

HYPERLINKED RELATIONAL DATABASE VISUALIZATION SYSTEM

BACKGROUND

The present invention relates to business intelligence tools for building applications on a database management system (DBMS).

The advent of powerful, yet economical computers made possible by advances in processor, memory and data storage devices has made computers an integral part of modern companies. An important class of application for these computers includes a DBMS where information is collected and organized according to a data model and searched using queries. The DBMS allows users to perform operations such as locating, adding, deleting and updating records stored in the computer without a detailed knowledge of how the information making up the records actually is stored in the computer.

One powerful type of DBMS is known as a relational DBMS where stored information appears to the user as a set of tables, each of which is termed a relation. In each relation, the information appears to be arranged in rows and columns, with columns of data being related to each other by one or more predetermined functions.

To access particular information in the relational DBMS, a query compiler converts a user request, typically expressed in a query language such as a Structured Query Language (SQL), into a set of operations to be performed on one or more input relations to yield a solution responsive to the user's request. Using the query language provided by the DBMS, the user may develop application programs which facilitate retrieval of the data from the DBMS, processing of the data, and organization of the data into reports.

One issue in developing business intelligence tools is the type of reports that the tool is to generate. Typically, the tool generates certain pre-formatted reports using the query language. Although the query language is easier to use than conventional programming languages such as Basic or C, the generation of each new report still requires a certain programming expertise and can often take a substantial amount of time.

SUMMARY

The invention provides user access to data through information spaces called scenes that allow the user to understand, view and navigate data. In one aspect, a hyperlink system is provided for viewing in context information associated with an application with a plurality of scenes from a viewpoint. The hyperlink system has a first scene having a first zoom factor from the viewpoint to the first scene; a second scene nested in the first scene, the second scene having a second zoom factor from the first scene to the second scene; and a wormhole projecting from the first scene to the second scene based on the first and second zoom factors.

Implementations of the invention include the following. The wormhole has an attribute property for setting the value of the one or more parameters associated with the second scene. The wormhole provides context information from each subsequent scene to each previous scene. Further, each scene may be displayed in one or more nested windows and may contain one or more objects representing the content of the database. The first elevation $elev_1$ may be determined in accordance with the first zoom factor $Zoom_{scene\ 1}$ as follows:

$$elev_1 = \frac{1}{Zoom_{scene1}}$$

The second elevation $elev_2$ may be determined in accordance with the second zoom factor $Zoom_{wormhole}$ as follows:

$$elev_2 = \frac{1}{Zoom_{scene1}} + \left[\frac{1}{Zoom_{wormhole}} - 1 \right].$$

Further, a query generator dynamically triggers queries to a database based on the first and second zoom factors. Additionally, one or more notification events may be generated as a user navigates the hyperlink system for triggering an object's behavior.

In a second aspect, a method for navigating a virtual world having a first scene and a second scene nested therein is disclosed. The virtual world has a wormhole projecting from the first scene to a second scene, each scene having one or more objects and each object having a behavior. The method involves interacting with the one or more objects; and generating notification events to trigger the behavior of the object depending on whether the object is viewed from the first scene or the second scene.

Implementations of the method include one or more of the following. The behavior is determined by setting the value of the scene or one or more global parameters. The object's behavior is determined by executing an action. Further, the first scene has a first zoom factor from a viewpoint to the first scene, the second scene having a second zoom factor from the first scene to the second scene, and wormhole projects from the first scene to the second scene based on the first and second zoom factors. The first elevation $elev_1$ is determined in accordance with the first zoom factor $Zoom_{scene\ 1}$ as follows:

$$elev_1 = \frac{1}{Zoom_{scene1}}$$

and the second elevation $elev_2$ is determined in accordance with the second zoom factor $Zoom_{wormhole}$ as follows:

$$elev_2 = \frac{1}{Zoom_{scene1}} + \left[\frac{1}{Zoom_{wormhole}} - 1 \right].$$

Additionally, the parameters include a global parameter, a scene parameter, or a query parameter. The method may dynamically trigger queries to a database based on the first and second zoom factors. The method also generates one or more notification events as a user manipulates the one or more objects.

Advantages of the invention include one or more of the following. The invention is a visual business intelligence tool for building applications that extend beyond the limitations inherent in conventional forms-based or report-based applications. Specialized programmers are removed from the application development process and users are moved closer to the data so that application development time is reduced. User interfaces can be created quickly and easily for information rich databases and for applications such as data warehousing and decision support.

The invention's hyperlinks provides context and "look-ahead" information to applications. This capability supports several powerful advantages in building data-driven applications. First, users can see through portals into other views