A CONVERSATIONAL AGENT TO HELP NAVIGATION AND COLLABORATION IN VIRTUAL WORLDS

OLIVIER BERSOT[1], PIERRE-OLIVIER EL GUEDJ[2], CHRISTOPHE GODÉREAUX[3], AND PIERRE NUGUES[1]

1. Institut des Sciences de la Matière et du Rayonnement, GREYC, 6, boulevard du Maréchal Juin, F-14050 Caen, France.
2. GESTINFOR, 10, rue Alfred Kastler, F-14000 Caen, France.
3. Université de Caen, GREYC, 6, boulevard du Maréchal Juin, F-14050 Caen, France.

(Authors are in alphabetical order)

Reprint from *Virtual Reality*, 3(1):71–82, 1998

Abstract. This paper describes the prototype of a conversational agent embedded within a collaborative virtual environment. This prototype – *Ulysse* – accepts spoken utterances from a user enabling him or her to navigate within relatively complex virtual worlds. It also accepts and executes commands to manipulate objects in the virtual world. We are beginning to adapt our agent to parse certain written descriptions of simultaneous actions of world entities and to animate these entities according to the given description.

The paper first describes what we can expect from a spoken interface to improve the interaction quality between a user and virtual worlds. Then it describes Ulysse's architecture, which includes a speech recognition device together with a speech synthesiser. Ulysse consists of a chart parser for spoken words, a semantic analyser, a reference resolution system, a geometric reasoner, a dialogue manager, and an animation manager, and has been integrated in the DIVE virtual environment. Ulysse can be 'personified' using a set of behavioural rules. A number of tests have demonstrated its usefulness for user navigation. We are currently developing simulations of written reports of car accidents within Ulysse; such simulations provide dynamic recreations of accident scenarios for individual and collaborative reviewing and assessment.

Keywords: Conversational agents; Spoken navigation; Simulation; Semantics of space; Planning.

1. Collaborative Virtual Environments and Linguistic Interaction

Collaborative virtual environments enable the re-creation of offices, meeting rooms or more complex scenes. In such environments, distant participants are embodied within a virtual workspace using more or less realistic representations. Participants can move about or go from one virtual room to another, share tools, and discuss, in real time, projects or ideas. Examples of three-dimensional virtual environments include research prototypes such as DIVE [1], MASSIVE [2], and SmallView [3]. Their acceptance is being boosted by the development of the VRML standard [4] and its incorporation in many Web browsers.

Virtual environments can also include videoconferencing facilities. Participants' images taken from a video camera then replace the face of user embodiments in the virtual world [5]. Using metaphors, such environments improve the comprehension of videoconferencing sessions and forums, or provide an interactive visualisation of information [6].

Visualisation gives virtual environments an indisputable capacity to represent and to communicate knowledge [7], [8]. Provided that a scene is well rendered, users can relate the computer simulated situation to their working environment, understand it, and realise its complexity. Virtual environments may also help improve cognitive development [9]. The counterpoint is that it is much more difficult to interact with the interface. While making an extensive use of 3D icons, commercial products or prototypes, such as AT&T's MultiMedia Communication eXchange (MMCX) or France Telecom's Varèse, remain in a two-dimensional space.

Speech and language appear to be 'natural' means for interaction and description and therefore possible interfaces to a virtual world. Language lets us refer to objects that are not immediately visible, encapsulate complex groups of objects, and support reasoning [10]. However, in spite of these advantages, speech and language interfaces are not common in virtual or simulation environments. Notable exceptions include Put-that-there [11], Diverse [12], Persona [13], and Nautilus [14]. Possible reasons of this rarity are that reliable speech recognition devices have only recently become available and the exact role speech could play in a virtual environment has not been well defined.

In the article, we investigate the "usability" of spoken interaction in virtual environments for teleconferencing. An experiment is described that enabled us to clarify appropriate uses of speech interfaces in a collaborative virtual environment. Drawing on conclusions drawn from this experiment, we describe the design and implementation of Ulysse, a conversational agent.

2. The Experiment

We built a virtual world – Ithaque, (Figure 1) – using the Distributed Interactive Virtual Environment (DIVE) [1] to evaluate and compare the quality of interaction using speech and mouse. DIVE enables users to share virtual worlds where they can connect from a remote location. They can move into these worlds and meet other participants. Participants share the same geometric model of the world with a different viewpoint. Modifications of the world from user interactions are replicated to the other participant's sites to keep the world consistent.

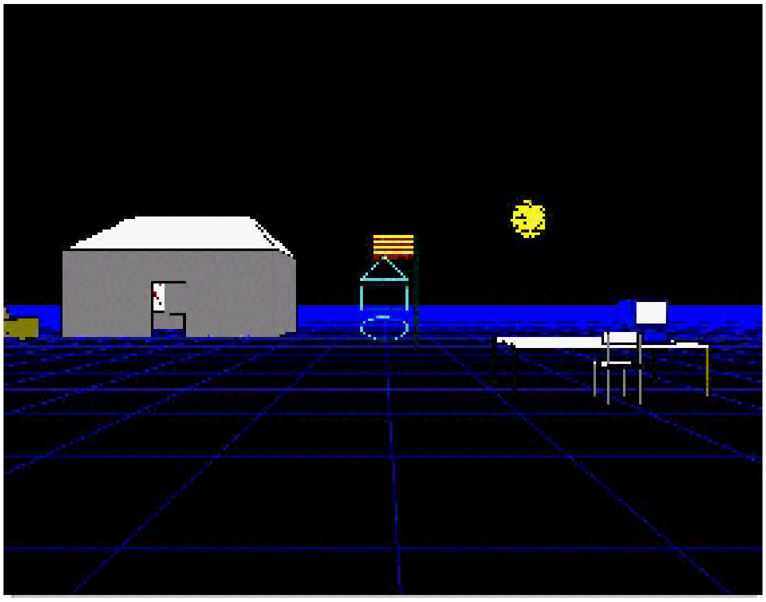


Figure 1. The Ithaque world.

We collected a corpus of spoken dialogues involving two experienced (computer scientists with no background on virtual environments) and two novice users (word processing level). We recorded these dialogues in four sessions of interaction [15]. Each dialogue involved two participants: the interacting user, and another who played the role of the agent by acting on the virtual world. We plotted two scenarios. In the first one, novice users had to move around and discover the world and in the second one, more experienced users had to discover a treasure hidden in the world.

In comparing mouse and speech interactions, we found that mouse navigation was a major difficulty. Navigating with devices such as mice, space balls, is one of the trickiest issues for new users. Certain motions are very difficult to carry out and a novice user can easily get seasick with her/his 'body' upside-down within a two-minute session.

More experienced users can move relatively accurately in horizontal or vertical planes after an adaptation time. However, they find it difficult to align with an object or to go around it using a mouse. In addition, it is impossible to look at a specific location while moving using a single mouse. This makes some motions clumsy, for instance when the user is going around an object, and it makes it impossible for a user to check easily how many users there are in a room.

Conversely, many of these motions are easy to describe using speech. This is particularly true when the user wants to align with an object, to go around it or to turn around. We found that in many situations the user prefer 'to say it' rather than to 'do it'. However, it does not seem desirable to try to substitute completely pointing devices because it is sometimes easier to point at an object rather than to describe it in a verbal way.

Our conclusion is that mouse navigation presents difficulties for novices without considerable training. Spoken dialogue interfaces improved the usability of virtual environments for both novice and experienced users. Many actions proved easy to formulate verbally. Thus, virtual environments with classical input devices may cause interaction difficulties, hindering effective collaborative activities. We propose retaining the benefits of pointing devices, but supplementing this interaction style with an interface that recognises spoken commands. Choice of interaction style can be left to individual users.

3. Ulysse System Architecture

We designed and implemented a conversational agent – Ulysse – understanding spoken natural language to navigate in a virtual world. Ulysse is incorporated within a user's embodiment and offers assistance within the world by responding to motion commands. Ulysse acts on the command and transports the user within a virtual environment on his/her behalf [15] [16]. Ulysse works in co-ordination with other interaction modes: in our prototype a mouse.

Ulysse consists of commercial speech recognition and speech synthesis devices. Like other spoken dialogue systems [17], Ulysse features a chart parser and a semantic module analysing the word stream and building a logical form from it (Figure 2). The chart was adapted to accept spoken utterances [18]. It is using phrase-structure rules and a 400 word lexicon that we derived from our corpus.

Ulysse's navigation capabilities require reasoning on the objects and on the geometry of the scene. In consequence, the architecture is complemented by a reference resolver that maps the entities mentioned in utterances to geometric objects from the virtual world and to actions. It also features a geometric reasoner to understand the world. The navigation – the motion of the embodiment – is carried out by an animation planner, which utilises behavioural rules to carry out the user's movement commands.

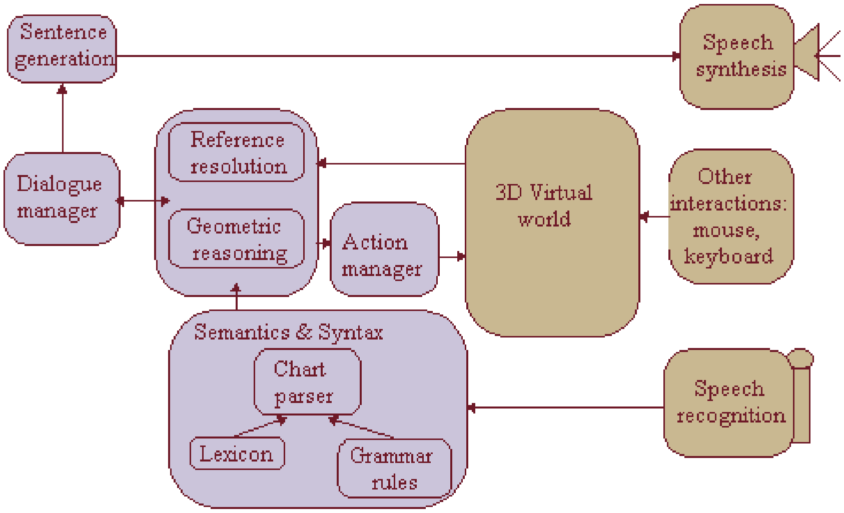


Figure 2. Ulysse system architecture.

4. Syntactic and Semantic Parsing

The user activates Ulysse using a 'push-to-talk' scheme. It presses a button to signal the beginning and the end of the utterance. Speech recognition is carried out using the IBM's VoiceType commercial device. VoiceType is operating on isolated words, that is, the speaker must pause between words. A chart parser is connected to the recognition device output and analyses the words. This chart [18] adopts a classical bottom-up algorithm with a dual syntactic formalism: It can operate using phrase-structure rules and a dependency formalism [19].

We used a constituent grammar to encode the lexicon and phrase-structure rules. Together they accept all the 400 utterances of the corpus we collected in the experiment [15], [16]. The lexicon is using parts-of-speech that are a variation of MULTEXT-TEI categories [20]. Phrase-structure rules are rewriting the utterance structure using unification constraints and non-terminal categories such as noun groups, verb groups, prepositional groups, determiner groups, adverb groups, and adjective groups. Rules were adapted to accept missing and unknown words. They also include a large number of prepositional, adverbial, and demonstrative locutions.

The semantic analyser splits the chart returned by the syntactic parser into a list of clauses. Each clause is mapped to a structure whose members are the subject, verb group, and a list of complements. Complements are annotated with a semantic tag such as time, manner, or location. Verb groups are also annotated with a semantic tag and packed with possible adverbs and clitic pronouns. We classified verbs according to six categories:

1. go (*aller, entrer, avancer, monter*) corresponds to a change of location with a possible rotation of the embodiment;
2. return (*retourner, revenir*) corresponds to change of location, but visibility of object does not matter;
3. rotate (*tourne, regarde*) corresponds to a simple rotation;
4. look (*regarder, voir*) corresponds to a rotation of the head;
5. stop (*arrêter, stopper*) stops the current action;
6. continue (*continuer, poursuivre*) resumes the action.

We use a case grammar similar to that of [21] to represent each clause into one or more predicate. Each predicate consists of a triplet:

`<verb type, preposition type, object or direction>`

A simple verb is directly mapped to one verb type and corresponds to one triplet. More complex verbs such as *contourner* (*skirt around*) are expanded into a sequence of these basic verbs.

Clauses are rearranged according to the "connectors" that link them. Connectors are adverbs, conjunctions, or syntactic forms. These connectors are associated with list operators such as append, delete, replace, or insert. For instance adverb *puis* (then) in the sentence:

> *Monte sur la maison* ***puis*** *va devant l'ordinateur*
> (Go above the house and then go in front of the computer)

results into the appending of the second action after the first one:

1. go above the house
2. go in front of the computer

Gerund *en passant* (in passing) in the sentence:

> *Va vers la maison* ***en passant*** *devant le drapeau*
> (Go toward the house in passing in front of the flag)

results into the insertion of the last motion before the first one:

1. go in front of the flag
2. go toward the house

5. Navigation Capabilities

We gave Ulysse reasoning capabilities to control the navigation of a user's embodiment. In other projects [12], [14], the user can talk to the 'world' and expect an agent behind the scene to have the ability to answer the orders. In these projects, commands are related to navigation, but also to object manipulations and to queries.

We preferred to single out one capability – navigation – to let the user have a clear idea of what the agent can do. With navigation, domain reasoning consists essentially in resolving references and understanding the geometry of the world. The referencer and the geometric reasoning modules map the linguistic description – resulting in a predicate list – onto the virtual space. They de-index the sequence of logical forms and transform them into a sequence of 3D co-ordinates.

Referencing consists in associating a name with an object of the geometric database. It is not as straightforward as it may appear. For instance *a computer* can be encoded in the database as a single object or as a set of polygonal lines. Plurals such as *the cubes* or parts such as *the front of the computer* must also be handled. In addition, *the front of a computer* is not the same as *the front of a cube.*

In the present prototype, we addressed this denomination problem by carefully associating a name with the entities of the world database. We structured the database to keep the most consistent relations between names and world entities according to our corpus. We also gave a main orientation to each object whenever possible and reference axes originating at its gravity centre.

The referencing module considers the sequence of action predicates – the logical forms – coming from the semantic part. When one or more nouns occur in a predicate, it searches the compatible objects in the geometric database and constructs a list from it. It is not always sufficient to de-index the references. Given the structure of the world, multiple choices are often possible such as in *go around the house* (with several houses). In addition, deictic references such as *go there* are frequent. They are often associated with a mouse pointing. Ulysse solves these references using a salience value [22] consisting of two criteria:

- the object visibility,
- a focus coefficient keeping the record of user interactions with objects [12]. Each object is assigned an integer representing when it has been accessed the last time. When the user mentions an object or points at it, its focus is set to the time. After each user turn, the time is incremented.

Using these criteria, the referencer can de-index adverbial description without object or a simple noun group (an utterance with only one noun group):

- If no object is mentioned (*go there*), the object with the highest focus is retained. The user usually clicks on an object at the same time it utters such a command, and it corresponds to the highest focus.
- If one object is mentioned, according to the verb, one or both of these criteria is taken into account. For instance with verbs of type *go*, only the visible object with the highest focus is retained in case of ambiguity.

The referencer relies on the geometric reasoner to identify objects in compounded noun groups (recursive noun phrases). In a recursive noun phrase, core noun groups are linked together with prepositions. Therefore, we distinguish between position prepositions linking a noun group to a verb, for example, *devant* in the sentence:

> *Va devant la maison*
> (Go in front of the house)

and relation prepositions linking a noun group to another one, for example, *devant* in the sentence:

> *Va dans la voiture devant la maison*
> (Go into the car in front of the house).

Compounded noun groups are analysed recursively from right to left. The rightmost noun group is selected first and filters compatible objects from the geometric database. The relation preposition sets constraints on the potential objects of the second rightmost noun group and so on. At the end of the analysis, if there is more than one object remaining, the referencer uses the salience criteria to select one.

Once the referencing has been done, the geometric reasoning module is used again to produce a set of co-ordinates that delineates the motion path. The action depends on the verb type, on the preposition, and on the object type. *Go in front of the cube* (the user should go to an intermediate position between the cube and her/him) is not the same as *Go in front of the house* (she/he should go in front of the entrance door).

Reasoning is carried out considering the triplets:

`<verb type, preposition type, object type>`

Verbs have been itemised into six categories and are described in the previous section. We classified the prepositions in ten categories. Database objects are also classified according to their overall shape and whether they have a front or not. Each of the triplets is mapped to a specific type of reasoning. When the verb corresponds to a change of location, the preposition, together with the centre and the overall shape of the object enable to compute an acceptable distance to position the user relatively to the object. We consider a group of objects (plurals) as a unique entity using the gravity centres. This process results in an array of 3D positions representing the motion segments with a start location and an arrival location for each of the logical forms [15].

The sequence of 3D positions contains also the body direction and the sight direction of the agent. We use the sight direction parameters to implement the *look at* verb category. We use it also to direct the vision field toward the main object of the utterance. Thus, the user keeps the eye on it while moving. When going around a house, the user won't lose it from sight. We use the body direction to animate the embodiment with more 'natural' motions. Each element of the array finally contains:

- the sight direction of the agent;
- the body direction of the agent;
- the location where the agent is moving.

6. Planning

Once the action has been understood and possibly clarified by the dialogue manager, the planning and action manager takes the user to the location where she/he wished to go. Agent embodiments consist of hierarchies of components that are articulated. Notably, body and head are rotated separately to enable the sight to have a different direction to that of the main motion. Embodiments feature a different component structure according to the articulations wanted and their overall shape: human-like or car-like. The navigation of the whole set is carried out using a planner and actions are applied separately to each component of the embodiment.

The planner structure is generic in Ulysse and according to the type of embodiment, it includes behavioural rules that are specific. So far, we wrote rules for simplistic human-like and car-like embodiments. The human-like body notably has two legs with ankle and hip articulations (Figure 3). The planner executes the whole motion by articulating consistently all the components according to the rules. For instance, *marcher* (walk) consists in extending one leg, then the other one and doing it again.

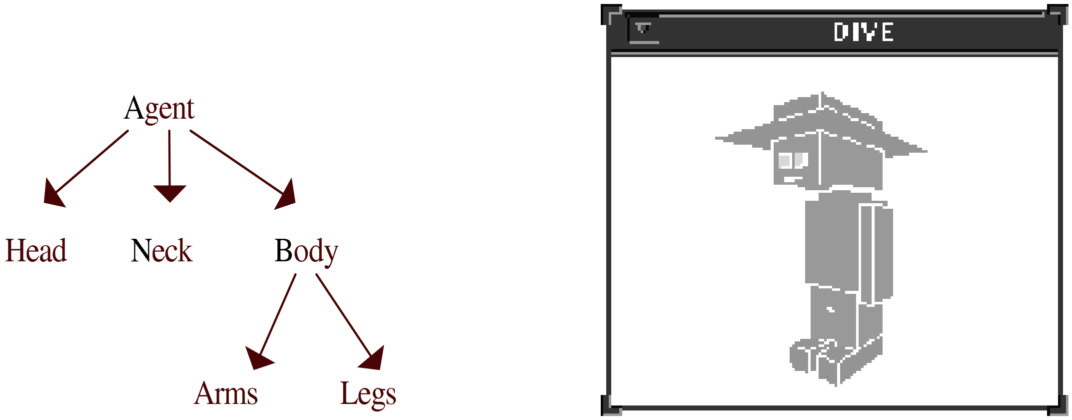


Figure 3. Agent hierarchy.

6.1. **Generating a plan.** We have implemented a linear and hierarchical motion planner that decomposes a problem into a sequence of independent more basic sub-problems [23], [24]. From the set of problems, the planner builds a chain of operators describing the intermediary positions of the virtual agent. The chain is finally submitted to a cinematic simulator that fractionates it into 3D geometric actions to yield a relatively fluid motion.

The plan generator receives a set of orders from the geometric reasoning module. These orders correspond to changes of location and changes of posture that concern the body and the head. As an initial state, the generator has a set of orders sorted in a stack. Each of these orders corresponds to a function with parameters. For example, going to a location while simultaneously turning the head in a definite direction corresponds to the following sequence:

```
((go_to {4 0 2.2})
(turn_head {0 0.75 0}))
```

The planner processes functions sequentially, starting from the first one, here go_to. It expands them into more basic actions. Expansion is carried out using the behavioural rules. Rules are triggered when preconditions can be matched. They are applied until the planner reaches basic 3D motions and can execute them.

Figure 4 shows a simplified rule describing how an agent can walk to a specific location ?x represented by 3D co-ordinates. The precondition verifies that ?x is in the horizontal plan. If so, go_to is translated into two new sub-plans (:new_pb) that specify that the embodiment must turn its whole body in the direction of the end location (turn_to ?x) and then go to it (go_straight_ahead_to ?x).

```
#RULE
:name     go_to;
:var      (?x);
:precond  ((horizontal ?x));
:new_pb   ((turn_to ?x) THEN (go_straight_ahead_to ?x));
```

Figure 4. A behavioural rule (simplified).

6.2. **Planning basic motions.** Once the planner has produced sub-plans, it converts them into an animation description. Sub-plans from Figure 4

```
((turn_to ?x)
(go_straight_ahead_to ?x));
```

correspond to: turn head and body in the direction to go. Then step forward and do it over and over. Each sub-plan is translated into basic symbolic motions that apply to the components of the embodiment (the real plan is one page long):

```
((move #BODY {x, y, z})
(move #HEAD {0, 0, 0}))

(REPEAT
  (move {#LEFTLEG {x', y', z'}) (move {#RIGHTLEG {x'', y'', z''})
  THEN (move #LEFTLEG {x'', y'', z''}) (move #RIGHTLEG {x', y', z'}))
```

The translation of the first sub-plan tells the agent to move simultaneously its body and head toward the target. When finished, the second sub-plan tells it to repeat the following sequences: move simultaneously the left leg and the right leg, then the reverse. Each sub-plan is assigned a goal and the planner loops on the code until the goal is satisfied: here when the walker has reached the final destination.

When the planner has finished building the symbolic plan, a chain of operators is compiled and simultaneous actions are assembled describing intermediary positions. Then, a cinematic simulator is run to carry out the motion between the intermediary positions the agent has to reach (Figure 5).

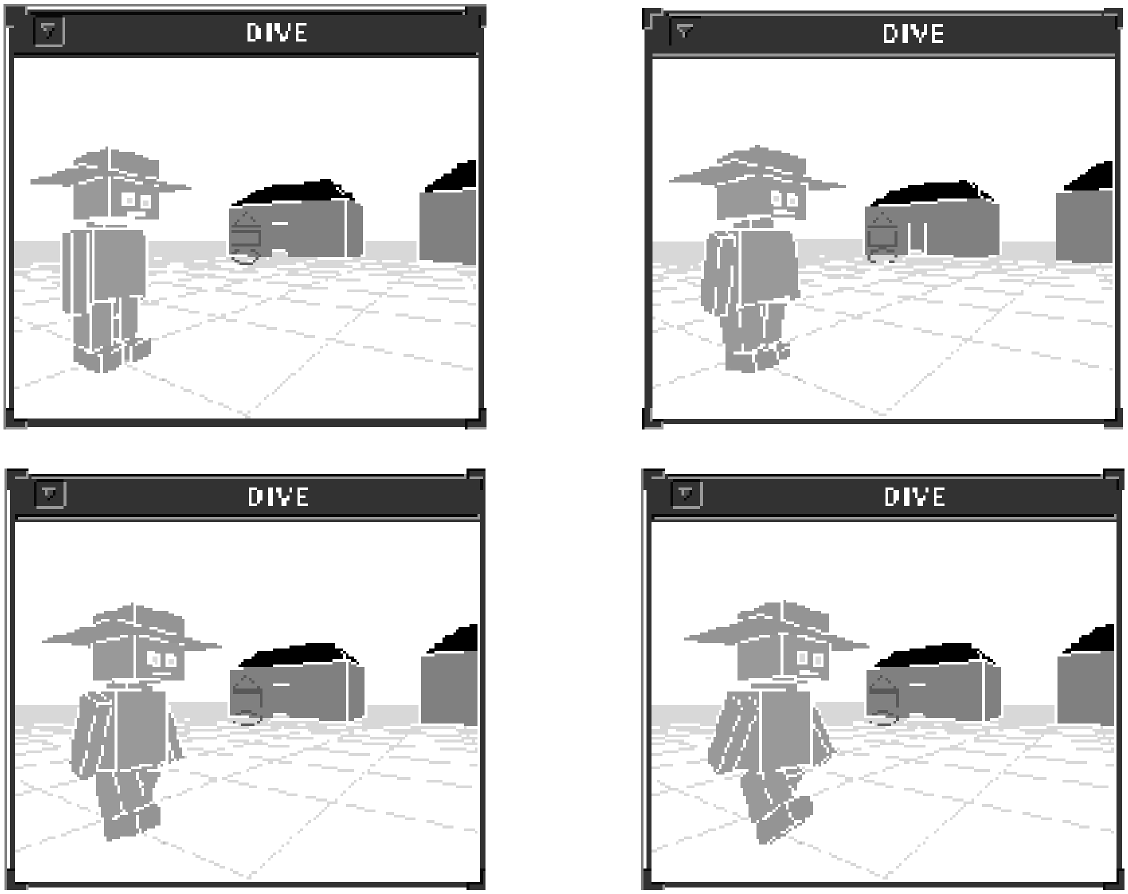


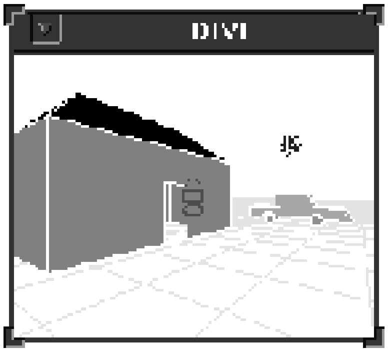
FIGURE 5. A walking motion.

7. AN EXAMPLE OF DIALOGUE

Figure 6 gives an example of what Ulysse can do. It pictures a dialogue between the user and its navigating agent. Comments in column 1 are the users and those in column three are from Ulysse. The changes in the virtual environment are depicted pictorially. The user moves in the world, goes around objects, and visits them. In the experiments we conducted, novice users were not able to do these actions using the standard DIVE finder and a mouse.

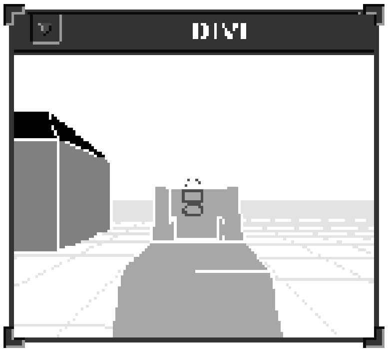
8. CONCLUSION AND PERSPECTIVES

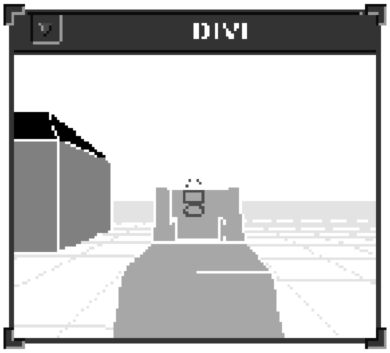
In this paper, we have reported an experiment to assess the usability of speech and language in a 3D collaborative virtual environment and the development of a conversational agent, Ulysse. Ulysse was designed by considering data collected from our experiments. These experiments led to insights into appropriate vocabulary, into input preferences and speech and navigation. Our experimental work suggested that more classical DIVE classical vehicles with mouse-only input proved hard for novice to use. Only extensive training led users to become proficient at mouse driven navigation. We believe that the navigational difficulties we observed could seriously interfere with collaborative activities in virtual environments.



Va devant cette voiture (Go in front of this car)

Voilà (Here it is)





Tourne à droite (look on the left)

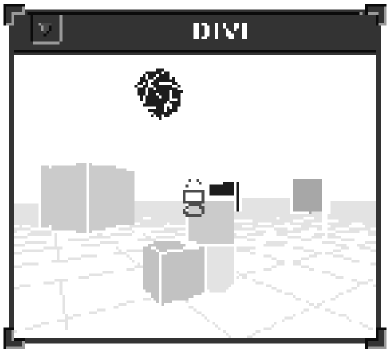
Voilà

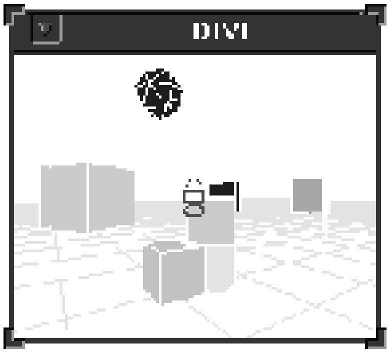
Va vers le cube (Go to the cube)

Il y en a plusieurs (There are several cubes)

Va vers les petits cubes
(Go to the small cubes)

Voilà



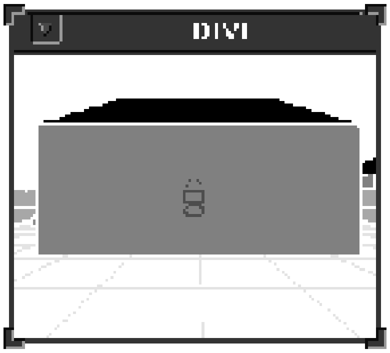


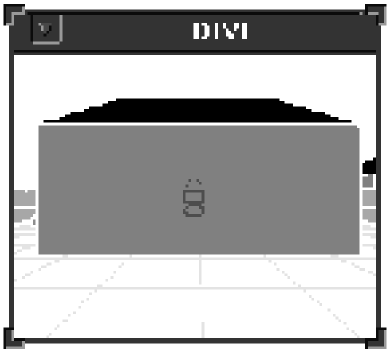
Retourne devant la maison
(Come back to the house)

Voilà

Va derrière

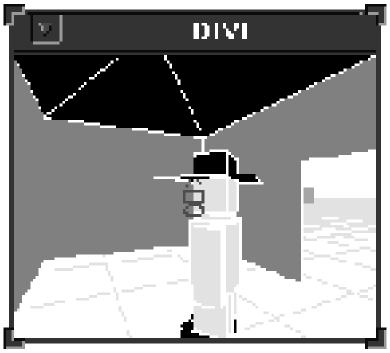
Voilà

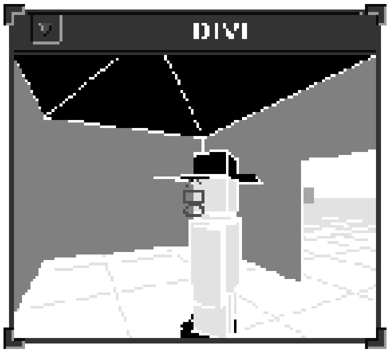




Entre dedans
(Go into)

Voilà





Va devant Jo
(Go in front of Jo)

Voilà

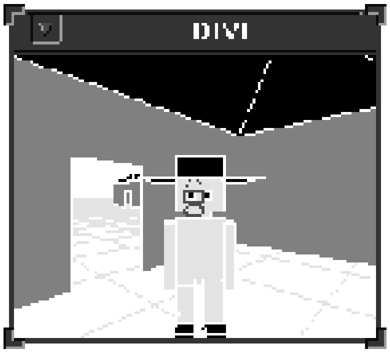


FIGURE 6. A dialogue example: The user asks Ulysse to carry out actions to discover the world.

Ulysse's architecture includes a speech recognition device together with a speech synthesiser. It consists of a chart parser for spoken words, a semantic analyser, a reference resolution system, a geometric reasoner, a dialogue manager, and an animation manager. We think this architecture is relatively generic and we have adapted it to navigate into a virtual brain [25].

Ulysse is incorporated into the user's embodiment and the user can navigate in the world through a spoken dialogue, which is parsed by Ulysse. Acting in co-ordination with other input devices, Ulysse can carry out motions such as going around an object or looking around a room. For instance, the user can say 'go around this house' (while possibly pointing at it) and Ulysse will execute the command while maintaining a visual display of the referent (the house). The user can also look around or go into a room. Such easy navigation and movement makes observing what else exists in the world (for example how many other people are in the room) very easy. Such motion, navigation and vision is crucial to the discovery of a virtual environment and thus to an efficient collaboration. Users are not constrained to speech only input; they can select between mouse and speech input, and thus selects the most appropriate method for the task at hand, rather than being distracted by a cumbersome mechanism.

We plan to embed Ulysse in various objects of virtual environments. The user would then address the object from the 'outside' and order it to carry out specific motions or actions. Continuous improvements in speech recognition make it possible to implement reliable speech interfaces to these objects. However, adapting Ulysse to a variety of objects and actions requires extending its reasoning capabilities. For the moment they are limited to geometric faculties such as navigation and spatial manipulations. Scheduling an agenda, for instance, is impossible because it would require to reason about time.

In more detail, we are now beginning to implement capabilities to co-ordinate a set of agents to interpret written car accident reports and to replay the accident. Each actor of the scene (cars, pedestrians) incorporates its own agent that determines its behaviour in the world. This new system includes a generic agent – the master agent – that is to process the report and to split it into subsets of orders. Orders will be routed to the scene agents that will animate accordingly. We are implementing synchronisation rules to co-ordinate the agents and we are using reactive planning to make agents track moving entities. These rules control motions such as overtaking or the behaviour at cross-roads. The overtaking action notably is parameterised using details from the report that take into account the side of this action (on the left hand side, on the right hand side) or the number of lanes. Although they are preliminary, we have already obtained some promising results in the co-ordination of agents to simulate simple reports.

In conclusion, we believe that utilising conversational agents like Ulysse within virtual environments brings many benefits. Mouse and speech input devices have different characteristics such that each may be best suited to different tasks. Within our system, users can select the most appropriate input style for different tasks and thus can concentrate on collaboration and harvest the full benefits of visualisation and navigation capabilities offered by 3D collaborative virtual environments without the drawbacks.

9. Acknowledgements

This work was carried out within the framework of the COST-14 Computer Supported Co-operative Work project [26] from the European Commission.

References

[1] Magnus Andersson, Christer Carlsson, Olof Hagsand, and Olov Ståhl. Dive, the distributed interactive virtual environment. Technical reference, Swedish Institute of Computer Science, Kista, 1994.

[2] Steve Benford, Chris Greenhalgh, and David Lloyd. Crowded collaborative virtual environments. In *Proceedings of CHI'97*, Atlanta, Georgia, 1997.

[3] Wolfgang Broll. Populating the Internet: Supporting multiple users and shared applications with VRML. In *Proceedings of the VRML'97 Symposium, ACM SIGGRAPH*, pages 87–94, Monterey, CA, February 24-28 1997. ACM.

[4] Mark Pesce. *VRML – Browsing and Building Cyberspace.* New Riders Publishing, 1995.

[5] Steve Benford, John Bowers, Lennart E. Fahlén, Chris Greenhalgh, and Dave Snowdon. User embodiment in collaborative virtual environments. In *Proceedings of CHI'95*, Denver, Colorado, 1995.

[6] George G. Robertson, Stuart K. Card, and Jock D. Mackinlay. Information visualization using 3D interactive animation. *Communications of the ACM*, 36:57–71, 1993.

[7] Stephen Michael Kosslyn. *Ghosts in the Mind's Machine.* Norton, New York, 1983.

[8] Michel Denis. Imagery and thinking. In Cesare Cornoldi and Mark A. McDaniel, editors, *Imagery and Cognition*, pages 103–132. Springer Verlag, New York, 1991.

[9] Kimberley Osberg. Cognition in the virtual environment. HITL Report R-97-18, University of Washington, 1997.

[10] Don Gentner and Jakob Nielsen. The anti-mac interface. *Communications of the ACM*, 39(8):70–82, 1996.

[11] Richard A. Bolt. Put-that-there: Voice and gesture at the graphic interface. *Computer Graphics*, 14(3):262–270, 1980.

[12] Jussi Karlgren, Ivan Bretan, Niklas Frost, and Lars Jonsson. Interaction, models, reference, and interactivity in speech interfaces to virtual environments. In *Proceedings of the 2nd Eurographics Workshop*, Monte Carlo, 1995. Darmstadt: Fraunhofer IGD.

[13] Gene Ball, Dan Ling, David Kurlander, John Miller, David Pugh, Tim Skelly, Andy Stankosky, David Thiel, Maarten van Dantzich, and Trace Wax. Lifelike computer characters: the Persona project at Microsoft. In Jeff M. Bradshaw, editor, *Software Agents*. AAAI/MIT Press, Menlo Park, California, 1997.

[14] Stephanie S. Everett, Kenneth Wauchope, and Manuel A. Pérez. A natural language interface for virtual reality systems. Technical report, Navy Center for Artificial Intelligence, US Naval Research Laboratory, Washington DC, 1995.

[15] Christophe Godéreaux, Korinna Diebel, Pierre-Olivier El Guedj, Frédéric Revolta, and Pierre Nugues. An interactive spoken dialog interface to virtual worlds. In John H. Connolly and Lyn Pemberton, editors, *Linguistic Concepts and Methods in CSCW*, Computer supported cooperative work, chapter 13, pages 177–200. Springer, London, 1996.

[16] Pierre Nugues, Christophe Godéreaux, Pierre-Olivier El-Guedj, and Frédéric Revolta. A conversational agent to navigate in virtual worlds. In Susann LuperFoy, Anton Nijholt, and Gerd Veldhuijzen van Zanten, editors, *Proceedings of the eleventh Twente Workshop on Language Technology, TWLT 11, Dialogue Management in Natural Language Systems*, pages 23–33, Enschede, 1996. Universiteit Twente.

[17] James F. Allen, Lenhart K. Schubert, George Ferguson, Peter Heeman, Chung Hee Hwang, Tsuneaki Kato, Marc Light, Nathaniel G. Martin, Bradford W. Miller, Massimo Poesio, and David R. Traum. The TRAINS project: A case study in building a conversational planning agent. TRAINS Technical Note 94-3, University of Rochester, Rochester, 1994.

[18] Pierre-Olivier El Guedj. *Analyse syntaxique par chart combinant règles de dépendance et règles syntagmatiques*. PhD thesis, Université de Caen, 1996.

[19] Lucien Tesnière. *Éléments de syntaxe structurale*. Klincksieck, Paris, 1959.

[20] Nancy Ide and Jean Véronis, editors. *The Text Encoding Initiative: Background and Context*, Dordrecht, 1995. Kluwer Academic Publishers.

[21] Marion Mast, Franz Kummert, Ute Ehrlich, Gernot A. Fink, Thomas Kuhn, Heinrich Niemann, and Gerhard Sagerer. A speech understanding and dialog system with a homogeneous linguistic knowledge base. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):179–194, 1994.

[22] Carla Huls, Edwin Bos, and Wim Claassen. Automatic referent resolution of deictic and anaphoric expressions. *Computational Linguistics*, 21(1):59–79, 1995.

[23] Richard Fikes and Nils J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–208, 1971.

[24] Earl D. Sacerdoti. Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, 5(2):115–135, 1974.

[25] Matthias Ludwig, Alexandre Lenoir, and Pierre Nugues. A conversational agent to navigate into MRI brain images. In *Proceedings Interfaces 97, 6th International Conference*, pages 207–210, Montpellier, May, 28-30 1997.

[26] CoTech. Minutes of the COTECH workgroup, virtual and augmented environments for CSCW. Technical report, University of Nottingham, Department of Computer Science, Nottingham, 1995.