation and Mental States) and Environment Ontology. For the development of the ontologies, the concepts were based on both existing models and models that were developed to attend the requirements of the problem that we are trying to cover.

The **Multi-Modal Visualisation Ontology** enables us to express the different modalities of delivery considered in this approach. As the essence of the framework is to be generic, a broad range of modalities are considered. The definition of this model is based on previous classifications of information visualisation categories existing in the literature [Card, 99], while it also tries to incorporate a diversity of modalities that fulfil the framework's requirement of generality. The model has three main concepts defined by the following classes (and their respective children in the class hierarchy): Multi-Modality, Interface Component and Interface Operator.

Regarding the *Multi-Modality* conceptualisation (Figure 1), at the first level, the information visualisation modalities are categorised into simple structured and complex structured classes. At the second level, however, the modalities are categorised according to their dimensional representation. At the final level, the own modalities are categorised. This model contains the following modalities of

information delivery: Textual, Sound, Tabular, Graphical, Map-Based, Spatial, Virtual Reality, Tree, Network, Temporal and Natural Language.

The second main concept in the semantic modelling is the Interface Component. This class (and its children) is related to the Multi-Modality class by the restriction "Multi-Modality hasComponent InterfaceComponent". That means, an instance of the Multi-Modality class has at least one (is related to) Interface Component. For example, a textual modality of information visualisation would have text as interface component. In other words, each of these components acts as primitive elements during the creation of a specific interface.

The definitions regarding the classes' hierarchy are: Interface Component, Plain Component, Structured Component, Text Component, Sound Component, Voice Component, Table Component, Graphical Component, Map Component, Three Dimensional Graphical Component, Virtual Reality Component, Tree Component, Network Component, Time Component and Natural Language Component [Lino, 07].

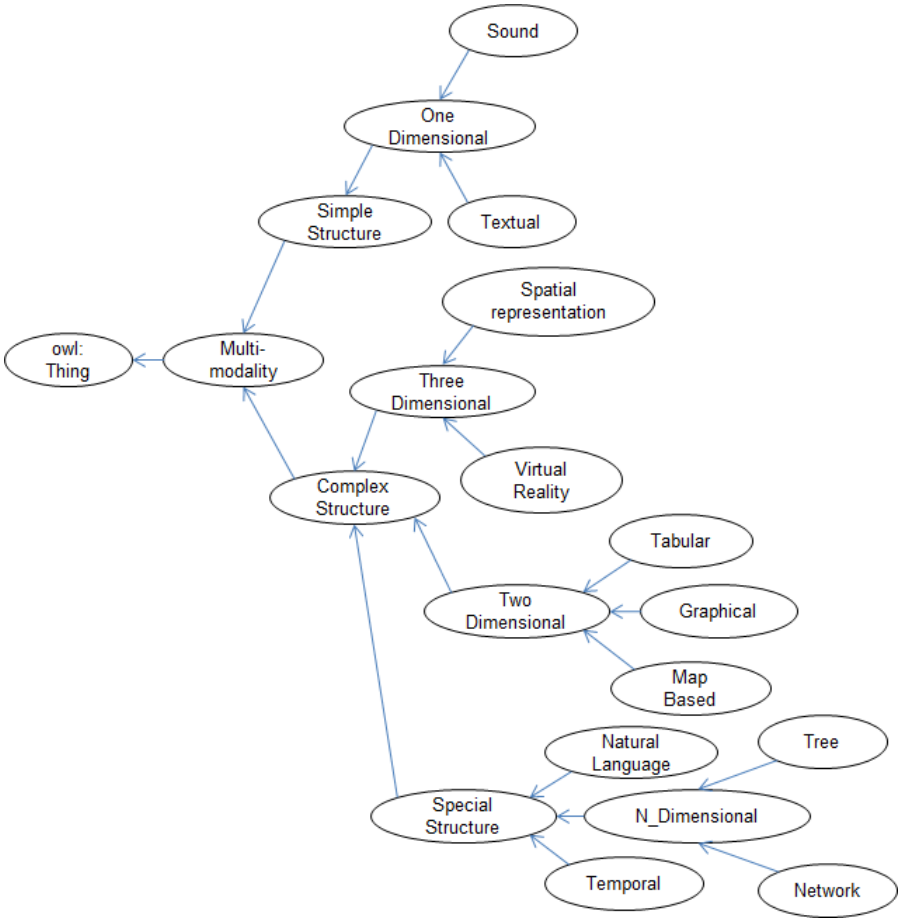

Figure 1: Multi-Modal visualisation ontology classes hierarchy. Note that other classes and details were hidden here for legibility reasons

The last main concept to be discussed in the Multi-Modal Visualisation Ontology is Interface Operator. This class (and its children) is related to the Multi-Modality class by the restriction "Multi-Modality hasOperator InterfaceOperator". That means that an instance of the Multi-Modality class has (is related to) an Interface Operator. For instance, a map modality of information visualisation may have zoom as interface operator. This class hierarchy conceptualises the operations that can be performed by users on information visualisation modalities. The concepts regarding the classes' hierarchy of Interface Operator are described below:

- Obtain Details: selects an item or group and gets details when needed;
- Extract: allows extraction of sub-collections and query parameters;
- Filter: filters out uninteresting items;
- Obtain History: keeps a history of actions to support undo, replay and progressive refinement;
- Overview: creates an overview of the entire collection;
- Relate: views relationships among items;
- Zoom: zooms in/out on items of interest.

The **Planning Information Ontology** models information related to the planning process. It categorises, at a high level, planning information as one of the following natures:

- Domain Modelling: this category includes concepts of planning information related to domain modelling, involving, for example, description of goals, resources, etc;
- Plan Generation: in this nature, the semantic modelling is concerned with plan generation information concepts and abstractions;
- Planning Execution: includes vocabulary regarding information about planning execution.

This ontology is based on <I-N-C-A> (Issues-Nodes-Constraints-Annotations), a general-purpose ontology that can be used to represent synthesised artefacts, such as plans and designs, in the form of a set of constraints on the space of all possible artefacts in the application domain [Tate, 03]. Each plan represented via <I-N-C-A> is made up of a set of issues, a set of nodes and a set of constraints, which relate these nodes and objects in the application domain. Annotations can be added to the overall plan, as well as any of its components.

The main focus of the planning information ontology is on allowing a generic conceptualisation of planning information, so that the visualisation process can reason about the plan components (activities, constraints, etc.) and decide on the best option to show this plan. The clear specification provided by <I-N-C-A> supports this process because the components are explicitly represented. We consider that planning information can be used to meet different aims such as planning modelling, generation and execution.

According to the literature and existing planning systems, planning information is approached in different ways, depending on the aim. Thus, information delivery for planning modelling is not the same as delivery for planning generation. Using <I-N-C-A> we can easily identify the plan components that are most related to the current aim. For example, if the system is in the execution stage, some important information to be displayed corresponds to the report-back of activities and their progress status. Apart from the planning aim, it is possible to identify and classify planning information via the analysis of an instance of the model. For example, we can identify a group of temporal constraints, which have a different strategy of visualisation, if we compare these constraints with world-state constraints or a set of annotations. All decisions, based on a particular plan description, will be performed by the reasoning mechanism (Section 3.2), which needs to present an understanding of planning information from a visualisation perspective. Note however, that such reasoning and decision making process is performed after considering the entire context, which is modelled via the ontologies presented in this section.

Works on **Devices Ontology** [Lino, 03] have been investigating approaches for knowledge representation of devices capabilities and preferences concepts. An important example of such ontology is the CC/PP [W3C, 04], a W3C standard for device pro- filing. The approach of CC/PP has many positive aspects. First, it can serve as a basis to guide adaptation and content presentation. Second, from the knowledge representation point of view, since it is based on RDF, it is a real standard and permits integration with the concepts of the Semantic Web construction [Fahad, 11]. Third, another advantage of CC/PP is the resources for vocabulary extension, although extensibility is restricted. On the other hand, CC/PP has a limited expressiveness power that does not permit a broader semantic representation and, consequently, it restricts reasoning possibilities. Based on this investigation, we have incorporated other semantic elements to CC/PP model, which enhance the CC/PP representation and semantics [Lino, 07].

The **Agent Ontology** intends to satisfy the needs for reasoning about agents (software and human) roles in the organisation when participating in collaborative processes of planning and all aspects related to it. The concepts modelled in this ontology, and how they influence in the visualisation, are: mental states, roles, relationships and preferences profiling. The development of this ontology is based on two existing model concepts: BDI [Thangarajah, 08] and I-Space [Tate, 03]. BDI (Belief-Desire-Intention) is the most popular concept used in the agent-based modelling and programming. Each agent has its own BDI model and, in order to achieve some goal (Desire), the agent can analyse its related data (Belief) and choose an appropriate plan (Intention). The I-Space approach supports the arrangement of collaborative teams, allowing the management of organisational relationships such as superior-subordinate or peer-peer. Considering an agent *ag*, I-Space shows the kind of relationship that *ag* has with other agents of the team (superior, subordinate or peer). For each of these relationships we can associate specific forms of interaction, which specifically characterise each relationship. In addition, I-Space also shows the capabilities of each agent that composes the contact list of *ag*.

Finally, the **Environment Ontology** accounts for enabling the expression of environment awareness. The main concept modelled in this ontology is *Geographic Location*. According to our model, every environment should have a location system,

which can be one of following four classes: GPS, reference-based, descriptive or special. GPS gives the location of objects via the latitude and longitude attributes. The reference-based system gives the position of every object in the environment as the orthogonal distance (axis x, y and z) between this object and a referential fixed point. The descriptive system is represented by a natural language description of a position or place. This category can be decomposed into two subclasses: formal and informal descriptions. The formal description is mainly represented by addresses. The informal description does not have a pre-defined format and can look like: *I am in the Highlands on the West shore of Loch Ness, four kilometres south of the Urquhart Castle*. Special location systems are associated with environments where the representation of objects is given in a more complex way. Deep-space exploration missions are examples of domains where the environment, in this case the space, does not have a common way to represent positions of its objects. Thus, different approaches for each case must extend this class to define appropriate location systems.

It is important to note how different location systems can affect the visualisation decision process. Consider, for example, the use of a referential-based system during a rescue operation inside a big building such as a tower with several levels. For this scenario, a 3D representation is the most appropriate strategy due to the importance of the three dimensions during navigation inside this building. In another example where we have an informal description of a location, a textual visualisation could be a simpler way to deliver this information, due to the fact that the reasoning may not place this position in a map or any other visualisation resource.

In the modelling methodological process followed, it would be possible to consider a foundational ontology for providing basic concepts, and to support all domain-specific ontologies to be built. An example of such ontologies is the UFO [Guizzardi, 04] ontology proposed by Guizzardi. However, this work considered a set of ontologies and its vertical growth of concepts and relationships among these concepts was substantial. In addition, not using an upper ontology was not an impediment for the purpose of this work, despite the benefits of formal semantics and model checking that it would add for instance. Considering the life cycle of the ontology set proposed in this work, the refinement of these ontologies according an upper ontology is possible, and will surpass some of the limitations of this work. The refinement of these ontologies, the possible adoption of an upper ontology and the modelling details will be explored in another paper. In this paper the focus was in the big picture of the proposed work.

3.2 The Reasoning Mechanism

The reasoning approach performs information delivery decisions via a Production System [Cao, 10], where a set of rules represents the knowledge about which is the most appropriate form of visualisation in a specific context. This context is specified in a pre-defined way via the ontologies described in the previous section. A Production System is a specific class of rule-based systems that consists of a set of IF-THEN rules, a set of facts, and an interpreter that controls the application of the rules according to the facts. The left hand side of a rule contains information about certain facts and objects, which must be true in order for the rule to potentially execute. Any rule whose left hand side matches is placed on an agenda. Then, when one of the rules

on the agenda is picked, its right hand side (implication) is executed in the agenda. The agenda is then updated and a new rule is picked to execute. This process continues until there are no more rules on the agenda. In a more formal way, the elements and operation of a production system can be defined as follows:

- Production systems: work memory (WM) + production rule set (PRS);
- Work memory: set of work memory elements (WME), where each WME is of the form (type $attribute_1$ $value_1$ $attribute_2$ $value_2$ … $attribute_n$ $value_n$), where *type*, $attribute_i$ and $value_i$ are atomic elements. For example:

(Device-x displaySize 10 memory 4Gb).

This construction can be understood as:

$\exists x [type(x) \wedge attribute_1(x) = value_1(x) \wedge attribute_2(x) = value_2(x) \wedge \ldots \wedge attribute_n(x) = value_n(x)]$

The Product rule set contains several rules in the form IF *conditions* THEN *actions*, where:

- conditions: they are tests on WM. For example, "displaySize > 10"
- a*ctions*: they are changes to WM via Insert(WML_x), Delete (WML_x) or Modify(WML_x) functions.

Then, the basic operation of a production system is defined as:

Cycle of {
1. **Recognize**
   agenda ← selects subset of PRS whose *conditions* are satisfied by current *WM*
2. **Resolve**
   action ← selects which rule of *agenda* will fire (conflict resolution)
3. **Act**
   WM ← Perform required changes to WM according to *action*

} Stop when no rules fire.

The idea proposed for the delivery reasoning process is to allow that it creates the most appropriate interface in accordance with the scenario and knowledge specified via the ontologies. The first step to understand this process is to associate groups of rules with the information codified for each ontology. Then, the reasoning can deal with group of rules, giving priority to some of them. Based on this introduction, the reasoning process works on four principal groups of rules, which are:

- The device-restriction rules analyse the device specification to decide which categories of visualisation are allowed, thus filtering the rules that can infer a suitable option(s);

- The planning-information-restriction rules mainly consider, but not only, the type of planning information being visualised to take decisions about convenient methods;
- The agent-restriction rules analyse the agent requirements regarding its needs and preferences for the task that is being executed. Based on that, suitable methods of information visualisation are proposed;
- The environment-restriction rules decide the appropriate forms of visualisation, based on awareness and characteristics of the environment and restrictions that it may impose.

Instances of these four ontologies (Device, Agent, Planning Information and Environment) define a scenario. Given a scenario definition, the reasoning mechanism infers a suitable information visualisation modality, semantically expressed by classes of the Information Visualisation ontology. For that end, the reasoning mechanism occurs in two main phases (Figure 2). In the first phase the Scenario Rules are applied. As a result, several suggestions of suitable information visualisation are proposed as output. In a second phase, optional Disambiguation Rules can be applied to choose only one modality of information visualisation among the proposed output set.

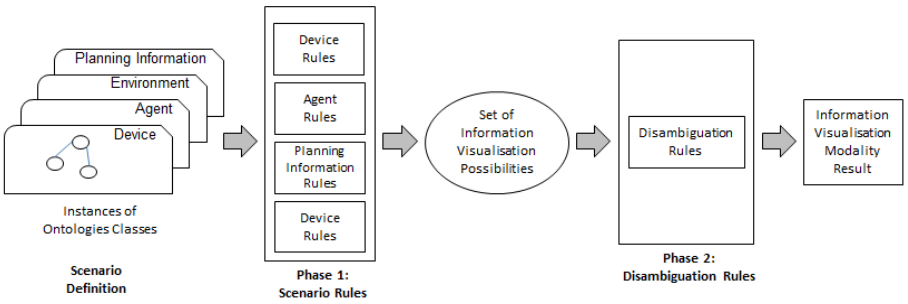

Figure 2: Reasoning mechanism overview

The set of device-restriction rules is the first to be applied. Initially, all the modalities are inserted to the base, together with the device instance that is going to be used. In this way, the device rules must indicate which modalities are supported for it. The next rule (1), for example, codifies the conditions that a device needs to have to support the 3D (virtual reality) modality. Such conditions are, for example, physical constraints (video data transference rate) and existence of support library (OpenGL or DirectX). If this rule holds, its consequence is the assertion of a new fact to the basis saying that the modality "m" is now enabled to be used. In this way, only the enabled modalities will be used for the remainder rules during the reasoning process.

$$\forall d,m \text{ device(d)} \wedge \text{modality}(m) \wedge \text{isModality}(m, \text{3D}) \wedge \text{hasMinimumVideoDataTranfer}(d,m) \wedge \text{hasOpenGlOrDirectXLibrary}(d) \Rightarrow \text{enabled}(m) \quad (1)$$

After that, the reasoning mechanism analyses the plan itself. According to our approach, every plan p is composed of n elements e_i, where $i \in [1..n]$, according to the <I-N-C-A> ontology. When a plan p is created, its elements are inserted to the knowledge base as facts, which will validate one or more rules during the reasoning process. For example, consider the next rule (2):

$$\forall p,e,m \text{ plan}(p) \wedge \text{elementOf}(e,p) \wedge ((m = \text{Textual}) \vee (m = \text{Tabular}) \vee (m = \text{NLP}) \vee (m = \text{Sound})) \Rightarrow \text{displayEnabled}(e,m) \quad (2)$$

According to this rule, for every instance of the plan class, the information related to any of the plan elements of this instance can be delivered via a textual, tabular, NLP or sound representation. In other words, we are saying that these modalities are appropriate to deliver any kind of plan information represented by <I-N-C-A>. This rule, in particular, only considers the kind of plan element (issue, activity, constraint or annotation) to generate a conclusion. However, other rules need to analyse specific features of each plan element.

The reasoning mechanism uses the agent rules to analyses the agent requirements regarding its needs and preferences for the task that is being executed. This mechanism considers two optional concepts that are related to the planning process: organizational structure of the group and description of agents. The first concept is important because it places each agent in the planning process, highlighting its role. The second shows the preferences and mental state of each agent, stressing what they can do or intend to do during the planning from its own perspective. The next rule (3), for example, says that if there are two options to visualise a same planning element, the agent preference could be used to decide for one of them.

$$\forall p,e \text{ visualisation}(e,p,m_1) \wedge \text{visualisation}(e,p,m_2) \wedge \neg(m_1 = m_2) \wedge \text{agentPreference}(e, m_1) \Rightarrow \text{retract}(\text{visualisation}(e,p,m_2)) \quad (3)$$

The reasoning mechanism uses the environment rules together with the planning rules to configure appropriates manners to deliver the planning information. For example, the next rule means that for all instances of a plan class, if this plan has two constraints that refer to the same object and such constraints has latitude and longitude as attributes, then the object of these constraints has a 2D position (4).

$$\forall p,c_1,c_2 \text{ constraintsOf}(c_1,p) \wedge \text{constraintsOf}(c_2,p) \wedge (\text{objectOf}(c_1) = (\text{objectOf}(c_2)) \wedge \text{attribute}(c_1, \text{Latitude}) \wedge \text{attribute}(c_2, \text{Longitude}) \Rightarrow \text{has2dPosition}(\text{Object}(c_1)) \quad (4)$$

After the application of such rules, we have a set of information visualisation possibilities. Then, rules of disambiguation can optionally be applied, so that only one modality is returned as outcome of the reasoning mechanism.

4 A Practical Application

The application summarised here is based on a disaster relief operation where several agents are carrying out tasks in a collaborative environment [Siebra, 06]. A disaster relief domain is a good example for our demonstration because it involves agents using several types of devices and dealing with different parts of a plan. Consider that each member of a disaster relief team has an assistant agent α running in a device *d*, dealing with a subplan *p* in an environment *e*. To run our framework we must have: a description for α, according to the agent ontology, which must be loaded to *d*; a description for *d*, according to device ontology, which must be acquired from the own device; a description for *p*, according to plan ontology, which is produced by a planning process running inside *d*; a description for *e*, according to environment ontology, which must be loaded to *d* before the start of the operation.

The first step of the visualisation mechanism is to transform all these descriptions into objects to be inserted into a knowledge base. For example, the device is one object and the attributes of such an object represent the features of the device. Then, the reasoning process acts to generate the possible forms of information visualisation to each kind of plan element. If just a simple visualisation modality is required, then the disambiguation rules can be applied. The outcome of the reasoning is one or more mappings from visualisation modalities to the plan or, more commonly, parts of the plan. Table 1 describes the test scenarios used during the system validation, while the next subsections detail the elements of such scenarios and the system execution.

Scenario	Agent	Device	Disambiguation rules
1	Operation commander	C2 Room	No
2	Fire Station	Personal Computer	No
3	Fire Brigade	Mobile Device 1	No
4	Fire Brigade	Mobile Device 1	No
5	Operation commander	C2 Room	Yes
6	Fire station	Personal Computer	Yes
7	Fire Brigade	Mobile Device 1	Yes
8	Fire Brigade	Mobile Device 2	Yes

Table 1: Definition of scenarios in terms of agents, devices and employment of disambiguation rules.

4.1 Agents Characterisation

We can classify the agents that are performing in disaster relief environments into three representative classes: (1) Central command and control agents, (2) Local command and control agents and (3) Execution agents. Note that inside each of these

classes can coexist several command and control levels. However the basic three levels idea is still the same. The important point of this classification is that agents in each group are likely to use different devices, depending on the role that they are performing in the organization and their location. While central command and control agents commonly have powerful resources available, execution agents will have limited type of devices that do not disrupt their mobility and action. Local command and control agents could have an intermediary kind of device between powerful and limited ones.

Another important point in this discussion is that the planning process, performed for each of these classes, is also different. The next three tables (Table 2, 3 and 4, based on [Siebra, 2006]), describe this difference. The central command and control level (Table 2), or strategic level, accounts for developing plans at a high level of abstraction, or "what-to-do" plans. In other words, this level specifies what must be done, but it does not give details about how something must be done. In this way, the principal tasks are related to analysis, directions and comparison of courses of actions.

Features	**Description**
Input	Generally a complex and abstract task
Output	Requests for the performance/filling of "what-to-do" plans
Time	Long-term goals
Influence	The entire collaborative team is affected by its decisions
Knowledge	Global, diversified and non-technical
Processes	Problem analysis, definitions of directions and priorities

Table 2: Central command and control agents (Strategic level).

Considering a disaster relief domain, this level could be represented by the *Search and Rescue Command Centre* (SRCC). Just after an earthquake, the SRCC receives the tasks of rescuing injured civilians and limiting the damage to the city. Analysing the problem, the SRCC decides to divide the city into regions and set priorities for each of them (some regions can be more critical than others because they have a higher probability of having buried civilians, historic value such as museums and monuments, or present risks of increasing the catastrophe such as deposits of fuel and explosives). The SRCC can also analyse global information, such as speed and direction of wind to predict the fire behaviour and generate tasks to avoid future causalities. Possible outcomes of its deliberative process are: avoid the fire spread to region x, look for buried civilians in buildings of region y, keep unblocked the road z (because it is an important path to access resources), and so on. Note that such outcomes say what must be done without references on how they must be done. Furthermore they are long term goals, which can affect the entire collaborative team.

The local command and control level (Table 3), or operational level, could be composed of local units such as fire stations and hospitals. When such components receive subgoals from the strategic level, they start by checking the necessary conditions and options to reach the subgoals, according to their available resources. In this way, operational components are taking decisions at a different level because they are thinking about how the activities could be carried out. Each local unit has the function of employing its subordinates to attain specific goals through the design,

organisation, integration and conduct of sub-operations. For that end, each unit has its own skills and abilities so that its knowledge is more specialised in the field in which it is operating. This level also pays significant attention to the resource/time relation. This means an efficient and balanced use of resources. Thus, processes such as automatic task allocation and load balancing are very useful.

Features	**Description**
Input	What-to-do plans and possible restrictions on their performance
Output	Requests for the performance of specific tasks
Time	Mid-term goals
Influence	One or more sub-teams are affected by their decisions
Knowledge	More specialised, mainly on operation environment and resources
Processes	Synthesis of plans, resource allocation, load balancing, etc.

Table 3: Local command and control agents (Operational level).

The level of execution (Table 4), or tactical level, is where the execution of operations actually takes place. For this reason the degree of knowledge of tactical components is very specialised within the domain which they are operating and their decisions are generally taken on sets of atomic activities. As the components are performing inside a dynamic and unpredictable environment, their reactive capabilities and speed of response are very important so that the use of pre-defined procedures could be an useful alternative. The output of this level is a set of atomic activities that are commonly executed by the own components.

The execution level, in a disaster relief operation, could be composed of fire brigades, paramedics and police forces for example. For the performance of their tasks, these components could need specific intelligent processes such as a pathfinder, which looks for best routes to specific destinations, or patrolling mechanisms to trace routes that efficiently cover search areas. The tactical level is also the principal source of new information to the coalition because its components are in fact moving through the environment. In this way they are more propitious to discover changes and new facts that must be shared among their partners.

Features	**Description**
Input	Specific tasks and possible restrictions on their performance
Output	Primitive operations (atomic activities)
Time	Short-term goals
Influence	Decisions should not have influences on other levels
Knowledge	Very specialised
Processes	Pathfinder, patrolling, reactive procedures, knowledge sharing, etc.

Table 4: Local command and control agents (Tactical level).

From this discussion, the diversity of information and planning processes in a disaster relief domain becomes clear. However, as discussed before, this is not an

exclusive feature of this domain, so that several collaborative planning domains present this same diversity.

4.2 Devices Characterisation

The experiments have used the following device profiles in an emulation[1] mode:

- C2 Room: command and control room with processing power of 2 parallel processors of 6.0GHz, 2GB RAM memory and four 40" (1920x1080) LCD Flat Panels. Hard memory of 300GB, containing all libraries;
- Personal Computer: a Pentium 4 Processor 3.0 GHz, with 512MB memory and a 20" (1280x1024) LCD Flat Panel as display. Contains the following visualisation libraries: GUI, DirectX and Map;
- Mobile Device 1: V980 Handheld with processing power of 200MHz, 2MB memory, a 30x20 display, CLDC configuration and Java enabled. It does not contains any special library;
- Mobile Device 2: Palm Intel XScale 416 MHz, 4GB memory, display 60x50 TFT, CLDC configuration and Java enabled. Contains special libraries to manipulate tree and network representations.

4.3 Experiments Execution

Each of the scenarios is an experiment and all of them use the same instances of visualisation modalities: textual, tabular, sound, graphic, network, tree, spatial, virtual reality (3D) and natural language. After running the experiments, the system returns the options for each kind of plan element[2] in accordance with the rules. All the next figures (Figures 3 to 6) show indications of visualisation modalities, which are returned by the system to plan elements (activities, issues, constraints and annotations).

Figure 3 shows the results to Scenario 1. As the command and control rooms are very well equipped in terms of hardware and software, they enable any kind of planning visualisation. So we can see several visualisation options as follows.

[1] Emulation mode means that we installed emulators of these device profiles to conduct experiments.
[2] Plans can be seen in http://www.aiai.ed.ac.uk/project/ix/project/lino/resources/plans.zip

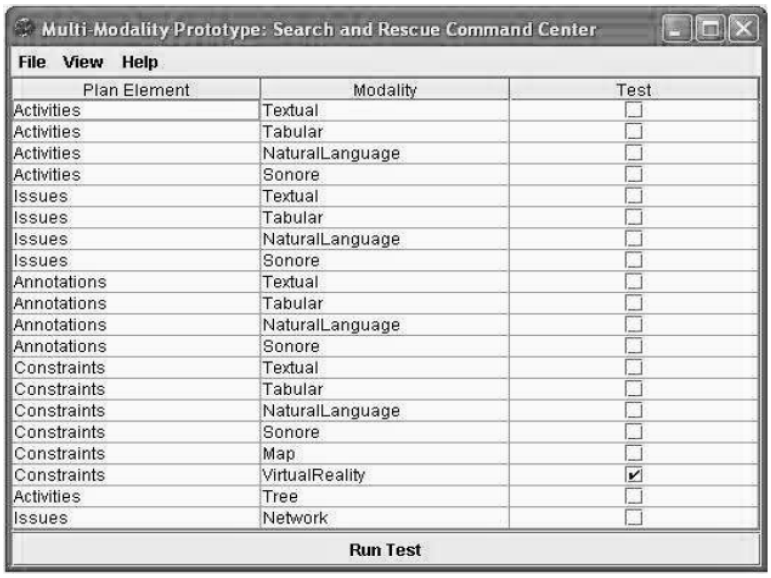

Figure 3: Results for visualisation modalities to Scenario 1

Figure 4 shows a smaller set of visualisation options (from now on we are no longer considering issues and annotations for simplification reasons). There are two reasons for that. First the device resources are more limited, mainly in terms of libraries. Second the user has set a visualisation preference constraint so that if this option is available (in this case the map modality), only this option is returned.

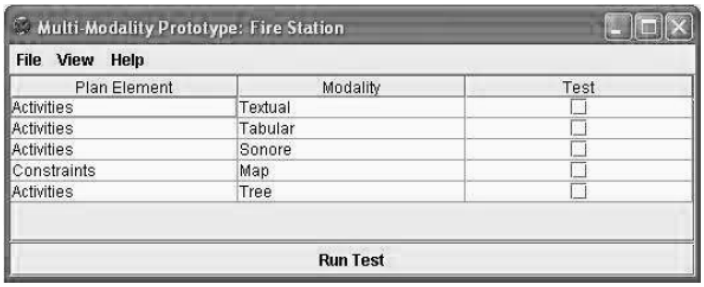

Figure 4: Visualisation results to Scenario 2.

Figure 5 shows results of experiments that use the same agent profile running in different devices. The second device is more powerful, however it returns less options because it provides the kind of visualisation modality that was set by its user (Map modality). Note that, if the system infers that the first device does not support the map modality, then the agent preferences cannot be applied and all other possible options are returned (Figure 5, top). Differently, the second device is more powerful and

supports this kind of modality. Thus, it returns only this visualisation option to the Constraints plan element (Figure 5, bottom).

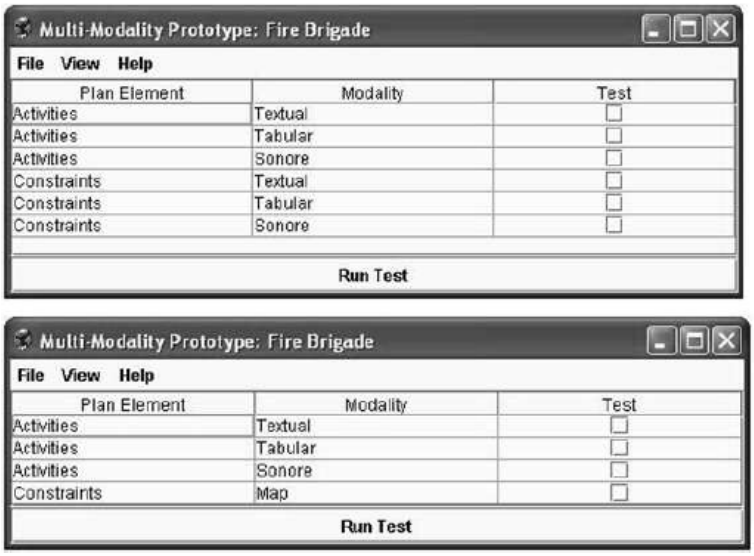

Figure 5: Results for visualisation modalities to Scenarios 3 (top) and 4 (bottom)

In the majority of planning systems, one kind of visualisation is enough for each plan element. Thus, cases like the one represented in Scenario 1 (Figure 3) should be refined. The refinement process is carried out via filtering rules, as previously explained. Note that user preferences can be considered a kind of filtering approach, however in situations that they cannot be applied, then the system must offer some filtering strategy. The strategy used here to exemplify the idea of filtering is simple. If there is one or more special structure modalities, one of that modalities is randomly chosen. Otherwise, the system tries one of the complex structure modalities. If both options fail, then one simple structure modality is used. In brief, the idea is to try more specialised modalities before the simple ones.

Note that for this kind of reasoning, the system needs to understand the hierarchical relation between the classes (Figure 1). For example, it needs to know that if the *Tree modality* is part of the *N_Dimensional* set and the *N_Dimensional* set is part of the *Complex Structure* set, then the *Tree modality* is also part of the *Complex Structure* set. Adding the set of filtering/disambiguation rules to the rule base, we have the following results (Figure 6). Note that the system returns only one visualisation modality for each category, according to the new set of rules.

This information visualisation prototype also provides a way to see the resultant interface. For that end, the user should check the respective option (right columns in Figures 3 to 6) and press the button to run the demo. Examples of interfaces are shown in Figure 7.

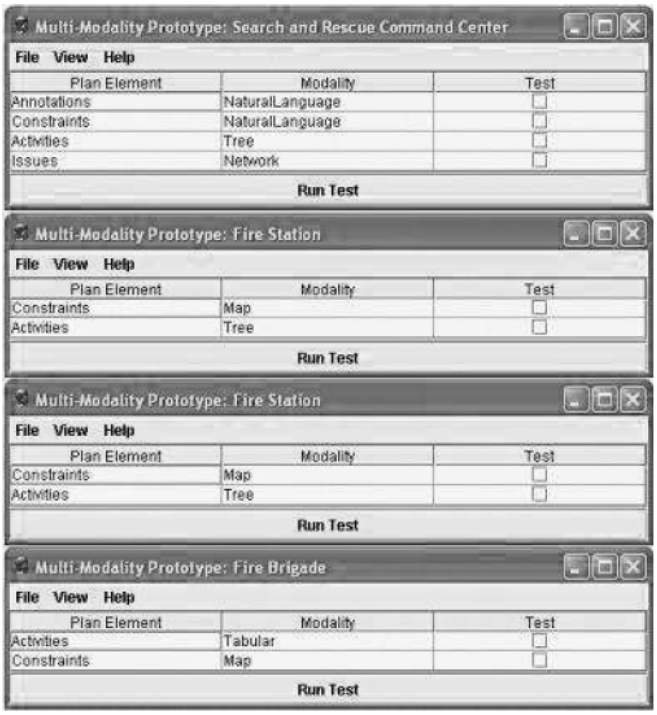

Figure 6: Results for visualisation modalities for Scenarios 5 to 8

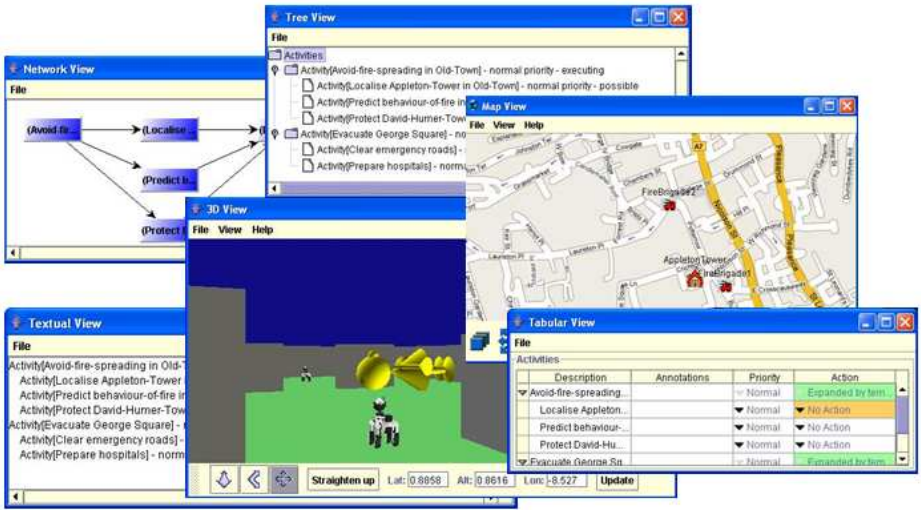

Figure 7: Examples of interfaces generated by the system prototype.

4.4 Empirical Evaluation

This section discusses an empirical evaluation of our framework, which uses results derived from experiments of Section 4.3 and related observations. For that end, we follow the methodology of first defining the scope of the framework. Then, we list the set of requirements that the framework tries to fit, showing if they are fulfilled.

The idea of our framework is to consider any kind of collaborative planning domain, which can be defined via a planning representation language. Once we are using a specific representation, the <I-N-C-A> ontology, as a basis for our planning model, we can say that the scope of our framework is delimited by the scope of <I-N-C-A> regarding its representation of planning domains.

Based on this assumption, we need to analyse the scope of <I-N-C-A> itself. The proposal of <I-N-C-A> is to be a general ontology for representation of plans. In this way, it is based on general objects (e.g., activities, constraints, etc.) rather than concepts coupled to particular domains. To represent a broad set of planning domains, <I-N-C-A> objects are specified in a very open way. The content of constraints, for example, is defined by a list of parameter elements, where parameter is an open kind of element that will be defined according to the constraint to be created.

While this kind of definition provides enough freedom to create several kinds of constraints, the semantics of new constraints cannot be directly used by the reasoning mechanism. In this way, it is important that a more refined definition of constraints is given, via the definition of types such as world state, temporal or resource constraints. Then, the reasoning process can correctly use the elements of these definitions. For example, the definition of temporal constraints allows that a set of this type of constraint can be analysed and the system can create or choose a customised form of visualisation delivery to this specific set.

A conclusion to this discussion is that the scope of our framework is restricted to all kind of domains that can be specified via the current version of the <I-N-C-A> ontology and its pre-defined constraint types. Note that extensions in its representation will not have an impact on our framework. However such extensions will not aggregate value to the visualisation reasoning process, just because the framework will not recognise them. Considering this scope, we can evaluate our framework according to five requirements: coverage, extensibility, soundness, completeness and quality.

The evaluation of coverage tries to investigate if the framework covers all possible scenarios, or if there is any type of problem/event that such a framework does not cover and why. As discussed before, the scenarios represent domains of collaborative planning, such as the Search and Rescue instance. This domain has been used because it is a complex real world area of concern, involving several agents and types of devices. In this way, its employment was useful because we could verify that the models were able to represent the significant domain features from the point of view of the visualisation needs. For example, we have used very different visualisation devices to see how they could be modelled. In fact, independently of the device type, all of them have a subset of features that can be specified by the framework models. Examples of these features are display size, sound support or processing power.

The evaluation of extensibility examines if the framework can easily be modified to consider extensions in its models. This requirement is closely related to coverage.

The current framework has a specific coverage given by the models and rules. If the framework has a good extensibility, then it is also easy to update the coverage of the framework. The design of our framework has mainly considered this requirement via the use of a rule base to maintain knowledge about visualisations. As discussed before, a rule base can easily be extended and maintained. Also, the categorisation of these rules and reasoning, proposed in this work, enables a better understanding of the process and, consequently, supports the insertion or modification of new rules. We could undergo these features during our experiments when the filtering rules were used. This new set of rules had a significant impact on the results, however its design and integration into the framework was simple and natural.

The evaluation of soundness examines if the framework behaves correctly and as expected. An advantage of this framework is that the models can be previously tested via RACER [Haarslev, 03], which provides a way to test for inconsistencies and structural errors in the models. Related to the inference process and rules, we have used eight instances to test the scenarios to verify the correctness of the rules. Using simple observation of the outcomes, we could verify if such outcomes are actually appropriate and follow the ideas codified via the rules specifications. Note however, that this is not an exhaustive kind of test, so that the use of multiple variations may bring some unexpected result.

The evaluation of completeness examines if the framework covers all of the necessary concepts and functionalities. At its current stage, our framework is not meeting this requirement. There exist concepts associated with the environment and agents that are not being explored in their entirety. These concepts may have an influence on the visualisation process, apart from the fact that they are not fundamental for such a process. In fact, several concepts could be added to the models, as well as rules to augment the quality of reasoning.

The evaluation of quality examines how well the framework covers/supports the problem domain. In other words, it examines the quality of results. We have noticed that quality is closely associated with the definition of rules. Note that the soundness of the framework does not imply that the results are the most appropriate for a given scenario. During the development of the experiments, we have considered the search for quality when we try to match the best form of visualisation to each plan element. For example, the match of temporal constraints elements to the temporal modality of visualisation. Thus, the rules are mainly in charge for the results quality. However, an interesting situation noticed in our experiments is when users have visualisation preferences. Because such preferences have priority in the inference process, the system cannot ensure quality, which becomes responsibility of users. Maybe it sounds contradictory that the framework generates a final result that is different of the user preference, because we are claiming that the framework always looks for the best visualisation modality. This could indicate that users do not know the real capability of their devices, or they do not feel comfortable with such a specific modality. This last case shows that quality is a subjective parameter so that the same result could be attested as high quality for some user and not so good for another.

Additionally, further work has been conducted on the use of the concepts described in this paper via the conduct of some trial exercises involving the US Army Research Labs and civilian professional emergency responders operating in a virtual collaboration environment [Wickler, 14; Tate, 14b].

5 Conclusions and Future Work

This work proposed an integration of ontologies and reasoning mechanism for multi-modality visualisation in collaborative planning environments. The set of ontologies and their integration will permit the expressiveness of several aspects related to real world applications in environments of mixed initiative planning from a visualisation perspective. Meanwhile, the reasoning mechanism will allow a tailored delivery and visualisation of planning information. The main contributions of this framework are: a proposal for a general framework for planning information visualisation; an ontology set that supports the organisation and modelling of domains from the visualization perspective; and a reasoning mechanism that permits a proper presentation of information according to each scenario. As the framework is based on real standards (W3C), it facilitates communication and interoperability with other services and systems, also supporting extensions for its use on Semantic Web applications.

The originality of this work is because general approaches to planning information visualisation cannot be found in the literature since intelligent planning research predominantly focuses in the core planning aspects, such as algorithms, heuristics, knowledge engineering for planning or applications. Consequently there are planners with an absence of solutions for information visualisation, or the solutions are directly associated with the planning approach and the design made for a specific aspect of the planning process, for instance planning debugging. This article presents a general framework for planning information delivery, which is divided into two main parts: a knowledge representation aspect based on an ontological set, and a reasoning mechanism for multimodality visualization.

This work was also motivated by the lack of flexibility and extensibility in solutions; the use and integration of different modalities for information visualisation; and addresses issues regarding collaboration and different types of user involved in the planning process. The knowledge-based solution proposed guarantees flexibility and extensibility. It permits knowledge engineering additions to be made with minimal adjustments, since the focus of knowledge-based solutions is in the data and not in the procedures. Consequently, procedures, or reasoning mechanisms remain sound with extensibility.

Apart from the Semantic Web applications opportunity for the usage of the framework, we can list other directions of research. For example, extensions of the models so that they can mainly consider more features for the agents, environment and devices models. This implies an extension of the rule base, so that it also reasons on new classes and instances. We can also improve the evaluation tests, which must consider the requirements of coverage, extensibility, soundness, completeness and quality. Such tests should, for example, consider more than one planning domain to verify the behaviour of the framework and to prove its generality.

Other future work suggested is regarding the multimodality visualisation ontology. This work was carried out from an intelligent planning information visualisation perspective. Hence, it was conducted as an evidence-based study, considering the most common used visualisation modalities in AI planning systems. To extend this ontology a systematic review should be made about different graphic representations and their proposals, requirements and usage. That will permit a good coverage of possibilities and appropriate mapping via the reasoning mechanism and

other knowledge involved, to the appropriate information to be delivered. These works can give insights in this direction [Judelman, 04] [Burkhard, 05] [Tergan, 05].

Acknowledgements

The results presented in this paper have been developed as part of a collaborative project between Artificial Intelligence Applications Institute (AIAI - University of Edinburgh) and Federal University of Paraíba (CI/UFPB). This work was partially financed supported by the National Institute of Science and Technology for Software Engineering (INES –www.ines.org.br), funded by CNPq, grant 573964/2008-4. The University of Edinburgh and research sponsors are authorized to reproduce and distribute reprints and on-line copies for their purposes notwithstanding any copyright annotation hereon.

References

[Ai-Chang, 04] Ai-Chang, M. et al.: MAPGEN: Mixed-Initiative Planning and Scheduling for the Mars Exploration Rover Mission. Intelligent Systems, 2004, 9(1):8-12.

[Allen, 01a] Allen, J., Byron, D., Dzikovska, M., Ferguson, G., Galescu, L., Stent, A.: Towards conversational human-computer interaction. In AI Magazine, 2001, 22(4):27-37.

[Allen, 01b] Allen, J., Ferguson, G., Stent, A.: An architecture for more realistic conversational systems. In Proceedings of Intelligent User Interfaces 2001 (IUI-01), Santa Fe, NM, USA.

[Bandara, 04] Bandara, A., Payne, T. and et. al, An Ontological Framework for Semantic Description of Devices, The 3rd International Semantic Web Conference, Japan, November 2004.

[Blum, 97] Blum, A., Furst, M.: Fast Planning Through Planning Graph Analysis. Artificial Intelligence, 1997, 90:281-300.

[Burkhard, 05] Burkhard, R. A. Knowledge Visualization: The Use of Complementary Visual Representations for the Transfer of Knowledge: a Model, a Framework, and Four New Approaches. Ph.D. Thesis, Swiss Federal Institute for Environmental Science and Technology, 2005.

[Cao, 10] Cao, B., Yin, J., Zhang, Q., Ye, Y.: A MapReduce-Based Architecture for Rule Matching in Production System. IEEE Second International Conference on Cloud Computing Technology and Science, 2010, pp. 790-795.

[Card, 99] Card, S., Mackinlay, J., Shneiderman, B.: Readings in Information Visualization, Using Vision to Think. Los Altos, CA, Morgan Kaufmann, 1999.

[Chen, 04] Chen, H., Finin, T. and Joshi, A. Semantic Web in the Context Broker Architecture, IEEE Conference on Pervasive Computing and Communications (PerCom), Orlando, March, 2004.

[Daley, 05] Daley, P., Frank, J., Iatauro, M., McGann, C. and Taylor W. 2005. PlanWorks: A debugging environment for constraint based planning systems. In Proc. of the 1st Int. Competition on Knowledge Engineering for Planning and Scheduling.

[Fahad, 11] Fahad, M., Moalla, N., Bouras, A., Qadir, M., Farukh, M. Towards Classification of Web Ontologies for the Emerging Semantic Web, Journal of Universal Computer Science, 2011, 17(7): 1021-1042.

[Gerevini, 08] Gerevini, A. and Saetti, A. An Interactive Environment for Plan Visualization and Generation: InLPG, Working notes of Eighteenth International Conference on Automated Planning & Scheduling (ICAPS-08) - System Demonstration, Sydney (Australia), 2008.

[Gerevini, 11] Gerevini, A. E. and Saetti, A. An Interactive Tool for Plan Visualization, Inspection and Generation. Proceedings of the ICAPS Workshop on Knowledge Engineering for Planning and Scheduling (KEPS 2011), Freiburg, Germany, 2011.

[Ghallab, 04] Ghallab, M., Nau, D., Traverso, P.: Automated Planning: Theory and Practice. Morgan Kaufmann, 2004.

[Guizzardi, 04] Guizzardi, G., and G. Wagner. A Unified Foundational Ontology and some Applications of it in Business Modeling. Proceedings of the CAiSE'04 Workshops. Faculty of Computer Science and Information Technology, Janis Grundspenkis and Marite Kirikova (Eds.), Riga Technical University, Riga, Latvia. Volume 3. pp. 129–143. Riga, Latvia. June 7–11, 2004.

[Haarslev, 03] Haarslev, V., Möller, R.: Racer: A Core Inference Engine for the Semantic Web. Workshop on Evaluation of Ontology-based Tools, 2003.

[Judelman, 04] Judelman, G. B. Knowledge Visualization - Problems and Principles for Mapping the Knowledge Space (2004). MSc Thesis Dissertation.

[Kautz, 98a] Kautz, H., Selman, B.: The Role of Domain-Specific Knowledge in the Planning as Satisfiability Framework. In Proceedings of the 4th International Conference on Artificial Intelligence Planning Systems, Pittsburgh, PA, USA, 1998.

[Kautz, 98b] Kautz, H., Selman, B.: BLACKBOX: A New Approach to the Application of Theorem Proving to Problem Solving. Working Notes of the Workshop on Planning as Combinatorial Search, Pittsburgh, PA, USA, 1998.

[Kim, 04] Kim, H., Gratch, J.: A Planner-Independent Collaborative Planning Assistant. Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems , 2004, volume 2, pp. 766 - 773.

[Lino, 03] Lino, N., Tate, A., Chen-Burger, Y.: Improving Semantics in Mobile Devices Profiling: A Model Conceptual Formalisation and Ontology Specification, In Workshop on Semantic Web Technology for Mobile and Ubiquitous Applications at the 3rd International Semantic Web Conference, Hiroshima, Japan, 2003.

[Lino, 07] Lino, N.: Semantic Based Support for Visualisation in Complex Collaborative Planning Environments. PhD Thesis, The University of Edinburgh, UK, 2007.

[Mahmoudi, 07] Mahmoudi, M. T., Badie, K. and Neshatian, K. An Ontological Model for Knowledge Management through Concept Composition, Proceedings of IEEE First International Conference on the Digital Society, EKNOW 2007, Guadeloupe, French Caribbean, January 2-6, 2007.

[Mizoguchi, 00] Mizoguchi, R. Kitamura, Y., "Foundation of knowledge systematize-tion: Role of ontological engineering", Industrial Knowledge Management - A Micro Level Approach, Springer-Verlag, 2000.

[Penalver, 13] Penalver, A., López, J. Botella, F., Gallud, J.: Defining Distribution Constraints in Distributed User Interfaces, Journal of Universal Computer Science, 2013, 19(6):831--850.

[Siebra, 06] Siebra, C.: A Unified Approach to Planning Support in Hierarchical Coalitions. PhD Thesis, The University of Edinburgh, UK, 2006.

[Tate, 03] Tate, A., Levine, J., Dalton, J. Nixon, A.: Task Achieving Agents on the World Wide Web, In Spinning the Semantic Web, Fensel, D., Hendler, J., Liebermann, H. and Wahlster, W. (eds.), Chapter 15, pp. 431-458, MIT Press, 2003.

[Tate, 14a] Tate, A. Using Planning to Adapt to Dynamic Environments, in Suri, N. (ed.) (2014) Adaptive, Dynamic, and Resilient Systems, Chapter 13, pp. 237-251, Taylor & Francis, 2014.

[Tate, 14b] Tate, A., Hansberger, J.T., Potter, S. and Wickler, G. (2014) Virtual Collaboration Spaces: Bringing Presence to Distributed Collaboration, Journal of Virtual Worlds Research, Assembled Issue 2014, Volume 7, Number 2, May 2014.

[Tate and Drabble, 1995] Tate, A. and Drabble .B. O-Plan's PlanWorld Viewers, Proceedings of the Fourteenth UK Special Interest Group on Planning and Scheduling, Wivenhoe House Conference Centre, Essex University, 22-23 November 1995.

[Tergan, 05] Tergan. S. O and Tanja, K. Knowledge and Information Visualization: Searcjong for Synergies. Springer Verlag, Berlin/Heidelberg, 2005.

[Thamboulidis, 07a] Thamboulidis, Kleanthis C., Doukas, George S. and Koumoutsos, Giannis V. Device Modeling for a Flexible Embedded Systems Development Process. Proceedings of the 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC'07), 2007

[Thamboulidis, 07b] Thamboulidis, Kleanthis C., Koumoutsos, Giannis V. and Doukas, George S. Semantic Web Services in the Development of Distributed Control and Automation Systems. Procedings of 2007 IEEE International Conference on Robotics and Automation Roma, Italy, 10-14 April, 2007.

[Thangarajah, 08] Thangarajah, J., Harland, J.,Morley, D. Yorke-Smith, N.: Suspending and resuming tasks in BDI agents. Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems, 2008, voulme 1, pp. 405-412.

[Veloso, 95] Veloso, M., Carbonell, J., Perez, A., Borrajo, D., Fink, E. and Blythe, J.: Integrating Planning and Learning: The PRODIGY Architecture. Journal of Theoretical and Experimental Artificial Intelligence, 1995, 7(1):81-120.

[W3C, 04] World Wide Web Consortium (W3C). Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. W3C Recommendation, 15 January, 2004. http://www.w3.org/TR/CCPP-struct-vocab/

[Wickler, 06] Wickler, G., Potter, S., Tate, A.: Using I-X Process Panels as Intelligent To-Do Lists for Agent Coordination in Emergency Response, International Journal of Intelligent Control and Systems (IJICS), Special Issue on Emergency Management Systems, 2006, 11(4):248- 259.

[Wickler, 14] Wickler, G., Tate, A. and Hansberger, J. (2013), Using Shared Procedural Knowledge for Virtual Collaboration Support in Emergency Response, in Special Issue on Emergency Response, IEEE Intelligent Systems, Vol. 28, No. 4, July/August 2013, IEEE Computer Society.Journal of Virtual Worlds Research, Assembled Issue 2014, Volume 7, Number 2, May 2014.

[Wilkins, 01] Wilkins, D., desJardins, M.: A call for knowledge-based planning. AI Magazine, 2001, 22(1):99-115.