Biological Journal of the Linnean Society (2001), *73:* 81–98. With 5 figures
doi:10.1006/bijl.2001.0527, available online at http://www.idealibrary.com on IDEAL®

Nomencurator: a nomenclatural history model to handle multiple taxonomic views

NOZOMI YTOW[1], DAVID R. MORSE[2] and DAVID McL. ROBERTS[3]*

[1]*Institute of Biological Sciences, University of Tsukuba, Tsukuba 305-8572, Japan*
[2]*Computing Department, Faculty of Mathematics and Computing, The Open University, Walton Hall, Milton Keynes MK7 6AA*
[3]*Department of Zoology, The Natural History Museum, Cromwell Road, London SW7 5BD*

Received 4 July 2000; accepted for publication 14 February 2001

Evolutionary studies are generating increasing numbers of phylogenies which, in turn, sometimes result in changes to hierarchical organization and therefore changes in taxonomic nomenclature. A three-layered data model for a nomenclature database has been developed in order to elucidate the information structure in nomenclature and as a means to organize and manage a large, dynamic knowledge-base. In contrast to most other taxonomic databases, the model is publication-oriented rather than taxon-oriented and dynamic rather than static, in order to mimic the processes that taxonomists use naturally. The three-layered structure requires data integrity localized to each publication, instead of global data integrity, which relaxes constraints common to taxonomic databases and permits multiple taxonomic opinions: taxon names are made available as metadata within the model. Its prototype implementation, written in C^{++}, has an autonomous self-identification mechanism to avoid spurious data-inflation in a publication-oriented data model. Self-identification is also desirable for distributed implementations of the nomenclature database. Publication-oriented design also will make maintenance easier than for taxon-oriented databases, much of the maintenance workload being amenable to automation. The three-layered data model was designed for use by taxonomists, but is also able to provide concise, reduced expression for non-experts required in biodiversity research, for example.

ADDITIONAL KEYWORDS: dynamic knowledge base; species names; temporal database; name scope

INTRODUCTION

New technologies, especially DNA sequencing and analysis, are creating a new perspective on relationships between taxa and an unparalleled burst of knowledge on the nature of the biological world. In particular we are seeing a rapid increase in phylogenetic studies and a blossoming of understanding of evolutionary relationships. In practice, establishing relationships on the basis of genomic data has weakened the emphasis on phenotypic or morphological descriptions and thus on our ability to circumscribe these new taxa (Young, 2000). New methods often produce results which are not congruent with existing schema and the resolution of these conflicts takes time, but the immediate impact is seen in the nomenclature. These data do not reduce but increase the importance of classical taxonomy and nomenclature which provide the framework for communication of biological concepts. Such is the rate of technical advancement and data accumulation that some sequenced data are even being accumulated without a sufficient description of their source organism. The name of the species from which the sequence is derived is a sufficient specifier if and only if the name is unique to the species. This creates a logical contradiction: new knowledge advances our understanding of relationships and this is reflected in changes to the taxonomic organization, which often means that the names change. Thus we should not expect that there will be an unique, unchanging map of species names to species themselves (Alberdi & Sleeman, 1997; Härlin, 1998). The creation of computer databases containing taxon names as searchable fields is exposing the shortcomings of the system and revealing that the name alone is often not a unique access point to a species (Berendsohn, 1999; Berendsohn *et al.*, 1999). The importance of DNA

* Corresponding author. E-mail: dmr@nhm.ac.uk

0024–4066/01/050081+18 $35.00/0

sequencing to phylogenetics, however, is precisely because it is novel knowledge which exposes new understanding of the relationships between taxa. This understanding may result in changes to taxon concepts, of course, which require changes in taxon names to reflect these newly-recognized relationships: this is an intrinsic property of the nomenclatural system and not a shortcoming, as it is sometimes portrayed.

There are efforts to build name lists of species, such as Species 2000 (Brugman, 1999), some of which are also driven by the necessity to build a basic data set for biodiversity research and its applications (Anonymous, 1999; Rees & Sadka, 1999). Such name lists can help to ensure the uniqueness of each name and the process of building the lists often reveals ambiguity in the underlying taxon concepts, which act as a focus for further research. But when the names and their relationship to one another are subject to frequent, often radical, reorganization or dispute (in other words, unstable; Alberdi & Sleeman, 1997) the construction and maintenance of such lists rapidly outstrips the resources available to them. In practice this means that lists can be maintained for well studied, established groups where new data confirm our basic understanding of the evolutionary relationships. New data and new analyses may, however, provide insight that changes the concept of a taxon: consequently there are tight links between descriptive data, the concept of the taxa, the taxon names, and the physical specimen from which the data were gathered. The scale, that is the highest taxonomic rank affected, of such a process is likely to be less in more stable groups including mammals and higher plants, but it is likely to be greater in groups that are not yet well studied such as the protozoa. Stable lists of species names and a stable hierarchical organization are unlikely to be tractable for the latter groups. Some botanical taxonomic databases (Beach, Pramanik & Beaman, 1993; Berendsohn, 1997; Jung *et al.*, 1995; Pullan *et al.*, 2000; Raguenaud, Kennedy & Barclay, 1999a,b; Zhong *et al.*, 1996) do support multiple taxonomic opinions, but whether this is sufficient for these latter groups, particularly the zoological taxa, is yet to be established. These groups will surely require the most flexible databases capable of capturing the dynamic interaction between each taxon concept and the data linked to it.

This contribution will describe the results of a feasibility study on such a database. The primary objective is a carefully-built data model which can be implemented in an appropriate database management system. The design is intended to mimic the process of tracing nomenclatural history in a library, giving regard to the nature of the available data and the way in which they become available.

CONCEPTUAL DATA MODEL FOR THE DATABASE

The term 'taxonomic database' can take on a variety of meanings depending on the purpose for which it was designed. Some databases are intended to store descriptive data for taxonomic purposes, most commonly either identification or classification analysis. These are a special case for present purposes, because they link specimen information directly to the names and they contain information through which taxon concepts may be directly assessed; an example of this type is PANDORA (Pankhurst, 1993). The more common situation is databases which simply, often simplistically, use taxonomic nomenclature and which can best illustrated by two extreme examples.

First, there are floral or faunal lists, which can simply be a list of the taxa known to occur in a geographical region (e.g. http://fff.nhm.ac.uk/cheklist.htm). At the extreme, because stability of name-usage is a high priority for their purpose, such lists may not comply with the latest taxonomic or nomenclatural practice and may be using names and classifications which are regarded as outmoded or inadequate in other contexts: none the less, they fulfil the purpose for which they were built (cf. the US Federal Geographic Data Committee's interest in establishing a Biological Nomenclature and Taxonomy Data Standard, see http://www.fgdc.gov/standards/status/sub5__8.html).

Second, there are formal taxonomic reviews which maintain a consensus list of valid names, that is names currently accepted by the international taxonomic community. Typically these databases provide a trace for synonyms and other names not currently considered valid members of the group concerned. The International Legume Database & Information Service (ILDIS; http://www.ildis.org/) is an example of this type of list, where members of the legume family (Leguminosae; peas, beans etc.) are divided between panels of experts and each panel periodically reviews the composition of their group.

The distinction between them is the data model they employ and the consequential maintenance of those data. For the most part, data are gathered for purposes other than systematics and the resource to expand the data-collection component of the task is not available, as discussed in detail below. For the purposes of what follows, the important point is the nature of the available data, other differences being unimportant and the term 'taxonomic database' will be used in its most general sense.

Taxonomic databases often use the concept of a taxon as the principal data container. This is a naïve conjugation between a fundamental database constraint, the demand that each primary data container be unique, and an implicit assumption that a taxon must be

uniquely defined. This scheme implicitly requires a clear definition of each taxon in order to store any piece of data, including its name, within a taxonomic database. Although the definition of a taxon may not be formally stated (Härlin, 1998), it must be possible to decide whether two or more taxa are the same, whether data entries are duplicated and where these data should be stored; otherwise data cannot be reliably, or usefully, stored in the database. Taxonomic database schemes do not normally allow multiple taxonomic opinions because that would violate the uniqueness assumption. The potential taxon concept (Berendsohn, 1995) can relax this restriction by combining the name with the citation in which it appears. When the publication is not the authority for the name, then a decision must be made as to whether the name refers to the same taxon as the authority, which can be referred to by a status flag for that record. If these decisions cannot be made, then the number of potential taxa will inflate (Berendsohn, 1997; Berendsohn *et al.*, 1996). Clear, exclusive definitions are rarely available for taxonomic groups under intense investigation, where interactions between the taxon concept itself and the data that are to be attributed to the taxon are active and dynamic. It is obvious that such taxonomic groups would greatly benefit from taxonomic databases in order to store such information appropriately, otherwise the recovery of data when it is needed will be hard: indeed this is the situation now where the necessary data are randomly scattered throughout the world's scientific literature. This is a chicken-and-egg problem. These taxonomic groups require more flexible data structures which are not based on a set of taxon concepts that are mutually exclusive, without any overlap or conflict, as the principal data container. This means that we can use neither an exclusive taxon nor the valid name as the primary component of the database.

A taxon concept proposed in a publication can be expected to be consistent, unique and exclusive at least within that publication (Fig. 1). The reader in the figure may use the name to access specimens, for example indicated by the 'access by name' arrow. The name can also be used inversely, for example, the reader may have started from a comparison between a set of given specimens and specimens in a collection. The name is being accessed through the specimen in this case, indicated by the 'access by instance' arrow. The reader forms a taxon concept which is an association of the name, the description in the publication and the reader's own contextual knowledge. Even though the author and the reader cannot share taxon concepts directly, the author's taxon concept plays a key role in linking the name and instances of the taxon, e.g. (type) specimen (cf. the Circumscribed Taxon, Pullan *et al.*, 2000). The taxon concept is a crucial link between the name and resources

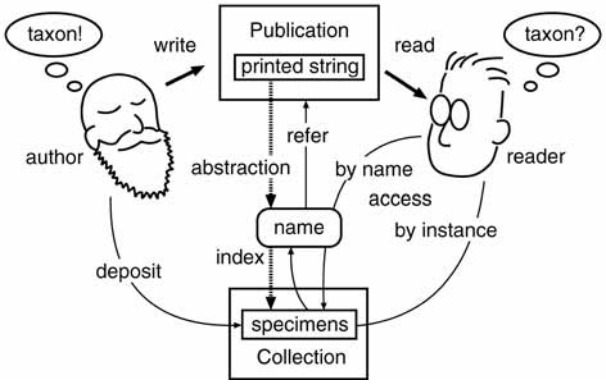

Figure 1. The concepts relating to a taxonomic name. An author proposes a taxon according to a study of its instances including specimens (if the taxon is a species or lower) or lower taxa (for taxon higher than species). The author gives a name to the taxon and the name appears printed in the publication. A reader of the publication, who may be reading it long after the death of the author, finds the printed name. The description in the publication is an abstract form of the original taxon concept (left oval) that the author had in mind when the manuscript was written. The description is not a taxon concept itself. There is no direct link between the taxon concepts of the author and the reader (right oval), hence there is no way to verify that these two concepts are identical.

described in the publication. The name, through its hierarchical position and its relationship to other names, is an abstraction of the original taxon concept which is, in fact, an access key to other resources. It is important to appreciate the information content embedded in the names themselves: the existence of a classification system enables the names to carry information and makes them more than mere labels.

It is important to note that the taxon concept itself cannot be completely defined, because it depends on the examination of a finite number of specimens and discrimination from closely related and potentially unknown related specimens. For this reason, taxon concepts preserved as circumscriptions need to be updated as new specimens are studied. It is, in essence, a rather slippery concept defying precise definition in a manner analogous to finding a usable universal species definition. In some fields, where preservation of specimens is both practical and practiced, taxon concepts can be described by the voucher specimens that they encompass (Pullan *et al.*, 2000). Practising taxonomists lacking this advantage still use taxon concepts on a day-to-day basis without difficulty, but the lack of a rigorous definition or universally accepted understanding does not make them a natural choice for a primary data container.

Another difficulty exists in the use of circumscriptions, which are descriptive lists of characters. Where the target taxa are highly variable or the scope of the database is broad, then these lists share very little, if any, commonality, which seriously compromises their utility. Compilation of such lists is also a laborious task and within the context of a broad nomenclatural database it is unclear what benefit would accrue from an effort to capture the taxon concept in this way. This, again, suggests that the taxon concept is not itself a strong candidate for the primary data container.

The taxon concept involves the delineation of the taxon and its relationship with other taxa. As knowledge advances and understanding develops, the taxon concept will also develop, so the term 'taxonomic opinion' will be used to describe the taxon concept as it existed for an author at the time of publication. A taxonomic opinion can be identified without ambiguity by specifying a pair of tangible objects; the name as printed and the publication in which it appeared. This pair can be used as the principal container in a more flexible database although there is a risk that it may result in an inflation of data, as suggested for the potential taxon concept by Berendsohn (1995), because two or more taxonomic opinions may refer to the same taxonomic object. Each of these name-publication pairs will result in a different entry in the database. Berendsohn (1995), focusing on the taxon specified in the name-publication pair, suggested that there should be 'taxonomic control' by taxonomists as a guard against this inflation, although this implies some level of restriction on taxonomic opinion, which would be inappropriate for flexible data models not based on the taxon concept as the primary data container. By focusing on the taxonomic opinion, i.e. the publication, rather than the taxon, inflation is a failure to recognize the same name-publication pair when derived from multiple sources. An autonomous method needs to be provided to avoid such inflation, which will enable us to trace the development of taxon concepts. The combination of a name and a publication as a key (the principal data container or primary key in database terminology) provides an autonomous method for data identification which is important for distributed taxonomic databases. Taxonomic databases are going to be of greatest value when they can be shared and distributed.

Inflation of database records is a problem if and only if the added records contain no new information. The model proposed here seeks to store information derived from publications with the nomenclatural operation given by the author, similar to the 'taxon view' approach described by Zhong (1996). In this way, each occurrence of a name in a publication which also makes some nomenclatural statement is a valid record. Without the nomenclatural statement, a record may provide no more than the existence of the name, but if it is the only record of that name, it is still of value. Thus criteria exist for autonomous management of the database size and control of inflation. It is of interest to note that for disciplines without a strong specimen-based tradition, such as the soft-bodied Protista, the publication of a taxonomic opinion, especially with an accompanying figure, stands proxy for actual specimens (cf. Pullan *et al.*, 2000).

HOW A NOMENCLATURE DATABASE CAPTURES TAXA: AN EXPLANATION USING THREE LAYERED THREADS

A taxon is defined in a descriptive publication as an abstraction of one or more individuals for species or lower classification levels, or taxa of lower rank for classification levels higher than species: collectively, these are referred to as instances. Note that an instance is a conceptual object, not a physical specimen. The taxon name is defined by the publication, expecting there to be an unique relationship between the name and the taxon: this expectation must be met within the publication for it to be intelligible. Meaningful circumscription of the taxon relies on a knowledge of closely related taxa which may not be available at the time of original definition. Further, in under-worked, poorly described groups this comparative basis is often lacking but it is in precisely these circumstances that a nomenclature database would be of greatest benefit to taxonomic research.

A name in a publication is a proxy for a taxon concept, expressing the taxon's relationship to other taxa within a classification system. It does not have material existence and is expressed as a set of printed Roman characters (a literal string). Descriptive publications contain a reflection of the taxon concept, but they are at best merely a summary of it as perceived by the author at the time of publication and in the context of contemporary knowledge. Similarly, each instance of the taxon is not the entire taxon, and a name is an abstraction of the taxon but not the taxon itself. This distinction is important because nomenclatural databases seek to catalogue names through an understanding of the taxa they represent. The name of a taxon serves as a link between each instance and the taxon concept. A taxon concept and a name cannot be regarded as a single object because, for instance, two or more different names can be linked to a single taxon concept; this is known as synonymy. This is technically expressed as the lack of a one-to-one mapping between names and taxon-concepts. It is exactly the failure of this mapping that needs to be addressed by nomenclatural databases.

Each descriptive publication links a name, a taxon and some material entities at the time of publication. The name and the taxon concept become relevant to nomenclature instantaneously at the time of first publication but they also persist thereafter and, significantly, the taxon concept will change with increasing

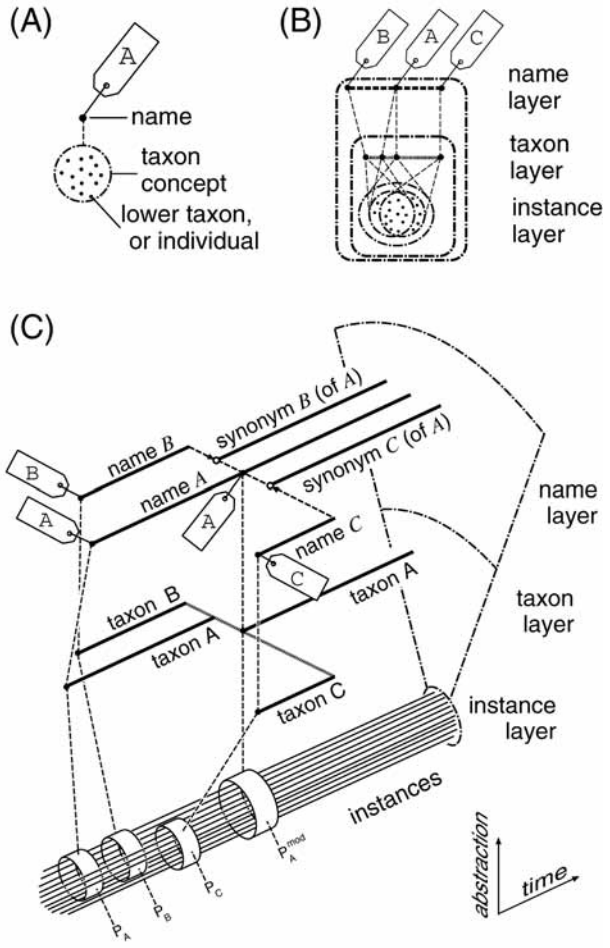

Figure 2. Three layer model. (A) The model deconvolves taxonomic naming into three encapsulating components: the instances (specimens or lower taxa) are encapsulated within a taxon concept which unites them and which is, in turn, encapsulated by the name itself, which provides a tag (*A*). (B) Different taxonomic opinions can ascribe different taxonomic concepts to a set of instances. Here taxon concepts B and C are intersecting while A unites all of B and C. There is no indication here whether A is the oldest concept which has been subdivided into B and C or whether A is a recent concept revising the older concepts B and C. (C) The three layers are drawn out into a temporal span to indicate the relationship between the names *A*, *B* and *C*. The information link from a specific publication (indicated as P_A, P_B, P_C and P_A^{mod}) is shown by a broken line. Existence is shown by solid horizontal lines. Instances are shown by a bundle of horizontal lines. It is important to note that there are no links between the taxon, name and instance except through publications. Taxon A was described, while taxon B was a later description made in ignorance of taxon A, and thus erroneous (a clerical error). Taxon C was described later still with new instances, with a proposal to remove some of the instances from A, so re-defining the taxon concept of A. This represents multiple taxonomic opinion. A later reviser takes the view that the division of A and C is unjustified and declares C to be a junior synonym of A, but with an emended definition of the taxon concept A. After this only a single taxon thread (A) continues, whereas in the name layer, the names *B* and *C* continue to exist as junior synonyms of *A*. Note that the figure shows only one opinion. Other opinions, with different arrangement of links, are also possible.

knowledge. It is one of the rules of the taxonomic codes of nomenclature (Sneath *et al.*, 1992; Murphy *et al.*, 1995; Ride *et al.*, 1999; Greuter *et al.*, 2000) that, with certain exceptions, once a name is used it should not be used later to refer to a different taxon within the scope of the rules. Contravention of this rule creates a homonym which will require nomenclatural correction when it is discovered. The exceptions, for instance the use of the same name to refer to an animal and a plant, are particularly troublesome and will be dealt with later.

The link between the name, the taxon and its instances may be amended for several reasons, for example the recognition of synonymy, a revised description and so on. These later proposals modifying the taxon concept cannot affect previous publications, of course, because they exist as material entities, but they are intended to affect the taxonomic concept. The histories of these amendments are records of the change in understanding of Nature as well as correcting clerical inconsistencies essential for the effective operation of the system as a whole. These histories contain information on the development of related taxon concepts and can be captured as directed links between names from the later publication to the earlier publications. Combining citation references with the names removes any ambiguity which might be seen if a database is searched by name alone.

In the real world, a publication has two types of link: internal and external. Internal links join a name, a taxon concept and its instances cited within the publication (lines in a vertical plane in Fig. 2C). External links join two or more publications through names and citations (lines in an horizontal plane in Fig. 2C), so names work as interface nodes to external links, since publications dealing with the same name usually appear at different times conceptually (even though it is rare to know the exact date of appearance and hence the date is commonly approximated to the nearest year or so). Publication-to-publication links via name infer the external relationships between taxon concepts, because it is assumed that an author is expressing an opinion relating the name to the taxon concept within the publication, and an opinion of the taxon concept in another publication. Although the objective is to catalogue taxa, i.e. the collected set of taxon concepts, there can be no direct link between taxon-concepts in one publication and another: links can only be established through a

name. External links, both explicit and implicit, make up threads that join objects, i.e. the name, taxon and instance threads. These threads remain bundled and do not cross threads of a different type: hence we recognize three layers: the name, taxon and instance layers, as shown in Figure 2.

A taxon concept is a discriminator that is used to determine whether an object is an instance of the taxon or not. It is referred to by its name. It is not a definition of the taxon, in the sense of a circumscription, but rather is defined by comparison with a set of exemplars. Multiple opinions and clerical errors complicate the information relationships. For example, although the name A is used twice with different taxon concepts, *sensu stricto* and *sensu lato*, the name-line A appears continuous in the figure. Note that each layer is encapsulated by a layer of higher abstraction; a taxon is an abstraction of its instances, and the name is an abstraction of the taxon. For taxonomic ranks higher than species, instances are lower taxa, and hence the encapsulated layers are nested; magnification of an instance will show the same structure for lower taxa, ultimately arriving at the resolution of individuals, i.e. physical specimens (see Fig. 3).

Navigation in the name layer enables us to track the history of a taxon concept and reconstruct the taxonomic view. It is, however, an implementation issue for software which will navigate the database, rather than a conceptual design of the data structure and will not be examined further here.

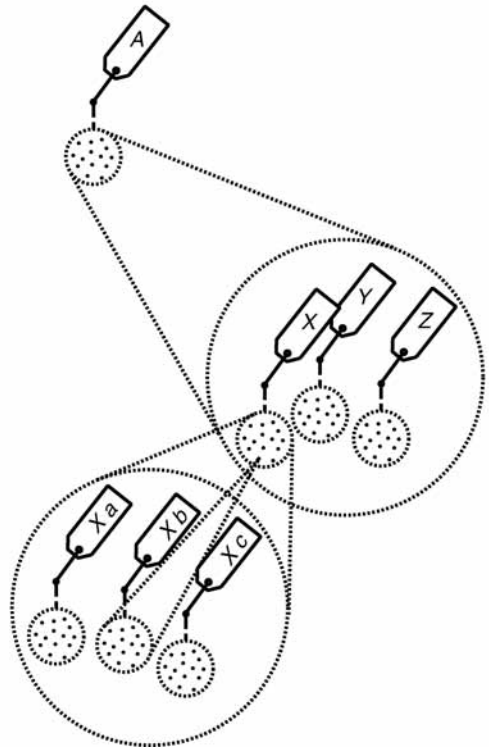

Figure 3. Taxon concept encapsulation captures a taxonomic hierarchy. A taxon concept (broken circle) can be expressed as a set of instances (dots in the circle) with a name tag. For instance, family A is a taxon concept embracing genera (≡ instances), *X*, *Y* and *Z*. Instances of genus *X* are themselves lower level taxon concepts, i.e. species *X a*, *X b* and *X c*. This chain may be extended upwards as far as necessary, although its lower boundary will be individual specimens. Each combination of a name and a taxon concept is specified in a publication, so not only taxon concepts but also the range of the hierarchy is publication dependent, enabling the database to hold multiple taxonomic views, since each encapsulation is publication specific. Even if the publication does not specify all the hierarchy levels necessary to fill the levels between *A* and *X*, the data must reproduce this structure without adding 'missing' levels. Gap-filling between them is a responsibility of the hierarchy navigator (if the user requires it to estimate the missing nodes), but each record should hold no data that are not in the publication. Scalability, by using the same data structure for any level of the classification, enables flexible handling of missing levels.

EXPLANATION OF THE THREE-LAYER MODEL

Suppose a group of authors proposed a taxon and gave it name A in publication P_A. This publication can be expressed in Figure 2 as a pair of lines in a single vertical plane (i.e. a time slice), indicating an instantaneous event, one linking the bundle of individuals to a node corresponding to a taxon concept, and the other linking the taxon node to a name tag. The way of bundling individuals depends on the taxonomic concept described in the publication and exists from then onwards. This taxon concept is expressed as a line extending in time rather than as an instantaneous node. Another group of authors proposed a taxon with name *B* in publication P_B. Their taxon concept coincided with that proposed earlier in the opinion of a later author and, although it may be obvious that *A* and *B* are the same thing, both names exist validly and independently until the later publication P_A^{mod} formally submerges name *B* as a junior synonym. When yet another group of authors proposed a similar but different taxon concept for part of the same group of individuals and gave it a name *C* in a publication P_C, this can be captured by a name tag *C* and a pair of links which are anchored to an overlapping set of instances. In summary, three taxa A, B and C were proposed, taxon *A* was also named *B*, a matter of clerical error, but taxon C represents a difference of taxonomic opinion.

Thereafter, suppose yet another group of authors recognize that all these taxa refer to the same group of instances, and conclude in a publication P_A^{mod} that the name *A* should be used and the names *B* and *C* are junior

synonyms of *A*. This recognition may create a new taxon concept when the conclusion differs from all of the original taxa. For example, if taxon A (and hence B) and C differ in the morphology of the individuals due to environmental factors, a new concept covering both taxa would be created. This opinion can also be expressed by a pair of links and a name tag. This introduces a link from the later name tag *A* to all three previous name tags *A*, *B* or *C*, each of which was defined in publication P_A, P_B or P_C. These links ensure that all three names were recognized in the nomenclatural antecedence of A^{mod}. It implies that the taxon lines and name lines must be extended to at least when $P_A{}^{mod}$ was published. $P_A{}^{mod}$ links the later name tag *A* to the newer taxon concept, but it does not link the name tag *B* nor *C* to any taxon concept. Instead, the name lines *B* and *C* are linked to the name line *A* as synonyms. One might reasonably expect that, for most practical purposes, it is the taxon layer that non-taxonomists users would find most useful. Note that this figure represents the single opinion of the authors of $P_A{}^{mod}$; other opinions are also possible and each opinion gives its own arrangement in the taxon layer.

It may appear that distinguishing between the taxon and the name introduces an unnecessary complexity, because a taxon and a name are so tightly bound. A consideration of synonymy and homonymy explains why the two should be separated. Synonymy is a situation in which two or more different names are assigned to the same taxon; homonymy is where the same name is given to two or more taxa. The detection of homonymy requires that we can detect a name being used to refer to more than one incompatible instance. In practice the information to detect this situation normally comes from the hierarchical information (see below), so that for instance the genus name *Pieris* refers both to a member of the Ericaceae (a large group of shrubs including *Rhododendron*) and to a member of the Pieridae (a group of butterflies, including the cabbage white). This homonymy is allowed because the instances fall under separate Codes of Nomenclature, i.e. separate namespaces. Even within the scope of a single Code, homonymy may not be recognized for many years, demonstrating that, at least in the past, it has often not been a practical problem because taxa referred to by the name never appear in the same context, i.e. the name is used in different name-scopes (see *Curiosities of Biological Nomenclature*, http://www.best.com/~atta/taxonomy.html, for further examples). In classical taxonomy, where most taxonomists concentrated only on a specific, well-defined taxon, such undetected homonomy could persist for many years. In this age of DNA sequencing, however, taxonomists may come across homonymy more readily through database searching. The relevant point here is that it is possible to detect some cases of homonymy automatically within the database through hierarchical information, provided it is present. Homonomy in closely related taxa cannot, of course, be detected in this way.

There are two types of synonymy: heterotypic (also called taxonomic or subjective) and homotypic (also called nomeclatural or objective). Automatic detection of homotypic synonymy is possible within a database provided the necessary data are present, normally the unique identification of the type specimen. Heterotypic synonymy, on the other hand, is very much more difficult to detect because it requires taxonomic judgement to place two or more instances within the same taxon concept, which should therefore carry a single name. In a limited number of cases, such judgement may be published but unrecorded directly and may be inferred from the data held; this is particularly the case in higher taxa where the publication is re-arranging lower taxa. In general, however, the recognition of heterotypic synonymy requires taxonomic judgement and the necessary data for such a decision are not present in the database as described here. Contingent on this, statements of heterotypic synonymy are the published opinion of an author and can be encoded into the database. Authors may not agree, so that the simple situation described in Figure 2 represents a single opinion, a single horizontal line after $P_A{}^{mod}$, and in reality there may well be contrary views being expressed. The key purpose of a nomenclatural database is to make these views accessible to the researcher and to minimize the number of irrelevant citations that need to be retrieved from the library (Maurer, Firestone & Scriver, 2000).

In the example given in Figure 2, the last publication $P_A{}^{mod}$ contains such a statement of synonymy. Before publication of $P_A{}^{mod}$ there was no way of knowing from the literature that the taxa referred to by the name A and B belonged to the same concept, nor that taxon C referred to more instances of the taxon. The last publication $P_A{}^{mod}$ may state that: "No differences of taxonomic significance between A, B and C were found, thus *B* and *C* are junior synonyms of *A*", or words to that effect. This statement should be understood as follows: instances of the taxa referred to by the names *A*, *B* and *C* belong to the same taxonomic concept, therefore name A should be used for the taxon, names *B* and *C* are aliases, which should no longer be used, of the name *A*. This change effectively modifies the definition of taxon A. Although the name A remains unchanged, note that it now represents a different taxon concept. The latter statement is of course an over elaboration for taxonomists who understand the implication of the former statement. It should, however, be stated clearly that in building a data structure for a flexible database, such implicit understanding by taxonomists is not obvious to the database architects nor the database management system. Assumptions made in taxonomy need to be explicitly implemented in the data structures.

Although the example figure was restricted to species, the same explanation can be applied to the higher taxonomic ranks such as a genus, a family, and so on by replacing the individual threads in the instance layer with the lower rank threads, each of which is a shrink-wrapped thread of the lower three layers (Fig. 3). Climbing up the taxonomic ranks, the three layers of the lower rank are nested as threads in the instance subspace of the higher rank. Here the names also provide interfaces between the ranks. This taxonomic hierarchy can be captured by links between names belonging to adjacent ranks, and hence the same structure can be used for any rank. The hierarchical linkages may be stated explicitly in publications or inferred only by reference to other taxa in other ranks. Further exploration of this scalability and inference are beyond the scope of the current discussion.

IMPLEMENTATION OF THE CONCEPTUAL MODEL

The model described above and summarized in Figure 2 is intended to mimic the process of a taxonomist tracing nomenclatural relationships through a library search. As such, the nature of information and the manner in which it becomes available are important considerations. It is not normally possible to start from the original authority publications and work forwards, rather one starts from a summary or revision and works backwards and then forwards again. Information becomes available in small snippets and, in notebook fashion, is stitched together in a later synthesis phase. Information about the taxon concepts under investigation is not immediately accessible and certainly not easily encoded in a general scheme. The taxon concept layer of Figure 2, although crucial to the process and the ultimate goal of taxonomic research needs to be inferred from statements made about the names and their relationships. The database design follows a scheme based on accessibility and utility rather than trying to adhere to a conceptual model whose data units would be difficult to capture. In particular, the design is intended to be usable with data of variable quality, or indeed missing altogether.

DESIGN OF THE DATA CONTAINERS

The Entity-Relationship (ER) diagram and its Extended version (EER) are commonly used to describe the data and their relationship in a database: taxonomic databases are not an exception to this approach (Berendsohn, 1997). In the following description of the data model, however, an object-oriented approach will be used because it is sensitive to the identity of each of the data records (the objects). Such usage is intended for clarity of description rather than as an implementation imperative. As Berendsohn (1995) stated in his discussion of the potential taxon concept, data models based on publication may easily result in inflation of data, that is new database records will be created which contain no new information. New data records need to have an unique identity, thus must contain unique information. The object-oriented paradigm (OOP) is particularly sensitive to the identity of the objects, but the use of OOP does not mean that the data model requires object-oriented database management systems (OODBMS). Indeed, the described data structures can be converted into the normalized relations (tables) of a relational database (RDB). The description will begin with the basic data structures comprising the data model, which will be refined to avoid the problems suggested by the potential taxon concept.

The OOP uses data structures called objects as building blocks of software (Budd, 1997, 2000). The object is an abstraction of a real thing, which may be either a concept or a physical matter, so the OOP requires the identification of real things for the design of software objects. The focus of the model is on the identification of real things and the relationships between them, rather than on implementation details for each software object (Booch, 1991, 1994, 1996).

DATA AVAILABILITY

Taxonomic and nomenclatural information is scattered through the literature, both as primary descriptions, (i.e. first record of new taxa), or as revisions (i.e. reorganization or updating of existing descriptions). This means that when undertaking nomenclatural research or building a database the data become available in small and often incomplete pieces. The data model must be sufficiently flexible to accept such data fragments and to store them sensibly so that more complete statements can later be built and there is not an unwanted inflation of recorded objects presenting aspects of the same conceptual item. To be of interest, there must be a taxon name and some form of publication reference, of course. The description of the data model that follows is intended to allow such flexibility and there are no mandatory components specified.

THE BASIC DATA COMPONENTS

A nomenclature database is designed to manage names. To exist, according to the rules of nomenclature, the name must be properly defined in the literature, so the data source is a publication (Greuter *et al.*, 2000; Ride *et al.*, 1999; Sneath *et al.*, 1992). The data model has an object named *Publication* intended primarily to identify the publication in the real world, i.e. a bibliographic citation. Thus the basic datum for the database is the publication in which names appear.

To be of interest, each publication must contain at

least one taxonomic name. A combination of each name and *Publication* should enable unique identification of records in the database, in essence the potential taxon concept (Berendsohn, 1995). The data structure combination of the name and its publication is referred to as a *NameRecord*. The *NameRecord* may contain additional information, such as the location of the type specimen, for example. The name itself is only a literal string and there must be some contextual information present to specify the way the name was used. For instance, nomenclatural hierarchical information, some nomenclatural statement, such as *nov. sp.* (new species), or links pointing to other publications where the name was used are useful extra information. Note that such information need not be unique to the *NameRecord* and that a taxon can be described as *nov. sp.*, for example, in multiple publications, e.g. *Phisteria* Burkholder & Glasgow, 1995; Burkholder, Glasgow & Hobbs, 1995.

Different publications which include any given name are of relevance to understanding the relationship between the objects described and, unlike the potential taxon model (Berendsohn, 1995), do not represent empty inflation of the database. Inflation can occur when more than one record refers to a single publication-name object, which can occur when the publication is captured indirectly, by citation from another work. This issue will be discussed further below.

The *NameRecord* structure and the *Publication* structure were not combined because a single *Publication* may contain multiple names and a *NameRecord* object relates to a taxon rather than the *Publication*. A *NameRecord* must, therefore contain a link to the *Publication* rather than being part of the *Publication* object itself. This pairing creates an unique identifier for each record and is given the notation (*NameRecord, Publication*), for instance, taxon A in publication P_A is represented by (A, P_A).

Even a single statement of a name in a publication may contain multiple names. For example,

> "**Lembus (Vibrio) verminus** (O. F. MÜLLER, 1786) (*Lembus elongatus* CLAP. u. L., 1859; *L. velifer* COHN, 1866; *itermedius* GOURR. u. R., 1886; *striatus* COHN-FABRE, 1885; *ornatus* SMITH 1899; *infusionum* CALKINS, 1903)"

which appeared in (Kahl, 1930–35: 369). The statement followed contemporary taxonomic convention for the discipline (ciliates) and means that the species *Lembus verminus* was originally described by O. F. Müller in 1786 under the name (the basionym) *Vibrio verminus*. Kahl does not tell us who first moved the taxon from the genus *Vibrio* into the genus *Lembus*. This relationship is not technically a synonymy but a recombination, notwithstanding that we end up with two binomen representing a single taxon. In Kahl's view, it represents a single taxon concept with those named *Lembus elongatus, Lembus velifer, Lembus intermedius, Lembus striatus, Lembus ornatus* and *Lembus infusionum*, as described by the various authors listed, which are all considered junior synonyms and in the absence of other information these statements are assumed to represent heterotypic synonymy.

This statement illustrates a number of data-encoding issues. First, it contains eight names that are structured to indicate the relationship between those names. An additional data object called an *Appearance* was used to capture this statement, being another abstraction of the publication, to contain the appearance of the name cited. The *Appearance* object is designed to record the page number where the taxon concept first appears, in order to specify uniquely the taxon concept referred to, and to hold the name appearance datum to capture cases where a single description contains multiple names. The name appearance can also be used to handle simple clerical mistakes such as typographical errors. A single statement in an *Appearance* may contain multiple names, so the *Appearance* should not generally be held by a single *NameRecord* but should be shared by the *NameRecords* of the names listed in the statement. On the other hand, an *Appearance* must be unique to a single publication but a publication may own several *Appearances*. *Appearance* contains a list of links to the *NameRecord* objects associated with each name identified from the statement and a link to the *Publication* in which the statement appeared. Each *NameRecord* points to an *Appearance* rather than the *Publication* itself, so a *NameRecord* is linked to a *Publication* indirectly through an *Appearance*. Each *Publication* has a list of links to the *Appearances* which it contains, which may in practice be added at different times by different users. The data structure *Appearance* also contains the precise page where the description appeared first in the *Publication* as an auxiliary link to the *Publication*. This page number link can help in the identification of names when a *Publication* contains several usages of the name; for example, a larger volume of revision work written by several authors, or a list of names such as Zoological Record may contain the same name referring to different taxon concepts on different pages. In the rare instances where unconventional characters, ligatures or diacritics are used, the referrng work will rarely quote the exact typographical form of the original name. The page number enables the precise location of the original statement so that such data may be captured.

The three data structures, *Publication, Appearance* and *NameRecord* provide three forms of abstraction from the original data source, i.e. a publication which you may find in the library. This linked abstraction enables context-dependent self-identification of names. The same name, i.e. the single literal string, appearing in multiple publications may be distinguished in the database as

multiple *NameRecord* objects each linked uniquely to different *Publication* objects through *Appearance* objects. Relationships between those abstracted *NameRecord* objects, however, are not yet established. Relationships expressed in the appearance of the name, such as a synonym list in the above example, are also lost. The example given above, with eight names embedded in it, expresses synonymy between the names. Without retaining those relationships, a collection of *NameRecord* objects does not work as a database designed to help taxonomists. These relationships must be captured by other data structures.

The second issue highlighted by the example is the use of the diacritic in Müller. Although this might seem to be an issue of implementation, it has structural significance because of the constraints it imposes on information flow. As discussed above, the rules of nomenclature specify that organism names are to be spelt without accents or other non-Roman marks. There remains a serious problem in string-matching of author names because earlier attempts at machine-encoding information used a variety of methods to represent the ü. The most common were Mueller or Muller. A limited set of diacritics is available in the extended ASCII set, but it is dominated by Western European languages and omits those of Eastern Europe. Even though this encoding issue can be solved by use of appropriate code sets (either ISO10646/Unicode or ISO 2022 style code set switching), translated author names in data sources remain problematic. Transliterated author names, for example from Russian, are also highly variable, depending on the phonetics of the target language (in taxonomy most commonly German, French or English). The problem is well known in computer science and is beyond the scope of this study, but the nature of matches that can be made between database objects demands that we be aware of the problem.

The third issue illustrated by the example, again raised here because of constraints on information flow, is the abbreviation of well-known author's names. Being well-known is a function of the field of study, of course, but it is common practice to truncate names, such as 'CLAP. u. L.' which is short for Claparède & Lachmann. Kahl is comparatively unusual in including the publication date in the statement and the reference to the listed authors in his bibliography. If neither the date nor the reference are given, then the capacity for matching is handicapped and must rely on a more flexible strategy. It is noteworthy that the Botanical code (Recommendation 46A; Greuter *et al.*, 2000) encourages such abbreviation.

RELATIONSHIPS BETWEEN THE DATA COMPONENTS

Clearly the principal information required to manage names and their relationship to instances is the taxon concept (Fig. 2). Database applications, such as PANDORA (Pankhurst, 1993) and descriptive structures, such as DELTA (Dallwitz, 1980) have been built to capture the taxon concept more directly by means of a circumscription, though none has been used to examine how the concepts change over time. The labour of completing such descriptions is substantial, especially in poorly worked groups, and they are of very limited value when only partially complete. Taxonomists begin the process of building these taxon concepts with library searches and, if they are lucky enough, with access to a specimen collection. The model described here seeks to imitate the library-search process. The data components that are available are linked as shown in Figure 4 and are related as follows.

To be of interest, each publication must contain at least one statement operating on a taxon, such as creation of a taxon, merging two or more taxa, partition of a taxon into two or more taxa, and so on. Besides the bald creation of a taxon, i.e. a new description, other statements include annotation to a previous statement on a taxon. Creation of a taxon should refer to previous works to differentiate it from other described taxa. These relationships are captured by a data structure named *Annotation*, which is an abstraction of an action in the publication, i.e. a summary of the taxonomic operation performed. In the previous example (Fig. 2), the authors of a publication P_A^{mod} stated that taxa (A, P_A) and (B, P_B) cannot be distinguished and they proposed synonymizing them under the name A. The merged-pair taxon may not represent an identical taxon concept—at the very least there are now more instances—so the new taxon is represented by a new *NameRecord* (A, P_A^{mod}). Although the *NameRecord* (A, P_A^{mod}) and *NameRecord* (A, P_A) objects share the same name A, they express different notions. This relationship was captured in an *Annotation* object with the reasoning attribute 'synonym' and links from (A, P_A^{mod}) to both (A, P_A) and (B, P_B). The data structure *Annotation* retains the links as a list because there may be multiple *NameRecords* linked to a given *NameRecord*. The actions being encoded appear in publications; so, naturally, the *Annotation* object also has a link to an *Appearance*. The linked *Appearance* may have a list of links to other *Annotation* objects because a single publication may perform many taxonomic actions.

The most important aspect of building a data model is the recognition of the information to be held and the creation of suitable containers. Taxonomic publications, especially those dealing with nomenclature, are rich in information but it is often expressed with such brevity and couched in discipline-dependent language, so that its meaning and implications can be obscure. The *Annotation* object was created to hold that information and is crucial to the ability of the

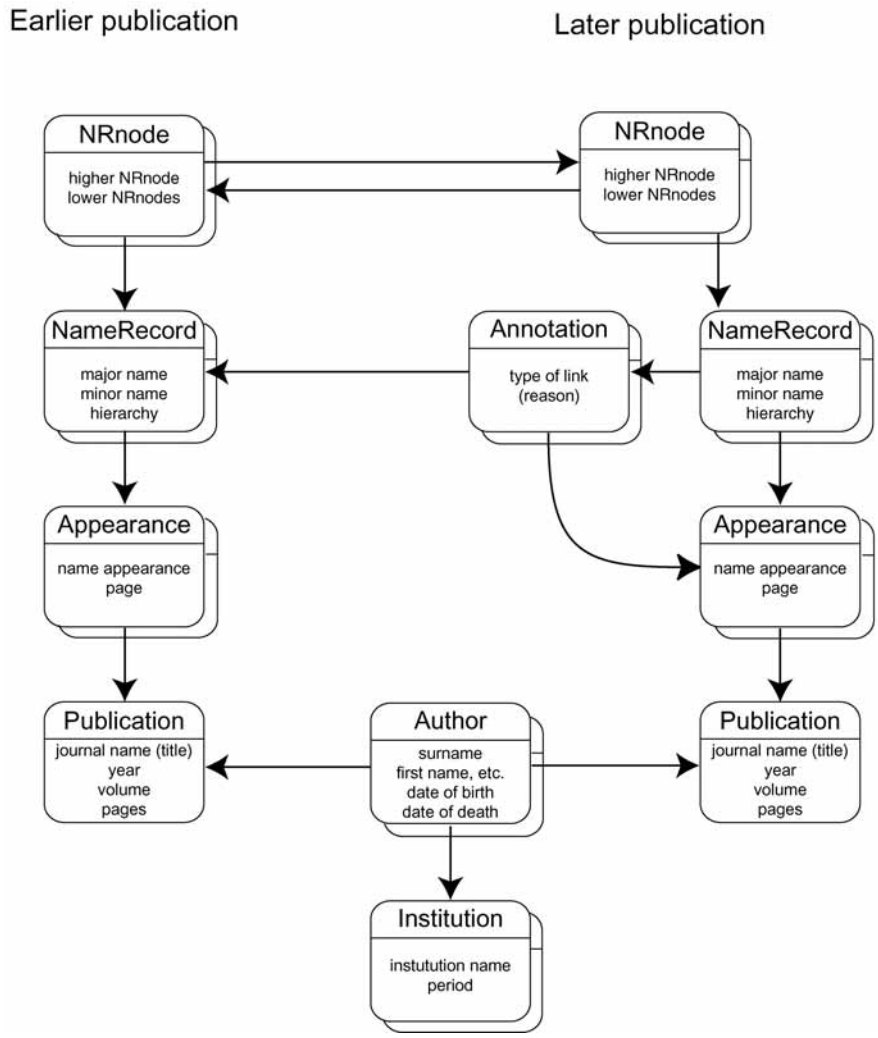

Figure 4. An object-oriented diagram showing an example implementation supporting the nomenclature data structure. Each round-cornered box indicates an *object*, a unit of data storage. The name and typical attributes of the object are shown in each box. Links between the objects are expressed by arrows between the boxes. The nomenclature data structure is publication oriented, rather than taxon oriented, hence the objects can be grouped by publication base (shown as columns in the diagram), which can be separated in time. An unique link between the name and the publication is the primary implementation issue and must provide an identification for a name based on a publication. Note that each *Publication* must possess one or more *Appearance* objects and each *Appearance* must possess one or more *NameRecord* objects. A column composed of these three objects corresponds to a publication which is the core unit in the structure. The structure does not have a centrally-controlled taxon concept, so links between names are based on publications. A publication often refers to previous publications and these references are captured by *Annotations* pointing from the later publications (right column) to the previous publications (left column). Since the annotation is publication specific, the *Annotation* object is linked to (or, owned by) the later *Appearance* object. The *Annotation* object has an attribute of link type: the richness of this link type is crucial to capturing the taxonomic relationships between the names (see Table 1 for details). The *Annotation* is unidirectional: it never refers to future publications, but always refers from the publication giving the annotation to previous publications. Since this is inconvenient for navigation in the name layer, the *NRnode* was created to facilitate finding publications which refer to a given publication, thus the network of the *NRnodes* provides the primary working domain for the navigator software. The *NRnodes* record the *NameRecords* referring and referred to by a given *NameRecord*, so are bi-directional. Notwithstanding its disadvantages in navigation, the unidirectional property of *Annotations* simplifies maintenance of the core data records because addition of new records does not modify the core data records and requires only modification of the pointer list in the *NRnode* objects, even if the new record proposes drastic change of taxa. *Appearances*, especially when abbreviated, are sometimes inconsistent or ambiguous which presents a serious problem to a publication-based database. *Authors* can be used to relax this constraint because the identification of authors can lend support to a link between publications that is inconclusively identified. *Affiliations* can also help identification of authors. These objects are intended to be used by navigator programs for arbitration.

Table 1. Examples of link types used in *Annotation* objects with the cardinality of the links with the *NameRecord* object. This list contains only a simple set of relationships and must be extended to encompass the richness and subtlety of taxonomic relationships

Type of linkage	Number of NameRecords	
	Refers from	Refers to
Reference	1	n
Revision	1	1
Synonym	1	n
Homonym	n	m
Partition	n	1
Equivalence	n	n
Nov	1	n
Assignment	1	0
Propagation	1	1

model to encode relationships. Taxonomic language has evolved to contain a high degree of implication and subtlety which should properly be expressed in the *Annotation* object, since this is the location of taxonomic actions. As the database expands to encompass a broader range of taxonomic groups with their nomenclatural conventions, new descriptors will be needed. The descriptors that currently can be recognized by the prototype implementation are listed in Table 1 and represent a crude set of simple relationships. Enriching the link type descriptor will increase the number of different relationships that can be captured and is clearly an area for future development. Ideally, these link types and their equivalence will be encoded as a data object to be used by the implementation software so that the practices and requirements of different user communities can be both accommodated and shared.

Publications have one or more authors and an individual author is another route to associate taxonomic names with publications. Under the rules of nomenclature, authors should quote the authority for a given name, i.e. the name of the author who created the taxonomic name. No date is formally required and authorities are rarely cited in the bibliography of a publication. This presents a common missing-data problem so a data structure named *Author* is defined to facilitate links being made through the name of individual taxonomists, especially in the absence of information about the publication itself. It contains a parsed name (i.e. a surname, a first name, etc.) of the author and a list of links to *Publication* objects. It also has a slot for the list of affiliations and the date of birth and death to enable identification of the author, even though this information is often not available. This configuration was designed to allow the navigating software to infer likely publications even though the taxonomist's name is not unique. The *Author* object is not intended to hold authorities specifically: this information is available through a combination of the *NameRecord*, the *Annotation* and the *Appearance*. In those cases where an authority (say, Smith) is not also an author of the publication, then a *Publication* object must be created of the form 'Smith in Jones (date)' where Jones (date) is the traceable citation.

The link-structures *Annotation* and *Author* enable bibliographic tracking of the *NameRecord* objects in the temporal domain. A nomenclatural database naturally focuses on the taxon name rather than its data source, i.e. publication, so the *Author* objects linking *Publication* objects are not used so intensely. The *Author* objects, however, provide an alternative, indirect way to track *NameRecord* relationships when the annotation link is not readily available. This strategy can also be used to relieve the diacritic and abbreviation difficulties discussed above.

TRACING THE BASIC DATA COMPONENTS

Annotation objects and *NameRecord* objects are abstractions from publications. No publication can contain information on what might happen in the future, such as 'this taxon will be amended 14 years later', so it is logical that no such future information is located in *Annotation* or *NameRecord* objects. This restriction permits only backward tracing, i.e. tracking to past name usages, so another data structure, *NRnode* (which is an abbreviation of *NameRecord* node), was devised to trace a *NameRecord* both backwards and forwards in time. It is a meta data structure, used to integrate the abstracted objects into a database, containing a link to a *NameRecord*, and two lists of links to other *NameRecords*, one list corresponding to *NameRecords* which refer to the *NameRecord* in question and the other to *NameRecords* which are referred to by it. The *NRnode* might be viewed as an implementational detail because it is purely a metadata construct; it is described here in order to clarify the location of information within the data, which is the prime purpose of the model.

By design, the name is the predominant query key in a nomenclature database, so a lookup table from names to a list of *NRnodes* is sensible to assist performance. Database management systems will look for a queried name in the table, then track from each *NRnode* to find the whole history of the name. The query result, a list of names relating *Publications* and the relationship between them, is probably sufficient to reduce significantly taxonomists' drudgery in tracing taxonomic history through the library; it should give

rapid access to a list of sources that need to be consulted to resolve a nomenclatural question. It is, however, too terse for ordinary public use. More general users will require a more structured output better tailored to their needs, presenting a sub-set of the total information in a clearer manner. How this is done will depend on an understanding of the needs of the general user and will probably be controllable by the users themselves in an intelligent manner. This is a function of the navigating application and is beyond the scope of this description of the basic database structure.

PROTOTYPE IMPLEMENTATION: NOMENCURATOR

A comparison of Figures 2 and 4 shows that the central element of the three-layer model, the taxon concept, is missing from the object-oriented model. As explained above, despite the fact that users would undoubtedly want to navigate though the taxon concept layer, this layer is not capturable directly. It can be seen, however, dimly reflected in the taxonomic operations recorded in the publication and held in the *Annotation* object. The concept of a 'nomenclature database which is free from taxa' is unusual. It may make experienced taxonomists or taxonomic database-researchers uncomfortable, although it is intended to mimic the thought processes of a taxonomist doing bibliographic research and is a consequence of the potential taxon concept. A prototype database was implemented using the programming language C^{++} to explore further the concept of mimicking the taxonomical approach.

IMPLEMENTATION USING OOPL

Object-oriented programming languages (OOPL) were chosen because they facilitate a more straightforward implementation of the data model. Of the available OOPLs, C^{++} was chosen because the model is rich in links between objects (in the terminology of C^{++}, pointers to objects), and C^{++} has good performance in pointer handling. Existing database management systems (DBMS) including relational and object-oriented databases were not used in order to make clear the requirements of the data structures themselves rather than the technical restrictions of the DBMS.

Implementation of each attribute of the data structure (a class, in the terminology of C^{++}) was straightforward. The source code and supplementary material are available from http://www.nomencurator.org/. The code can be compiled using C^{++} compilers supporting ANSI C^{++} 3.0, such as gcc 2.9. The code was tested on Linux version 2.0.36 and the 2.2 series (except 2.2.8). It has a stand-alone mode and a server mode which can be switched by command-line options. Further details are available in the source code and accompanying documents.

It would be quite exceptional for the majority of (or in some cases' any) records to have complete information. Again, this mimics the process of taxonomic bibliographic research as it actually takes place. The Nomencurator model requires that publications are the source of records, although it is commonplace to generate a record without access to the publication itself, most commonly from a citation in other publications. These citations may not contain sufficient information to satisfy fully the requirements of the Nomencurator model, but the data may be stored and handled in this form until they can be supplemented from other sources, essentially by matching the core elements of *Publication*, *Appearance* and *NameRecord*, and then comparing other fields. This process represents a real risk of inflation when multiple partly-complete records representing a single publication are not recognized as such. In the real world, these partial citations are often combined to help in location of the complete reference and so it should be in the data model. This aspect works poorly in the current prototype which defaults to assuming records to be different unless they match exactly. There is clearly a balance to be struck between this position and assuming that any citations which match partially but without conflict are the same. Development of matching algorithms is clearly an area demanding further work, especially if the database is distributed over several nodes.

MAPPING TO AN RDB

The prototype implementation was written in an OOPL because of the straightforwardness of this approach and the primary objective was to explore the information structure present in the available data. The Nomencurator model can, however, be mapped to a relational data structure as a set of relations (i.e. a set of tables) in a relational database (RDB). Well-designed data structures (or classes, in the terminology of C^{++}) can be mapped directly into normalized relations in an RDB. The corresponding ER diagram is shown in Figure 5. Although it shows the independence of the model from a DBMS architecture, it may prove inappropriate, or at least inefficient, to use an RDBMS because of the recursive, pointer rich structure of the data model. Indeed, the choice of a suitable DBMS is restricted by the need to navigate through the *NRnode* structures which express a network of data paths.

DISCUSSION

The Nomencurator model set out to elucidate the information processes which underlie taxonomy, resulting in the relationships depicted in Figures 1 and 2. The rôle of the taxon concept in the relationship

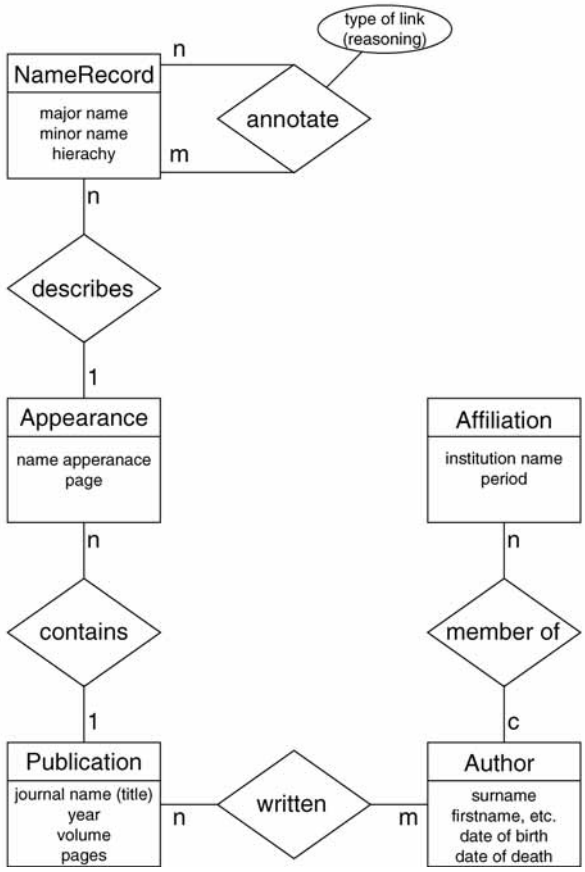

Figure 5. The object model shown in Fig. 4 can also be used in a relational database management system (RDBMS) through a mapping to Entity Relation (ER) diagrams. The object model provides an already normalized ER diagram. Note that the temporal structure shown in the object diagram does not appear clearly in the ER diagram. Such a phenomenon may happen, depending on the programming paradigm used in the design phase, and underlines the value of a pluralistic approach.

between names and instances was seen to be crucial in understanding the information flow. A name is tied to an instance, ultimately to a specimen, and not to the taxon concept itself. It is assumed that the name is unique and the taxon concept to which it relates is non-overlapping with other taxon concepts. Where these assumptions fail taxonomy and nomenclature become inconsistent and unreliable.

Taxonomic opinions revolve around relations between taxon concepts and may result in alternative hierarchies. Conflict in taxonomic opinions requires resolution if the taxon concept is the primary data container, making it difficult, but by no means impossible (e.g. Beach *et al.*, 1993), to hold multiple taxonomic opinions. The model described here does not use the taxon concept as a data container and hence does not require opinions to be resolved. Relationships in this model are held within the *Annotation* object in the form of links and multiple opinions result in a more complicated linkage map.

The use of the taxon concept as primary data container forces users to make taxonomic decisions on the validity and appropriateness of data at the time of data entry, i.e. the data must be allocated to a particular taxon concept. In the early phase of taxonomic research such decisions are not normally taken since they are usually the purpose of the research: it was the objective of the Nomencurator approach to remove or avoid such decision-making. Such prerequisite taxonomic decisions were decomposed separating the taxon concept from the name explicitly, locating the taxon concept into a temporal thread. This projection of the taxonomic link onto the temporal thread of references enables the database to hold information on taxon concepts without an explicit data container. The taxon concept will be revealed by navigation over these threads.

COMPARISON WITH OTHER DATA MODELS

The Nomencurator model retains information on taxa as a concept inferred by annotation links between names under the assumption that a single taxon has a single name in the publication where the name appeared. The captured taxon is a dynamic meta-object because it is not explicitly defined in the data structures, but its existence is inferred through a thread of *Annotations*, where annotation is a temporal action that is performed on names and publications that already exist. Thus, Nomencurator works in a referential model (Ghiselin, 1997; Härlin, 1998) and does not attempt to hold descriptive information on the taxa themselves, unlike ReTAX (Alberdi & Sleeman, 1997). Nomencurator was designed to hold nomenclatural information from diverse published sources, made available in the fragmentary fashion typical of a primary-level literature search. It was not designed to accommodate more richly structured data or descriptive data which are in general less readily available or which require synthesis by the user as part of the data gathering process. It could, of course, be used as the nomenclatural component of a more complex application.

It is not uncommon for only the scientific name and authority, without date or bibliographic citation, to be given in a publication. Even with such poor source information, the prototype database was implemented to accept and handle these data because in the Nomencurator model not only each *NameRecord* but also the links between them, i.e. the *Annotation* data structures, play important rôles. This active use of incomplete data to accumulate information is a particular feature of Nomencurator.

Most taxonomic databases provide a data structure to retain information on a taxon as a core object. A taxon in data models of this type is static compared to that inferred in the potential taxon model and its successors including the Nomencurator model described here. When a new taxon is proposed (and accepted in the case of static data models such as Berendsohn, 1997), it is necessary to create a new taxon entry in the database and map pre-existing taxa to the new taxon where necessary. The mapping is unnecessary in the Nomencurator model: when a new datum entry is created, corresponding to a new publication, annotation links are created that join directly to the referred records by name and citation. This significantly reduces the maintenance load in keeping the database up-to-date and is one of the advantages of the Nomencurator model which is particularly important in those areas of taxonomic instability which *de facto* are most in need of nomenclatural research. Mapping from old to new taxa is impossible in the real world because the newer publication cannot affect previous publications, it can only change the context of the existing publications. This is what is implemented in the Nomencurator model but not in static taxon models.

The static taxon model, on the other hand, provides a convenient summary for inexperienced users. A database using the Nomencurator model can produce such concise information quickly and easily so that a taxonomic expert can prepare a static view for general application. The Nomencurator model can generate a static taxon summary by projecting threads in a given 'time slice', although it is obviously incapable of exercising taxonomic judgement and thus is incapable of producing an automatic review. There can be several modes of such data reduction; not only an 'accepted view' (if available) but also 'current', 'all' or some specific taxonomist's view, although it must be pointed out that 'current' means most recent and does not always mean 'correct'. An important application of the 'all' would be the recovery of data from databases using names as a key field which are not nomenclaturally validated, such as the molecular sequence repositories. For organism groups which have not had the benefit of a modern taxonomic review, this route may be the only one possible. This multiplicity of options for data recovery may not be appropriate for use by non-experts, as required by Species 2000 for instance (Brugman, 1999), although it is desirable for databases which are a basic data provider for conservation of species.

There is another taxonomic database that has a rather more temporal approach to taxonomic data, namely HICLAS (Jung *et al.*, 1995; Zhong *et al.*, 1996). Detailed comparison between the nomenclature data model and HICLAS is beyond the scope of this article because HICLAS is weighted towards hierarchies of taxa rather than the taxa themselves. Here we focus on the dynamism of taxa in both models; the hierarchy issue will be discussed elsewhere. HICLAS has a more dynamic taxon concept than a static one, of course, but it is not as dynamic as the Nomencurator model. For example, consider a genus to which a few species belong when one of the species is moved to another genus. In HICLAS, this modification does not affect the genus. In the Nomencurator model, it evokes the creation of two genera because each taxon is linked to the name through the *Publication*, and the *Publication* proposing to move the species may also propose modifying the concept of the genera to which the species belonged, and to which it has been moved.

The separation of the naming system from the taxon concept can be generalized to the separation of expressions from content. In computer programming, for example, the same concept and technology could be used for revision tracking in software maintenance. Modern programming languages refer to data parameters, functions and operations by their names for convenience of programmers. These names have their own scope, within which each name has its own meaning: outside of the name scope, the names are not defined or may be used with a completely different meaning. References between components of a computer program are solved statically when the program components are linked, or dynamically when the program is executed. Databases based on taxon concepts are similar to a static, linked program in which each component must be defined clearly before linkage. The Nomencurator data model is similar to dynamic linkage because it uses names as interfaces and does not require formal definition.

The recently published Prometheus model (Pullan *et al.*, 2000) embodies many of the key differences that Nomencurator has with earlier models. Compared with Prometheus, the most important differences in Nomencurator are:

(1) The focus of attention on fragmentary, incomplete information which is used to support inferential chains, no matter how weak.
(2) The use of name (or ascribed name in Prometheus' terminology) as an interface to the taxon concept in the publication versus as an attribute of Circumscribed Taxon in Prometheus model.
(3) The encapsulation of names, taxon concepts and instances compared to Prometheus' partial overlap of Nomenclatural Taxon and Circumscribed Taxon.
(4) The temporal extension of the name (versus the taxon concept in Prometheus) to allow the construction of an audit trail of an instance's nomenclatural history.
(5) The use of the *Annotation* object to build a representation of taxon concepts via name(s).

It is reasonable to suggest that Prometheus' focus on herbarium-based taxonomy and Nomencurator's focus on micro-invertebrate, library-based taxonomy have given rise to much of the difference. Historically, it was common practice for botanical collectors to prepare multiple, equivalent herbarium sheets and to distribute them to various herbaria, sometimes unnamed. This is one source of the problem of the objective synonym, but also allowed taxonomists in different herbaria to handle directly equivalent specimens. Thus tracking names through specimens is feasible. Practices in Zoology are more varied than in Botany, reflecting the greater diversity of form and function. In some areas, for instance the protozoa, preservation of specimens has only relatively recently become practical, although when preserved they do not retain the full gamut of characters relevant to taxonomic analysis. In this case, use of the specimen as the primary data container would require a corruption of the concept of specimen because in most cases specimens do not exist. New material is identified by comparison with the published description, which is ultimately the only data source.

Nomenclature, on the other hand, is defined in the Codes of Nomenclature as being about publications. The focus of Nomencurator extends the potential taxon model away from the taxon and towards the publication with the intention of supporting the taxonomist in managing the highly fragmentary literature, both primary and revisionary. Unlike Prometheus, it does not attempt to aid the taxonomist directly in the resolution of a taxonomic problem.

The difference is significant, but should not mask the fundamental similarity of the analysis of the problem domain and, as Pullan and colleagues point out, there is much common ground with Berendsohn's (1995) insightful analysis, which provides mutual support for the general structure.

DATA INFLATION

One of the problems with the potential taxon concept and its derivatives is that it can suffer from inflation of records: 'Taxonomic monitoring' (Berendsohn, 1995) is recommended to avoid this inflation, by which is meant manual verification of records by someone who can exercise judgement to unite records representing a single taxon. In the nomenclature database described here, lacking an explicit data container for taxon, 'taxonomic monitoring' is logically impossible. 'Inflated' is an appropriate word from the viewpoint of taxonomic databases, because they are designed to handle taxa rather than names so the number of names and the number of records is expected to be concordant. In contrast, a database designed to handle publications, and thence names does not view it as inflation, but an essential richness of the nomenclature database.

The publication-localized design of the data model makes maintenance easier by removing the necessity for regular judgemental review. The data structure does requires a mechanism to avoid data duplication because the publication-based design has the possibility of data inflation, but not in the same way that Berendsohn (1995) pointed out. The acceptance of incomplete, fragmentary data can result in the same name-publication combination appearing more than once because of a failure to recognise the publication data as the same. As discussed above, the default position is to assume that citations which do not match exactly are different. The reason for this choice is because the name-publication is used as a self-identification mechanism of each datum. This self-identification mechanism is particularly desirable in a distributed database, in combination with the naming mechanism of CORBA, for example (see http://www.corba.org/). The combination of a data model which supports multiple opinions and a distributed database system based on self-identification makes distributed maintenance of the database easier because each taxonomist can maintain a personal database according to their own taxonomic requirements, but much of the older, summary literature can be entered once and become an institutional resource. Given that the data entry was accurate, no data can ever become redundant in this data model. Data accumulation is the most important and most expensive phase in the construction of the database. Distributed construction would make it easier, in the same way as the open source paradigm works well.

CONCLUSION

At the heart of this model, each *NameRecord* (potential taxon) is a manageable information unit, to be assembled into taxa under some organizational scheme or opinion. Exactly how this is done is beyond the scope of the current discussion, which is focused on how the information is to be make available to analysis.

The Nomencurator model was designed to mimic the way taxonomists work and record their data during the bibliographic phases of their studies. It decomposes the structure of nomenclatural information held in taxonomic publications and establishes relationships between the component parts. It can work as a flexible nomenclature database allowing multiple taxonomic opinions. The model requires implementation of a means to avoid data duplication: the current implementation used self-identification of a data object to satisfy this requirement, although it is acknowledged that this scheme is, as yet, imperfect. Although the implementation was written using an object-oriented programming language, it can be mapped to a relational database. Therefore, the

Nomencurator model provides a general foundation for a nomenclature database.

ACKNOWLEDGEMENTS

We are grateful to the University of Tsukuba, and in particular to Prof H. Seki, for the support for Nozomi Ytow to work in The Natural History Museum for a year (from 3rd November 1998 to 2nd November 1999), where this work was conceived. We would also like gratefully to acknowledge the contribution made by three anonymous reviewers, who pointed some of our lapses in clarity of expression.

REFERENCES

Alberdi E, Sleeman DH. 1997. ReTAX: A step in the automation of taxonomic revision. *Artificial Intelligence* **91:** 257–279.

Anonymous. 1999. Final report of the OECD megascience forum working group on biological informatics. http://www.oecd.org/dsti/sti/s_t/ms/prod/birepfin.pdf.

Beach JH, Pramanik S, Beaman JH. 1993. Hierarchical Taxonomic Databases. In: Fortuner R, ed. *Advances in Computer Methods for Systematic Biology: Artifical Intelligence, Databases, Computer Vision.* Baltimore: John Hopkins University Press, 241–256.

Berendsohn WG. 1995. The Concept of Potential Taxa in Databases. *Taxon* **44:** 207–212.

Berendsohn WG. 1997. A taxonomic information model for botanical databases: the IOPI Model. *Taxon* **46:** 283–309.

Berendsohn WG. 1999. Names, Taxa, and Information. In: Blum S, ed. Taxonomic Authority Files Workshop. Washington, DC. http://research.calacademy.org/taf/proceedings/Berendsohn.html.

Berendsohn WG, Anagnostopoulos A, Hagedorn G, Jakupovic J, Nimis PL, Valdés B. 1996. CDEFD Information Model for Biological Collections. Disseminating Biodiversity Information. Amsterdam: Proceedings of the European Science Foundation Workshop. (avaliable from http://www.bgbm.fu-berlin.de/CDEFD/CollectionModel/cdeFd.htm).

Berendsohn WG, Anagnostopoulos A, Hagedorn G, Jakupovic J, Nimis PL, Valdés B, Güntsch A, Pankhurst RJ, White RJ. 1999. A comprehensive reference model for biological collections and surveys. *Taxon* **48:** 511–562.

Booch G. 1991. *Object oriented design with applications.* Redwood City, CA: Benjamin/Cummings Pub. Co.

Booch G. 1994. *Object-oriented analysis and design with applications.* Redwood City, CA: Benjamin/Cummings Pub. Co.

Booch G. 1996. *Object solutions: managing the object-oriented project.* Menlo Park, CA: Addison-Wesley Pub. Co.

Brugman ML. 1999. Species 2000 Home Page: Species 2000 Secretariat. http://www.sp2000.org/.

Budd T. 1997. *An introduction to object-oriented programming.* Reading, MA: Addison-Wesley.

Budd T. 2000. *Understanding object-oriented programming with Java.* Reading, MA: Addison-Wesley.

Burkholder JM, Glasgow HB. 1995. Interactions of a toxic estuarine dinoflagellate with microbial predators and prey. *Archiv für Protistenkunde* **145:** 177–188.

Burkholder JM, Glasgow HB, Hobbs CW. 1995. Fish kills linked to a toxic ambush-predator dinoflagellate – distribution and environmental conditions. *Marine Ecology-Progress Series* **124:** 43–61.

Dallwitz MJ. 1980. A general system for coding taxonomic descriptions. *Taxon* **29:** 41–46.

Ghiselin MT. 1997. *Metaphysics and the origin of species.* Albany, NY: SUNY Press.

Greuter W, Mcneill J, Barrie FR, Burdet HM, Demoulin V, Filgueiras TS, Nicolson DH, Silva PC, Skog JE, Trehane P, Turland NJ, Hawksworth DL eds. 2000. *International Code of Botanical Nomenclature (Saint Louis Code) adopted by the Sixteenth International Botanical Congress St. Louis, Missouri, July–August 1999.* Königstein: Koeltz Scientific Books.

Härlin M. 1998. Taxonomic names and phylogenetic trees. *Zoologica Scripta* **27:** 381–390.

Jung S, Perkins S, Zhong Y, Pramanik S, Beaman J. 1995. A new data model for biological classification. *Computer Applications in the Biosciences* **11:** 237–246.

Kahl A. 1930–35. *Urtiere oder Protozoa. I: Wimpertiere oder Ciliata (Infusoria), eine Bearbeitung der freilebenden und ectocommensalen Infusorien der Erde, unter Ausschluss der marinen Tintinnidae.* Jena: G. Fischer.

Maurer SM, Firestone RB, Scriver CR. 2000. Science's neglected legacy. *Nature* **405:** 117–120.

Murphy FA, Fauquet CM, Bishop DHL, Ghabrial SA, Jarvis AW, Martelli GP, Mayo MA, Summers MD. 1995. *Virus taxonomy: classification and nomenclature of viruses: sixth report of the International Committee on Taxonomy of Viruses.* Wein: Springer-Verlag.

Pankhurst RJ. 1993. Taxonomic databases: the PANDORA system. In: Fortuner R, ed. *Advances in Computer Methods for Systematic Biology: Artificial Intelligence, Databases, Computer Vision.* Baltimore: Johns Hopkins, 229–240.

Pullan MR, Watson MF, Kennedy JB, Raguenaud C, Hyam R. 2000. The Prometheus taxonomic model: a practical approach to representing multiple classifications. *Taxon* **49:** 55–75.

Raguenaud C, Kennedy J, Barclay PJ. 1999a. The Prometheus Database Model (Report). Edinburgh: Napier University. http://www.dcs.napier.ac.uk/~prometheus/publications.html.

Raguenaud C, Kennedy J, Barclay PJ. 1999b. Query Language for Prometheus (Report). Edinburgh: Napier University. http://www.dcs.napier.ac.uk/~prometheus/publications.html.

Rees R, Sadka M. 1999. *The Postcode Plants Database.* London: National History Museum. http://fff.nhm.ac.uk/.

Ride WDL, Cogger HG, Dupuis C, Kraus O, Minelli A, Thompson FC, Tubbs PK. 1999. *International code of zoological nomenclature: adopted by the International Union of Biological Sciences.* London: International Trust for Zoological Nomenclature.

Sneath PHA, Lapage SP, International Committee on Systematic Bacteriology, Judicial Commission, International Union of Microbiological Societies, Bacteriology and Applied Microbiology Section, American Society for Microbiology. 1992. *International code of nomenclature of bacteria: and Statutes of the International Committee on Systematic Bacteriology: and Statutes of the Bacteriology and Applied Microbiology Section of the International Union of Microbiological Societies: bacteriological code.* Washington, DC: American Society for Microbiology for the International Union of Microbiological Societies.

Young JM. 2000. Recent developments in systematics and their implications for plant pathogenic bacteria. In: Priest FG and Goodfellow M, eds. *Applied Microbial Systematics.* Dordrecht: Kluwer, 135–163.

Zhong Y, Jung SW, Pramanik S, Beaman JH. 1996. Data model and comparison and query methods for interacting classifications in a taxonomic database. *Taxon* **45:** 223–241.