Conceptual Modelling for Applied Bioscience: The Bacteriocin Database

C. Maria (Marijke) Keet

School of Computing, Napier University, 10 Colinton Road, Edinburgh EH10 5DT, Scotland

Abstract

Semantic aspects are discussed for constructing a conceptual model capturing data across biological disciplines (biochemistry, microbiology, genetics) and including applied sciences (food science), by taking the development of a relational database for bacteriocins as a test case.

Capturing the subject domain semantics of an applied bioscience faces different problems compared to conceptual modelling for the primary biological sciences, as the former requires an emphasis on practical solutions conceptually representing the integration of various disciplines, necessarily reducing representation of biological complexity, whereas the latter stresses conceptually and ontologically comprehensive models within their primary specialisations such as biochemistry and genetics.

1. Introduction

In recent years, growth in availability of biological data has been exponential, and it is expected to continue at the same, if not faster, pace. It is a natural step to organise these vast amounts of data by making use of developments in the field of computing, where the combination of biology and computing gave rise to the discipline of bioinformatics. Viewed from the IT angle, it covers computational chemistry, neural networks, evolutionary computing and software and database development. The latter will receive special attention in this article, first by providing a brief overview of the current bioinformatics databases related research and, secondly, by considering the conceptual modelling aspects when constructing a model capturing data across biological disciplines (biochemistry, microbiology, genetics) and including applied sciences (food science), taking the development of a relational database for bacteriocins as an example.

2. Bioinformatics and conceptual modelling for biology

The division of bioinformatics concerned with structuring biological data and research output into databases is extensive. There are long established databases on DNA, protein sequence and genome mapping [37] and relatively more recent developments concerning metabolic pathways, protein interactions (e.g. [41]), gene expression and function databases. These probably will expand to encompass the emerging epigenetic data, which are relatively more challenging due to the increasing levels of interaction and relationships between the objects/entity types. Further examples of biological database applications are phylogenetic databases, which involve additional neural network-type query and search tools, and protein structure databases, which are primarily focussed on multimedia and representational factors of the data (e.g. [40]). All of these databases can be further categorised into *data type* specific (like GenBank and Swiss-Prot)[1], *species* specific (FlyBase, ACeDB) or *subject matter* specific (REBASE), at least partially requiring horizontal and/or vertical linking of data, addressing not only social issues of interdisciplinary cooperation, but also posing "hard scientific questions" [14, 25]. Macauley [25] defines 'horizontal' as sequence, structure, mapping, position and phenotype and 'vertical' linking as related elements of the same type that pertain to other genes in the same or other organisms. However, one could also interpret horizontal concerning the same components (e.g. DNA with DNA and so forth) and vertical as DNA-RNA-protein etc, akin to a (complicated) "biological OSI model"[2]. Krishnamurthy [22] refers to organising the pathways at "different levels of biological function". On top of the aforementioned divisions, there are so-called primary source databases (TIGR) as well as "boutique collections" to meet specific requests of smaller research communities (such as the bacteriocin database).

Aside from the complication that different databases describe different aspects of the same biological unit, there are definitional problems and a general lack of standardization in nomenclature ([14, 23, 25, 40], among many others.): "anarchy" according to Drysdale [8], although the FlyBase she describes adds to this problem because its creators devised their own keyword system. The Microbial Genome Database elevates this to a feature: the user can create his/her own classification table [38]. There are a few coordinated attempts to unify data formats via Abstract Syntax Notation I [3, 14], the NEXUS file format [26] and the establishment of the Gene Ontology Consortium (GOC)[3]. Criticisms on the latter approach is that it may be criticised for 'dumping' semantic and conceptual disagreements of research groups in the lap of ontologists, there is an apparent lack of cooperation with its implementers and, more importantly, ontology efforts use divergent approaches. There are distinctions from function-based vocabularies (GOC) to descriptive-hierarchical (in taxonomy[4]), where the former devises a vocabulary with e.g. an 'energy generating device' (covering organelles like mitochondria), whereas descriptive ontologies drill down from 'flower' to 'petal' and so forth, alas in some cases introducing new incompatibilities, the very aspect they try to solve.

One can look in more detail into the modelling aspects of the data. For example, compare the common entity type `Person`: in a company or club conceptual database model the `Person` is either `M` (male) or

[1] For a more comprehensive list (not exhaustive), consult Frishman and Infobiogen [14, 17]. Biological databases mentioned in this article are listed at the end after the references.

[2] The OSI [Open System Interface] model is an abstraction of the 7 layers of communication (within ICT): physical – data link – network – transport – session – presentation – application, with horizontal (virtual) communication between e.g. two network layers each residing on a machine and vertical communication occurs between the layers, e.g. data link – network.

[3] More information on the Gene Ontology Consortium is online available via: http://www.geneontology.org/, [12], and for an example of its use with pathway databases, see [22]. There are longer established nomenclature attempts in naming enzymes and coordinated bacterial nomenclature (the latter subject to excessive re-classifications resulting form molecular biology, analogous to the "New Drude" in plant taxonomy [13]).

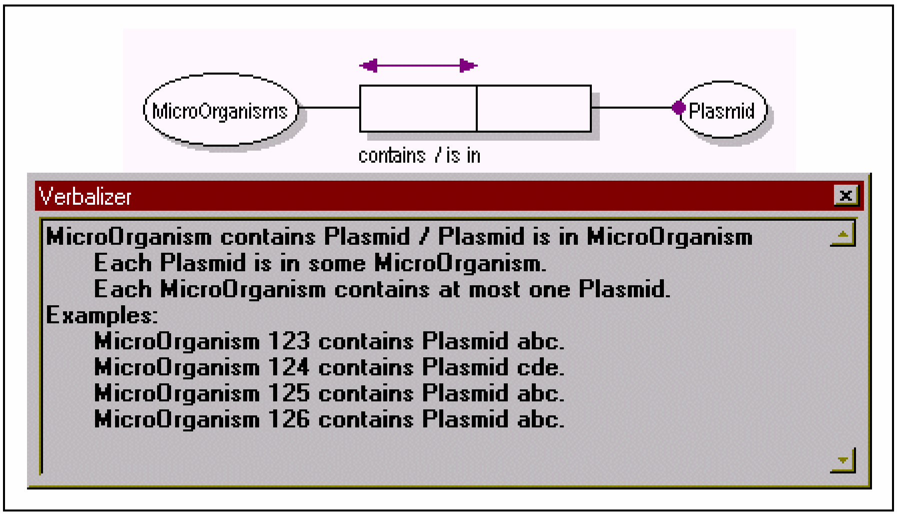
[4] For example the PrometheusDB Project, a collaboration of the Royal Botanic Gardens in Edinburgh and Napier University.

F (female) but not 'mostly M, depending on some factors', whereas a molecule, like a bacteriocin, can be coded 'usually' on plasmids and transposons, though 'rarely' on chromosomal DNA, plus a transposon can insert itself into a plasmid: should one classify the gene location as transposon or plasmid, or both? Further, bacteriocin inhibition can have 'stronger' effects in some environments and 'weaker' when for example the membrane potential is lower [30]. How much weaker or stronger, how to represent gradations, non-discrete data, in relationships? There is no such equivalent in, say, hockey club membership: either you are a member, or you are not. These examples raise only some of the questions about facilitating exceptions and occasional relationships. How ought one to represent environmental conditionality, heterogeneous information and fluctuating data quality? This is a serious design consideration, especially prevalent in attempting to meet requirements of biological science researchers, primarily because this kind of data cannot easily be generalised. Alternatively, for example an address from a company: one knows the components (attributes), *all* of them *and* modelled numerous times before. On the contrary with biological data: there is no substantive legacy to draw from, e.g. a mode of action of a bacteriocin can be 'postulated', i.e. there is a requirement to document a plethora hypotheses by researchers; how can one anticipate attributes and entity types if researchers do not precisely know the parameters? Another issue is the disparity between the need to store extensive knowledge of one bacteriocin, nisin (the most researched bacteriocin), compared with other bacteriocins where hardly any information is known (e.g. reuterin), thereby leaving 95% of the attribute values empty – a waste of resources of the table. However, note that the latter would not occur to such extent if one were to implement an object-oriented database as opposed to a relational database [36], because instances are only created on demand.

In a wider perspective, one can look at conceptual modelling systems, like the Entity Relationship modelling deployed in this project. A disadvantage of ER modelling is that in an early stage decisions have to be made on what will be an entity and what its attribute(s). One cannot know or predict which factor will prove to be important or subject to modification, but ER 'fixes' the diagram and once implemented, is difficult and laborious, if not impossible, to change. This can be partially addressed by resorting to Object Role Modelling (ORM, refer to [15] for an explanation) to reveal intricacies and postpone design details. Further, a limitation of ER is that it does not allow relationships of any arity, whereas ORM does. ORM can include attribute restrictions more clearly, and the use of sample data accompanied with the model (see *Figure 1*) aids domain experts. Halpin [15], North [32] and Ter Hofstede and Proper [35] elaborate further on this aspect and it provides the opportunity to design and implement an ORM-model in either a relational or an object database[5]. One can argue that an iterative process between different conceptual data modelling tools ought not to be necessary: a single conceptual modelling technique should be sufficiently expressive to be able to capture everything[6].

[5] The interested reader may like to read an example of ORM to ER mapping in Halpin ([15], p343-346) and ORM to UML mapping is addressed on pp396-397.

[6] Or at least biological semantics. It falls outside the scope of this research to assess other modelling methods, like graph theory or formal concept analysis, on their suitability. This could be an interesting avenue for further research.



MicroOrganism contains Plasmid / Plasmid is in MicroOrganism
    Each Plasmid is in some MicroOrganism.
    Each MicroOrganism contains at most one Plasmid.
Examples:
    MicroOrganism 123 contains Plasmid abc.
    MicroOrganism 124 contains Plasmid cde.
    MicroOrganism 125 contains Plasmid abc.
    MicroOrganism 126 contains Plasmid abc.

Figure 1. Example of the verbalizer of the fact type between `MicroOrganism` *and* `Plasmid`.

Concerning the pros and cons between ER and Object-Oriented (OO) data modelling. Thierry-Mieg [36] claims

> [r]elational systems are best when the schema is simple, the data is regular and successive queries are independent. Object systems are best when the schema is complex, the data irregular and the queries correlated.

and with OO it is easier to "search the neighbourhood". Although this has not been substantiated by experimental comparative research on biological databases, Uchiyama's [38] MSGD discusses "similarity relationships", Thierry-Mieg [36] addresses "progressively explor[ing] the surrounding area" in relation to the AceDB, and Raguenaud [33] also addresses "localised" searches. This leaves open the question as to what type of database is faster and more flexible when the same data is captured in a OO and in a relational database. Another factor governing the suitability of either ER or OO is the primary requirement for its intended use: the most commonly used methodology in molecular biology is gene comparison, which both ER and OO can facilitate. However, the recent development of metabolic pathway databases try to capture far more complex information than simple gene sequences because of the type of interactions between the molecules (chemical reactions): the objects forming the data are nodes of networks linked by edges representing the chemical reactions [14, 22, 40]. However, others (e.g. [29]) refute the non-suitability of ER[7].

Additionally, and no less important, which model is more understandable for the domain expert to accurately capture the variations in semantics in aiding the iterative analysis process? Again, opinions vary. Bornberg-Bauer and Paton [5] have conducted a limited comparison between ER and OO (using UML) on a theoretical level, though by discussing what *is* possible in biological data modelling, but not what *should be* in order to meet database requirements of biologists. Is one or the other merely the 'lesser of two evils'? On the other hand, it seems that requirements set by the various sub-disciplines of

[7] Another related modelling technique, the object-relational approach, is not further discussed here. BIND [3] and the *Arabidopsis thaliana* database [14] make use of this modelling approach. A hierarchical (tree) model is not discussed, because integrative data does not fit such a modelling approach and graph theory is beyond the scope of the project.

biology are not compatible with one another and that further standardisation in definitions and data formats would be required before the next step towards designing consistent and compatible databases can be taken.
These are some of the serious questions and problems, and as of yet unanswered, which affect the conceptual data modelling of the bacteriocin database.

The more practical problems of conceptual modelling, design and maintenance of biological databases are addressed annually in the January issue of *Nucleic Acid Research* (see e.g. [3, 38]) and in a less fragmented manner by Letovsky [24]. Whilst these provide a topical analysis of a single database, both theoretical and practical, or the problems arising with a few related or similar databases, they do not provide a structured approach in categorizing which type of database faces what kind of problem(s), apart from generalizations on issues of data duplication, redundancy and inconsistency between related databases. These arise, at least partially, because it is very tempting not to link, but to *copy* the few sections of relevance from a primary source database into the communal database. The 'advantage' of copying data is that you can change the data format in whatever way you prefer for your own database, but of course that does not aid data(base) integration. Consult Shoop [34] for a comprehensive discussion on this matter and related integration problems of biological databases. The height of (loose) data(base) integration is the Sequence Retrieval System (SRS) [6, 7], linking 150 databases and more than 42 million unique records of information via one web interface [6].

In summary, the development of sophisticated information systems for the biological sciences is still in its infancy and IT/computing cannot yet live up to requirements, uncertainties, inconsistencies and fragmented nature of data of biology. Biology requires standardization and cooperation between closely related research subjects, but also improved modelling and implementation methods from a computing perspective are required to meet the demands of a (research) society of the future.

3. Modelling the bacteriocin database

The bacteriocin database, which from a computer science perspective functions as a first attempt of its kind to represent biological semantics with the application of food science into a conceptual model, was developed on request for Amalia Scannell, from the Department of Food Science, University College Dublin. Her main requirements were to have an easily accessible, structured and searchable repository for bacteriocin-related data extracted from the vast amount of journal articles she gathered over the years. The iterative development process of the analysis, design and implementation is addressed elsewhere [20]. The database is operational and may be made available to the wider bacteriocin-research community at a later data when it contains a larger volume of data. Here, attention is given to the analysis considerations regarding this data and how to represent this conceptually in a manner that captures the involved semantics within the narrow field from the perspective of a food scientist.

3.1 Bacteria and Bacteriocins

Bacteriocins are compounds similar to antibiotics, inhibiting growth of other, often closely related, bacteria, though unlike antibiotics, they are functionally non-therapeutic so that there is potential to use bacteriocins as a natural ingredient in food produce for food safety and preservation. Most research on the application of (microorganisms producing) bacteriocins is still in the early stages and information in journal publications is scattered around across specialisation boundaries, ranging from food research to genetics journals.

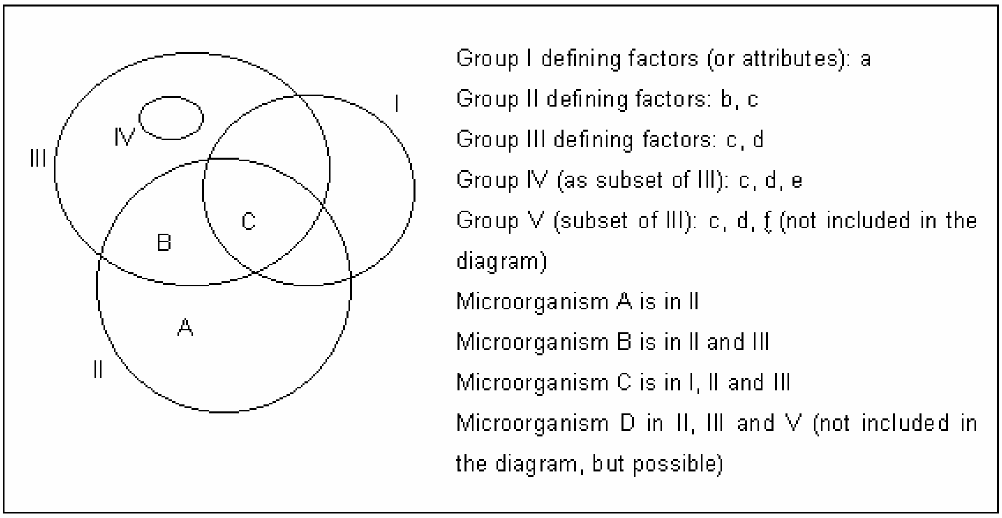
Data about bacteriocins suffers from two problems important for undertaking modelling: the molecular mechanisms of production of, and inhibition by, these peptides are not fully understood [30]) and due to a lack of conformity in the definition, naming, and categorization of these molecules and their corresponding genes, the term bacteriocin covers a range of chemically diverse substances [10, 18]. However, this should not be interpreted as a “we don’t know what we have, nor what it does”, but that over time, types of bacteriocins and their fragile classification is subject to change, and the importance of various parameters (potential attributes of an entity type) may change in emphasis and new relevant factors can emerge. For example the mode of action of a bacteriocin, *how* it targets the inhibition of (predominantly) bacteria (‘often’ by disrupting the membrane potential) and *what* environmental parameters, attributes, are relevant to include (e.g. pH and moisture content). In addition to this, alternative modes of action can be postulated and be confirmed, where both are deemed sufficiently relevant to merit inclusion in the data analysis.

The other main component is how to represent the microorganisms, relevant not only for production and inhibition, but also allowing for possible future expansion of the data model to include more general microbiology data and/or food microbiology-related data. (There does not yet exist a general database containing data on microbiological applications (other than bibliographical repositories)). Compared to plant taxonomy, microbiological nomenclature is straightforward, with Latin names consisting of genus, species, subspecies and optionally a ‘sub-sub’ species and a designate. The designate is a number like ATCC 6633, indicating the origin of the microorganism from some culture collection (a *Bacillus subtilis* from the American Type Culture Collection), or a construct (genetically modified strain). For example *Lactobacillus delbrueckii bulgaricus* LMG 6901 is valid, so is *Lactobacillus helveticus*. Bacteriocins do not inhibit only single (sub)species of bacteria, but may have a more generic “target group” of bacteria [1], addressed in the next section.

3.1.1 Groups of microorganisms

The idea to represent “groups of microorganisms” as a separate entity type `MOGroup`, was removed from a preliminary explorative ER-model because it was not considered essential data in the present context of applied microbiology. However, there appeared to be ample existing published data indicating that a bacteriocin can inhibit a group of microorganisms (e.g. Gram-positive bacteria), meriting the inclusion of “groups” – at least in the data model – as not to address this matter at least in the conceptual model would be a semantic hiatus of the overall subject matter.

What defines “groups of microorganisms”? They are microorganisms that have one or more “aspects” in common. This might be the genus (*Listeria* sp.), isolation source (soil bacteria), growth condition (thermophiles), biochemistry (lactic acid bacteria – LAB), morphology (cocci), Gram staining (positive or negative, depending on the murein content in the cell wall), or other determinants. One can think of conceptualising this as a large entity type, `MicroOrganism`, containing many nested subtypes, that is, if distinctions were this straightforward. However, for three reasons this would not reflect the groups correctly. First, microorganisms are more often than not a member of more than one group (a LAB coccus); second, some of these groups can be subtyped further (streptococci, staphylococci etc.) and, third, there are overlapping traits that define a group. Schematically, this can be represented as sets of microorganisms as depicted in *Figure 2*.



Group I defining factors (or attributes): a
Group II defining factors: b, c
Group III defining factors: c, d
Group IV (as subset of III): c, d, e
Group V (subset of III): c, d, f (not included in the diagram)
Microorganism A is in II
Microorganism B is in II and III
Microorganism C is in I, II and III
Microorganism D in II, III and V (not included in the diagram, but possible)

Figure 2. Schematic representation of groups of microorganisms.

Hence, sets have overlapping as well as distinct identifying attributes (which themselves can be a subtype of another hierarchy). If one were to model an entity type `MicroOrganism` and any/all of its “groups” as subtypes of `MOGroup`, it might obfuscate data, whereas separate entity types could clarify matters while maintaining semantic correctness (though one may wish to pursue this matter further to devise a comprehensive categorisation model for groups of microorganisms). The ‘separate entity types’ chosen here, are greatly simplified from the biological situation, because a) including everything, i.e. trying to solve too large a problem, is not feasible within the limitation of the current research project; b) categorising groups of microorganisms is not a main purpose of this database and c) the customer did not perceive (lack of) categorisation to be a problem in the first place: to her, including groups of microorganisms in the entity type `MicroOrganism` made perfect sense. Note here, that this is not an exercise in trying to create a solution to a non-existing problem. Within the present context of a database model, covering a relatively specialised subject area, the groups categorisation is less important, but not if one were to decide to integrate this database with either primary source databases or something like an “SRS for food science”: in order to keep the conceptual model as flexible as possible one should bear in mind the larger picture.

Considering aforementioned background and modelling aspects, there are three parts to include MOGroup in the conceptual model: 1) optional participation condition and *m:n* multiplicity between MOGroup and MicroOrganism to cover the 'groups'. In theory, this would facilitate an intersection relation containing the identifier of a microorganism (surrogate key IDMO) with a GroupName where multiple IDMOs can be in one group of microorganisms and allow an IDMO to belong to more than one group. 2) A relationship between MOGroup and Inhibits, covering the semantics that more than one bacteriocin can inhibit a group and 3) (following from 2) Inhibits has either a relation with a MicroOrganism or an MOGroup associated with a Bacteriocin, but not both at the same instance (same tuple in the table). Refer to *Figure 3* for the ER representation.

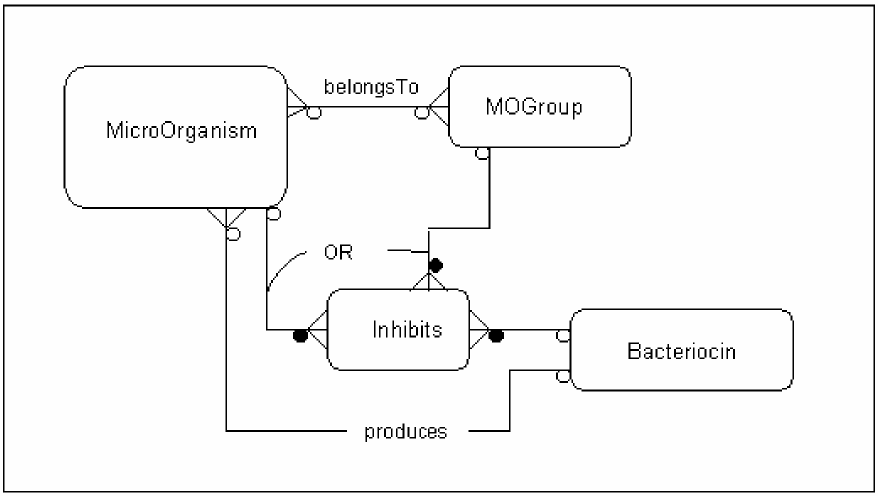


Figure 3. Section of the ER-diagram related to microorganisms, their groups and bacteriocins.

3.1.2 Genetic determinants of bacteriocins in bacteria

The usage of the rather vague term 'genetic determinant' as an alias for the specific genes encoding for the bacteriocins may seem a little odd, but is widely used in bacteriocin-related food science to encompass "some name and location where the genes(s) may reside". Whilst it may be less relevant to be accurate in food science (abstracted away), the details *are* relevant in the sense when one would want to explore the option to take the gene encoding for a bacteriocin and insert it into a target bacterium, an activity more common in the field of bacterial genetics, and if the gene(s) is(are) located in a more stable area, chromosomal DNA versus plasmids and transposons. Plasmids, mobile DNA fragments, can transfer to other bacteria, giving the new host the possibility to produce the same bacteriocin. This process can occur naturally (either conjugative or non-conjugative) as well as via engineering efforts. Transposons are even more mobile than plasmids, in that they can move independently between chromosomal and extra-chromosomal elements. (One could visualise this as a transposon inserting itself into e.g. a plasmid, where it subsequently can 'hitchhike' with that plasmid to wherever the plasmid goes.) These three locations, chromosomal DNA, plasmids and transposons are referred to as 'genetic determinant'[8]. Normally, a genetic determinant does not contain sequences encoding for another bacteriocin, though it is possible. If this genetic determinant contains code for more than one bacteriocin, it is highly unlikely that both are active at the same time; that is, it has never been observed (yet).

ORM enforced stricter description of entity type attributes (than ER), requiring introduction of the intersection relation MOContainsGD between MicroOrganism and GeneticDeterminant: a plasmid may be in more than one microorganism (via horizontal transfer), there may be more than one plasmid in a microorganism, and a particular gene encoding for a bacteriocin can reside on more than one genetic determinant. Bearing in mind these high-level statements, the customer considered it desirable to include "some data on genetics" in the data model.

Basic data include the name of the gene encoding for the bacteriocin, its location (chromosomal DNA, plasmid or transposon) and name of the location, whereas details such as the amount of base pairs, start/stop codon and promoter region are too specific. The gene name is an attribute of a bacteriocin, but where should the location and its name reside? The idea of including a type of genetic determinant, TypeOfGD, as attribute, in the intersection relation MOContainsGD was considered, because the gene can be located on the chromosome of a bacterial strain – in that particular instance – hence requires the information to be stored in MOContainsGD as it is an asset of the microorganism. However, a genetic determinant, say, Tn*5301*, can define the location of the gene coding for the bacteriocin (nisin), where the prefix Tn stands for transposon, thereby the name defining the location – thus an attribute of GeneticDeterminant. To complicate matters a little further, as mentioned a transposon can mobilise and insert itself into a conjugative plasmid; should the location, the value of TypeOfGD, be stored as plasmid or transposon, or both? It will be stored both, where the plasmid and transposon must have different names. Secondly, it is far more widespread that the gene is located somewhere on a mobile DNA fragment, i.e. *not* chromosomal, thus not uniquely linked to a specific microorganism, and TypeOfGD in an entity type GeneticDeterminant can store the value chromosome anyway. This is not ideal, as one can imagine the hypothetical situation where two different strains have the same gene on another locus on their chromosome. See also paragraph 4 and *Figure 6* for the chosen and alternative model on these aspects.

3.2 Food

Categorising and modelling food can be as simple or as complicated as one would like. The simplest is one entity type, Food, with an identifying attribute FoodName. Food subsequently could provide one side of the intersection relation with MicroOrganism to represent bacteria involved in fermentation that also happen to produce a bacteriocin to kill their close relatives in order to achieve a competitive advantage for the resources in the produce (e.g. lactic acid bacteria fermenting milk to produce yoghurt or cheese), and an intersection relation with bacteriocins that can be added to a food product as an

[8] There exist other genetic determinants as phages and insertion elements, but are not considered to play a role in the genetics of bacteriocins.

ingredient. However, this is prone to error not only due to spelling mistakes and (partial) synonyms of food products, but it could not capture the processing involved of the food product (or the product-to-become-ingredient) either, e.g. freeze-dried or comminuted meat. Alternatively, one could create an extensive vocabulary, or even an ontology, specifying how to construct and adequately represent the semantics of food (products, groups, types, processing, ingredients) via `is-a` and `has-a` relationships (generalisations and aggregations). However, one should not get carried away merely to satisfy a computer scientist's desire for a proper structure. One of the requirements by the customer was to be able to 'drill down', zoom in, from high-level food groups down to a particular product, alike searching the neighbourhood as mentioned in §2, e.g. "meat – pork – Frankfurter sausage". This can be met by introducing a straightforward cascade of entity types `Food`, `FoodSubGroup` and `FoodGroup`, with a separate entity type `Processing` to store aspects of the food production process. Again, as with the microorganisms, this is quite a simplification, where it most certainly is arguable if it is a proper representation of "true semantics", to which I will return in the discussion section.

3.3 Generalisation to compounds produced by microorganisms used in the food production process

What §3.1 and 3.2 address are examples of microorganisms producing a compound, where the compound can be added as an ingredient to a food product, or the whole microorganism that produces the desired compound grows in the food product as part of a fermentation process. The compound in this example database is a bacteriocin, but this principle occurs more often in the food industry. One may recollect cheap fruit yoghurts with slogans of "contains no artificial flavours" as a marketing ploy for the 'all natural' trend in the 1990s. The bacteria used for this fermentation are genetically modified in such a manner to include the capacity to produce the flavour that would otherwise have been added to yoghurt. Without digressing fully into the area of food biotechnology, microorganisms and their produce are deployed for a wide variety of production processes. Then, to facilitate maximum flexibility of the conceptual model, would it not be better to include, say, an entity type `ProduceByMO` with `Bacteriocin` as subtype? The produce ranges from simple peptides to enzymes, sugars to polysaccharides, and aroma components, alcohols, lipids etc. In other words: to truly reflect the biological components, it would require a complete categorisation of molecules, greatly complicating the model. *Figure 4* shows a biologically more inclusive categorisation related to peptides and proteins, which begins to address classification of biological components as can be defined within an ontology, but is only a very small fraction of the wide range of compounds microorganisms are capable of producing.

A simplified concept with a direct relationship without 'higher' hierarchies was chosen, which still leaves ample room for future expansion of the model and focuses more on the *application* of science than on biochemistry. New products can be added to the conceptual model as new entity types, analogous to the bacteriocins. Although not entirely inclusive, it *does* represent the semantics of food microbiology correctly in the conceptual model.

As an aside, note that this is a different consideration than the `GeneticDeterminant`-associated data in *Figure 6*: those details are abstracted away because it is not deemed relevant, whereas in this case, it is relevant with respect to future additions to the database within the realms of food science.

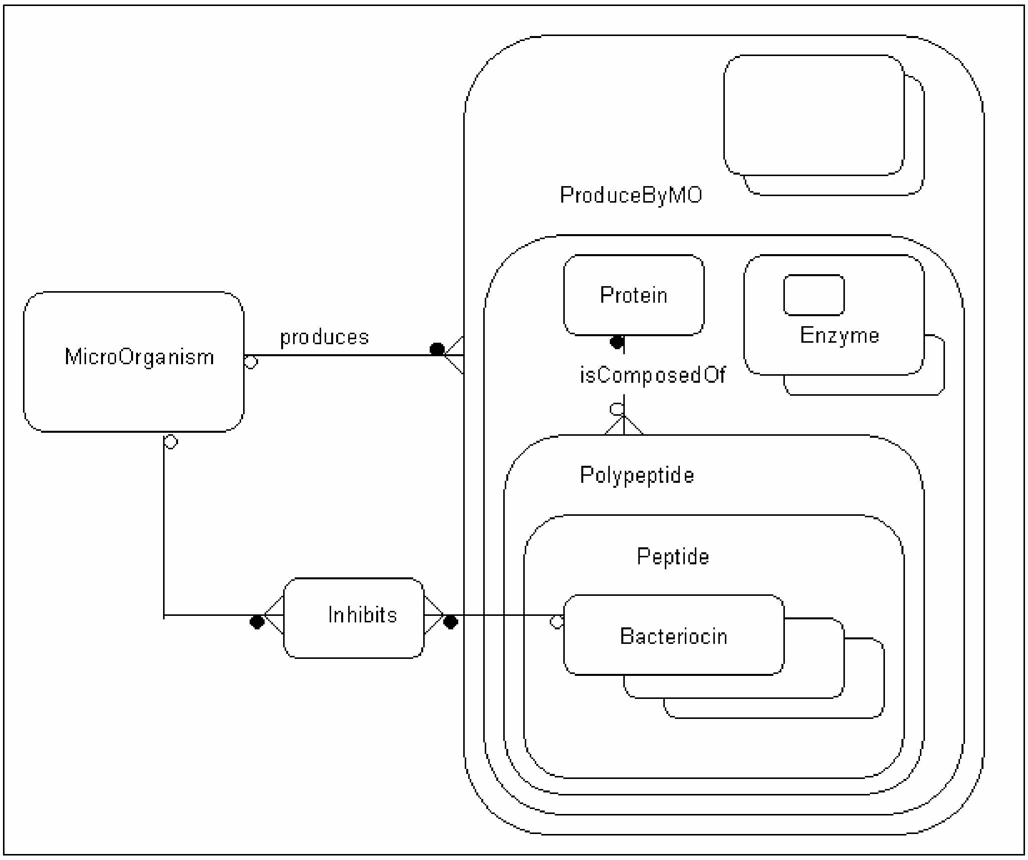


Figure 4. Example of an untenable partial categorisation structure of produced compounds.

3.4 Other modelling aspects

Although the primary for this database concerns are bacteriocins, microorganisms and food, several other facets were included for extra convenience. These include commercialised bacteriocins, disease-causing microorganisms (Bad Bug Book [39]), purchase options from the ATCC culture collection, and data on starter culture usage of the bacteriocin-producing microorganisms. The latter three are straightforward entity types with a relationship to the entity type `MicroOrganism`, which in the logical design can be modelled with a foreign key of the microorganism identifier `IDMO`. Essentially, any microbiology feature could have been, and can be, introduced in such a manner.

One particular aspect of the logical model requires attention, which is the design of the `Inhibits` relationship from a bacteriocin to either an instance of a `MicroOrganism` or a `MOGroup`. The conceptual model represents this as a primary and an alternate key[9]:

```
                  ◄──────────────────────────────────────────────►
Inhibits(IDMO, BactName, TargetProduces, MOGroupName, Publications)
```

However, it was not possible to code more than one key in SQL, have it accepted by InfoMaker and provide data on either one of the key: the software cannot anticipate when a user wants to provide data on the (`IDMO, BactName, TargetProduces`) in one instance and (`BactName, TargetProduces, MOGroupName`) in the other. In theory, there are four options for a workaround in order to meet the customer's requirement to be able to retrieve data when answering the query "*bacteriocin x inhibits y*": 1) use the microorganism table to store the groups of microorganisms, where each group, like a single microorganism, is stored in a separate tuple. 2) Make a separate entity type `MOGroup`, create a surrogate key in `Inhibits` relation and make only `BactName` and `TargetProduces` as not null and explain on the data entry form that users should enter either an `IDMO` or a `MOGroup`, but not both. 3) As in 2, but include a procedure, that if `IDMO` is an empty string, then `MOGroup` must be non-empty and vice versa, and that one cannot save a record that have both cells as an empty string or 4) create two inhibits relations, one `InhibitsMO` and a second `InhibitsMOGroup` and devise application support to notify the user of the difference. Options 2, 3 and 4 require additional entity types, tables and for option 3 a procedural constraint to implement plus additional support in the application (data entry form) to inform the user not to enter data in both cells and not to leave them empty. Two notes on the latter, is that first, one ought to avoid reliance on implementation to accommodate aspects that should have been addressed in the conceptual and computational model and second, it is a known fact that users *do* make mistakes, therefore option 2 can be ruled out. Option 4 is suboptimal as it creates an artificial separation that does not exist from the perspective of a microbiologist; furthermore, it relies on additional application support. As it is possible that any future user group will not work with the database on a daily basis, it cannot be expected they are fully conversant with this limitation and option 1 would appear to be the most straightforward workaround and the least prone to errors or inconsistencies. However, it makes the conceptual model and the design inflexible, for example if a future requirement arises from the customer to add groups of microorganisms for other purposes. However, this may be unlikely because the entity type `MOGroup` was dropped from the preliminary model, as it was considered non-essential (irrelevant). With the reservation that the design does not fully capture the nature of the microorganism groups, the `Inhibits` relation was designed in line with option 1.

Appendix A contains the complete ER diagram.

[9] In line with devising a 'minimal' key, the attribute `TargetProduces` is strictly not necessary (`TargetProduces` captures bacteriocin produced `by` or is active `against`). However, in the design, logical model, the relationships `Inhibits` and `produces` are combined into one table definition, `BactAndMO`, where `TargetProduces` is not allowed `null`, therefore included in the key.

4. Discussion

The centrality of MicroOrganism in the conceptual model is based on the Tree of Life design (see *Figure 5*) to allow flexibility to 'plug-in' any microbiology-related data in the bacteriocin conceptual model and database, as well as linking it to other (Internet) databases. The conceptual model and design still allows future expansion to connect with, for example, data on antibiotics, fermentations, metabolism and so forth.

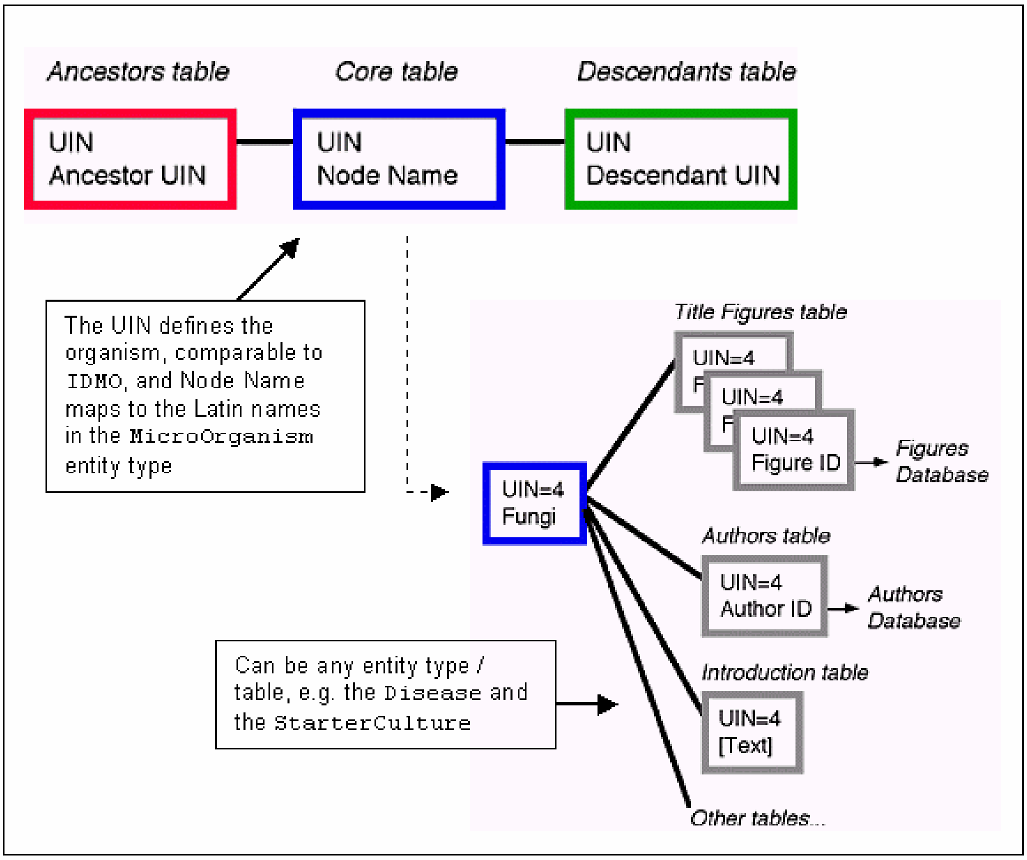


Figure 5. The picture represents the core Tree of Life data model and how this maps to the conceptual model of the bacteriocin database. Source: [27].

The entity type Bacteriocin is similar to the IDMO of the MicroOrganism table, with its related entity types ModeOfAction, BacteriocinType and CommercialProduct. However, Bacteriocin is more restricted than the MicroOrganism table in that it is further down in the specification and detail hierarchy. Ideally, plasmids would be designed in an analogous fashion, but this was complicated by the introduction of the theoretical possibility that the gene encoding for a bacteriocin might reside on chromosomal DNA. From the author's perspective of molecular microbiology, a preference existed for a semantically more comprehensive representation as in *Figure 6b*, but the customer interpreted this from the angle of food science, which abstracts away these details of genetics. The section of the model surrounding GeneticDeterminant (*Figure 6a*) captures what is necessary from a food scientist's perspective, but thereby losing possible future expansion in functionality of the system. This serves as an interesting example of the difficulties facing data modellers of biological databases: closely related disciplines interpret subject matter "roughly the same", yet with minor, though significant, differences in emphases.

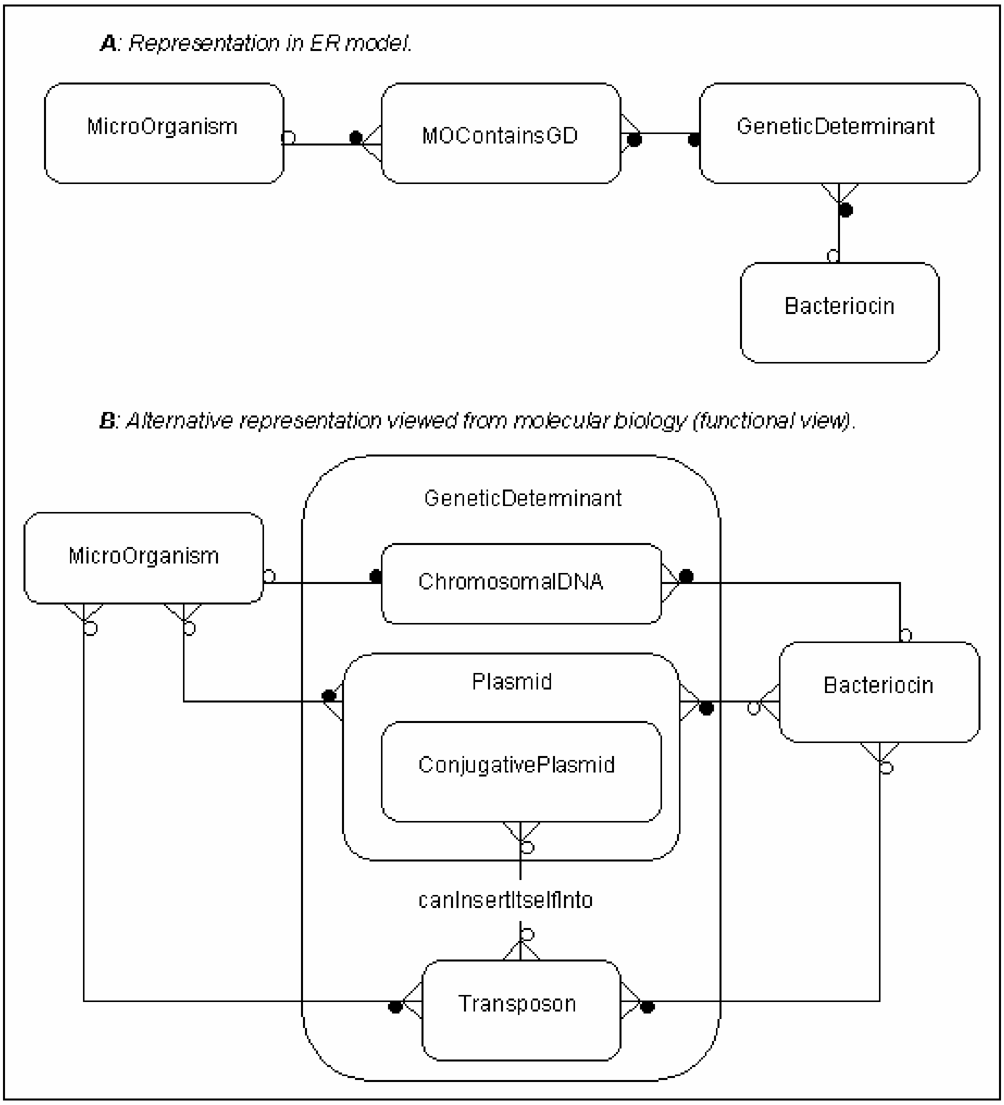


Figure 6. Modelling options surrounding the `GeneticDeterminant`*. The m:n relations between* `Plasmid` *and* `Bacteriocin` *and* `Transposon` *and* `Bacteriocin` *are unlikely, but theoretically not impossible. Further, these are not the only existing genetic determinants, but I have not come across genes for bacteriocins coded on e.g. bacteriophages or IS elements.*

Equally possible, though not elaborated on here, would have been, like the molecular biology example, a slightly higher emphasis on protein composition and structure and/or the metabolism of bacteriocin production, or modelling the environmental factors affecting bacteriocin production and inhibition, thus further and more detailed specifications surrounding `StarterCulture` and `ModeOfAction`. For example, the stability of a bacteriocin is not only affected by pH and depending on temperature, but responds differently to various proteolytic enzymes [4], can have different optima for bacterial growth compared to ideal circumstances for bacteriocin production [19] or the effectiveness of production can be a trade-off when varying several parameters [31]. Considering bacteriocin metabolisms, McAuliffe [30] could serve as a good start to explore these kinds of factors (entity types, attributes, sort of values) one may expect for the proteins, composition and production of bacteriocins. For general problems and complexities of representing metabolisms in databases, refer to e.g. [22].

Aside from emphases and in/exclusion of details as described in the previous paragraph, is that ontology definitions were not adhered to for several reasons. First, the functional approach of the GOC was considered too abstract by the customer, as she preferred recognisable entity type and attribute names that match her view of the problem domain. Secondly, only sections of the persistent data could have benefited from a separate newly devised ontology (the food section). Third, the overall contents of the data are 'cross-boundary' (integrated food science), covering organism, molecule, type (food, food groups etc.) and parameter structures, hence would have required an unsatisfactory attempt to integrate incompatible ontologies. Even when restricting ourselves to a conceptual model and including all existing categorisations as suggested in §3, this would make the bacteriocin database unreasonably extensive if it were to include a full categorisation and model including molecular compounds (to cater for the microbiological produce), a complete structure of food *and* covering the whole genetics field. The majority of these concepts are not used with the same rigor as in the 'core' sciences. Even these basic biological science disciplines do not combine categorisations to such an extent and are still in the stages of devising their own data structures. Hence, considering these facets, attempting to construct a conceptual model for an applied science that encompasses data from several specialisations might well be alike "picking data of one's own preference", thereby acknowledging that this database model does not aid in solving any of the nomenclature problems. However, and this is an important point: *it does meet the stated customer's requirements*. This notion can easily lead to a discussion of "what is more right: a computational categorisation representing data semantics to a fuller and more comprehensive extent and following from this, highlighting ways to generate new information, *or* that the customer has a useful tool to organise the scientific data related to bacteriocins to satisfaction?" which I leave to the reader to answer for him/herself.

Clarification of the particular area in the conceptual model, together with the semantics of microorganisms and bacteriocins, has primarily arisen from ORM (experiments not included in this report). ORM forces one not only to *think* about an attribute as in ER modelling, but also actually to *write down* (by drawing and providing example data for fact types) and specify the relationship between an entity type and its attribute(s), thereby making explicit what was implicitly assumed. This can go unnoticed in an ER model: the modeller and domain expert each may assume something without specifically communicating it, thus the possible difference in interpretation remains unnoticed until implementation. By experimenting with ORM in addition to ER modelling, several of these variations in thought processes surfaced from the modelling exercise and the fact that the customer understood the model descriptions with examples and verbalizers better, therefore these assumption could be resolved in the modelling stage, saving considerable time.

However, one aspect that was conveniently set aside and not suggested to the customer as a practical option during the development process (the theoretical side was touched upon briefly), was the effectiveness of a bacteriocin inhibition of a microorganism: sometimes they are a nuisance for a particular strain, whereas other bacteriocins, or the same but acting on other bacteria or in another environment, can kill microorganisms. This inhibition depends on environmental factors and is, as well as bacteriocin production, regularly expressed in AU (e.g. ml^{-1}), where AU means Arbitrary Units, which, as the name suggest, varies from publication to publication. Inhibition is captured in the present model and implementation as a yes/no `Inhibits` relationship, but ideally, one would want to include the *severity* of inhibition. Xenarios and Eisenberg [41] discuss the inclusion of "confidence levels" in the DIP database[10], alike the 'strength' of a relationship, and Bornberg-Bauer and Paton [5] have an extra attribute "accuracy" in the PRINTS-S database[11], that equally could be transformed to an attribute in the `Inhibits` relation with either a categorisation of, say, a 'nuisance', 'moderate inhibition', 'strong' and 'kill', or a percentage between 0 and 100. Alternatively, one could spend time trying to integrate a fuzzy logic algorithm or a neural network classification to accommodate the gradations in inhibition of microorganisms. With fuzzy logic, one can think of the fuzzy sets in a range of 0 to maximum inhibition based on the cell count of the inhibited bacterium and a rule-based system to accommodate the relevant environmental factors. A neural network type of classification would take the values of the environmental parameters and then calculate the class in the output values with an appropriate scoring system (e.g. a radial basis function network to deal with the complex data). Refer to e.g. [16] for an explanation of the technologies. Theses approaches present some fascinating topics for further research, because the inhibitory activity can be additionally dependent on environmental factors.

Further, the difficulty of the `Inhibits` relation has not been addressed to satisfaction because of the misuse of `MOType` and `MicroOrganism` to store data on groups of microorganisms. In this instance, it can be argued this is primarily a problem of design, hence SQL and SQLAnywhere limitations of not allowing overlapping primary and alternate keys (the database was implemented with InfoMaker). On the other hand, a request to expect software to be able to guess what the user wants one time or the other is unrealistic. The complaint of "impossible requirements" requested by the biological sciences has been mentioned in various publications (e.g. [28]), although note that it is an impossible requirement *viewed from a computing perspective*, because semantically, a single bacterium is different from (has different attributes than) a group of bacteria. On the other hand, the customer, from her *food science perspective*, interpreted the matter as 'bacteriocin x inhibits y', and as long as the database answers what is inhibited, regardless if it is a single microorganism or a group, it meets her requirement, thus placing groups of microorganisms in the `MicroOrganism` table in the implementation was deemed no problem whatsoever. Again, this poses the question if computing scientists introduce at least part of the claim of impossible requirements themselves because they are more focussed on ontology, classifications and rigid semantics than the more fluid concepts and functionalities in biology.

[10] DIP – Database of Interacting Proteins – is online available at http://dip.doe-mbi.ucla.edu.

[11] `accuracy` as an attribute of the `assignment` relationship between entity types `Sequence` and `Fingerprint` in the ER diagram of the PRINTS-S database, which is database of protein family fingerprints, online available: http://www.bioinf.man.ac.uk/dbbrowser/PRINTS/. Refer to [2] for more information.

A few notes on finding and adding data to the database is relevant. Information on bacteriocins is scattered around journals form different disciplines, but with access to multiple e-journals from several universities[12], this is merely time consuming and not a major problem. However, there is an interesting point to explore. Whilst microbiology articles are useful and readily interpretable information sources to researchers in that discipline[13], the information is not represented in a clear fashion and structure that allows one to take an article and 'step' through the database tables to add the published information. The author experimented with building use cases and flow charts to emulate the data entry process, but there are as many approaches as there are articles, which have a negative impact on usability, no matter how sophisticated the interface is designed.

A further concern related to availability of data, was demonstrated when populating the `ToLLink` table, referencing related Tree of Life web pages (later replaced by `ATCCLink`, for reasons addressed here). This table intended only to serve as example to provide easy access to other Internet-based databases with further information on microorganisms. The overlap between the bacteriocin-producing microorganisms and the Tree of Life was small; therefore, an extensive search for other microbiology databases containing sufficient overlap was conducted. Despite that there are over 511 biology databases[14] and dozens related to microorganisms, there is no single database that simply contains all known/researched microorganisms with which to produce a listing with their morphology, environmental parameters and so forth. The MicrobeLibrary provides some of the desired information, though unfortunately tailored for a high school classroom audience and is in a germinal stage. DSMZ [9] is 'close' with over 6000 microorganisms, but only provides information for purchase. TIGR, the leading source on bacterial genomics, has a mere 26 completed and 50 ongoing bacteria and GOLD documents 432 bacteria[15], but with many dead links to 'further information'. There exist other microbiology databases by culture collections[16] (similar to DSMZ) and some are topic specific like the Bad Bug Book [39][17], as used in the `Disease` table. Therefore, the aim of linking a communal database to a *general* repository of microorganisms is still in the future at the time of conducting this research.

From the customer's perspective, the primary importance for the near future is filling the database with content, as the current features and information utility meet the customer's needs. However, a future 'ideal solution', which is well outside the field of database development, would be a database system she can query in natural language instead of having to resort to InfoMaker's `ToolBox` entering only keywords.

[12] The Open University UK, and the University of Limerick and University College Dublin both in Ireland.

[13] E.g. Keet [21].

[14] Catalogued by Infobiogen [17].

[15] Genomes OnLine Database completed 110 bacteria and has 322 bacteria ongoing (Date accessed: 20-7-2003).

[16] A listing is accessible via http://wdcm.nig.ac.jp/DOC/menu3.xml; chosen was to link the American Type Culture Collection, because the UCD Department of Food Science (i.e. the customer) regularly uses and orders bacteria from this company.

[17] Another database similar to the BBB is the "Gateway to Food Safety Information" (accessible via http://www.ces.ncsu.edu/depts/foodsci/agentinfo/org/staph.html), maintained by the North Carolina State University, but is not as famous as the BBB and not an official USFDA site (as is the BBB).

From the author's point of view, the fact that the customer's statement of requirements was met does not imply a well-designed database, particularly with respect to represented semantics, like the problem surrounding `MOGroup`, and to what the author considers 'gaps' in interesting data on the molecular structure of bacteriocins and the details of genetic determinants. Further, although this author is convinced the database will be useful for the bacteriocin research community, aspirations were higher in that the likelihood of database linking and/or integration might/could have been within reach, which is more difficult to achieve because the present model is tailored for the particular task (albeit with flexibility to expand). On the other hand, modelling of biological data is difficult primarily because of gradations (in inhibition, mode of action), endless parameters, varying levels of extant information (on specific bacteriocins, genes), and lack of standardisation. The contents of this bacteriocin database cut through these distinct sub-disciplines in life science research to achieve integration, which is, in hindsight, an ambitious goal at the current time.

On the database development process itself, an iterative process was deployed. This development process is exceedingly suitable for bioinformatics, because one of the obstacles is that at the beginning it is often not known by both the modeller *and* domain expert what parameters *exactly* are relevant to capture and sufficiently abundant to include.

It was therefore some consolation that the literature research revealed that even bioinformatics experts struggle with the very same problems encountered in this research project, but it is also exactly these complications that make the combination of computing with life sciences very fascinating and an exciting field where there is ample room for promising further research.

5. Conclusions

A main aim of this research project was the consideration of how modelling aspects from an integrated (food) science perspective contribute to bioinformatics research. Capturing the subject domain semantics of an applied bioscience faces different problems compared to conceptual modelling for the 'core' life sciences, as the former requires an emphasis on practical solutions conceptually representing the integration of various fields, whereas the latter stresses conceptual and ontological "all-inclusive" models within their primary specialisations such as biochemistry and genetics.

Although the field of (micro)biology, and bacteriocins in particular, is a *tabula rasa* in modelling, design and database implementation, literature research revealed interesting trends in both modelling applied to the area of bioinformatics and modelling methods themselves, which contributed positively towards creation of a relatively flexible conceptual model. The principal factors contributing to this is the centrality of the `MicroOrganism` entity type and its subsequently connected types, the use of ORM in addition to ER modelling and the database subject knowledge of the author. All of the customer's data requirements could be modelled, designed and implemented. There were design and implementation problems related to structuring the overlapping sets/attributes of a `MOGroup`, but this was not a specific requirement from the customer as such but a computing interpretation desiring semantic correctness.

Due to the nature of the database topic, several aspects can be further investigated, aside from filling the database with more bacteriocin-related content (pursued at the time of writing). Relatively simple improvements are:

- Addition of new entity types such as antibiotics and other biosynthesized compounds and attributes (e.g. more data related to plasmids);
- Expanding the database in the direction of a 'general' microbiology database (including yeast, archae and fungi) where bacteriocins are only one aspect of (the use of) microorganisms;
- Include record ownership as an attribute per table.

This can then be made available to the wider research community, preferably via a web interface. With an expanding database, it may be advisable to separate literature references into a connected database of bibliographical data and ensuring compatibility with primary biological databases.

It will be challenging to attempt to model the groups of microorganisms properly, represent gradations in inhibition of microorganisms and, together with the mode of action of bacteriocins, the relevant environmental factors including scrutiny of degrees of certainty of the published bacteriocin-research information. This has the potential of creating new knowledge with the aid of computing, as opposed to merely structuring the extant information of bacteriocins.

At the time of conducting this research, full database connectivity and integration of this bacteriocin database with primary biological databases and other communal databases related to (micro)biology and biochemistry are not feasible. The wider database-focussed bioinformatics community will need to come to an agreement with life science researchers on standardizing computing-suitable structures, nomenclatures and conceptual modelling approaches based on preferably one type of ontology, or at least several that can communicate with one another without duplication, inconsistencies and 'translational' problems between as of yet incompatible models, designs and implementations. On top of this comes usability, reminding ourselves that *any* successful bioinformatics implementation should not only overcome aforementioned technical hurdles, but also that it actually will aid the life- and applied science researchers and benefit their work of unravelling the mysteries in biology and contribute to the application of this knowledge.

Acknowledgements

The author wishes to thank the customer of the bacteriocin database, Amalia Scannell (Department of Food Science, Faculty of Agriculture, University College, Dublin), for her cooperation and enthusiasm for the project and Trevor Paterson for providing textual and grammatical improvements to the manuscript, which made it all more readable.

References

[1] Ahern, M., Verschueren, S. and Van Sinderen, D., (2003), 'Isolation and characterization of a novel bacteriocin produced by *Bacillus thuringiensis* strain B439'. *FEMS Microbiology Letters*, 220(1), 127-131.

[2] Attwood, T.K., Croning, M.D.R., Flower, D.R., Lewis, A.P., Mabey, J.E., Scordis, P., Selley, J.N. and Wright, W., (2000), 'PRINTS-S: the database formerly known as PRINTS'. *Nucleic Acids Research*, 28(1), 225-227.

[3] Bader, G.D., Donaldson, I., Wolting, C., Ouellette, B.F.F., Pawson, T. and Hogue, C.W.V., (2001), 'BIND — The Biomolecular Interaction Network Database'. *Nucleic Acids Research*, 29(1), 242-245.

[4] Bizani, D and Brandelli, A., (2002), 'Characterization of a bacteriocin produced by a newly isolated *Bacillus sp*. Strain 8 A'. *Journal of Applied Microbiology*, 93(4), 512-520.

[5] Bornberg-Bauer, E. and Paton, N.W., (2002), 'Conceptual data modelling for bioinformatics', *Briefings in Bioinformatics*, 3(2), 166–180.

[6] Brooksbank, C., Camon, E., Harris, M.A., Magrane, M., Martin, M.J., Mulder, N., O'Donovan, C., Parkinson, H., Tuli, M.A., Apweiler, R., Birney, E., Brazma, A., Henrick, K., Lopez, R., Stoesser, G., Stoehr, P. and Cameron, G., (2003), 'The European Bioinformatics Institute's data resources'. *Nucleic Acids Research*, 31(1), 43-50.

[7] Carter, P., Coupaye, T., Kreil, D.P. and Etzold, T., (1999), 'SRS – Sequence Retrieval System'. In: *Bioinformatics – databases and systems*. Letovsky, S.L. (ed.). Massachusetts: Kluwer Academic Publishers. pp 213-231.

[8] Drysdale, R., (2001), 'Phenotypic data in FlyBase'. *Briefings in Bioinformatics*, 2(1), 68-80.

[9] DSMZ, Deutsche Sammlung von Mikroorganismen und Zellkulturen, (2002), *Bacterial Nomenclature Up-to-date, November 2002*. http://www.dsmz.de/bactnom/bactname.htm. Date accessed: 1-3-2003.

[10] Ennahar, S., Sashihara, T., Sonomoto, K, and Ishizaki, A., (2000), 'Class IIa bacteriocins: biosynthesis, structure and activity'. *FEMS Microbiology Reviews*, 24(1), 85-106.

[11] Gene Ontology Consortium. http://www.geneontology.org/. Date accessed: 12-6-2003.

[12] Gene Ontology Consortium, (2001), 'Creating the Gene Ontology Resource: design and implementation'. *Genome Research*, 11(8), 1425-1433.

[13] Graham, M., Watson, M.F. and Kennedy, J.B., (2003), ‘Novel visualisation techniques for working with multiple, overlapping classification hierarchies’. *Taxon*, 51, 351-358.

[14] Frishman, D., Heurmann, K., Lesk, A. and Mewes, H.-W., (1998), ‘Comprehensive, comprehensible, distributed and intelligent databases: current status’. *Bioinformatics*, 14(7), 551-561.

[15] Halpin, T., (2001), *Information Modeling and Relational Databases*. San Francisco: Morgan Kaufmann Publishers. 761p.

[16] Hopgood, A.A., (2001), *Intelligent systems for engineers and scientists*. Boca Raton, Florida: CRC Press, 2nd ed. 467p.

[17] Infobiogen. *DBCatalog - Database of Databases*. http://www.infobiogen.fr/services/dbcat/. Date accessed: 20-7-2003.

[18] Jack, R.W., Tagg, J.R. and Ray, B., (1995), [Abstract of ‘Bacteriocins of gram-positive bacteria’. *Microbiological Reviews*, 59(2), 171-200], [Electronic]. Available via: http://www.sciencedirect.com/. Date accessed: 23-7-2003.

[19] Juarez Tomas, M.S., Bru, E., Wiese, B., de Ruiz Holgado, A.A.P. and Nader-Marcias, M.E., (2002), ‘Influence of pH, temperature and culture media on the growth and bacteriocin production by vaginal *Lactobacillus salivarius* CRL 1328’. *Journal of Applied Microbiology*, 93(4), 714-725.

[20] Keet, C.M., (2003), ‘The use of bacteria and bacteriocins in the food industry – modelled and documented in a relational database’. BSc Final Year Project, Department of Technology and Department of Computing, Open University, UK. 149p.

[21] Keet, C.M., (1998), ‘Effect of maize rhizosphere on degradation of 3-chlorobenzoate by *Pseudomonas* B13 or *Alcaligenes* L6’. MSc Dissertation, Department of Microbiology, Wageningen Agricultural University, the Netherlands. 106p.

[22] Krishnamurthy, L., Nadeau, J., Ozsoyoglu, G., Ozsoyoglu, M., Schaeffer, G., Tasan, M. and Xu, W. (2003), ‘Pathways database system: an integrated system for biological pathways’. *Bioinformatics*, 19(8), 930-937.

[23] Laser, U., Lehrach, H. and Roest Crollius, H., (1998), ‘Issues in developing integrated genomic databases and application to the human X chromosome’. *Bioinformatics*, 14(7), 583-90.

[24] Letovsky, S.L. (ed.), (1999), *Bioinformatics – databases and systems*. Massachusetts: Kluwer Academic Publishers.

[25] Macauley, J., Wang, H. and Goodman, N., (1998), 'A model system for studying the integration of molecular biology databases'. *Bioinformatics*, 14(7), 575-582.

[26] Maddison, D.R., Swofford, D.L and Maddison, W.P., (1997), 'NEXUS: an extensible file format for systematic information'. *Systems Biology*, 46(4), 59-621.

[27] Maddison, D.R., (2001), *The Tree of Life Project: Creating an Open Phylogenetic Database of Biodiversity Information*. http://tolweb.org/tree/home.pages/nsfproject.html. Date accessed: 23-2-2003.

[28] Marcotte, E.M. and Date, S.V., (2001), 'Exploiting big biology: Integrating large-scale biological data for function inference'. *Briefings in Bioinformatics*, 2(4), 363-374.

[29] Markowitz, V.M., Chen, I.A., Kosky, A.S. and Szeto, E., (1999), 'OPM: Object-Protocol Model Data Management tools '97'. In: *Bioinformatics – databases and systems*. Letovsky, S.L. (ed.). Massachusetts: Kluwer Academic Publishers. pp 187-199.

[30] McAuliffe, O., Ross, R.P. and Hill, C., (2001), 'Lantibiotics: structure, biosynthesis and mode of action'. *FEMS Microbiology Reviews*, 25(3), 285-308.

[31] Nel, H.A., Bauer, R., Vandamme, E.J. and Dicks, L.M.T., (2001), 'Growth optimization of *Pediococcus damnosus* NCFB 1832 and the influence of pH and nutrients on the production of pediocin PD-1'. *Journal of Applied Microbiology*, 91(6), 1131-1139.

[32] North, K., (1999), 'Modeling, data semantics and natural language'. *New Architect*, 7 [Electronic]. http://www.webtechniques.com/archives/1999/07/data/. Date accessed: 27-4-2003.

[33] Raguenaud, C., (2002), *Managing complex taxonomic data in an object-oriented database*. PhD Thesis, Napier University, Edinburgh. 196p.

[34] Shoop, E., Silverstein, K.A.T., Johnson, J.E. and Retzel, E.F., (2001), 'MetaFam: a unified classification of protein families. II. Schema and query capabilities'. *Bioinformatics*, 17(3), 262-271.

[35] Ter Hofstede, A.H.M. and Proper, H.A., (1998), 'How to formalize it? Formalization principles for information systems development methods'. *Information and Software Technology*, 40(10), 519-540.

[36] Thierry-Mieg, J., Thierry-Mieg, D. and Stein, L., (1999), 'ACEDB: The ACE Database Manager'. In: *Bioinformatics – databases and systems*. Letovsky, S.L. (ed.). Massachusetts: Kluwer Academic Publishers. 265-278.

[37] Uberbacher, E., *Computing the Genome*. http://www.ornl.gov/ORNLReview/v30n3-4/genome.htm. Date Accessed: 24-8-2002.

[38] Uchiyama, I., (2003), ’MBGD: microbial genome database for comparative analysis’. *Nucleic Acids Research*, 31(1), 58-62.

[39] US Food & Drug Administration, (1992), *Bad Bug Book*. Online version available at: http://www.cfsan.fda.gov/~mow/intro.html.

[40] Wittig, U. and De Beuckelaer, A., (2001), ‘Analysis and comparison of metabolic pathway databases’. *Briefings in Bioinformatics*, 2(2), 126-142.

[41] Xenarios, I. and Eisenberg, D., (2001), ‘Protein interactions databases’. *Current Opinion in Biotechnology*, 12, 334-339.

Index to consulted Internet biological databases

ACeDB – A *C. elegans* DataBase (genome project): http://www.acedb.org

ATCC – American Type Culture Collection: www.atcc.org

BBB – Bad Bug Book: http://www.cfsan.fda.gov/~mow/intro.html

BIND – Biomolecular Interaction Network Database: http://www.bind.ca/

DIP – Database of Interacting Proteins: http://dip.doe-mbi.ucla.edu

DBCatalog: http://www.infobiogen.fr/services/dbcat/

DSMZ – Deutsche Sammlung von Mikrobiologische Zellkulturen: http://www.dsmz.de

Entrez-PubMed: http://www.ncbi.nlm.nih.gov/entrez/

FlyBase – Drosophila genome: http://flybase.bio.indiana.edu/

Gateway to Food Safety Information:
http://www.ces.ncsu.edu/depts/foodsci/agentinfo/org/staph.html

GenBank: http://www.psc.edu/general/software/packages/genbank/genbank.html

GOLD – Genomes OnLine Database: http://wit.integratedgenomics.com/GOLD/

IntEnz – Enzyme database: http://www.ebi.ac.uk/IntEnz/

InterPro – Proteins: http://www.ebi.ac.uk/interpro/index.html

MicrobeLibrary: http://www.microbelibrary.org/

MBGD – MicroBial Genome Database: http://mbgd.genome.ad.jp/

PRINTS-S – Protein motif fingerprint database: http://www.bioinf.man.ac.uk/dbbrowser/PRINTS/

PrometheusDB: www.prometheusdb.org

REBASE – Restriction Enzyme database: http://rebase.neb.com/rebase/rebase.html

ScienceDirect: http://www.sciencedirect.com

SRS – Sequence Retrieval System: http://srs-mips.gsf.de / http://srs.ebi.ac.uk/

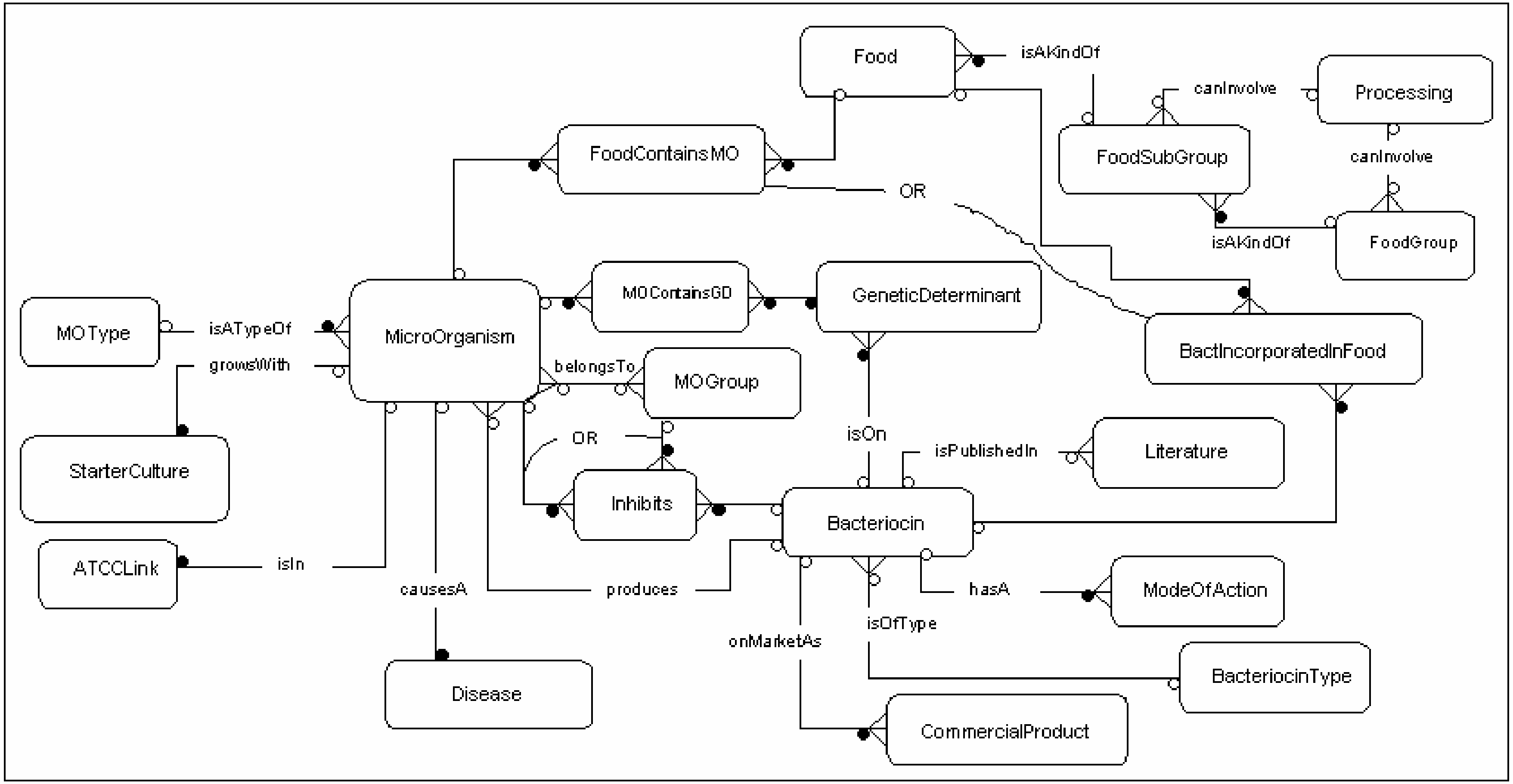
Swiss-Prot – Protein knowledgebase: http://www.ebi.ac.uk/swissprot/index.html

TIGR – The Institute of Genomic Research: http://www.tigr.org

Tree of Life: http://tolweb.org

Appendix A

ER Diagram of the Bacteriocin Database


Food
isAKindOf
canInvolve
Processing
FoodContainsMO
FoodSubGroup
canInvolve
OR
isAKindOf
FoodGroup
MOContainsGD
GeneticDeterminant
MOType
isATypeOf
MicroOrganism
BactIncorporatedInFood
growsWith
belongsTo
MOGroup
StarterCulture
OR
isOn
isPublishedIn
Literature
Inhibits
Bacteriocin
ATCCLink
isIn
causesA
produces
hasA
ModeOfAction
isOfType
onMarketAs
Disease
BacteriocinType
CommercialProduct