Three-Dimensional Visualization Of Relational Databases

Tibor I. Kesztyüs & Klaus P. Kratzer
Fachhochschule Ulm

Abstract

In recent years, databases have reached unprecedented complexity and volume. A database structure comprising tens of thousands of tables with a staggering number of inter-table relationships to match can hardly be understood and managed by the human mind. This holds in particular, if two-dimensional visualizations of such a structure are used. This paper describes a project to enhance human comprehension by another dimension enabling database users to move among database tables in three dimensions. The project applies techniques of virtual reality enabling users to control the individual aspect and the level of detail while gaining valuable insights into the structure, the contents, and the retrieval strategies within their databases.

1. Introduction

From a technical point of view, databases, namely relational databases, are meanwhile well understood after more than thirty years of research and extensive application. However, the cognitive exploration of a database as the foundation of an application system is becoming more and more difficult, since the complexity of the average database has increased tremendously due to the ever-extending technical limitations of such systems. In the Seventies of the last century, a database comprising 50 to 100 tables was considered complex --- nowadays, a full-fledged ERP system database will hold close to 30,000 tables with a matching number of inter-table relationships. The tools used to manage such intricate data structures have basically remained the same: mainly two-dimensional dynamic diagrams based on Peter Chen's ([Chen 1976]) notation depicting the database in different levels of detail.

It would be rather arrogant to assume that the cognitive capability of the human mind has remained in step with this ever-growing complexity. Therefore, news ways of visualization have to be investigated to support human administrators. One could argue that database design and administration simply has to follow the lead of other fields of expertise, e.g. precision engineering and medicine, where a three-dimensional approach to visualization induced a paradigmatic step towards management of extremely complex information.

The project outlined in this paper aims at a three-dimensional immersive visualization of databases, on different levels of detail and with various functional properties. Of course, there is a wild side to this approach, fully in the tradition of the *cyperpunk* movement of the Eighties (as manifested e.g. in *Neuromancer* [Gibson 1984]) but it can be safely expected that in the near future users of network clients will be capable of (and expecting to) use three-dimensional interfaces with the same ease as today's users use graphical interfaces based on the two-dimensional desktop metaphor.

This approach has already been used in many instances (see e.g. [Morcrette 1999], [Nakazato 2001]). However, the emphasis is invariably directed at visualization of data to increase its transparency, not at understanding of the conceptual structures underneath, whereas the project described in this paper is mainly focused on the administrator's view of a database.

2. Tools and Methods

The term “Virtual Reality” is used in a variety of ways and often in a confusing and misleading manner. Originally, the term referred to “Immersive Virtual Reality”. In immersive VR, the user becomes fully engulfed in an artificial, three-dimensional world that is based on a geometric model and completely generated by a computer. The immersive quality is supported by special peripheral hardware that tries to overwhelm the sensory apparatus of the recipient ranging from conventional displays with synchronized shutter goggles to elaborate CAVEs (see Fig. 1) with projection surfaces all around the user --- possibly augmented by pressure gloves and pads to stimulate the tactile senses and reflective gear to located the user precisely and allow for interactive behavior. Immense processing power is needed to create that fictitious impression; therefore, just a few years ago VR has become widely available at acceptable cost.

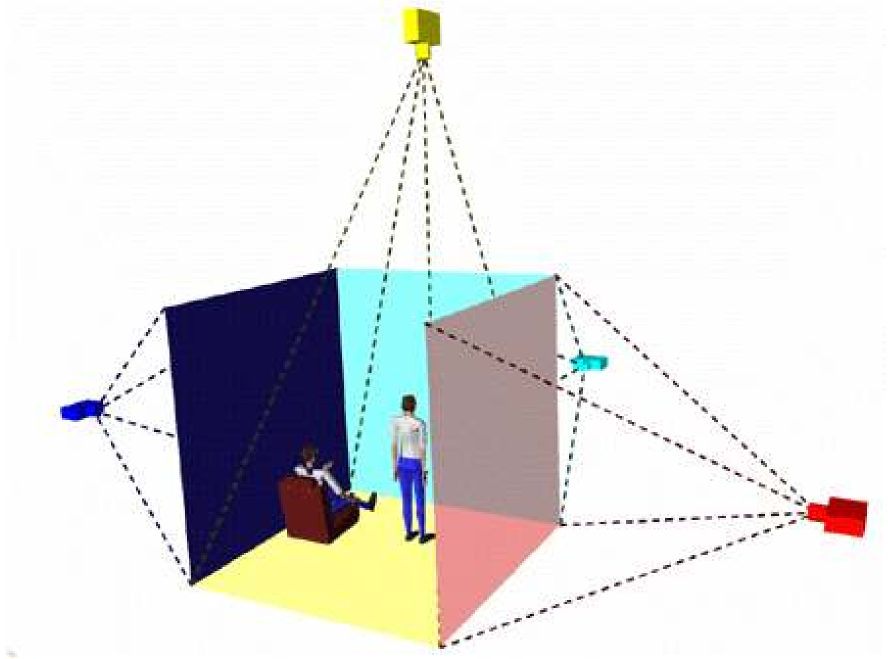

Fig. 1: Schematic of a CAVE

The universal standard for the propagation, export/import, and screening of virtual worlds is the Virtual Reality Modeling Language (VRML, see e.g. [Carey 1997]). This descriptive language can been seen as a 3-D visual extension of the World Wide Web. People can navigate through 3-D space and interact with objects that respond by pre-programmed animations or lead to other virtual world represented by URLs. VRML includes sensory components, interpolating animations, and a script language for flexible control. In its most basic and low cost version, a virtual world can be viewed with a conventional browser software plus VRML plug-in, for instance Cosmo Player.

Figure 2 shows a sample of VRML code and a view of the corresponding primitive world including the navigational controls of a mouse-controlled “player”. The ASCII file (whose name, by convention, ends in *.wrl*) first defines some light sources, then a cone and a sphere of different material.


```
#VRML V2.0 utf8

DirectionalLight {
  direction 1 0 0
}

DirectionalLight {
  direction -1 0 0
}

Transform {
  translation -2 0 0
  children Shape {
    appearance Appearance {
      material Material {
        diffuseColor 0.3 0.3 0
      }
    }
    geometry Cone {}
  }
}

Transform {
  translation 2 0 0
  children Shape {
    appearance Appearance {
      material Material {
        diffuseColor 0 0 0.4
      }
    }
    geometry Sphere {}
  }
}
```

Fig. 2: Sample Of A Virtual World And Its Visualization

The programming environment to bridge the gap between databases and virtual reality was chosen to be Java (see e.g. [Eckel 2002]), mainly for its platform-independence and the standardized JDBC database interface for relational databases. In particular, JDBC provides extensive metadata information via Java method calls to obtain in-depth information on all available database structures and their properties and parameters.

3. System Design and Functionality

The basic version of the system uses a stand-alone generator written in Java which retrieves all data and metadata information from the relational database and creates a VRML virtual world as ASCII output in a *wrl* file. This file, in turn, is interpreted by a standard viewer providing the user with fixed viewpoints for a walkthrough, but also with absolute navigational freedom to study the structures individually.
VRML provides a class concept to support different representations for tables, relationships, and table contents. The three-dimensional layout may be

modified by parameters to enable the user to deviate from the standard layout chosen by default. This layout would classify the tables according to the number of their relationships to other tables, use that classification to assign tables to layers, and arrange tables in a circular way. Figure 3 shows a global view at a database underneath a medical information system comprising 60 tables (depicted as cubes) together with field descriptions (ellipsoids) and inter-table relationships (edges).

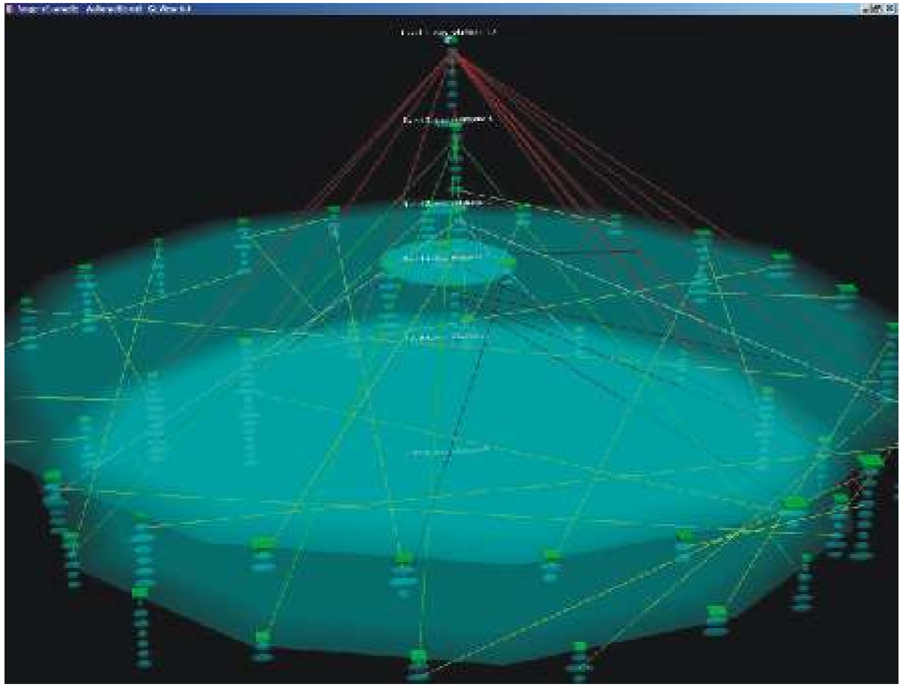

Fig. 3: Global View Of The Database Structures

A closer view is shown in Figure 4. The edges symbolizing relationships are colored differently depending on the level from which each relationship originates.

This prototype version of the system mainly serves as playground for “innocent” users to test how comfortable they feel navigating among tables and to see what insights they gain from the inner structures of their database. The generator has been successfully used with both Microsoft's *SQL Server* and *MySQL* based on the storage engine *InnoDB*.

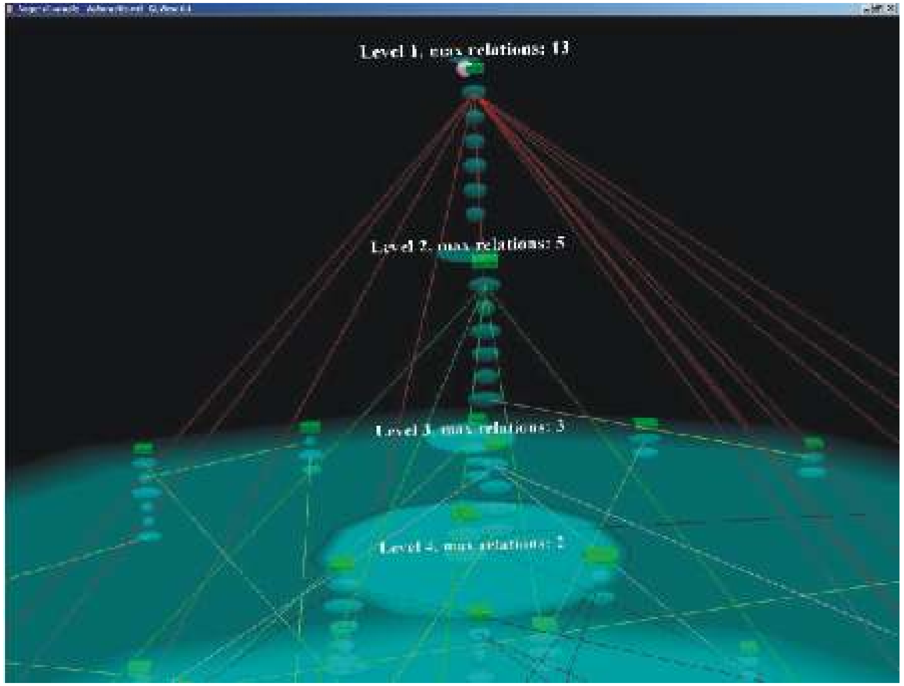

Fig. 4: Close-Up Of The Database Visualization

In the long run, however, mere navigation with no interaction will not satisfy. This, among other things, leads to the major requirements for future versions of the system:

- *Evaluation of layout and visualization:* By extensive experiments, the ideal arrangement for the majority of users has to be established. At the moment, there are no conclusive ideas, how users would like their 3-D interfaces to be. This will change, of course, as soon as 3-D visualization becomes the run-of-the-mill interface paradigm.

- *Seamless switch from data view to metadata view and vice versa:* The current system version strictly distinguishes between the data view and the metadata view of the database. First trials have shown, that ideally both views should be available at the same time. An implementation could follow the specification of the LOD (level of detail) feature in VRML to ease this restriction.

- *Interactivity within the VR environment with real-time reflection to database data and data definitions:* The future system versions should allow for interaction like the insertion, modification, or deletion of data or metadata objects. Such changes should be reflected immediately in the database.

- *Visualization of access plan data for query analysis and optimization:* A comprehensive view of the database can give crucial insights on how and how efficiently a query on a database is processed. An interactive version of the system could read access plan data provided by the database management system and visualize it within the framework of data and metadata on display. Such functionality, however, is beyond the scope of standardized database access tools like JDBC --- the DBMS has to be accessed via proprietary interfaces to retrieve access plan data.

These enhancements and additions imply a switch from the standardized VRML/JDBC approach towards an implementation based on proprietary VR programming systems like the *RealiMation* development suite.

4. Conclusion

Even in the current version, the generator has proven to be a tremendous support in the classroom teaching database classes due to its intuitive handling and the dynamic presentation of otherwise drab material. Students show considerable motivation to use the system themselves and seem eager to improve their navigational and interpretive skill.

Future generations will be at ease with three-dimensional virtual worlds as their workroom, just as the present generation is comfortable with the desktop metaphor. The results of the research presented in this paper are but a small step --- a step, however, in the right direction.

5. References

Carey, R. & Bell, G.
1977. *The Annotated VRML 2.0 Reference Manual*, Addison-Wesley Pub Co, Boston.

Chen, P.P.
1976. *The entity-relationship modeltoward a unified view of data*, ACM Transactions on Database Systems (TODS), 1 (1): 9-36.

Eckel, B.
2002. *Thinking In Java*, Prentice-Hall Pub, Englewood Cliffs.

Gibson, W.
1984. *Neuromancer*, Ace Books, New York.

Morcrette, C.
1999. VRML Generation tools for visualization of database content in three dimensions, Master's Thesis, Massachusetts Institute of Technology, Cambridge.

Nakazato, M.. & Huang, T.S.
2001. *3D MARS: Immersive Virtual Reality for Content-Based Image Retrieval*, Proceedings of 2001 IEEE International Conference on Multimedia and Expo (ICME2001), Tokyo.