Journal of Informetrics 4 (2010) 564–580

Contents lists available at ScienceDirect

Journal of Informetrics

journal homepage: www.elsevier.com/locate/joi

A relational database for bibliometric analysis

Nicolai Mallig

Fraunhofer Institute for Systems and Innovation Research, Breslauer Straße 48, 76189 Karlsruhe, Germany

ARTICLE INFO

Article history:
Received 22 December 2009
Received in revised form 18 June 2010
Accepted 21 June 2010

Keywords:
Bibliometrics
Relational database
SQL

ABSTRACT

In this article a relational database schema for a bibliometric database is developed. After the introduction explaining the motivation to use relational databases in bibliometrics, an overview of the related literature is given. A review of typical bibliometric questions serves as an informal requirement analysis. The database schema is developed as an entity-relationship diagram using the structural information typically found in scientific articles. Several SQL queries for the tasks presented in the requirement analysis show the usefulness of the developed database schema.

1. Introduction

Nowadays a widespread use of bibliometric indicators can be observed. Probably the ease of using web-based bibliographical databases is boosting this trend. Unfortunately, there is some evidence that not all bibliometric studies rely on sound methodological work (Larsen, 2008). On the one side, it is easy to blame the authors of these studies using naive bibliometrics for their ignorance of years of methodological bibliometric work and existing relevant literature. On the other side, this indicates a more severe problem: the lack of adequate bibliometric databases.

The problem of adequate bibliometric databases has already been discussed (Moed, 1988) and the proposed solution consists of downloading the data from on-line databases and storing it in a customized in-house database. Unfortunately, the literature on the construction of proper bibliometric databases is very scarce. Some pioneering articles exist (Fernández, Cabrero, Zulueta, & Gómez, 1993; Small, 1995; Winterhager, 1992; Zitt & Teixeira, 1996), indicating the powerful abilities of relational databases (Codd, 1970) for use in bibliometrics, but there is no comprehensive article that describes the construction of a relational database schema for bibliometrics.

A quite recent review article on databases for bibliometric purposes (Hood & Wilson, 2003) illustrates this lack of discussion about off-line processing using in-house databases in the bibliometric literature. While there is only a short section about off-line processing and nothing about the construction of in-house databases, the largest part is dedicated to on-line databases and their limitations. Indeed, it seems true that there has been much progress made in the use of on-line databases (Marx, Schier, & Wanitschek, 2001). While all that can be done with on-line databases and the sophisticated tools that have been developed (Neuhaus, Litscher, & Daniel, 2007) is impressive, it seems that much of the effort has been invested just to circumvent their biggest limitations. Despite all this progress made, these articles show that bibliometric analysis is still a cumbersome task using on-line databases and the problem of data quality is still present.

In order to overcome these problems, one has to resort to a solution already performed 20 years ago (Moed, 1988): downloading the data or buying them from one of the providers of bibliographic data, cleaning it, and storing it into a

E-mail address: nicolai.mallig@isi.fraunhofer.de.

1751-1577/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.joi.2010.06.007

database appropriate for bibliometric tasks. The main question is how such a database should be constructed so that it best meets the needs of bibliometricians. Here I offer a solution based on relational database technology.

So the aim of this article is to depict the structure of a relational database that is very suitable for most bibliometric analyses, with a focus on the computation of bibliometric indicators.

Section 2 gives a short review of the relevant literature on related work. In Section 3 a short introduction to bibliometric indicators and methods is given. In Section 4 a relational database schema for bibliometrics is developed. Section 5 gives a proof-of-concept for the relational database schema: Several examples of SQL code covering many typical bibliometric analyses are shown.

2. Related work

There are several research groups in the field of bibliometrics using in-house databases that are based on relational database technology. Nevertheless the information in the literature as to how these databases are constructed is very scarce. In fact a debate on how a relational database should be designed to support advanced bibliometric methods is still lacking. This is particularly odd since bibliometric databases build the foundation of all the bibliometric work that is based thereupon. In the following a review of the few articles dealing with the use of relational databases in bibliometrics is given.

A very first testimonial of the use of *relational structures* in bibliometrics can be found in Moed (1988), dating back more than 20 years. While there is not yet a genuine relational database management mentioned in the article, the described organization of data clearly resembles the structure one would use in a relational database.

The first articles describing the use of relational databases in bibliometrics appeared in the early nineties in line with the rise of relational database management systems. Motivated by the problems and limitations encountered when existing document-oriented bibliographic databases were used for bibliometric purposes Winterhager (1992) introduces the concept of a relational database for bibliometric uses and presents a relational database schema for the SCI data. Almost at the same time another relational bibliometric database is presented (Fernández et al., 1993). The main challenge described in this article is the integration of data from several different sources. The authors emphasize the problem of data quality and the needs for standardization. So the focus of this article lies on the procedures employed for standardization and codification of the institutional information and journal names. Information about another relational database for bibliometrics built around the same time can be found in Zitt and Teixeira (1996), although the article was published some years later. It describes the process of calculating bibliometric and other macro-indicators based on a relational database. The focus lies on the description of the calculated indicators, only a rough overview of the relational structure used is given.

Only few years after the first articles reporting the use of relational databases in bibliometrics, the power of relational databases for advanced bibliometric analysis techniques like co-citation analysis or bibliographic coupling was recognized (Small, 1995). While the article focuses on the representation of the citation data in tables of a relational database and how the citation network can be navigated using SQL commands, it considers the generalization of these ideas to other bibliographic data elements as well. In particular, the article points out how a good relational design could facilitate further bibliometric work.

There follows a longer period when no use of relational databases is mentioned in the bibliometric literature. This might be interpreted as relational databases being an established tool used by the research groups operating in-house databases, but not worth mentioning explicitly. Another possible interpretation for the absence of relational databases from the bibliometric literature is the small community building in-house databases for bibliometrics, so the number of possible authors is very limited. In addition, the construction of the database might be seen just as a necessary preparatory step on the way to advanced bibliometric research questions and once an in-house database is set up there are so many new interesting research opportunities that the data model is of lesser interest. Another explanation for the scarceness of information about the construction of bibliometric databases in the literature may be attributed to the fact that the details of the implemention of a bibliometric database are considered as a kind of trade secret by some research groups operating bibliometric in-house databases.

It takes 10 more years until attention is drawn once again to relational databases (Wolfram, 2006). This time the article is not directly related to the construction of bibliometric databases, but to its uses in the more general field of informetric data processing using the query language SQL. The author uses the context of query data analysis for illustrative purposes, but emphasizes that the same methods can be applied to bibliometric questions. One of the examples shows how co-occurrence of terms can be calculated using SQL. A straightforward application of this example to co-word analysis seems possible.

Recently the need for specialized databases for bibliometric purposes has been recognized once again and a new interest in adequate data modeling was awakened (Yu, Davis, Wilson, & Cole, 2008). In this article an object-relational approach is chosen. However this does not mean that pure relational databases are obsolete, there are no stringent cases shown where a pure relational database would not have sufficed. An interesting new idea presented in this article is the modeling of authors' career paths.

Although this literature review may lead to the misleading conclusion that relational databases had already reached their peak of importance in bibliometrics several years ago, in my opinion it is too early to draw this conclusion. As far as the literature shows, it is not the case that relational databases have reached their limits in the field of bibliometrics. On the contrary, it seems that the bibliometric community on the whole is not yet completely aware of their powers. The current

article might give first insights and trigger further research on how relational databases could be constructed to best serve the needs of bibliometricians.

3. Requirement analysis

In this section I give a short review of the most important bibliometric indicators and methods. These serve as use cases of an informal requirements analysis. Completeness is not the focus here, but to give a representative set of indicators and methods used in bibliometrics. This approach is based on the following assumption: if the relational database schema is suitable for the indicators and methods mentioned here, then it can be utilized directly for most bibliometric problems without major modifications.

Bibliometric applications can be divided broadly into two parts: calculation of bibliometric (performance) indicators on different actor levels and analysis and visualization of bibliometric networks.

Analysis using bibliometric indicators can be further differentiated into *descriptive bibliometrics* and *evaluative bibliometrics* (van Leeuwen, 2004). While descriptive bibliometrics takes a top-down approach, trying to get the big picture, e.g. the research output of a country in different fields, the proportions of the different fields and their changes over time, evaluative bibliometrics is a tool to assess the research performance of smaller units like research groups or even individuals and uses an bottom-up approach collecting all the (relevant) publications of the respective unit. Obviously, evaluative bibliometrics poses stronger requirements on data quality. Therefore bibliometrics in research assessment typically involves a verification process to meet the required data quality, where the authors or institutions to be evaluated verify the publication data (van Leeuwen, 2007).

Calculation of bibliometric indicators is, from a technical point of view, basically counting numbers (publications and citations), although admittedly some preparatory work, like development of sophisticated classifications of journals, assignments of authors to organizations and linking of cited publications to citing publications is necessary for counting the right numbers. Especially if the citation linking has not yet been done by the data providers, e.g. for so-called non-source items, establishing this missing link requires sophisticated matching techniques and still needs human validation (Butler & Visser, 2006). Another important issue that should not be neglected is data quality. A comprehensive review of problems concerning data quality can be found in Hood and Wilson (2003). There are several publications which emphasize the aspect of data quality for bibliometric analyses (e.g. van Raan, 2005; Fernández et al., 1993). I will not expand on the topic of data quality here any further, since it is more a problem of data management than a genuine part of bibliometrics.

I assume in the following that these classifications and assignments are available, that the citation linking has been done and that the data has been adequately cleaned, so I can focus without distraction on the theoretical part.

3.1. Bibliometric indicators

Performance analysis with bibliometric methods is carried out with the idea of measuring output, impact (often used as proxy for quality) and collaboration. Bibliometric indicators are mainly based on publication and citation counts on different aggregation levels. While there is a plethora of bibliometric indicators, they have almost all one aspect in common: they are combinations of publication and citation counts on different aggregation levels. Some of them are normalized by expected citation counts and sometimes some transformations are applied.

With the idea of presenting typical examples of bibliometric indicators, I restrict myself basically to the indicators defined in (Moed, De Bruin, & Van Leeuwen, 1995), which represent a broad and quite elaborated set of indicators for the assessment of research performance, and follow their nomenclature. One reason for this choice is the availability of a precise and comprehensive description of how these indicators are calculated.

P Number of publications
C Number of citations received
CPP (Average number of) Citations per publication
CPPex (Average number of) Citations per publication, self-citations excluded
%Pnc Percentage of papers not cited (during the time period considered)
JCS Journal Citation Score (average number of citations per publication, per article type and journal)
FCS Field Citation Score (average number of citations per publication, per article type and (sub)field)
JCSm Mean citation rate of journal packet (weighted by number of publications of the article set under examination)
FCSm Mean citation rate of (sub)field(s) (weighted by number of publications of the article set under examination)
%SELFCIT Percentage of self-citations
CPP/FCSm Citations per publication, compared to citation rate of journal packet
CPP/FCSm Citations per publication, compared to citation rates of subfield(s)
JCSm/FCSm Citation rate of journal packet, compared to citation rate of subfield(s)

Since the last three indicators can be directly calculated from the others, I can safely ignore them in my approach to constructing a relational database schema, as long as I ensure that their components can be calculated.

I add another indicator to the set, which is often used as a measure of (international) cooperation (Glänzel, Schubert, & Czerwon, 1999a, 1999b):

CoP Number of co-publications (with another unit of analysis)

Depending on the research question, the unit of analysis used for measuring collaboration might be very different, ranging from single persons to whole countries.

This observation holds true in general for all the indicators mentioned above: bibliometric indicators are used on different aggregation levels, namely article, journal, author, institution, country, region, field of research (discipline, specialty). Thus it is important that the database schema covers all these aggregation levels.

3.1.1. Counting methods

Several different counting methods are being used in bibliometrics to count multi-authored publications. The two methods mostly employed are *full counting* and *fractional counting*. A deep and systematic analysis of the different counting methods is given in Gauffriau, Larsen, Maye, Roulin-Perriard, and von Ins (2007), identifying in total five different counting methods. I will follow their naming conventions here.

When counting publications one interpretation is *giving credit* per publication. The credits are aggregated to the appropriate aggregation level, e.g. author, institution or country. The counting methods differ mainly by the way in which credit is given per author for each publication. These counting methods are typically used:

Complete Each involved basic unit (author) gets a credit of 1.
Complete normalized Each involved author gets a credit of $\frac{1}{n}$, while n is the number of authors of the paper.
Straight The first author gets a credit of 1 the other authors receive a credit of 0.
Whole Each involved unit of the aggregation level under study (e.g. each country) receives a credit of 1.
Whole normalized Each involved object of the aggregation level under examination gets a credit of $\frac{1}{n}$, while n is the number of participating units of this aggregation level.

While both methods *complete normalized* and *whole normalized* are kinds of *fractional* counting methods, the term *fractional counting* is typically used with the meaning of *complete normalized* counting. The two methods differ in the way the fractions are accredited. For example, a paper with two authors from France and one author from Germany will get a credit of $\frac{2}{3}$ for France and $\frac{1}{3}$ for Germany if *complete normalized* counting is employed, whereas a credit of $\frac{1}{2}$ will be given to France and Germany each if *whole normalized* counting is used.

The existence of different counting methods is on the one hand motivated by the fact that there is a broad spread of research questions and no single counting method is perfectly suitable for all circumstances. Every method has its advantages and disadvantages, so the usage of a specific counting method is ideally motivated only by the research question. On the other hand, it is possible that the database used poses some restrictions on the counting method to be used. These restrictions might result from poor availability of data (e.g. address information only for first authors) or lack of power of the query language that make it impossible or really expensive to use the most appropriate counting method. Using on-line or web-based databases for example, *whole counting* is the easiest to apply because it involves only the use of adequate search terms and the number of matches is the sought result, whereas the application of *complete*-(*normalized*)-*counting* would involve the inspection of every element of the result set.

I will not go into detail of the advantages and disadvantages of the different counting methods mentioned above for different situations. Let's just state that it is desirable to have the opportunity to use each of these methods, so the choice of method only depends on the research question and is not imposed by some artificial constraints.

3.2. Bibliometric networks

Another subfield of bibliometrics consists of the study and structural analysis of *bibliometric networks* (Shrum & Mullins, 1988). Visualization of networks plays an important role and is known under the term *mapping of science* (Rip, 1988). In this case the network consists of scientific themes defined by sets of papers that are linked via citations or co-words. Techniques and tools from *social network analysis* (SNA) can be employed for analyzing bibliometric networks (Newman, 2001; Otte & Rousseau, 2002).

The types of bibliometric networks used are quite diverse, ranging from co-authorship networks (Glänzel & Schubert, 2004) over citation and co-citation networks (Small & Garfield, 1985) to co-word networks (Callon, Courtial, Turner, & Bauin, 1983).

A properly structured bibliometric database can provide an extensive resource of data for different bibliometric networks. Bibliometric networks can be differentiated by their type of links and their types of actors:

Type of links Co-authorship, citation, co-occurrence of (key) words (*co-words*), co-citations, bibliographic coupling
Type of actors Article, author, research group/institution, journal, region/country, key words

To analyze bibliometric networks, it is necessary that the network data can be extracted from the database. Analysis and visualization can be done subsequently with the aid of specialized SNA software.

4. Development of a relational database schema

Bibliometrics focuses on the statistical analysis of the bibliographical data of scientific articles typically published in journals. So our modeling approach starts with the information typically found in the front and back page of an article. This information will be arranged in an entity-relationship diagram (Chen, 1977). In the next steps I check if the model suits the requirements and adjust it as needed.

4.1. Basic data model

In this section the main structural components of an article and their relationships are identified. For this task let's have a look at some typical scientific articles:

- An article has a title and an abstract.
- An article is written by one or several authors. (The order of appearance may be important information.)
- An author is affiliated with an organization (or several) which has an address.
- An author may have an e-mail address.
- The article is published in a journal which has a name.
- The article is published in a specific issue of a journal identified by volume and issue number and has a publication date.
- The article contains the date when it was submitted by the author and received by the journal.[1]
- There are several key words (provided by the author) for information retrieval.
- On the back page there is a list of references to other articles (or article-like documents).
- Each reference contains mainly a condensed subset of the information contained in the front page of the cited article, usually sufficient to unambiguously identify it.

Let's summarize and organize the information found above:

The most important identified entities (objects of interest) are:
- article
- journal
- person (author)
- organization

The identified relationships (and the entities linked by them) are:
- authorship (linking person and article)
- publication (linking article and journal)
- affiliation (linking person and organization)
- reference/citation (linking article and article, with a *citing* and a *cited* direction)

I use entity-relationship (ER) modeling to visualize the entities with their attributes and the relationships identified above. An entity-relationship diagram is an abstract representation of data typically used for data modeling. The entities are displayed as boxes, the attributes as ellipses and the relationships as diamonds while edges link the corresponding elements.

The entities and relationships identified so far are displayed in Fig. 1. The fact that an article is published in a journal is expressed via the *publication* relationship linking *article* to *journal*. The *authorship* establishes a link between an *article* and the *person* having written it. A *person* belongs to an organization which is expressed by the *affiliation* relationship. The *reference* relationship links the citing to the cited article, i.e. it is linked twice to *article* once in the *citing* and once in the *cited* direction. The information that can be used to establish the citation linkage is stored as attributes of the relationship.

The reference relationship is somewhat tricky, so it is worth having a closer look at it. From a theoretical standpoint, the cited article is not structurally different from the citing one and can be perfectly represented by the article entity. So the attributes of the *reference* relationship are redundant information already contained in the data of the cited article. Practical reasons motivate the acceptance of this redundancy here: not every cited article will be contained in the database, because each article contains references to other articles and one has to stop following the references at some point. In the absence of the cited article in the database we need a place to store the reference data and the *reference* relationship is a natural choice for this. The other possibility is to store the basic information of the cited article in the database

[1] This information is very interesting because this date is much closer to the period in which the work was actually performed than the publication date. To my knowledge, this information is typically not available in the common bibliographic databases, so I'll skip this information in the following.

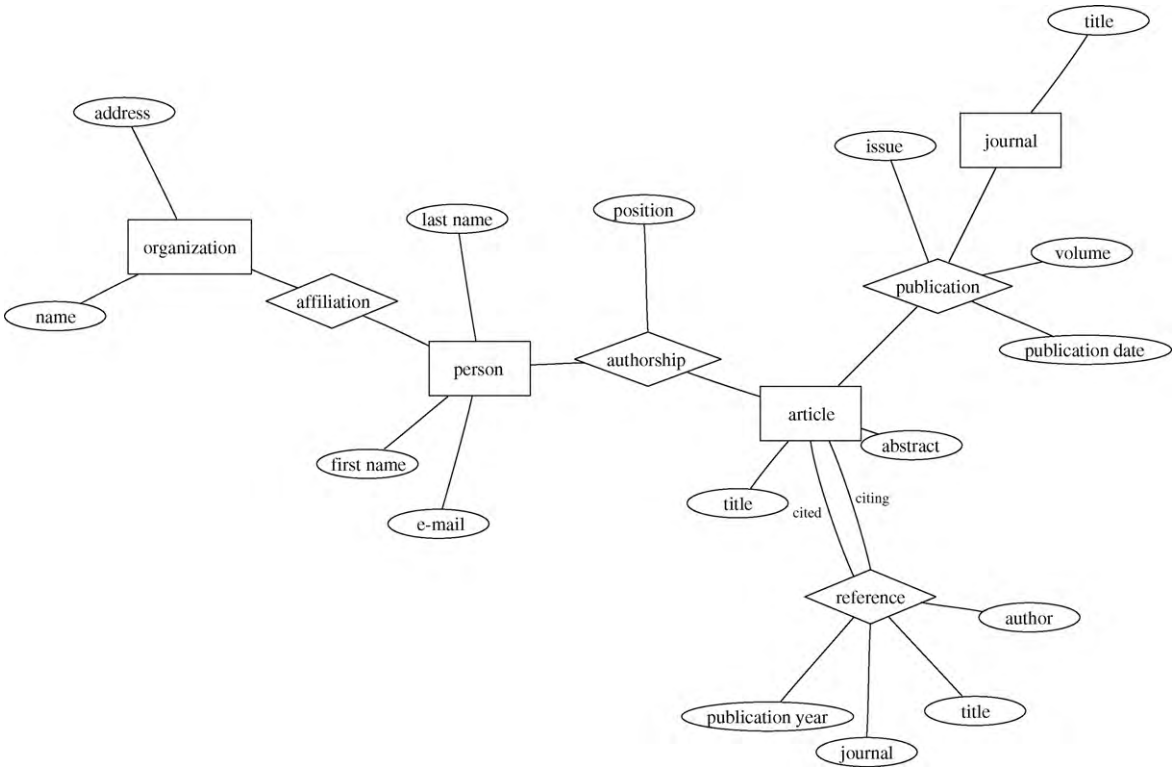

Fig. 1. Basic ER diagram.

(omitting its references to bring the recursive process to a halt) and add an attribute to the *article* entity for flagging these articles as so-called non-source items[2] (i.e. articles that are only covered as references and are not themselves sources of citations). When performing analyses based on non-source items (Butler & Visser, 2006) this solution might be preferable, since then all features of the database (e.g. classifications) can be utilized for the analysis. It is possible to combine both approaches, so the attributes of the *reference* relationship would store the original information as found in the reference list of the citing article while the information stored in the *article* entity might be an enriched or corrected version of the data. This would enable new interesting analyses based on differences or misspellings of references (Simkin & Roychowdhury, 2003).

To derive the relational schema, the entities, relationships and attributes of the ER model have to be mapped to the relational tables of the relational model. This transition is straightforward: entities and relationships are mapped to tables while attributes are mapped to the columns of the corresponding tables. The different instances of an entity are identified by a key (in practice the key is mostly an ID column, while in principle a combination of several columns which taken combined uniquely identify each row is also admissible). Relationship tables consist of the key columns of the related tables and the attributes of the relationship (if any).

The corresponding relational schema is displayed in Fig. 2. I shall continue to use ER diagrams in the remaining sections since these give a clearer presentation of the underlying data, distinguishing between entities and relationships.

The relational schema can be used to retrieve data in the following way: let us consider the task of retrieving all articles of a specific author: These articles can be found by first identifying the author in the *person* table. Using his ID it is possible to follow the link established by the *authorship* table to the *article* table. This is done by collecting all article IDs from the *authorship* table that are linked with the person ID of the author. Then these IDs can be used to retrieve the corresponding articles from the *article* table. Contemporary relational databases provide the query language SQL that is well suited to tasks like this. When using SQL it is not necessary to specify the process of retrieving the data step by step. Instead one specifies declaratively which information to return and the database management system handles the single steps internally. An SQL query for the question mentioned above would look like this:

[2] This flagging is important to distinguish articles that do not contain references from articles for which references are not recorded in the database, e.g. to avoid distortions when counting the average number of references.

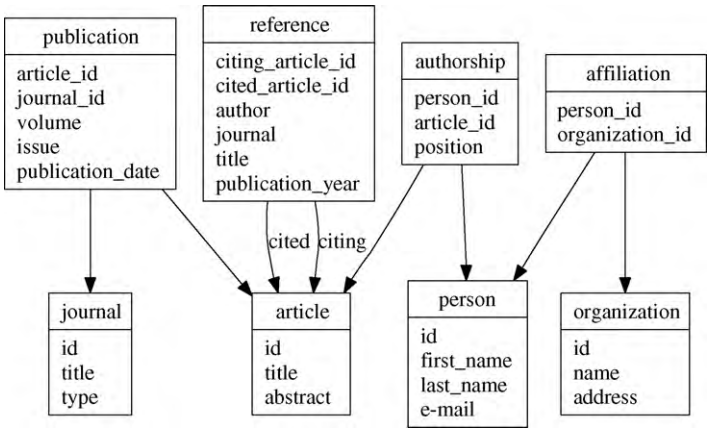

Fig. 2. Diagram of the basic relational schema.

```
SELECT
  person.last_name,
  article.title
FROM
  person
  JOIN authorship ON person.id=authorship.person_id
  JOIN article ON authorship.article_id=article.id
WHERE
  person.last_name='De Solla Price'
  AND person.first_name='Derek'
```

The *SELECT* clause specifies which data (columns) to return, namely the author's last name and the articles' titles. The *FROM* clause defines the tables to be used and how they should be linked (*JOIN...ON*). The *WHERE* defines the *filter* to be used; in this case the query retrieves only data for authors where first name and last name exactly match the given strings.

The usefulness for more complex queries will be shown below (see Section 5).

4.2. Refinements

4.2.1. Affiliations

Let us have a closer look at the affiliation relationship: this relationship is different from the others. While the other relationships are static, in the sense that they do not change over time (e.g. if an article is published in a journal, it will belong to this journal forever), the affiliation relationship is only valid for a specific period of time. If the person moves to another organization, the information of this relationship needs to be updated. But a real update (deleting old, inserting new affiliation) is not an option, because information of the old affiliation would be lost and if we looked at an older article we would wrongly associate it with the new organization. As a solution one could transform the relationship into a *temporal relationship* by adding two attributes to the affiliation relationship, namely *start* and *end*, denoting the start and end of the period for which this affiliation information is valid. The problem with this approach is how to get the correct affiliation data? Gathering this data from external sources would be only possible for a very limited number of authors. However, we are not dependant on external sources. All the information we need is in principle already contained in the data.[3] We are only concerned about these points in time when an article is written (published) by an author and the publication date is available. So for the use here, affiliation is a relationship that connects author (person) and article (i.e. the *authorship* relationship) to an organization. So we only have to adjust our schema slightly and model affiliation as a relationship between organization and the authorship relationship. Because the e-mail address used is typically an official one, it is modeled as an attribute of the affiliation relationship. The adjusted schema is shown in Fig. 3 (attributes have been omitted for clarity).

4.2.2. Classifications of journals

A topic not covered so far is the classification of articles into different scientific fields/disciplines. For a classification on a very broad level, it is not absolutely necessary to classify each article individually, instead a classification of journals like these already provided by vendors of the bibliographic raw data is sufficient.[4]

[3] Unfortunately some bibliographic databases do not provide the link between authors and corresponding institutions, although this information is available in the front page of the original journal article. In this case it may be difficult or even impossible to reconstruct this missing information in an automated way.

[4] Here I simplify the issue and ignore the existence of multidisciplinary journals on purpose.

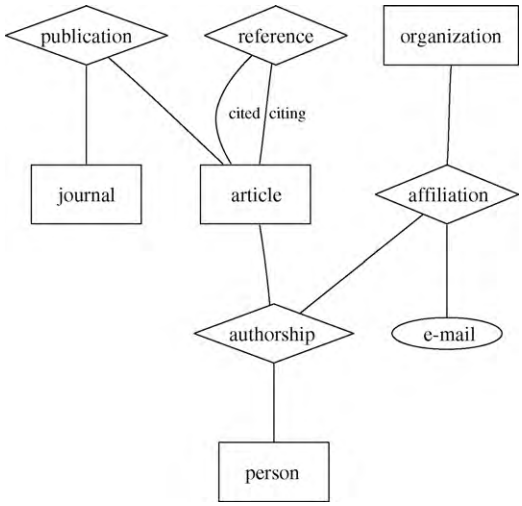

Fig. 3. ER diagram with adjusted affiliation relationship.

For the classification on journal level I introduce a new entity *classification*, consisting of the attributes *field* and *system* and a relationship *journal_classification* linking *classification* to the corresponding *journal*. The attribute *system* enables us to handle several different classification systems in parallel.

While not only a classification based on journals but on articles would be desirable, e.g. for dealing with articles of multidisciplinary journals or very specialized subfields of a discipline, I omit this classification in the model for the sake of simplicity. On the conceptual level there is practically no difference to the classification of journals, just the relationship *article_classification* would link from *classification* to *article* instead of *journal*. The differences lie only at the data level: a much higher number of objects has to be classified and there are no existing classifications available that can be directly used. For the classification of the individual articles a methodology like the one described in Glänzel et al. (1999a, 1999b) could be used.

4.2.3. Institutions/Organizations

Analyses should be possible on different institutional levels. While a fine-grained level, e.g. *departments* or *institutes*, may be necessary for some study, a coarser level (e.g. *university*) might be needed for another research question. So the data model should reflect these different needs.

For finer grained analyses in the organizational dimension, it is necessary to split up the *organization* entity a little bit. I introduce a new entity *organizational unit* (*org_unit*) which is connected via the relationship *belongs_to* to the *organization* (see Fig. 4). The *org_unit* stores the information of the lowest organizational aggregation level contained in the data and is linked to every higher organizational level stored in *organization*. An attribute *org_level* of the *organization* entity denotes the actual recorded aggregation level.

4.2.4. Name variants

Authors, journals and organizations may appear in different name variants in the bibliographic raw data, e.g. spelling variations with or without middle initial or due to a change of name. The task of assigning a unique standardized name for each of these entities belongs to the indispensable data cleaning process, and has been described thoroughly by Fernández et al. (1993).

It is advisable to keep track of all these name variants of an entity, so this information can be utilized in the data standardization process of new raw data. For this purpose it is sufficient to keep this information in the preprocessing system, so there is no need for extending the schema of the bibliometric database to cope with name variants.

4.2.5. Other publishing media

While I have focused my argumentation so far on articles published in journals, these are not the only publishing media. Other typical publishing media are edited books, conference proceedings or monographs.

All these publishing media can be easily incorporated into the model developed so far. The *journal* entity can be extended by adding a new attribute *type* to distinguish between those. The entity should then be renamed to something like *publishing_medium* to reflect its real content, but I prefer to keep the name *journal* to emphasize the fact that the typical articles analyzed in bibliometrics are published in journals.

Quite a new trend in publication behavior is the *Open Access* movement, making scientific articles freely available in preprint archives like *arXiv* prior to a publication in a reviewed journal. This diversification raises new interesting research questions like the influence of this additional publishing form on the citation counts of the articles (Davis & Fromerth, 2007).

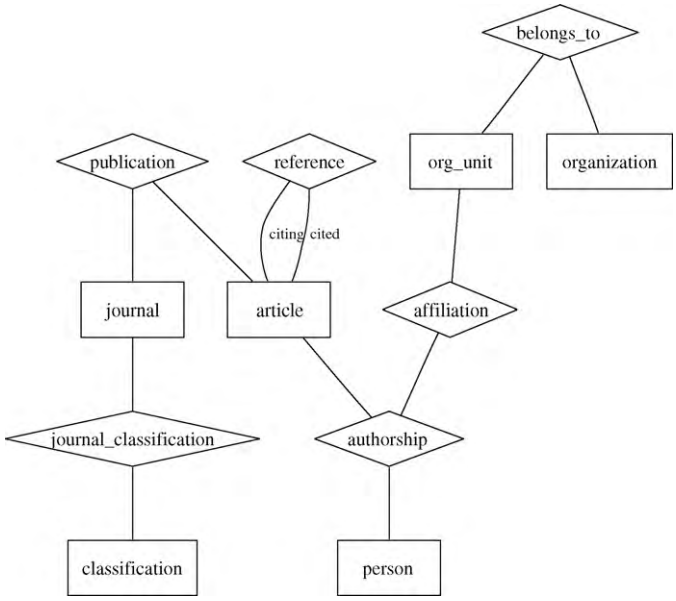

Fig. 4. Final ER diagram: *organizational unit* and *classification* added.

The possible integration of data from preprint archives into a bibliometric database leads to the phenomena of multiple instances of the same article.[5] Luckily the database schema developed so far naturally supports this extension. The integration of a preprint archive would result in a new entry in the *journal* (or *publishing_medium*) table, new entries in the *article* table for each article not yet covered in the bibliometric database and an entry in the *publication* table for each article contained in the preprint archive.

4.3. Adjustments to facilitate computations

In this section I shall discuss some enhancements to facilitate computations.

4.3.1. Fractional counting

When dealing with co-authored papers, fractional counting on the level of authors (*complete-normalized* counting) is often employed. For this purpose the number of authors for each article has to be calculated. While this can principally be done on-the-fly, it is a good idea to pre-compute this data and store it into the database. Therefore an additional attribute *author_count* is added to the *article* entity. Using this pre-computed data, the resulting SQL queries will be much shorter and computing time is reduced.

A pre-computation of the number of involved organizations per article seems also useful, speeding up queries using fractional counting on the institutional/organizational level.

4.3.2. Expected citation rates

For the calculation of normalized citation indicators like the *JCSm* (*FCSm*), the average (expected) citation rates per journal (subfield) have to be available while different publication years and citation window sizes have to be addressed.

The expected (average) citation rates are calculated separately for each different relevant article type (Moed et al., 1995) (e.g. normal article, letter, notes, reviews and proceeding papers). Sometimes citation counts excluding self-citations are used. To provide consistent normalized versions of this indicator, the expected citation should be calculated also excluding self-citations. So citation rates should be provided in two versions with and without self-citations. In addition, expected citation rates for different window sizes should be provided, offering the possibility to analyze short-term and long-term impact. So I add a new relational table as shown in Fig. 5 providing for each journal the expected citation score, distinguished by publication year, citation window, article type and self-citation handling. An example of how this data can be calculated from the database is provided in Section 5.5. The final version of the relational database schema is displayed in Fig. 5.

[5] To a small degree this phenomena already exists, e.g. when a conference paper is republished as a journal article.

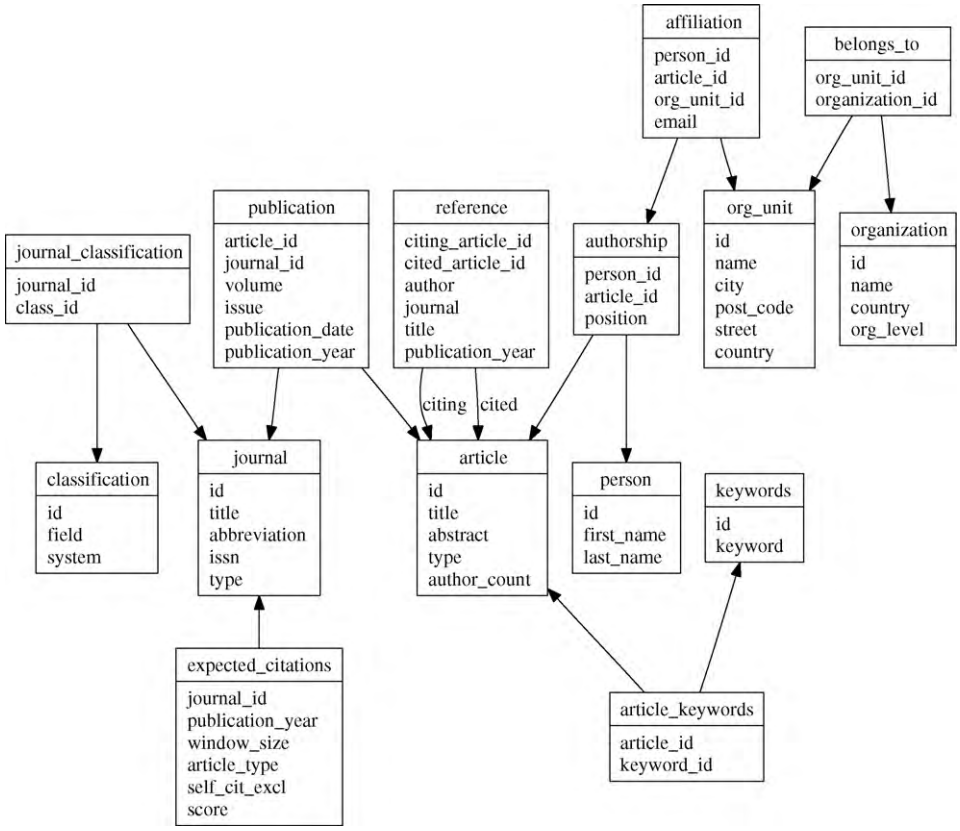

Fig. 5. Final relational database schema.

5. Proof-of-concept

In this section the usefulness of the relational schema developed in the previous section will be demonstrated. This will be done by showing SQL queries for the calculation of several of the indicators described in Section 3. The queries are based on the relational schema shown in Fig. 5.

5.1. Number of publications

One of the basic indicators is the *number of publications*. I'll show here how the number of publications per institution can be calculated. The query will be limited to institutions located in Germany, publication years between 2000 and 2005 and publications of type *article* or *letter*. The query is shown in Listing 1.

The query is straightforward. Starting from *organization*, we follow the chain of tables joining them until we reach the *publication* table. The filters mentioned above are applied. The results are grouped by organization and publication year and for each such combination, the number of distinct articles is counted.

5.2. Number of citations

The next example (Listing 2) shows the calculation of *number of citations*. A citation window of 3 years (current year plus two additional years) is employed.

In contrast to the previous example there is an additional JOIN to the *reference* table and the *article* and *publication* tables are joined twice, taking into account the double linkage of the *reference* table to *article*. To distinguish between them *aliases* are used for the second occurrence. Here not the number of articles is counted, but the number of different pairs of cited and citing articles. We have to use both, the ID of the citing and the ID of the cited article in the *COUNT*(*DISTINCT*. . .) expression. If only the ID of the citing article was used, the query would return the number of citing articles and not the number of citations. On the aggregation level *article* these numbers would be the same, but on higher aggregation levels (e.g. author or research group), there possibly are articles that cite several of the unit's articles (e.g. if an article A cites articles B and C of the same author, the number of citing articles is 1, the number of citations is 2), so the number of citing articles is different from the number of citations.

Listing 1
Number of publications.

```
SELECT
  organization.name,
  publication.publication_year,
  COUNT (DISTINCT article.id)
FROM
  organization
  JOIN belongs_to ON belongs_to.organization_id=organization.id
  JOIN org_unit ON org_unit.id=belongs_to.org_unit_id
  JOIN affiliation ON affiliation.org_unit_id=org_unit.id
  JOIN authorship ON authorship.person_id=affiliation.person_id
    AND authorship.article_id=affiliation.article_id
  JOIN article ON article.id=authorship.article_id
  JOIN publication ON publication.article_id=article.id
WHERE
  article.type IN ('article', 'letter')
  AND organization.country='DE'
  AND publication.publication_year BETWEEN 2000 AND 2005
GROUP BY
  organization.name,
  publication.publication_year
;
```

5.3. Number of citations, self-citations excluded

When excluding self-citations, the question is at which aggregation level self-citation is defined. In the next example (Listing 3) self-citations at author level are excluded. Self-citations at other levels can be excluded following an analogous procedure. According to Moed et al. (1995), a self-citation is defined as pair of a citing paper and a corresponding cited paper that has at least one author in common.

The extension compared to the query including self-citations consists of the additional sub-query in the *WHERE* clause that is connected via the condition *NOT EXISTS*. The sub-query basically calculates the intersection of author sets of the citing and the cited paper, i.e. it returns a non-empty set in the case of self-citation.

For this query it is essential that the quality of the data is very good, especially the query relies on the assumption that there are not two different IDs for the same person.

Listing 2
Number of citations.

```
SELECT
  organization.name,
  publication.publication_year,
  COUNT (DISTINCT reference.citing_article_id || reference.cited_article_id)
FROM
  organization
  JOIN belongs_to ON belongs_to.organization_id=organization.id
  JOIN org_unit ON org_unit.id=belongs_to.org_unit_id
  JOIN affiliation ON affiliation.org_unit_id=org_unit.id
  JOIN authorship ON authorship.person_id=affiliation.person_id
    AND authorship.article_id=affiliation.article_id
  JOIN article ON article.id=authorship.article_id
  JOIN publication ON publication.article_id=article.id
  JOIN reference ON reference.cited_article_id=article_id
  JOIN article citing_article ON citing_article.id=reference.citing_article_id
  JOIN publication citing_publication
    ON citing_publication.article_id=citing_article.id
WHERE
  article.type IN ('article','letter')
  AND organization.country='DE'
  AND publication.publication_year BETWEEN 2000 AND 2005
  AND citing_article.type IN ('article','letter')
  AND citing_publication.publication_year <= publication.publication_year+2
GROUP BY
  organization.name,    publication.publication_year
;
```

Listing 3
Number of citations, excluding self-citations.

```
SELECT
  organization.name,
  publication.publication_year,
  COUNT (DISTINCT reference.citing_article_id || reference.cited_article_id)
FROM
  organization
  JOIN belongs_to ON belongs_to.organization_id=organization.id
  JOIN org_unit ON org_unit.id=belongs_to.org_unit_id
  JOIN affiliation ON affiliation.org_unit_id=org_unit.id
  JOIN authorship ON authorship.person_id=affiliation.person_id
    AND authorship.article_id=affiliation.article_id
  JOIN article ON article.id=authorship.article_id
  JOIN publication ON publication.article_id=article.id
  JOIN reference ON reference.cited_article_id=article_id
  JOIN article citing_article ON citing_article.id=reference.citing_article_id
  JOIN publication citing_publication
    ON citing_publication.article_id=citing_article.id
WHERE
  article.type IN ('article','letter')
  AND organization.country='DE'
  AND publication.publication_year BETWEEN 2000 AND 2005
  AND citing_article.type IN ('article','letter')
  AND citing_publication.publication_year <= publication.publication_year+2
  AND NOT EXISTS (
    SELECT
      cited_person.id
    FROM
      authorship cited_authorship
      JOIN person cited_person
        ON cited_authorship.person_id=cited_person.id
    WHERE
      cited_authorship.article_id=reference.cited_article_id
    INTERSECT
      SELECT
        citing_person.id
    FROM
      authorship citing_authorship
    JOIN person citing_person
      ON citing_authorship.person_id=citing_person.id
    WHERE
      citing_authorship.article_id=reference.citing_article_id
)
GROUP BY
  organization.name,
  publication.publication_year
;
```

5.4. Percentage of publications not cited

The calculation of the *Number of papers not cited* is shown in Listing 4. In combination with *Number of publications* the indicator *Percentage of publications not cited* can be calculated.

The query is basically the same as the query for *Number of publications*, the only differences are the *LEFT JOIN* with the *reference* table and the additional condition *reference.cited_article_id IS NULL*. The former adds an additional (virtual) *citing_article_id* column to the intermediate result of the query, containing the IDs of the citing articles and having the value NULL for each article which is never cited. The additional filter condition restricts the result just to these rows.

5.5. Journal Citation Score

The next example (Listing 5) shows how the JCS (expected citation rates per journal) can be calculated.

The query consists of two nested SELECT statements. The inner calculates the number of citations per article, while some additional information (publication year, journal, article type) is kept. The outer groups the results by this additional information and counts the average citation rate for each of these groups. The use of *outer joins* (here: LEFT JOIN) in the inner query is crucial. Otherwise all articles with zero citations would not be contained in the result set of the inner SELECT.

The *Field Citation Score* can be calculated analogously. It needs just two additional JOINs with *journal_classification* and *classification* and the result has to be grouped by *classification.field* instead of *journal_id*. An extension to JCR excluding self-citations is straightforward using the ideas described in Section 5.3.

Listing 4
Number of papers not cited.

```
SELECT
  organization. name,
  publication. publication_year,
  COUNT (DISTINCT article.id)
FROM
  organization
  JOIN belongs_to ON belongs_to.organization_id=organization.id
  JOIN org_unit ON org_unit.id=belongs_to.org_unit_id
  JOIN affiliation ON affiliation.org_unit_id=org_unit.id
  JOIN authorship ON authorship.person_id=affiliation.person_id
    AND authorship.article_id=affiliation.article_id
  JOIN article ON article.id=authorship.article_id
  JOIN publication ON publication.article_id=article.id
  LEFT JOIN reference ON reference.cited_article_id=article.id
WHERE
  article.type IN ('article','letter')
  AND organization.country='DE'
  AND publication.publication_year BETWEEN 2000 AND 2005
  AND reference.citing_article_id IS NULL
GROUP BY
  organization.name,
  publication.publication_year
;
```

5.6. Mean citation rate of journal packet (JCSm)

For this example I use the author as *unit of analysis*. I assume that the JSC (expected citation rates) is already pre-calculated and stored in the table *expected_citations* as depicted in Fig. 5.

Listing 5
Journal Citation Score.

```
SELECT
  journal_id,
  publication_year,
  article_type,
  AVG (cit_cnt) AS JCS
FROM
(
  SELECT
    journal.id AS journal_id,
    publication.publication_year,
    article.type AS article_type,
    article .id AS article_id,
    COUNT (DISTINCT citing_publication.article_id) AS cit_cnt
  FROM
    journal
    JOIN publication ON publication.journal_id=journal.id
    JOIN article ON publication.article_id=article.id
    JOIN reference ON reference.cited_article_id=article_id
    LEFT JOIN article citing_article
      ON citing_article.id=reference.citing_article_id
    LEFT JOIN publication citing_publication
      ON citing_publication.article_id=citing_article.id
      AND citing_publication.publication_year <= publication.publication_year+2
  WHERE
    publication.publication_year BETWEEN 2000 AND 2005
    AND citing_article.type IN ('article','letter','review','note')
  GROUP BY
    journal.id,
    publication.publication_year,
    article.type, article.id
)
GROUP BY
  journal_id,
  publication_year,
  article_type
;
```

Listing 6
Mean Citation rate of journal packet.

```
SELECT
  person_id,
  publication_year,
  AVG (citn_cnt) AS CPP,
  AVG (jcs_score) AS JCSm,
  AVG (citn_cnt)/AVG(jcs_score) AS CPP_per_JCSm
FROM
(
  SELECT
    person.id AS person_id,
    publication.publication_year,
    article.id,
    COUNT (DISTINCT citing_publication.article_id) AS citn_cnt,
    expected_citations.score AS jcs_score
  FROM
    person
    JOIN authorship ON authorship.person_id=person.id
    JOIN article ON article.id=authorship.article_id
    JOIN publication ON publication.article_id=article.id
    JOIN journal ON journal.id=publication.journal_id
    JOIN expected_citations ON expected_citations.journal_id=journal.id
      AND expected_citations.publication_year=publication.publication_year
      AND expected_citations.article_type=article.type
    LEFT JOIN reference ON reference.cited_article_id=article_id
    LEFT JOIN article citing_article ON citing_article.id=reference.citing_article_id
      AND citing_article.type IN ('article','letter','review','note')
    LEFT JOIN publication citing_publication
      ON citing_publication.article_id=citing_article.id
        AND citing_publication.publication_year
          <= publication.publication_year+expected_citations.window_size
  WHERE
    person.last_name='Garfield' AND person.first_name='Eugene'
    AND article.type IN ('article','letter','review','note')
    AND publication.publication_year BETWEEN 2000 AND 2005
    AND expected_citations.window_size=2
    AND expected_citations.self_cit_excl=0
  GROUP BY
    person.id,
    publication.publication_year,
    article.id,
    expected_citations.score
)
GROUP BY
  person_id,
  publication_year
;
```

The query is given in Listing 6 and works as follows: the inner SELECT calculates for each relevant article the number of citations and extracts the corresponding expected citation rates. In the outer SELECT the JCSm is calculated (and CPP/JCSm as bonus). The FCSm can be computed analogously, given that the fields are properly defined and the expected citation rates per field are already pre-calculated.

5.7. Number of co-publications

The query in Listing 7 shows how to calculate international co-publications. Here the number of Germany's international co-publications for the year 2000 is calculated.

Starting from *organization* (restricted in the WHERE clause to those located in Germany) all relevant tables are joined until the *publication* table is reached. This table is needed to restrict the results to the publication year 2000. To identify the cooperation partners a second JOIN with the *organization* table (and the tables between) is needed. In the WHERE clause a filter condition is applied, retaining only the partners from foreign countries.

5.8. Bibliometric networks

In this subsection I show how data of bibliometric networks can be extracted from the database. The extraction of co-citation network data from a relational database has already been shown (Small, 1995), so it will not be repeated here but queries for different types of networks are shown.

Listing 7
Number of international co-publications.

```
SELECT
  organization.name,
  publication.publication_year,
  COUNT (DISTINCT article.id)
FROM
  organization
  JOIN belongs_to ON belongs_to.organization_id=organization.id
  JOIN org_unit ON org_unit.id=belongs_to.org_unit_id
  JOIN affiliation ON affiliation.org_unit_id=org_unit.id
  JOIN authorship ON authorship.person_id=affiliation.person_id
    AND authorship.article_id=affiliation.article_id
  JOIN article ON article.id=authorship.article_id
  JOIN publication ON publication.article_id=article.id
  JOIN affiliation affiliation_partner
    ON affiliation_partner.article_id=affiliation.article_id
  JOIN org_unit org_unit_partner
    ON org_unit_partner.org_unit_id=affiliation_partner.org_unit_id
  JOIN belongs_to belongs_to_partner
    ON belongs_to_partner.org_unit_id=org_unit_partner.id
  JOIN organization organization_partner
    ON organization_partner.id=belongs_to_partner.organization_id
WHERE
  article.type IN ('article','review')
  AND organization.country='DE'
  AND publication.publication_year=2000
  AND organization_partner.country != organization.country
GROUP BY
  organization.name,
  publication.publication_year
;
```

The first query (Listing 8) shows how to extract co-authorship networks. This query extracts the co-authorship network data for the field of *Information and Library Science* in the period from 2000 to 2002. The JOINs to *journal_classification*, *classification* and *publication* are needed to select the relevant articles restricted by *field* and *publication_year*. Since all pairs of co-authors should be retrieved, the *authorship* and *person* tables are joined twice. In the SELECT clause in addition to the IDs and names of the authors the number of co-authored articles is reported, indicating the strength of the co-authorship links.

Listing 8
Co-authorship network data.

```
SELECT
  person1.id,
  person1.last_name,
  person2.id,
  person2.last_name,
  COUNT (DISTINCT article.id) artcl_cnt
FROM
  journal
  JOIN journal_classification
    ON journal_classification.journal_id=journal.id
  JOIN classification
    ON classificatioin.id=journal_classification.classification_id
  JOIN publication ON publication.journal_id=journal.id
  JOIN article ON publication.article_id=article.id
  JOIN authorship authorship1 ON authorship1.article_id=article_id
  JOIN person person1 ON authorship1.person_id=person_id
  JOIN authorship authorship2 ON authorship2.article_id=article_id
  JOIN person person2 ON person2.id=authorship.person_id
WHERE
  classification.field='Information and Library Science'
  AND publication.publication_year BETWEEN 2000 AND 2002
  AND authorship2.person_id != authorship.person_id
GROUP BY
  person1.id,
  person1.last_name,
  person2.id,
  person2.last_name
;
```

Listing 9
Bibliographic coupling.

```
SELECT
  ref1.citing_article_id as article1,
  ref2.citing_article_id as article2,
  COUNT (DISTINCT ref1.cited_article_id) as strength
FROM
  reference ref1
  JOIN reference ref2 ON ref1.cited_article_id=ref2.cited_article_id
    AND ref1.citing_article_id != ref2.citing_article_id
GROUP BY
  ref1.citing_article_id,
  ref2.citing_article_id
;
```

The next query (Listing 9) extracts network data based on bibliographic coupling. Bibliographic coupling is defined as a link between two articles established by the presence of a common reference in their bibliographies. The strength of the bibliographic coupling is given by the number of common references. The query joins the *reference* relationship with itself where the *cited* end points to the same article. In addition it is ensured that the *citing* ends are different. For practical purposes the articles should be restricted to some manageable subset of the database, e.g. by restricting them to a certain subfield.

6. Conclusions

In this article a general-purpose relational database schema for bibliometric applications is presented. Based on a broad overview of bibliometric applications, the fundamental requirements were gathered. The data modeling was conducted using entity-relationship modeling techniques based on the structural data commonly found in journal articles. Where appropriate, refinements have been done. The soundness of the developed database schema has been shown by several SQL queries for common bibliometric questions. While this database schema does not cover data for every specialized bibliometric question, it is however well suited to the typical day's work and it provides a solid ground for extensions towards other problems.

References

Butler, L., & Visser, M. S. (2006). Extending citation analysis to non-source items. *Scientometrics*, *66*(2), 327–343.

Callon, M., Courtial, J. P., Turner, W. A., & Bauin, S. (1983). From translations to problematic networks: An introduction to co-word analysis. *Social Science Information*, *22*(2), 191–235.

Chen, P. P. (1977). The entity-relationship model—A basis for the enterprise view of data. In *AFIPS national computer conference* (pp. 77–84).

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, *13*(6), 377–387.

Davis, P. M., & Fromerth, M. J. (2007). Does the arXiv lead to higher citations and reduced publisher downloads for mathematics articles? *Scientometrics*, *71*(2), 203–215.

Fernández, M. T., Cabrero, A., Zulueta, M. A., & Gómez, I. (1993). Constructing a relational database for bibliometric analysis. *Research Evaluation*, *3*(1), 55–62.

Gauffriau, M., Larsen, P. O., Maye, I., Roulin-Perriard, A., & von Ins, M. (2007). Publication, cooperation and productivity measures in scientific research. *Scientometrics*, *73*(2), 175–214.

Glänzel, W., & Schubert, A. (2004). Analyzing scientific networks through co-authorship. In H. F. Moed, W. Glänzel, & U. Schmoch (Eds.), *Handbook of quantitative science and technology research: The use of publication and patent statistics in studies of S&T systems* (pp. 257–276). Dordrecht, The Netherlands: Kluwer Academic Publishers.

Glänzel, W., Schubert, A., & Czerwon, H.-J. (1999a). A bibliometric analysis of international scientific cooperation of the european union (1985–1995). *Scientometrics*, *45*(2), 185–202.

Glänzel, W., Schubert, A., & Czerwon, H.-J. (1999b). An item-by-item subject classification of papers published in multidisciplinary and general journals using reference analysis. *Scientometrics*, *44*(3), 427–439.

Hood, W. W., & Wilson, C. S. (2003). Informetric studies using databases: Opportunities and challenges. *Scientometrics*, *58*(3), 587–608.

Larsen, P. O. (2008). The state of the art in publication counting. *Scientometrics*, *77*(2), 235–251.

Marx, W., Schier, H., & Wanitschek, M. (2001). Citation analysis using online databases: Feasibilities and shortcomings. *Scientometrics*, *52*(1), 59–82.

Moed, H. F. (1988). The use of on-line databases for bibliometric analysis. In L. Egghe, & R. Rousseau (Eds.), *Informetrics* (pp. 133–146). Netherlands: Elsevier.

Moed, H. F., De Bruin, R. E., & Van Leeuwen, T. N. (1995). New bibliometric tools for the assessment of national research performance: Database description, overview of indicators and first applications. *Scientometrics*, *33*(3), 381–422.

Neuhaus, C., Litscher, A., & Daniel, H.-D. (2007). Using scripts to streamline citation analysis on STN international. *Scientometrics*, *71*(1), 145–150.

Newman, M. E. J. (2001). Scientific collaboration networks. I. Network construction and fundamental results. *Physical Review E-Statistical, Nonlinear, and Soft Matter Physics*, *64*(1 II), 016131/1–116131/8.

Otte, E., & Rousseau, R. (2002). Social network analysis: A powerful strategy, also for the information sciences. *Journal of Information Science*, *28*, 441–453.

Rip, A. (1988). Mapping of science: Possibilities and limitations. In A. F. J. van Raan (Ed.), *Handbook of quantitative studies of science and technology* (pp. 253–273). North-Holland.

Shrum, W., & Mullins, N. (1988). Network analysis in the study of science and technology. In A. F. J. van Raan (Ed.), *Handbook of quantitative studies of science and technology* (pp. 107–133). North-Holland.

Simkin, M., & Roychowdhury, V. (2003). Read before you cite!. *Complex Systems*, *14*, 269–274.

Small, H. (1995). Relational bibliometrics. In M. E. Koenig, & A. Bookstein (Eds.), *Proceedings of the fifth biennial conference of the international society for scientometrics and informetrics* (pp. 525–532).

Small, H., & Garfield, E. (1985). The geography of science: Disciplinary and national mappings. *Journal of Information Science*, *11*, 147–159.

van Leeuwen, T. (2004). Descriptive versus evaluative bibliometrics. In H. F. Moed, W. Glänzel, & U. Schmoch (Eds.), *Handbook of quantitative science and technology research: The use of publication and patent statistics in studies of S&T systems* (pp. 373–388). Dordrecht, The Netherlands: Kluwer Academic Publishers.

van Leeuwen, T. (2007). Modelling of bibliometric approaches and importance of output verification in research performance assessment. *Research Evaluation*, *16*(2), 93–105.
van Raan, A. F. J. (2005). Fatal attraction: Conceptual and methodological problems in the ranking of universities by bibliometric methods. *Scientometrics*, *62*(1), 133–143.
Winterhager, M. (1992). Towards bibliometric objects: A relational view to ISI's science citation index. In A. F. J. van Raan, R. E. de Bruin, H. F. J. N. A. Moed, & R. W. J. Tijssen (Eds.), *Science and technology in a policy context* (pp. 21–34).
Wolfram, D. (2006). Applications of SQL for informetric frequency distribution processing. *Scientometrics*, *67*(2), 301–313.
Yu, H., Davis, M., Wilson, C. S., & Cole, F. T. H. (2008). Object-relational data modelling for informetric databases. *Journal of Informetrics*, *2*, 240–251.
Zitt, M., & Teixeira, N. (1996). Science macro-indicators: Some aspects of OST experience. *Scientometrics*, *35*(2), 209–222.