

Descriptive Modeling Language based on Semantic

Lijun Xue^{1, a}, Lili Wang^{2, b}

¹Information and Technology Center, Jiangsu University, Zhenjiang, 212013, China

²School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang, 212013, China

^aemail: ljxue@ujs.edu.cn, ^bemail:xlj200203@163.com

Keywords: Semantic; Semantic Constraints; Virtual Reality; 3D Modeling

Abstract. Aiming at the shortage of current modeling by graphical modeling language, this paper presents a descriptive modeling language based on Semantic. This paper mainly introduces the design principle, modeling primitives to describe the modeling language and the design of the parser and the implementation process of this descriptive modeling language. The experimental results show that, the modeling language basically corresponds the thinking and operation habit of users, reduces complexity and difficulty of the virtual scene modeling, accelerate the speed of modeling.

Lack of Graphical Modeling Language Modeling

Graphical modeling language in the field of information system has had many years of research, such as the famous conceptual modeling language UML (Unified Modelling Language) [1], ER model (Entity Relationship) [2] and ORM model (Object Role Model) [3]. ORM and ER for static scenes modeling has enough power of expression, but the interaction and behavior description express has many limitations. UML has the ability of the virtual environment modeling[4] [5], but for the design of virtual reality, it lacks the ability to express.

The current graphical modeling language directly to virtual scene modeling is not suitable, we should design a more intuitive for the virtual scene modeling language. Therefore, we propose a declarative modeling language based on semantic. Using this modeling language to describe virtual scene required, the user modeling intentions directly is transmitted to the computer, then the computer automatically completes modeling tasks required by the user through the semantic reasoning.

Design Principles of Descriptive Modeling Language

The design of descriptive modeling language needs to follow the following principles:

A. Simple

Language must be concise, easy to learn[6]. Virtual scene modeling staff do not need to have the ability of programming, and will be able to grasp the method of using the declarative modeling language in a short time.

B. General

Descriptive modeling language must be versatile[7], not limited to a specific field. The modeling language can be used to model virtual scene in various fields.

C. Intuitive

Descriptive modeling language needs to accord with human cognitive ability of daily, avoid jargon used virtual reality or the graphics area, close to natural language expressions.

Modeling Primitives and Representing Method based on Descriptive Modeling Language

The main task of descriptive modeling language is to make the computer to complete virtual scene modeling process automatically. The syntax uses verb + object structure to describe the scene, and then the language parser generates virtual scenes automatically.

Modeling language has the following structure:

< object><action> [parameters]

Object indicates the subject object; Action indicates the subject object movement; Parameters indicate the action parameters. Descriptive modeling language mainly includes the following modeling primitives.

a) select scene objects

Select object;
indicates selected scene object.

b) set properties

The set command sets object attributes, such as size, color, material, texture, light, such as:
Set size=10,5,1;
Set the object size: length and width and height are 10,5,1.

c) add

Insert object;
indicates to add a object in the scene.

d) delete

Delete object;
indicates to delete the object in the scene.

e) move

The move command of scene object is precise movement, such as:
Move forward 2;

The statement indicates to move 2 units of distance to the front, forward indicates direction. The statement did not give specific units, at the beginning of the modeling needs to set the unit, otherwise it will use the default unit.

f) rotate

The rotate command of scene object is accurate rotation. Such as:
Rotate back 2;

indicates the backward rotation of 2 units, the default clockwise rotation, at the beginning of the modeling needs to set the unit.

g) assemble

The assemble command executes composition operation between scene objects.

Assemble {object1, object2,... , objectN};

indicates to join the scene objects object1, object2... objectN, What interface connection between objects is decided by interface class defined in the semantics library.

h) disassemble

The disassemble command makes an object get rid of grouping constraints from the other objects, in order to operate the object separately.

Disassemble object1 {object1, object2, object3,... ObjectN};

indicates to make the scene object object1 get rid of grouping constraints from {object1, object2, object3... objectN} .

Execution of Descriptive Modeling Language

Execution of descriptive modeling language is as shown in Figure 1.

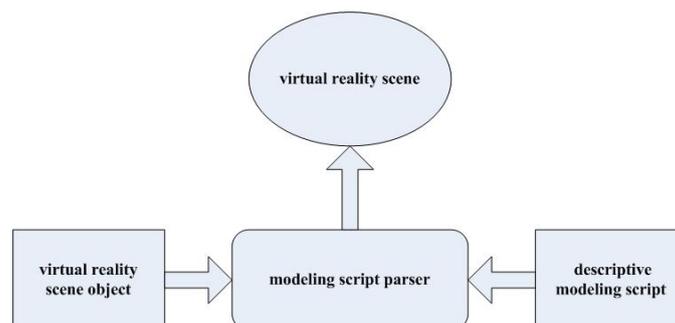


Fig.1. Implementation process of descriptive modeling script

The implementation of specific process is as follows:

a) the user edits the modeling script with the modeling script editor and transfers to script parser for processing.

b) the modeling script parser first judges object to perform by modeling scripts and then queries scene object library to judge the means of the object, namely it is representative of a specific example or a class of objects, and then infers the semantic information contained in the script, and extracts scene objects to place in the virtual reality scene in the corresponding.

c) modeling script parser parses semantic, and the analytic results X3D file is transferred to the scene generation module to automatically display the modeling scene.

Design of the Parser

The design of the parser is as shown in Figure 2.

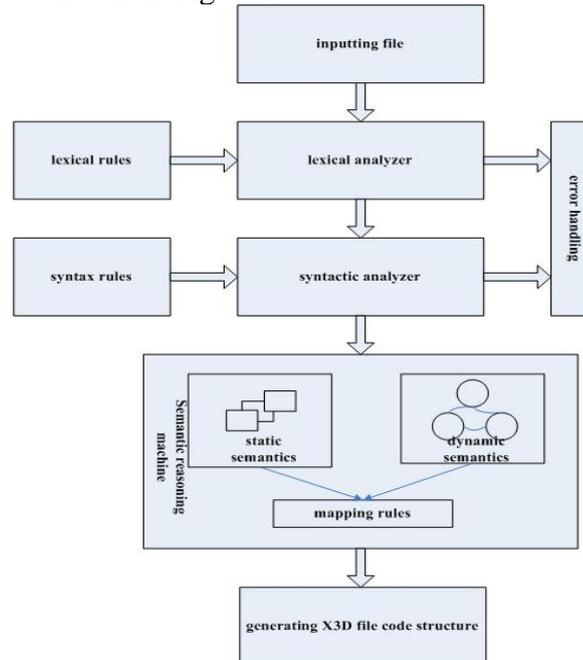


Fig.2. The parser design of descriptive modeling language

First reading the X3D model file, and then the file is transformed into the X3D document in memory, so that other components operate on it through the XML DOM API[4].

Querying and reasoning on the semantic by semantic inference engine. First, to ensure the semantic consistency, to check whether axiom constraints of the class, attribute and instance defined in system is conflicting. Second, to reason implicit knowledge in system.

The parse includes the parse of the class, equivalence class, attributes and instance. The class and equivalence class parse will get the subclass, the parent class or equivalent class, and horizontal and vertical relationship between scene object class, contains relation, equivalent relation, intersection relation, the relationship between the virtual interface class(including the transverse and longitudinal) and interface connection relation etc.. Attributes parse can judge all attribute of class, attribute types and attribute values. Instance parse can parse out all instances of a class, in addition, also can deduce the dependent relationship between instances and classes and the relationship between instances and instances.

Experiment and analysis

We first input text, and then system annotates and analyses sentence, converts the parsed output into a dependency structure, and the dependency structure is interpreted by semantic and converted into semantic expression. Under the rule mapping support, semantic expression is transformed into a set of low level 3D object description, including posture, spatial relations, the color attribute. Finally, according to the description the virtual scene is generated. As shown in Figure 3 is a scene instance generated by this modeling language.

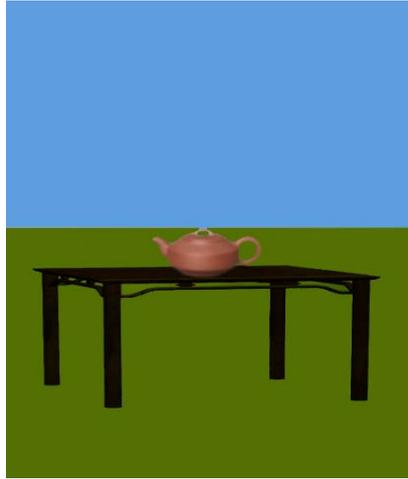


Fig.3. A scene instance generated by descriptive modeling language.

Input text of the scene:

Select background;Set skyColor=blue, groundColor=green;Insert background;Select table;Insert table;Move forward 2;Rotate left 3;Select pot;Insert pot;Move forward 2.

Experiments show that the modeling language reduces the virtual scene modeling complexity and difficulty, basically accord with the user's thinking and operation habit, method of modeling is easy to grasp, to speed up the speed of modeling, so that no 3D modeling experienced users can easily make the virtual scene modeling.

Conclusion

This paper first analyzes the deficiency of graphical modeling language, and then puts forward descriptive modeling language based on semantic to accomplish virtual scene modeling interactive task. This paper mainly introduces the design principles, modeling primitives to describe the modeling language, the design of the parser and implementation process of descriptive modeling language. Construction method of modeling language to introduce semantic information is discussed in this paper.

References

- [1] E. J. Braude. Software Engineering: An Object-Oriented Perspective. Wiley, 2000(1).
- [2] P. Susanne. Evolution of entity-relationship modeling. Data & Knowledge Engineering. 2006, 56(2): 122~138.
- [3] T. Halpin. Information Modeling and Relational Databases, From Conceptual Analysis to Logical Design. Morgan Kaufmann, 2001.
- [4] S. M. Kim, S. I. Yoo. Dom tree browsing of a very large XML document: design and implementation. Journal of Systems and Software, 2009,82(11):1843~1858.
- [5]<http://www.web3d.org/realtime-3d/specification/current>
- [6] D. Bowman, D. Johnson and L. Hodges. Testbed Evaluation of Virtual Environment Interaction Techniques. Presence: Teleoperators and Virtual Environments. 2001, 10(1):75~95.
- [7] D. Bowman, C. Wingrav. Design and Evaluation of Menu Systems for Immersive Virtual Environments. Proceedings of IEEE Virtual Reality. IEEE Press,2001:149~156.
- [8] Azuma R T. A Survey of Augmented Reality[J]. Teleoperators and Virtual Environments, 1997, 6(4): 355-385.