The Prometheus Database for Taxonomy

Cédric Raguenaud, Jessie Kennedy, Peter J. Barclay
Database and Object Systems Group, Napier University
219 Colinton Road
Edinburgh EH14 1DJ, UK
{cedric, jessie, pete}@dcs.napier.ac.uk

Abstract

This paper presents the work carried out in the Prometheus project and its motivation, taxonomy. Taxonomy presents challenges to common database systems. Because of its complexity and the necessary treatments applied to its data, common database models such as the relational, the object-oriented, of even graph models are not able to support taxonomic applications fully. Our approach is the extension of a object-oriented database model with explicit relationships in order to support new features and thereby offer the necessary level of service for developing taxonomic applications.

1. Introduction

The aim of the Prometheus project [5] is to design a database system that supports the working practice of plant taxonomists. An analysis of taxonomy resulted in the definition of a new model of plant taxonomy [6], and it was shown that existing taxonomic database systems do not accurately capture the semantics of taxonomic data nor the working practices involved in creating classifications [7]. This work allowed us to clearly express the requirements of a database system to support this process.

Taxonomy is the study of the general principles of scientific classification. Taxonomists arrange organisms into classification hierarchies according to various criteria (e.g. morphological similarities or, more recently, DNA relationships), which thereby depict their presumed natural relationships. Taxonomic classifications are used to name, refer to and aid the identification and understanding of organisms. As knowledge increases or opinions on the importance of certain criteria change, new classifications are generated which often lead to specimens known under one name now being known by a different name. Being able to refer to something under study uniquely and unambiguously is essential in any area of research or work. For example, if research is being undertaken into the chemical composition of a plant for pharmacological research or DNA sequencing for identifying genes with a particular characteristic, it is vital that the results from this research are related accurately to an organism that will refer to the same one in the future. Also, say in legal documents, it is important that a name of an organism covers all instances of that organism (even if known by a different name). In other words the name of the organism must be unique and reliable. It is important for researchers studying these organisms to be aware of the fact that classifications mean that organisms can have several names and that a name can apply to several organisms. The only way to use a name safely is to use that name in the context of the classification from which it was generated, and have systems that can relate names across different classifications. Taxonomy is a never-ending process, as new classifications will continue to be generated. Therefore it is not sensible to think of there ever being a definitive list of names for all species on Earth, although at any point in time a particular classification may be chosen to provide a *preferred* list for pragmatic purposes, but allowing cross referencing to other classifications.

This paper presents the motivation for the Prometheus project in section 2, then presents the approach chosen to solve the problem in section 3. We conclude in section 4.

2. Taxonomic data

Figure 1 depicts a simplification of the kind of scenario found in taxonomy. The information available grows over time, the criteria used for classification vary and the number of levels (ranks) used in the classification process varies. The grey shapes at the leaf nodes represent individual specimens to be classified.

The top left figure is the earliest classification and is based on a smallish set of specimens. The criterion used for this classification was the shape of the specimens which resulted in a two-level hierarchy. Square specimens are *typified* by the mid-grey square, triangular specimens by the dark equilateral triangle and circular specimens by the light-grey circle (i.e. these specimens are chosen by

the taxonomist as representatives of each taxon). Shapes in general are typified by squares and hence are represented by the mid-grey square. Subsequently, (top right) a second taxonomist decides that an intermediate level in the classification would make things clearer and introduces the general type square, triangle and circle and 2 sub-types of triangle, equilateral and right angle and two sub-type of round shape, circles and ovals. Owing to the naming conventions (defined in the Botanical Nomenclatural Code), squares are still typified by the same mid-grey square, triangles by the dark equilateral triangle, and circular shapes by the light-grey circle. However new types are required for right-angled triangles (the black one) and ovals. A third taxonomist (bottom left) finds some new specimens and decides that shape is not an important characteristic after all and reclassifies the larger specimen set according to their brightness. This creates a two level classification with five groups (he ignores one particular shade as there is only one instance of it). Co-incidentally each group contains an existing type specimen and therefore no new types need to be defined for the classification. In practice often several types will end up in one group, which then requires the oldest type specimen to be chosen as the type. Finally a fourth taxonomist (bottom right) comes along, and reclassifies the specimens by shape again.

The reality in taxonomy is much more complicated and involves many more specimens. However, the general principle and reason for the existence of multiple classifications should be clear. A taxonomy once created is never 'deleted'. Classifications reflect opinions and although opinions change they never replace previous classifications although a single classification may be 'preferred'.

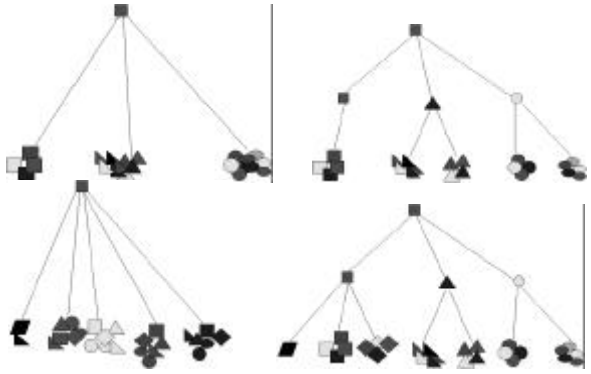

Figure 1 Four classifications with overlapping specimens and concepts

Figure 2 shows a visualisation of 4 classifications of the family Apiacaea. Specifically, the spread of the genera described by Berchtold and Presl in 1820 are depicted in the three subsequent classifications [2]. The variation in opinion of the classification of Apiacea and the naming of the associated taxa is apparent. This figure also shows the variation in the number of ranks used by different taxonomists. Each of the hierarchies start at the same taxonomic rank, family, and end with squares representing the taxonomic rank, genus. However, they vary in the number of intermediate ranks used to describe the classification, from 1 to 3. The shaded squares representing genera show the overlap between the 4 taxonomies. This visualisation is still simple in terms of the amount of data shown due to the number of levels represented and the lack of differentiation between classification and naming.

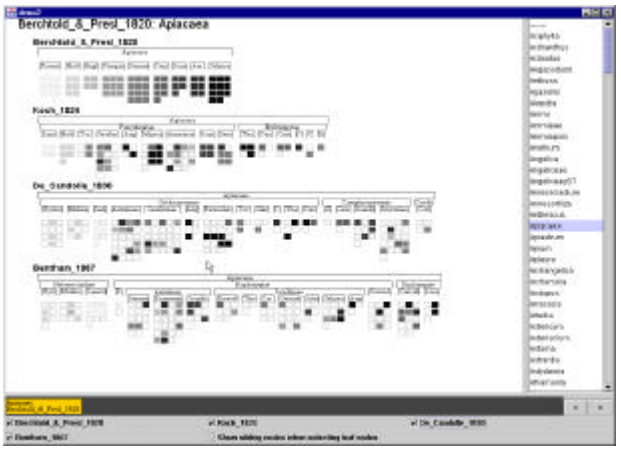

Figure 2: Four classifications of the family Apiacaea

It can be seen from this description that:

1. The data structures manipulated are highly hierarchical and many hierarchies are interconnected to form a complex network.
2. The names given to taxa are calculated using the Botanical Nomenclatural Code.
3. The place of a taxon in a hierarchy is significant.
4. The query language needs to manipulate hierarchies or more generally graphs.

From further analysis of the data, the requirements for a database system for managing taxonomic data are the following:

1. The ability to define hierarchies and graphs, which is possible in systems representing relationships as well as nodes.
2. The ability to manage complex objects, therefore differentiating between the kinds of relationships an object can be involved in (e.g. associations, aggregations).
3. The ability to define integrity constraints or rules, to support the Botanical Nomenclatural Code.
4. The ability to query the database recursively.

These properties make most existing database models unsuitable. For example, the relational model does not allow the explicit representation of the hierarchic aspect of the data and associated recursive/graph traversing

querying. Object-oriented models often lack the ability to traverse and explore graphs (although some models such as [3] offer regular path expressions that can simulate graph traversing, but are limited to recursive statements not involving joins). Neither model represents relationships explicitly nor their querying as first class objects. From our analysis of taxonomy and review of existing systems we identified and tested two approaches that could provide the necessary functionality: the design of an extended graph database model and the extension of an existing object-oriented database with relationships and manipulate graph structures.

3. Approach

The chosen approach incorporates the functionality achieved in graph databases into an existing object-oriented database system. This integration provides a means of benefiting from well-established object-oriented features (e.g. abstraction, classes, inheritance, and reuse) and at the same time supports graph manipulation.

In order to support graphs, we have defined an extended object-oriented model (POOM), which in order to support graphs, emphasises relationships and describes them as first class concepts. A generic simple object-oriented model has been described and we have shown how relationships could be introduced smoothly [9]. These relationships allow the representation of the various taxonomic hierarchies as an integrated graph where information is shared (multiple overlapping graphs), whilst keeping enough information in their attributes to distinguish them, e.g. relationships store publication information that is used to differentiate two classifications [8].

The development of the relationships and the recursive aspect of the treatments have required the extension of OQL (POOL) with features such as the implicit traversal of relationships, the extraction of graphs, the ability to navigate in one specific subgraph, type downcasting, implicit iteration over collections, the extraction of composite objects, recursive querying, and recursive joins [9].

A constraint language (PCL) has also been developed to support the constraints necessary to implement the Botanical Code.

4. Conclusions

Taxonomy, by its complexity and its history, poses a challenge to common database models. Its hierarchical aspect and its recursive operations require a database that is able to represent complex graphs, keeping enough information to distinguish logical graphs and query on these graphs.

We defined a new database model that emphasises relationships in order to allow the definition of these complex graphs, and a query language adapted to the new structures. Not only does this model offer tools for taxonomy, but as a generic extended object-oriented database system, it supports more expressive definitions and queries than traditional systems such as ODMG [1], i.e. it is closer to modelling and supports concepts such as aggregation, composition, association with their attributes, e.g. changeability, traversability, lifetime dependency. The model is an abstract model that can be implemented in numerous existing object-oriented database systems.

The query language developed extends OQL to support and manipulate the graphs defined with our relationships. Prometheus goes beyond systems such as OMS [4] in its ability to model and query complex graphs and object-oriented systems.

It has been fully implemented using Java and the OODB POET, and has been used to implement a taxonomic database. The query language has been shown to support the manipulation of classifications as required by the taxonomists.

5. References

[1] R. G. G. Cattell, D. Barry, D. Bartels, M. Berler, J. Eastman, S. Gamerman, D. Jordan, A. Springer, H. Strickland, D. Wade, "The Object Database Standard: ODMG 2.0", Morgan Kaufmann Publishers, Inc. (1997)

[2] M. Graham, J. B. Kennedy, C. Hand, “Visualising Multiple Overlapping Classification Hierarchies”, to appear in AVI 2000, Palermo, Italy (2000)

[3] M. Kifer, W. Kim, Y. Sagiv, "Querying Object-Oriented Databases", Proceedings of the 1992 ACM SIGMOD International Conference on Management of Data, San Diego, California, pp 393-402 (1992)

[4] M. C. Norrie, “An Extended Entity-Relationship Approach to Data Management in Object-Oriented Systems”, Entity-Relationship Approach - ER'93, 12th International Conference on the Entity-Relationship Approach, Arlington, Ramez Elmasri, Vram Kouramajian, Bernhard Thalheim Eds., Texas, USA, pp 390-401 (1993)

[5] Prometheus: http://www.dcs.napier.ac.uk/~prometheus

[6] M. R. Pullan, M. F. Watson, J. B. Kennedy, C. Raguenaud, R. Hyam, "The Prometheus Taxonomic Model: a practical approach to representing multiple taxonomies", Taxon Vol. 49 Issue 1, pp 55-75, February 2000 (2000)

[7] C. Raguenaud, J. Kennedy, P. J. Barclay, “Database support for taxonomy”, Prometheus technical report #1, School of Computing, Napier University (1999)

[8] C. Raguenaud, J. Kennedy, P. J. Barclay, “The Prometheus Taxonomic Database”, submitted to IEEE BIBE 2000 (2000)

[9] C. Raguenaud, J. Kennedy, P. J. Barclay, “The Prometheus Object-Oriented Database System”, submitted to VLDB 2000 (2000)