3D Information Visualisation:
Identifying and Measuring Success

Position Paper

Jessie Kennedy, K Mitchell, P Barclay, B Marshall
Object Systems Group

Napier University

2nd FADIVA Workshop
University of Glasgow, 20-22 July 1995

Introduction

The Object Systems Group at Napier University have been working on conceptual object modelling and interfaces to databases[10, 1, 2]. We are currently working on two projects related to interfaces to databases and data visualisation. The first is investigating the use of 3D interfaces to databases. We are developing our conceptual language NOODL [3] for use in the description of both the database schema and its interface[8]. To support this a generic framework for interfaces to databases has been proposed[9]. The example interfaces being developed to demonstrate the conceptual language and framework are based on desktop virtual reality, making use of both 2D and 3D environments in the interface. The second project is investigating mechanisms for the visualisation of object behaviour. This project is at an earlier stage and has been focusing on appropriate models for the description of co-operative object behaviour for use as the foundation for the visualisation.

This paper presents some of our views on information visualisation and interfaces to databases with respect to the theme of the workshop.

Successful Information Visualisation

What should count as a successful or unsuccessful information visualisation?

In our opinion the main criterion for defining whether information visualisation has been successful or not depends on whether users want to use the system. We believe that users will prefer to use an information visualisation system if it provides them with more information or answers to their questions than other existing systems can. For example, if the visualisation gives them insights into the data or shows patterns explaining the situation which otherwise is difficult if not impossible to visualise from raw data. For example, from the raw data collected by traffic sensors it is impossible to picture the effect of traffic flow under certain conditions, whereas an actual visualisation, even if not very sophisticated, can open up new understanding to the user [7]. Many scientific visualisations exist for the same reason. In addition the system providing the visualisation must provide the user with the ability to explore the data/information and as a result get ‘answers’ back rapidly.

If such visualisation systems provide the user with the ability to manipulate the data being visualised then it is imperative that the user be provided with immediate feedback.

What factors determine the success of 3D information visualisation?

Regarding the factors that determine the success of 3D information visualisation, we believe that these are much the same as for 2D systems and research done to date in 2D interfaces is applicable to 3D. However there may be two additional factors which need investigation. These are 3D navigation and the issue of immersion versus non-immersion. Users currently are familiar with non-immersive environments with 2D navigational controls which map closely to their visualisation. In 3D we can adopt either an immersive or non-immersive mode and from initial experimentation we believe that certain tasks are more suited to non-immersive mode than immersive mode. For example, designing a data model is easier in non-immersive mode as an overview is needed and the user will treat the data model as a real life model which they expect to be able to manipulate rather than explore in an immersive mode. Exploratory tasks are more suggestive of immersive mode interaction.

Can these factors be measured and if so, how?

The additional factors should be assessable through usability studies.

Generic vs. specific interfaces

To what extent is success related to details of the task? In other words, can we design generic tools to meet the needs of generic tasks or do the effects of the domain or particular features of specific tasks swamp these generic features?

We believe that databases must have support for both generic and specific interfaces. Certain users or tasks or domains or information to be visualised or combination of these may require either a general purpose/generic interface or a specialised/specific interface or some combination of the two. To this end we have developed a framework for interfaces to databases[9]. The primary organisational step for the creation of this framework was to determine the characteristic components of database interfaces (DBIs). These are detailed below.

- Database
    - Data Model
    - Schema
    - Objects
- User
    - Sophistication
    - Task
    - Authority
- Interface
    - Visualisation
        - Referent
        - Metaphor
        - Layout
    - Interaction
        - Intention
        - Medium
        - Effect
    - Style
    - Complexity

A general framework requires detailed specification for practical application. Recognising this, our framework relates both abstractly to the general features of a DBI and in depth to its atomic components. We have shown how a detailed classification of components under this framework may be mapped to a conceptual language, which embodies the relationships and dependencies among the components of a DBI.

An important property of this framework is that it is applicable to both general abstraction and concrete specification of the particular DBI in question. For example, the intention of an interaction component may be conceptualised at a general level as, *to provide data retrieval by means of interactively specifying a query,* or at a detailed level as, *to specify the intention of an atomic interaction component, such as pushing an 'ok' button to confirm the deletion of an object.* In this way, each of the four major components may be sub-divided to specify aspects of a DBI at successive levels of detail. A direct mapping to the conceptual language is achieved when the atomic sub-components of the database, user, interaction and visualisation components may be specified.

We believe that styles of visualisation can be generic or specific within given tasks and some may be generic across tasks while others not. A style of visualisation suitable for schema depiction may not be appropriate for instance visualisation due to the increased number of objects to be displayed on screen. Often the visualisation may be appropriate for differing tasks but the interaction mechanism may require to be different to permit ease of the task in hand, e.g. browsing data in 3D space might be suitable to flying through the 3D environment whereas specifying queries on the same visualisation would require a different interaction mechanism. Also a schema may be visualised in one way and the interaction mechanism may be determined by whether the user is manipulating the model or browsing it.

Certain visualisations are generic with respect to the application domain and the type of information [4], others are generic to the application domain and specific to the type of information being visualised, [10, 5], whereas others are specific to the application domain and type of information [8].

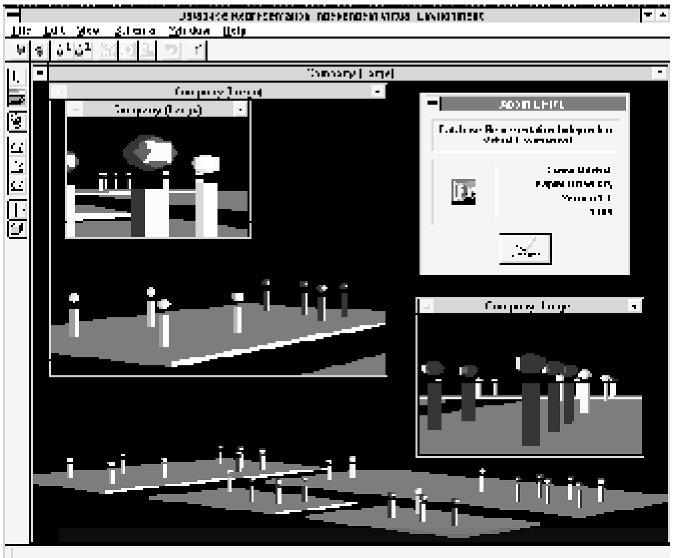

Fig. 1 Domain specific, information specific 3D visualisation

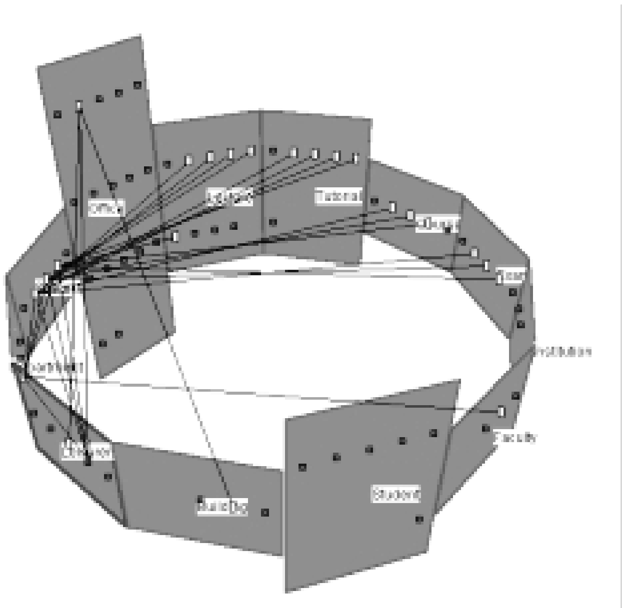

Fig. 2 Information specific, domain generic 3D visualisation

Our approach to developing interfaces by the provision of a generic framework for interfaces to databases provides a mechanism by which any type of interface for any type of task/user/information can be specified in a uniform manner by a conceptual interface/database description language.

Database Interface Tasks

Are there important differences between information retrieval (in the IR sense) tasks, query tasks using conventional database queries, other database operations such as data modelling and data entry and other information-related tasks, such as browsing?

We believe that there are differences in the tasks of browsing (which is akin to information retrieval), querying and data modelling. When a user is browsing a database, library or whatever, they have a fuzzy idea of what they are looking for and would therefore find it difficult to specify an exact query, therefore mechanisms to aid the exploratory process are required by the user. When querying, the user will more often have a clearer idea of what they want to know and will require aids to help them ask their question of the system. In data modelling the user is carrying out a design task which again requires different tools. However it is possible that the visualisation for say, the data model, may be informative enough to aid a user in the browsing or querying process. In [9], we refer to the different tasks as the intention of the user.

2D vs. 3D visualisation

Are there basic differences between 2D and 3D which make 3D better or worse for particular domains or tasks?

For certain domains and visualisations the provision of 3D space allows the visualisation to be closer to reality, in particular applications with inherent 3D spatial properties, e.g. archaeological data [6] or geological data. For any domain, the provision of a third dimension allows for additional information to be represented on screen. The argument regarding the obscuring of data is as valid to 2D as to 3D, however in 3D the data may be made visible by changing the viewpoint of the user. 3D also extends the features which may be used in the generation of information providing visualisations, e.g. the relative position of objects in 3D space or the depth of an object.

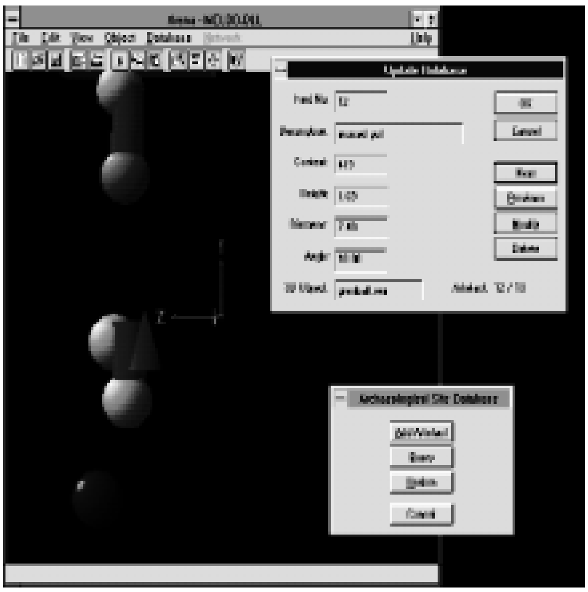

Fig. 3 Spatial information visualisation

Conclusions

Identify aspects of 3D visualisation which require investigation with the greatest urgency, either because they promise significant payoffs or because they address issues which are important for current visualisation designers

We believe that due to the predominance of PCs in the community it is imperative that the potential of desktop VR be investigated fully as this market will be the largest. Desktop VR also provides a natural stepping stone for users from Windows environments to using 3D in the integration of 2D and 3D environments. Associated with this we are investigating the use of multiple co-ordinated views with immediate feedback by providing 2D viualisation as e.g. maps to aid the user in the navigating and manipulating the 3D visualisation.

We are keen to discover useful techniques for use in identifying, assessing or measuring the success of our visualisations.

References

1. P.J. Barclay & J.B. Kennedy (1992). Using a Persistent System to Construct a Customised Interface to an Ecological Database, *1st International Workshop on Interfaces to Database Systems,* Springer Verlag.

2. P.J. Barclay (1993). *Object oriented modelling of complex data with automatic generation of a persistent representation*. PhD Thesis. Edinburgh: Napier University.

3.. P.J. Barclay & J.B. Kennedy (1994). A conceptual language for querying object-oriented data, *Proceedings of BNCOD*, 187-204.

4. S. Benford & J. Mariani (1994). Populated Information Terrains, *2nd International Workshop on Interfaces to Databases*,159-169, Springer Verlag.

5. J. Boyle, J.E. Fothergill & P.M.D. Gray (1994). Amaze: a three dimensional graphical user interface for an object oriented database, *2nd International Workshop on Interfaces to Databases,* 117-131, Springer Verlag.

6. P. Johnson, (1995). *Arena*, Technical Report, Dept Computer Studies, Napier University.

7. J. Kerridge, K H Lu, J Jones, (1994). Modelling Congested Road Traffic Networks Using a Highly Parallel System, in *Transputer Applications and Systems,* IOS Press Amsterdam, Sept 1994, pp 634-647.

8. K.J. Mitchell, J.B. Kennedy, & P.J. Barclay (1995). Using a Conceptual Language to Describe a Database and its Interface, in *Proeedings of the 13th British National Conference on Databases,* Springer Verlag.

9. K.J. Mitchell, J.B. Kennedy, & P.J. Barclay (1995). *A Framework for Interfaces to Databases*, Technical Report, Napier University (submitted to DOOD'95).

10. M.H. Rapley (1994). Three Dimensional Interface for an Object Oriented Database, *2nd International Workshop on Interfaces to Databases*, 133-158, Springer Verlag.