A Framework for a Multi-modal Query Interaction of the Virtual World Database System

Chiemi Watanabe†, Yoshifumi Masunaga‡
† Graduate School of Humanities and Sciences, Ochanomizu University
‡ Department of Information Science, Ochanomizu University

The VWDB is currently under development at Ochanomizu University with the aim of realizing a new generation database system for cyberspace applications. A prototype system is constructed by integrating a virtual reality system and an object-oriented database system. Since the VWDB provides a virtual world as the user's database interface, it must support a multi-modal query language. In this paper, we investigate a design of a multi-modal query language of the VWDB. We first analyze features of the multi-modal query interactions in the virtual world and identify that multi-modal interactions cause state transitions of the virtual world. Based on this observation, we introduce a multi-modal interaction model of the VWDB query language. A world algebra and an attribute grammar based approach to characterize the state transition are reported.

1 Introduction

Recent development of the virtual reality technology is great and they are used in many application areas such as simulation, communication and so on. And these applications require database functions for virtual world and its objects. For example, in a virtual shopping mall with database functions, customers can issue a query about the price of merchandise using their hands and voice, and owners can manage their products. The idea of integrating a VR system and a database system occurred quite naturally because it seemed that such an integrated system could inherit both the VR capability and the database capability so that it could provide a set of all necessary functions to describe a variety of database-based VR applications. We call this the Virtual World Database System (VWDB in short). We provide a query function to the VWDB. Users issue a variety of queries using their hands and voice in the immersive virtual environment. Though some VR applications that have query module are reported [4] [5] [9], users of these applications look the virtual world from the bird-view, and they issue queries by using 2D GUI form from outside of the virtual world. A user of these applications must get out of the immerive virtual environment once when he issues queries. Then we provide an environment in which users can be immersed and they can issue a variety of queries in a multi-modal manner. In this paper, we propose an framework for the multi-modal query interaction in the VWDB.

2 System Architecture

The VWDB is currently being developed at Ochanomizu University by integrating a VR system and an object-oriented database system. Figure 1 shows the current system architecture of the VWDB. We use a VR system as the front-end system and an object-oriented database system as the back-end system, and we integrate these two systems by developing VWDB client modules and server modules. The architecture is scalable in the sense that more than one VR system could be connected to an object-oriented database system via a network, which would be necessary in order to realize a distributed virtual environment (DVE) for collaborative work support. Figure 2 shows a set of input and output devices for users to wear when using the VWDB prototype system. By wearing an HMD, users can see the virtual world in 3-D. The 3-D 6DOF mouse enables users to move from their current position in any of six directions, i.e., up, down, right, left, forward and backward, in the virtual world. A magnetic sensor is attached to the top of a user's head to capture the direction of the user's head as well as to detect the user's or the avatar's position when processing location-sensitive queries and updates. These devices enable a user to walk through the virtual world as if he/she were in the real world. In addition to wearing the HMD and sensor and using the mouse, the user wears a data glove on his/her right hand and a headset on his/her head. The headset is connected to DS200, a voice recognition system from Speech System Inc. that processes a voice input based on the sentence pattern analysis method. These devices are necessary for measuring and recognizing the user's gestures and voice, and they ultimately allow the user to update and query the virtual world and its objects in a multi-modal manner. Figure 3 shows a snapshot of a screen image of the VWDB prototype system. A virtual office(students room) with database function is prototyped. User can be immersed in the virtual office(students room) and they can issue the queries. For example, when she points this object and says "who is the owner of this object", system answers the owner's name who has this object.

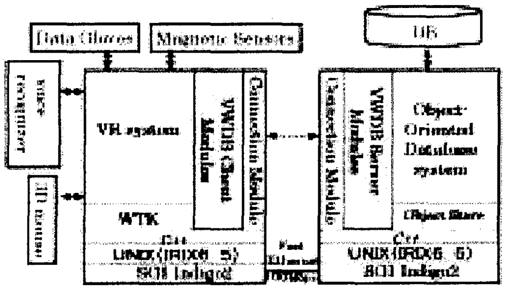

Figure 1: System Architecture

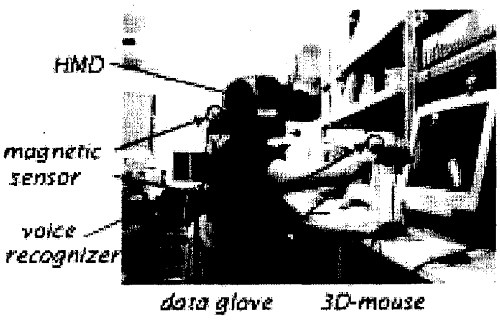

Figure 2: Input Equipments

3 Modeling the Virtual World and Its Objects

In the VWDB, user is immersed in a virtual world and issues a variety of queries about it virtual world

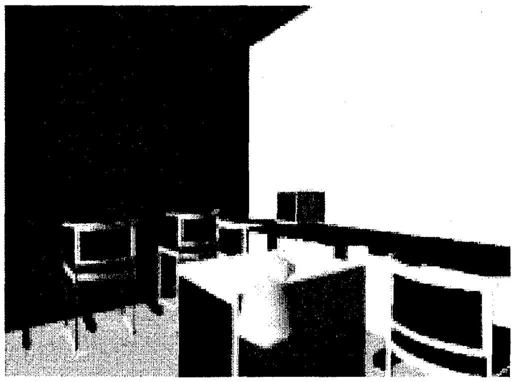

Figure 3: Screen Image of the VWDB Prototype System

and its objects. In order to realize to do that, we provide a model to store the virtual world and a virtual world object in the back-end data base. We defined primary model for virtual world and its objects by [3]. In this chapter, we describe characteristics of the virtual world and its objects and primary model for them.

3.1 Virtual World

Virtual world is the 3D space that has world coordinate system. Virtual world is represented by tree structure as shown Figure 4. We call this structure "World". World is based on scene graph that is hierarchical structure for creating a scene. Relationship between parents and children means IS-PART-OF relationship.

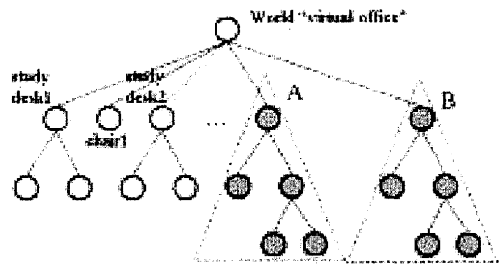

Figure 4: Tree Structure of the "World"

3.2 Sub world

We suppose that querying is to extract a set of sub-tree from World. For example, when a user ask "which table is what I bought yesterday?", a set of sub-tree shown in Figure 4 is extracted. We call this set of sub-tree "Sub World".

3.3 Virtual world object (VWobject)

VWobject is an element of World and it is an object which is in the virtual world. Each VWobject has spatial attributes such as position, direction and orientation, 3D shape, and non-spatial attributes such as owner, date of packing, name and so on.

4 Multi-modal Query Interaction

Suppose that one wants to distribute a box to each student desk in a virtual study room shown in figure 3. Suppose also that a human computer interaction might be done in a multi-modal manner. In this situation, he/she might interact with the VWDB in the following way:

User : He/she touches a box and says "this box".

System : It indicates the box by emphasis.

User : He/she asks "who is the owner of this object?"

System : It answers that "the name of the owner is Chiemi Watanabe."

User : He/she asks "which one is the student desk that this student has?"

System : It indicates the student desk by emphasis.

User :(He/she moves the box on the student desk.)

Figure 5 shows a workflow of query interaction in the VWDB.

The two main modalities in the VWDB are gestures made with the fingers of a data glove and user's voice(①). These two modalities, and avatar's data such as the avatar's position, direction and rotation are combined(②), and queries issued by means of such modalities are translated into OQL statements. The OQL statement is published to the back-end database(③). According to the query result(④), the system updates the 3-D world representation in the VR system(⑤). The multi-modal interaction manager has the information which objects match to the search condition of the last query. When a next multi-modal query is issued, this information is used to process the multi-modal input sequence to create an OQL statement.

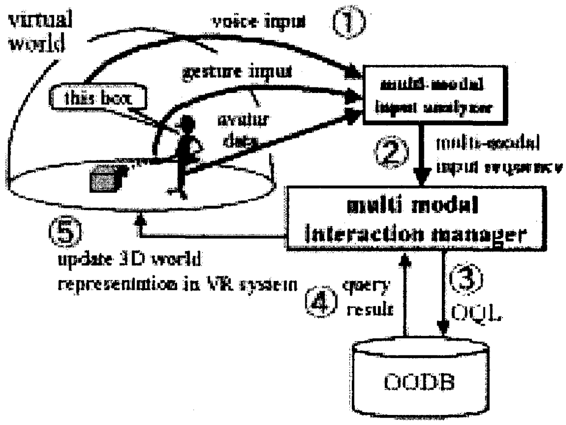

Figure 5: The Workflow of the Query Interaction in the VWDB

One might think that the above multi-modal query interaction can be substituted by a single query like "Which one is the desk that the owner of this box has?" by pointing a box. However, we believe that the multi-step multi-modal interaction is necessary in order to resolve uncertainty. That is, in the above interaction, we can't recognize explicitly which object the system has recognized, i.e. either the system has recognized the object that the user's finger pointed out, and also we are unsure either the system has recognized the user's speech correctly or not. In contrast, if a user interacts with the system step by step and gets feedback at each time from the system such as by indicating the pointed object or answering the desk owner's name, the user can check whether the input is recognized correctly or not. The multi-modal interaction in the VWDB is managed by the "multi-modal interaction manager".

5 Framework for a Multi-modal Query Interaction

In the previous section, we investigated how to process multi-modal query interactions in the VWDB. A framework for the multi-modal query interaction needs to describe how the VWDB system interacts with a user so that the user can get information that he/she wants. Corresponding to the interactions shown in Figure 5, the framework should be able to satisfy the following 5 requirements:

1. What information does the multi-modal interaction manager manage in order to issue queries to users interactively?

2. A grammar for generating multi-modal input sequences is necessary, which is Corresponding to (②) in Figure 5.

3. What OQL statement does the multi-modal interaction manager create based on the information about objects that are specified by the last query and the current multi-modal interaction? This is corresponding to (③) in Figure 5.

4. How does multi-modal interaction manager update information about objects based on query result? This is corresponding to (④) in Figure 5.

5. How does the multi-modal interaction manager update the virtual world as a result of a query? This is corresponding to (⑤) in Figure 5.

In this section, we design a framework for a multi-modal query interaction as follows :

- Characterize the relationship among the multi-modal input sequences issued by a user during interacting with the VWDB system, the states of the virtual world, and the outputs to the backend object-oriented database system.

- Define a grammar for generating multi-modal input sequences, OQL statements to the back-end object-oriented database system that corresponds to input sequences, and update-messages to the front-end VR system.

To design a framework, we first characterize the relationship mentioned above. In [2], we designed and implemented a framework for a multi-modal interaction for data manipulation in the VWDB. In designing the framework, we noticed that an interaction initiated by a multi-modal input sequence causes a state change of the target object, and we characterized the state transition of the target object using a set of automata. When a user issues a query, then all objects in the virtual world are checked each by each whether it satisfies the search condition or not. Then, a state list of all objects in the virtual world is defined, and we can characterize the state transition of the virtual world by using a set of Mealy type automata.

[Definition] State of the virtual world
Suppose there are objects $o_1, ..., o_n$ in the virtual world. Then S_i represents the state of the virtual world which is defined as a list of states of all objects $o_1, ..., o_n$. The state of an object is either (q_1) or (q_0) which corresponds to either the objects belong to the set of objects that satisfy the search condition of an issued query interaction.

Figure 6 shows the state transition of the virtual world caused by the interaction described in section 4.

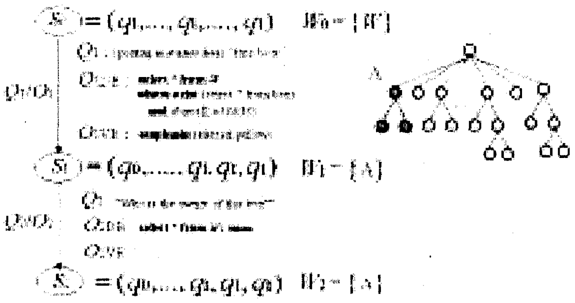

Figure 6: The state change showing interaction at the time of uttering "this box", while the user pointed at the box.

When a user says "this box" with pointing at the box, a system executes O_1. Output O_i consists of an OQL statement $O_{i_{DB}}$ and an update-message to the VR system $O_{i_{VR}}$. In Figure 6, $O_{1_{DB}}$ shows the message to select an box which is pointed by the user, and $O_{1_{VR}}$ is the message to indicate the selected box. When $O_{i_{DB}}$ is send to the backend database system, the system extracts a sub-world which consists of the selected objects, i.e. the objects with state q_1. In this example, the box pointed by the user and its part objects change its state from q_0 to q_1. Next, when the user says "who is the owner of this box?"(Q_2), the system sends message O_2 to the backend DB system and the front-end VR system. "This box" in Q_2 is the object which satisfies the previous query, and it is the element of the sub-world which is obtained by $O_{1_{DB}}$.

At the next step for designing a framework, we define a grammar generating a multi-modal input sequence (Q_i), an OQL statement corresponding to the input sequence $(O_{i_{DB}})$, and an update message to the front-end VR system$(O_{i_{VR}})$.

(1) Grammar for the multi-modal input sequence (Q_i) :
We define a grammar for the multi-modal input sequence for a query interaction(Q_i) using Multi-Modal Definite Clause Grammar (MM-DCG) [7]. The MM-DCG extended Definite Clause Grammar (DCG) which is a grammar description form for a

natural language for a multi-modal input sequence, and it can deal with different kind of modalities. The basic procedure of MM-DCG is described as follows.

```
head--> body.{ <semantic rule> }
```

This procedure means "the head can exist if the body, i.e. a set of goals is satisfied". A procedure consists of a syntactic rule and a set of semantic rules. Each Terminal or non-terminal symbol has its own attribute value. Attribute values of head is evaluated from the attribute values of goals in accordance with semantic rules. When body includes terminal symbols, the procedure is described as follows.

```
head--> modality:[word] {<semantic rule>}.
```

This means that the head exists if a user uses this modality to issue an input word.

We designed a grammar generating multi-modal input sequences in the VWDB. Figure 7 shows a part of the grammar.

```
Selection(aname,avalue,operator,category)
 --> pointing(objectID, position, direction),
    reference_word, category_name(category)
    { selection.aname= " objectID " ;
     selection.avalue=pointing.objectID;
     selection.operator=category_name.category;}

pointing(objectID,position,direction)
 --> gesture:[(-15<hand.finger.index.first_joint<15)]
{  objectID=hand.pointed_object;
   position=hand.finger.index.positon;
   direction=hand.finger.index.tip
              -hand.finger.index.base;
}

reference_word
 --> speech:[this] | Speech:[that]|...
category_name(category)
 --> speech:[student desk], speech:[box],
     speech:[student_chair]|...
...
```

Figure 7: A part of the grammar description for multi-modal input sequence in the VWDB

This description characterizes an input sequence which specifies an object by saying "this" pointing by user's finger. Non-terminal "Selection" has attribute values such as "aname" which is attribute name using in search condition, "avalue" which is attribute value using in search condition, "operator" which is operator comparing aname with avalue, and "category" which is category name. When user says "this box" pointing an box whose objectID is 398385, these multi-modal input sequence is accepted and system gets attribute values as follows : aname = ObjectID, avalue= 398385, operator = "=", category=box. These attribute values are used to creating a OQL statement corresponding to multi-modal input sequence.

(2) OQL statement corresponding to an multi-modal input sequence :

To create OQL statement corresponding to multi-modal input sequenceQ_i, attribute values which are evaluated from Q_i and sub-world W_i are used. We prepare an OQL pattern corresponding to a multi-modal input sequence in advance. Attribute values which are evaluated from the multi-modal input sequence are set inside the and the OQL pattern, an then an OQL statement is formed. The OQL pattern corresponding to a part of the grammar shown in Figure 7 is given below.

select *
from W
where exist (select * from < *category* >)
and < *aname* >< *operator* >< *avalue* >;

By setting attribute values and a sub-world, an OQL statement is created as follows.

select *
from W
where exist (select * from study_desk)
and objectID = 398385;

(3) Update-message to the front-end VR system

An update-message to the front-end VR system describes either an updating command to the display in the virtual world or a voice output command for displaying the query result in the virtual space. We provide a set of commands for describing update-messages. For example, the following command is prepared to indicate a set of objects that correspond to a query result by emphasizing tem in yellow color.

```
emphasis(selected,yellow);
```

`selected` represents a set of objects that match the query as well as the objects used as the elements of a sub-world extracted by the query.

6 Conclusion

We describe a framework for a multi-modal query interaction in the VWDB. This framework represents a relationship between a user and a system when they interact queries. It characterizes a multi-modal input sequence, and an OQL statement and an update-message to the front-end VR system corresponding to the multi-modal input sequence, respectively. Refinement of the proposed framework and its implementation are left as future work.

References

[1] Masunaga, Y. and Watanabe, C.: "The Virtual World Database System - Its Concept, Design and Prototyping - " , Advances in Multimedia and Databases for the New Century - A Swiss/Japanese Perspective -, pp.61-70, World Scientific (2000).

[2] Masunaga, Y. and Watanabe, C.: "Design and Imprementation of a Multi-modal User Interface of the Virtual World Database System(VWDB)", Proceedings of seventh International Conference on Database Systems for Advanced Application(DASFAA'01)(2001).

[3] Chiemi, W., Masunaga, Y.: "Design of a Schema and Domain Definition Language in the Virtual World Database System, 12th Data Engineering Work Shop 1B-3(2001).

[4] Gausemeier, J., Krumm, H., Molt, T. Gbbesmeyer, P. and Gehrmann, P : "A Database Driven Server For An Internet Based Plant Layout Presentation", Proceedings of VRML2000,pp.17-22 (2000).

[5] Kamiura M., Oiso H., Tajima K. and Tanaka K.: "Spatial Views and LOD-based Access Control in VRML-object Database",Worldwide Computing and Its Application, pp210-225, LNCS 1274, Springer (1997).

[6] Hirose M. eds. : "basic technologies of virtual reality system", baifukan, 195p. (in japanese)(2000).

[7] Shimazu H., Takashima Y. : Multi-modal Definite Clause Grammar(MM-DCG), The IEICE Transactions, D-II, Vol. J77-D-II, No.8, pp.1438-1446(in japanese) (1994).

[8] Egenhofer M. J. : Point-set topological spatial relations, Journal of Geographical Information Systems, Vol.5, No.2, pp.161-174 (1991).

[9] Van Ballegooij, A. and Eli?ns, A.: "Navigation by Query in Virtual Worlds", Web3D 2001 Conference, pp.19-22 (2001).