A Database Perspective of Social Network Analysis Data Processing

Mauro San Martín [*], [⋆] Claudio Gutierrez [⋆⋆]

Departamento de Ciencias de la Computación - Universidad de Chile
Avenida Blanco Encalada 2120 - Santiago - Chile.

Abstract

Interest in social networks is becoming pervasive and data volumes increase dramatically; however, current tools and models for data storage and manipulation in the area still lack established methodologies of the field of databases. For instance, there are no standard storage formats promoting reuse, sharing or combination of data sets from different sources, and available formats require adjustment of collected data to their representational capabilities, often excluding potentially useful data.

In this work we show the necessity and possibility of enhancing data management for social networks. The main technical challenge comes from the very nature of networked data and of the queries and analysis involved. We present preliminary results towards a data storage and manipulation model for social networks which natively supports attributed and dynamic multinets, using the full potentialities of standard database techniques.

Following Freeman's ideas on methodological aspects of social network analysis and based on current practices, we determine requirements, describe a suitable data workflow, and detect current limitations and needs. As a case-study we use DBLP, an online network of computer science authors and publications.

Key words: Data Management, Collecting Network Data, Academic Networks, Complete Networks, Data Representation

[⋆] M. San Martín was supported by a MECESUP scholarship and project FONDECYT N. 1030810.

[⋆⋆] C. Gutierrez was supported by project FONDECYT N. 1030810 and Proyecto Milenio P04-067-F, Center for Web Research.

[*] msanmart@dcc.uchile.cl

Article presented at Sunbelt XXVI 2006

1 Introduction

The prominence of social networks and their data volumes are increasing dramatically; however, current tools and models for data storage and manipulation in the area still lack established methodologies of the field of data management. To date most of the computer assisted social network analysis is being done with tools oriented solely to analysis itself, with little care about the inherent data management issues, like: data access in a proper level of abstraction, automatic collection, archiving and reuse, provision of a common ground to support network analysis over data incrementally collected from possibly different sources. Furthermore, the need for data management support is an explicit and urgent issue due to the need of automatic and continuous data collection for extensive analysis (Tsvetovat et al., 2005). This situation presents a number of open problems related to data management support of social network data, a situation that is occurring also in other fields that use network data too, like biosciences (Jagadish and Olken, 2003a; Gray et al., 2005).

As pointed out by Freeman (2004, ch. 1), the extensive work done by social network analysis community, since the 1930's (see also: Scott, 2000; Wasserman and Faust, 1994), has consolidated a characteristic data management workflow which is driven by a structural intuition, a systematic data collection and the use of visualization and mathematical models (Freeman, 2004).

In the past decades, *data/database models* has been devised by the database research community as the conceptual frameworks that provide the foundations to solve data management problems for a given domain. Social networks –independent of their origin– have common characteristics (Newman, 2003; Barabási and Bonabeau, 2003; Freeman et al., 1992) useful to provide a foundation for a common data model, as defined by Codd (1980), i.e. as a set of data structures, a collection of transformation and query operators, and integrity constraints.

The social network analysis data workflow can certainly benefit from data management techniques based on an appropriate data model.

Today there exists manifold *data models*, with different degrees of development, but they do not provide the needed support for social network analysis. For example, while the dominant *db-relational data model*[1] does not provide support to basic network operations (e.g. path finding and motif searching (Jagadish and Olken, 2003a)), other data models, like graph data models (Angles and Gutiérrez, 2005b) and semistructured data models (Abiteboul et al., 1999;

[1] In database literature this model is called *relational data model*; we call it in this paper *db-relational* to avoid confusions with *relational data*.

Suciu, 1998; Buneman, 1997), may offer a better support, but are not fully developed. A comprehensive review of recent years activity in database and data mining conferences, shows that database support for social networks, backed by a complete data model, remains an open problem.

In this work, we show how a specially tailored data model, on the lines of graph and semistructured data models, will benefit social network analysis. Our starting point is the Freeman et al. (1992) *maximal structure experiment* and the requirements collected from well known reference works like those of Wasserman and Faust (1994), Scott (2000) and Carrington et al. (2005), from recently published works (Butts, 2001; Jin et al., 1998; Newman and Park, 2003; Dodds et al., 1998), and from the features of existing computational tools, for instance, Pajek and Ucinet (see Huisman and van Duijn, 2005), and NetIntel and DyNetML (Tsvetovat et al., 2005, 2004).

Our main contribution is the definition of an improved data workflow for social network analysis based in an specialized social networks data model. To build this proposal we made an extensive survey of database support for social networks and an analysis on reported current practices in social network analysis, based on a sample of the works published in last three years in the journal Social Networks.

This work is organized as follows. In Section 2, we introduce data management topics and their relevance. In Section 3, we analyze the current social network analysis workflow from a data management perspective. We identify problems and opportunities, which are illustrated in the next section with use cases from DBLP. In Section 5, we summarize the main issues and benefits of a social network analysis data model, and state its main requirements. In the final section, conclusions and guidelines for further work are presented.

2 Data Management, Data Models and Databases

Even though not-automated data management is possible, it is practical only for small amounts of data. We have found enough evidence showing that social networks analysis has reached the point where automated data management is mandatory. Computer assisted management of data involves both defining structures for storage of data and providing mechanisms for its manipulation. In addition, the integrity and security of information stored must be ensured, even against system crashes or attempts at unauthorized access. Also, if data are to be shared among several users, the system must deal with a concurrency mechanism to avoid possible anomalous results (Silberschatz et al., 2001, ch. 1).

Tsichritzis and Lochovsky (1982) motivate data models as follows: “A perception of the world can be regarded as a series of distinct although sometimes related phenomena. From the dawn of time human beings have shown a natural inclination to try to describe these phenomena in some fashion whether they understand them completely or not. These descriptions of phenomena will be called data. Data correspond to discrete, recorded facts about phenomena from which we gain information about the world. Information is an increment of knowledge that can be inferred from data. It is apparent that an interpretation of the world is needed which is sufficiently abstract to allow minor perturbations, yet is sufficiently powerful to give some understanding concerning how data about the world are related. A data model is a model about data by which a reasonable interpretation of the data can be obtained.”

In practice, different data models are developed for different data managing problem domains, providing abstraction from low level data manipulation issues, and letting users specify data managing solutions in their own semantic context. For example: tabular relations for administrative data, objects for graphical elements, XML (extensible markup language) for documents, etc. . Thus, data models are the underlying structure of a database, a collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints (Silberschatz et al., 2001, ch. 1).

We are using here the expression *data model* in the sense of Codd (1980), i.e. consisting of three components:

(1) A collection of data structure types (the building blocks of any database that conforms to the model).
(2) A collection of operators or inferencing rules, which can be applied to any valid instances of the data types listed in (1), to retrieve or derive data from any parts of those structures in any combinations desired.
(3) A collection of general integrity rules, which implicitly or explicitly define the set of consistent database states or changes of state or both –these rules may sometimes be expressed as insert-update-delete rules.

Furthermore, as presented above, data models have been devised for computer-oriented representation of information.They are powerful conceptual tools for the organization and representation of information, yet they are translatable into structures that can be manipulated by computers (Tsichritzis and Lochovsky, 1982).

A database is a data collection that has a data source, some degree of interaction with real world events, and an audience which is actively interested in the database contents (Elmasri and Navathe, 2000, ch. 1).

A Data-Base Management System (DBMS) is a set of programs, designed under a given data model, to access a database. Its primary goal is to provide a way to store and retrieve database information that is both convenient and efficient (Silberschatz et al., 2001). Along with the database, there is usually stored a description of its contents (metadata) that allows the DBMS to manipulate the database, provided that both, DBMS and database, were defined under the same data model. This description is known as the database schema. During the operation of the DBMS a particular database goes through different states; each one of them is known as a database instance. Figure 1 depicts the relation between data models, database schemas, databases, and DBMSs.

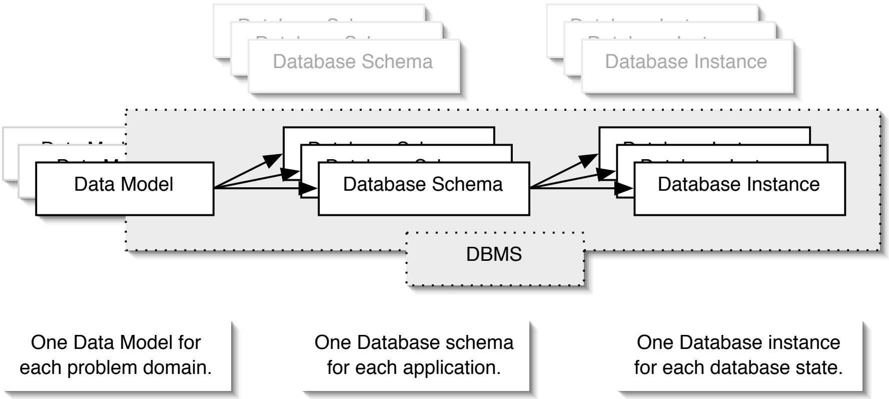
Database Schema
Database Instance
Data Model
Database Schema
Database Instance
DBMS
One Data Model for each problem domain.
One Database schema for each application.
One Database instance for each database state.

Fig. 1. Relation between Data Models, Schemas, Instances and DBMSs

With this approach, users do not need to program each required operation directly over the data, but interact with the data through the DBMS, which in turn handles all the low level operations on physically stored data (see Figure 2).

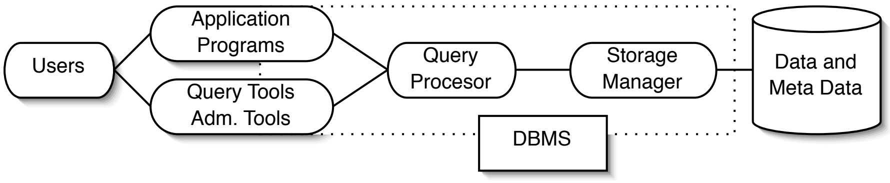

Fig. 2. Typical DBMS Architecture

The main benefits of a DBMS are:

- Independence between data and programs.
- Representation of complex relations between data.

- Data abstraction: A DBMS offers to users a conceptual representation of data, which hides the details about storage implementation.
- Different views over data to support the particular needs of different users (multiple user interfaces).
- Data interoperation and reuse: different data sources could be used as a consolidated database, and a database could be used in a different time and by different users, as long as enough metadata is available.
- Enforcement of integrity constraints to avoid misuse or degradation of data.
- Support for simultaneous users (access control and concurrent access).
- Backup and recovery to avoid loosing information.

3 Social Network Analysis Data Workflow

Social network study and analysis has a long tradition. This is one of the reasons it has a well defined set of data manipulation needs, which in turn support the idea of a dedicated data model.

3.1 Current Social Network Analysis Workflow

Freeman (2004) discusses the history of social networks analysis since long before Moreno started his works in the early 1930s, arguing that the foundational aspects were already present. Those aspects are: a structural intuition, a sustained effort towards systematic data collection, and the development of visualization devices and mathematical and computational models.

Given the amount of data and the processing tools available at the time, one of the main technical challenges was data reduction without loosing of relational meaning. Still today, there is a bias towards the feasibility of analysis instead of the richness of data.

Social network analysis is centered on analysis of data about relations, *relational data*. The common social network analysis practice (Wasserman and Faust, 1994; Scott, 2000; Hanneman and Riddle, 1994) follows the data workflow showed in Figure 3 .

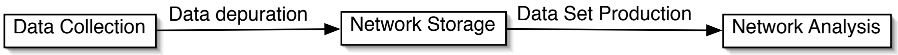

Fig. 3. Current SNA workflow

- Data Collection
  In this first stage a measurement unit is chosen, i.e., individuals or some kind of groups. Then a data collection device is designed and used to gather

relational data, for example, a form to record direct observation or a survey. In this stage, storage media varies greatly, from paper and pencil to some general purpose computational tools like word processors and spreadsheets. At the end of this stage data is checked and prepared to be stored as a network. From a data management point of view the main problem of the current practice is that data collection occurs as discrete events and, usually, there is not enough information to allow reuse and integration of data from different events or experiments.

- Network Storage
  After depuration, the data stored is complete from the perspective of the particular ongoing structural study. Some data manipulation is needed to obtain a network in terms of the desired unit of analysis (possibly different from the measurement unit). Given that interpretation of data (as metadata or a schema) is not available in standard terms, a set of custom made computer programs are needed to perform this manipulation. At the end of this stage, multiple networks may be produced, based on the stored data, in formats that an analysis program can understand. Some data management problems are evident: often collected data do not contain provenance metadata or it is just discarded after data sets production; reduction programs are related to specific data collections and not properly documented, and much information is lost in the reduction process, so most data collection effort is wasted; even when some data is preserved, almost no metadata is attached to it, rendering these data useless at large.
- Network Analysis
  Over each data set produced, typically in the form of one or two-mode networks, and ego-centered networks, some analysis is performed, and as results network, group and/or actor measurements are produced and interpreted, but they are not used to enrich the knowledge in the networks under analysis. As each data set require an ad-hoc program to be generated, each process is expensive, and hard to validate by others. Furthermore, provenance data is neither attached to data sets nor to results themselves making data sets and results hard to reuse in other studies or as reference.

3.2 Current Database Support for the SNA Data Workflow

Today, information management and database support for social networks is very limited. Most research conducted in social networks is performed with general purpose applications, like spreadsheets, or with software tools specially developed for specific tasks, often coded by someone in the research team itself (Jagadish and Olken, 2003b). The researcher must get involved in data preparation at a low level, in a per file and per experiment basis. This situation has a bad impact in environments with high data throughput and in interoperation tasks (Gray et al., 2005). It is also hard to validate new methods because standard data sets, usually hand curated, tend to be too small and simple.

Computational tools and data formats surveyed, which provide network storage and manipulation, rely directly on the filesystem of the operating system for data storage –with only few exceptions. These data file formats are based on adjacency matrices, lists of nodes and edges, or a combination of both. Some file structures are positional and others use different kinds of markup languages. Of these, just a few are supported by semistructured schemas (e.g. GraphML, DynetML and SBML). Formats surveyed were DIMACS, Dynagraph, DynetML, GraphML and SBML, and those used by Mage, Matlab, NetDraw, NetMiner, Pajek, STRUCTURE and UCINET.

Additionally, note that these file formats are suitable for network manipulation and analysis only in main memory. However, if the data set (a network or a set of networks) does not fit in main memory, internal memory graph algorithms like the ones described by Brandes and Erlebach (2005) are not efficient. There exists a relevant literature on data structures and algorithms for graphs in external memory (see e.g. Katriel and Meyer (2002); Sanders (2002); Pagh (2002)).

Among the few tools that use DBMS to some extent, NetIntel (Tsvetovat et al., 2005) uses a *db-relational* database to store relational data. Nevertheless, most network operations are performed outside the DBMS, loosing in this way most of the benefits of using a DBMS in the first place.

Instead of exposing increasingly complex file structures to the user, and forcing her to develop her own tools directly on them, it would be better to provide a higher level of service, raising the abstraction level of tools and languages. What is needed is database support for social networks analysis, via a data model.

In order to determine the current state of database support for social networks we surveyed the last editions of the main database and data mining conferences and associated events:

- CAiSE (2003): Conference on Advanced Information Systems Engineering.
- DMKD (2000-2004): Workshop on Research Issues on Data Mining and Knowledge Discovery.
- ICDT (1999-2005, biannual): International Conference on Database Theory.
- SIGKDD (2001-2004): ACM Special Interest Group on Knowledge Discovery and Data Mining.
- SIGMOD/PODS (2000-2005): ACM Special Interest Group on Management of Data / Principles Of Database Systems.
- VLDB (2000-2004): Very Large Databases Conference.
- WebDB (2001-2005): International Workshop on the Web and Databases.
- XSym (2003-2004): International XML Database Symposium.

From all research presented in these events, no single work addresses the topic of data management for social networks. There were only some partial references to graph or semistructured data models (e.g. Hidders (2003); Milo and Suciu (1999); Grahne and Thomo (2001); Kaushik et al. (2002); Yan et al. (2004, 2005); Wang et al. (2005); Topaloglou et al. (2004)). References on social networks were related to data mining on general graphs, complex networks and/or social networks (e.g. Dom et al. (2003); Cohen and Gudes (2004); White and Smyth (2003); Yan and Han (2003); Hopcroft et al. (2003); Noble and Cook (2003); Desikan and Srivastava (2004); Faloutsos et al. (2004); Horváth et al. (2004); Jeh and Widom (2004); Wang et al. (2004); Wu et al. (2004)).

This survey shows that a data model for social networks remains an open problem.

3.3 SNA Tools and Data Management

Huisman and van Duijn (2005) provide a complete review of 28 software tools and libraries for social network analysis. For each of the 6 most relevant tools, they discuss these capabilities: data entry and manipulation, visualization techniques, descriptive methods, procedure based analysis, and statistical modeling. Based on this review we conclude that, as analysis oriented tools, their data manipulation focus, as expected, is not on data management but in storing the data sets which are ready to be analyzed. However, it is often the case that more sophisticated data manipulation functions are needed, for instance, the generation of alternate analysis data sets –e.g. for different analysis units– from the same collected data. These tools do not provide support for this or other operations belonging to the first two stages of the data workflow (see Figure 3).

To some extent these pre-analysis operations can be performed manually or with ad-hoc programs but that approach is expensive, specially in large and complex data sets, and it difficults interoperation and reuse, as well as keeping adequate provenance metadata.

4 Use cases: DBLP

To illustrate the pitfalls of current data workflow, in this section we presents several typical use cases using DBLP as data set. DBLP (Ley, 2002), which today stands for "Digital Bibliography & Library Project", is a bibliographic project for scientific publications in computer science, with more than a decade of history (see http://dblp.uni-trier.de/). The DBLP database indexes more than 725000 articles and it contains also some information about authors (i.e. names and links to home pages). This database is in XML format and it is publicly available in the DBLP website (a DTD[2] schema is also available).

The DBLP schema is organized around different kinds of documents or publications, each of which in turn have several attributes. For the sake of clarity, Figure 4 depicts a simplified version of this schema, showing only the relations between the object *article* and its attributes, when every kind of publication –*inproceedings*, *article*, *proceedings*, *book*, etc.– is connected to each possible attribute. Each publication object has a unique id in the database.

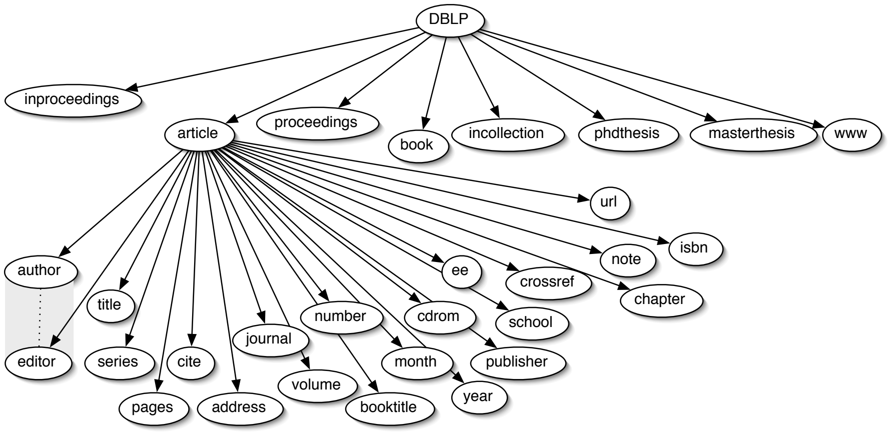

Fig. 4. Partial representation of DBLP XML Schema

DBLP can be seen as a multimode multinetwork with attributes and time information from which it is possible to extract for analytical purposes, different one and two mode networks, as well as more complex structures. See for example figure 5, for the ego-centered multinetwork around publication with key=*journals/algorithmica/NavarroB01.* It is possible to extract from it, for example, a one-mode coauthorship network, or a two-mode affiliation network of authors and journals as events, see Figure 6.

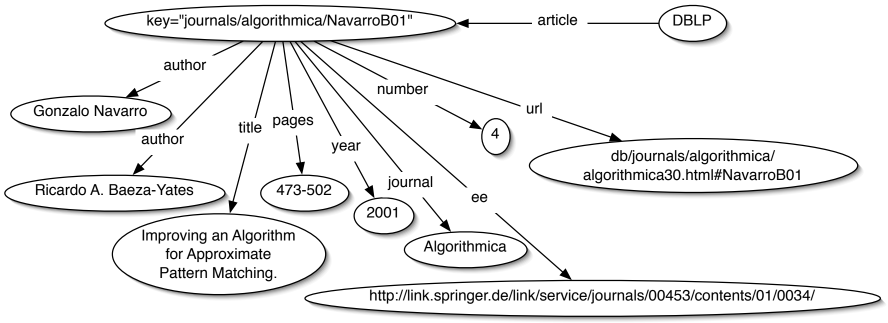
key="journals/algorithmica/NavarroB01"
article
DBLP
author
Gonzalo Navarro
author
Ricardo A. Baeza-Yates
title
Improving an Algorithm for Approximate Pattern Matching.
pages
473-502
year
2001
journal
Algorithmica
number
4
url
db/journals/algorithmica/ algorithmica30.html#NavarroB01
ee
http://link.springer.de/link/service/journals/00453/contents/01/0034/

Fig. 5. The network around an article in DBLP.

[2] Document Type Definition.

There exist in the database some implicit relations, that are not part of the schema, but that are potentially interesting, like persons being authors and also editors, and keyword occurrence across the titles.

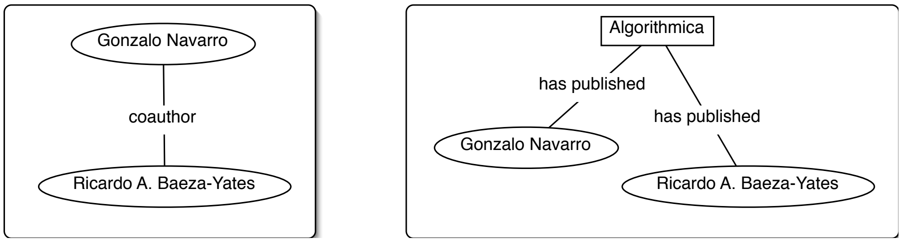

Fig. 6. One mode and two mode networks from network in figure 5

.

We will proceed now to discuss two use cases for each stage of the social network data workflow presented in Figure 3.

4.1 Data Collection

(1) Digital survey
One of the most time consuming tasks in the DBLP project has been data collection. Most of it has been done as part of the project itself by hired students. If there was a DBMS it would be possible to implement a digital survey system in which, for example, each interested publisher could upload the bibliographic information of its most recent publications. Such system requires that the DBMS, and the underlying data model, support automatic provenance data collection, incremental data collection, integrity constraints enforcement, concurrent data access, access control, and the development of user friendly interfaces.

(2) Previously collected data
It has been also the case that a publisher which has its own publications database grants permission to integrate it into DBLP. This has been done through extensive analysis and ad-hoc programs to translate the new data corpus to the format of DBLP. If there were explicit data schemas it would be possible to automate this process, even to the point that updates to the publisher database could be batch processed and automatically integrated to DBLP as they become available. Such integration requires that both databases have interoperable schemas or metadata, and it would be also desirable to have extensive provenance metadata.

4.2 Data Storage and Manipulation

(1) Groups definition and identification
The main user interface for DBLP is its web site, which offers different

views of DBLP data, for example: a page per author with all its publications, or pages per journals issues with their tables of contents. To build these views it is needed to extract from the database groups of information that meet some criteria. This requires operations to select portions of the data given a set of criteria.

(2) Data Maintenance
One of the hardest problems that the DBLP team must manage is author identification. In bibliographic databases authors are identified by their names, but names are not unique ids for persons, and worse, the same person may write down his or her name in different ways. Thus, in DBLP it sometimes happens that the same author has different keys, or different authors that share some name spelling share the same key, looking like only one person to DBLP. For example, if the DBLP team discovers that in the database there is only one author identified by the name "John Doe", but there actually are 2 authors with this name, the team should create a new author id, determine which publications belong to each author, and finally establish the correct relations. This requires operations to insert, delete and update actors and relations, and operations to select and identify the groups that will be object of these operations; integrity constraints must be enforced too before and after these operations.

4.3 Production of Data Sets for Analysis

(1) Different levels of analysis
DBLP has multiple modes and relations between those different sets of units. It may be desirable to explore the structure of the network at different levels, applying well established analytical tools. This involves at least two tasks related to preparing the data sets for analysis: to build units of analysis from measurement units, and to compute relational data between units of analysis. Consider for example these three levels of analysis: author and coauthorship relation, articles (as set of coauthors) related to all articles that share an author, and journals (as supersets of coauthors) related to all journals that published articles by the same authors. The DBMS should provide graph transformation operations that build the desired network from the collected data following some given criteria.

(2) User defined operations
In some cases, basic network manipulation operations (like those described above) are not enough for some domain specific data manipulation needs. Consider the case when a researcher wants to select from the database all actors that have a measure over some threshold for a given measurement. For example, a researcher wants to test a new centrality measure he or she has recently defined analyzing the network produced by selecting all authors from a coauthorship network, derived from DBLP, that has the new centrality over 0.5. This requires that there exists the

possibility and the means to define and use non standard operations in the DBMS.

It is clear that the requirements motivated by these use cases are in the same lines of the benefits provided by a DBMS, plus more specialized requirements related to network manipulation operations. We interpret this as evidence of the necessity of properly defined data management tools, i.e. a data model for social networks and a corresponding DBMS.

5 Social Networks Data Base Model: Issues and Benefits

To determine the requirements we analyzed the current SNA data workflow, considered use cases in the lines of those presented in the previous section, and we set as reference the Freeman's *maximal structure experiment*.

5.1 Maximal Structure Experiment

As a framework to define what would be desirable in terms of data management support for social network analysis we borrow the idea of *maximal social structure experiment* from Freeman et al. (1992, ch. 1). Freeman starts from the simplest case: a single relation recorded at a single time over an undifferentiated and unchanging population; defining an experiment which uses two kinds of information:

- A set of social units (which at the lowest possible level refers to individuals, i.e. persons).
- A set of pairs of social units that exhibit some social relation of interest between members of each pair.

From this basic setting, Freeman progressively builds the maximal social structure experiment adding the following elements:

(1) More than a single relation.
(2) Two or more types or levels of social units.
(3) Structures that changes through time.
(4) Sets of social units that grow or shrink.
(5) Attributes of social units.
(6) Attributes that change.

We think that it is possible to give support to the notion of maximal social structure experiment with an adequate data model, improving the first two stages of social network analysis data workflow: data collection, and data storage and manipulation; which in turn will leverage the possibilities for the analysis stage.

5.2 SNA DB Model Issues and Main Requirements

Our main goal is to design a data model at the proper level of abstraction to be compelling to SNA practitioners but still computationally viable. The main challenges to achieve this goal are the computational complexity of the desired model, and the reported lack of interest of potential users on sophisticated data managing tools despite the arguments in favor (Gray et al., 2005). It is expected that the special properties of social networks as complex networks –sparseness, low diameter, power law degree distribution and a high clustering coefficient– give design advantages over a generic graph database model, allowing the use of algorithms which are not usable in a more general context.

The main requirements for such a data model and DBMS follow from the use cases and the SNA workflow, and coincide with the usual benefits from dedicated DBMS:

- Support for storage of relational data –understood in the context of the *maximal structure experiment*– and its metadata (i.e. meaning and provenance).
- Incremental data collection and longitudinal data.
- Integrity constraints enforcement.
- Basic network manipulation operations.
- User defined network manipulation operations.
- Data export to analysis tools.
- Concurrent access.
- Access Control.

In particular, a data model for social network analysis should provide:

(1) Data structure types to represent networks, its components and relations.
(2) Operations to perform network data manipulation (selection, insertion, deletion and update) over units, groups and entire networks. It must be possible to compose these operations to build more elaborate and domain dependent operations over the network; for instance, for centrality computation.
(3) Integrity constraints to keep the network consistent as a network, and under domain specific restrictions.

Among the existing data models (Navathe, 1992), not all of them have the potencial to satisfy the requirements of a data model for social networks. Even though it is possible to store nodes and edges in relations of the db-relational data model, its set of operations does not provide support for network oriented data manipulation. Path finding, for instance, reduces to an undetermined number of joins of a "edge relation" over itself which makes it unfeasible under this model. Also, the ability of object oriented data models to support many classes of objects, each one with their own methods, do not offer any special advantage for social networks which are homogeneous, with all actors belonging to a small number of classes, and where operations are oriented to the structure.

Natural candidates to provide the needed support, given the intrinsic nature of network data, are graph data models (Angles and Gutiérrez, 2005b) and semistructured data models, like RDF (Angles and Gutiérrez, 2005a). However, there is no complete and implemented model of these types, nor a model that provides the required query specificity. A bottom-up approach, starting from a well defined and domain specific model like the one of social networks, can use the possible implementation advantages due to special properties of complex networks, and borrow the conceptual framework of these general models.

Summarizing, a social network data model should fulfill these general requirements: explicitly support of the structure, operations and constraints over network data, and promoting interoperation and reuse.

5.3 Expected Benefits of an Improved SNA Workflow

A specific data model, with a graph database point of view, will improve the support for the automation of data managing in the social network analysis field. As a result it is expected that productivity will improve, as in the well documented case of the *db-relational* data model and its application domain. (The argument is that a data model will let users and programmers to access data in an abstraction layer similar to the one used in the application domain, with all low level data managing hidden by the database management system (Codd, 1982).)

Hence, the availability of a data model and a DBMS for social network analysis should have a deep impact in its current data workflow (see Figure 7). The benefits from this improved workflow include: independence between data and programs, data interoperation and reuse, incremental and automated data collection, data security, redundancy control and automated integrity checking.

- Data Collection
  Data collection will still be based on discrete events but the data of each of them would be automatically integrated in the database as an incremental

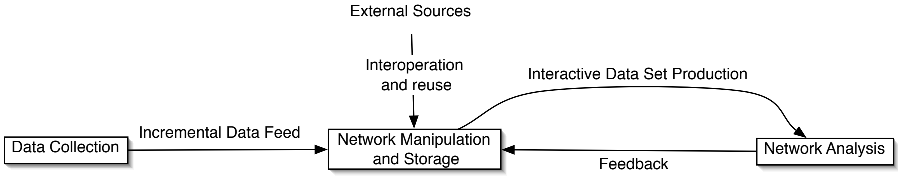

Fig. 7. Improved SNA workflow

feed. In this way the stored network will grow in size and detail over time. In addition, given the improved data manipulation capabilities, the measurement unit of choice should be the simplest available (the lowest possible level) leaving to the DBMS the building of more complex aggregations. The integration in the database of previously collected data from other sources could be automated too, if the adequate metadata is available. The DBMS will assure that integrity of data is preserved in every state of the database.

- Data Storage and Manipulation
  Data can be maintained and curated with the help of available operations in the data model. Furthermore, the relation with the analysis stage is a cycle: different data sets are generated as needed and the analytical results can be integrated to enrich the information in the database.
- Network Analysis
  This stage will keep using the well established tools of the field, but will be leveraged by the services provided to the previous stages by the DBMS.

6 Conclusions and Further Work

Our analysis of publications on Social Networks shows that there exists a characteristic social network analysis data workflow that is susceptible of improvement. Furthermore, the requirements not fulfilled by the current workflow are in the lines of data management issues solved by a DBMS. However, as our survey of database support for social networks shows, there is no specific database support for social networks. Current increasing trends in the data volumes of networks under study deepen even further the need for automated data management infrastructure in social network analysis, i.e. a social network DBMS.

Any DBMS requires as a foundation a full developed data model defined for a specific data managing problem domain. Social networks have an adequate set of characteristics to build a data model for them. Thus, it is needed and possible to build a data model for social networks to improve the data management workflow of social network analysis.

We are currently working on the complete specification of requirements for a social networks data model, and in its definition. We plan to implement it using RDF and to explore the possibilities of expanding the data management workflow to a knowledge management workflow.

References

Abiteboul, S., Buneman, P., Suciu, D., 1999. Data on the Web: From Relations to Semistructured Data and XML. Morgan Kaufman Ed.

Angles, R., Gutiérrez, C., 2005a. Querying RDF data from a graph database perspective. In: ESWC. pp. 346–360.

Angles, R., Gutiérrez, C., August 2005b. Survey of graph database models. TR/DCC2001-10 DCC. Computer Science Department Technical Report, Universidad de Chile.

Barabási, A.-L., Bonabeau, E., May 2003. Scale free networks. Scientific American, 50–59.

Brandes, U., Erlebach, T. (Eds.), 2005. Network Analysis: Methodological Foundations [outcome of a Dagstuhl seminar, 13-16 April 2004]. Vol. 3418 of Lecture Notes in Computer Science. Springer.

Buneman, P., 1997. Semistructured data. In: PODS. ACM Press, pp. 117–121.

Butts, C. T., 2001. The complexity of social networks: theoretical and empirical findings. Social Networks 23, 31–71.

Carrington, P. J., Scott, J., Wasserman, S. (Eds.), 2005. Models and Methods in Social Network Analysis. Vol. 27 of Structural Analysis in the Social Sciences. Cambridge.

Codd, E., 1980. Data models in database management. In: Workshop on Data abstraction, databases and conceptual modeling. pp. 112–115.

Codd, E., February 1982. Relational database: A practical foundation for productivity. Communications of the ACM 25 (2), 109–117.

Cohen, M., Gudes, E., 2004. Diagonally subgraphs pattern mining. In: Workshop on Research Issues on Data Mining and Knowledge Discovery proceedings. pp. 51–58.

Desikan, P., Srivastava, J., 2004. Mining temporally evolving graphs. In: WEBKDD proceedings. pp. 13–22.

Dodds, P., Muhamad, R., Watts, D. J., June 1998. Collective dynamics of 'small-world' networks. Nature 393, 440–442.

Dom, B., Eiron, I., Cozzi, A., Zhang, Y., 2003. Graph-based ranking algorithms for e-mail expertise analysis. In: Workshop on Research Issues on Data Mining and Knowledge Discovery proceedings. pp. 42–48.

Elmasri, R., Navathe, S. B., 2000. Fundamentals of Database Systems, 3rd Edition. Addison Wesley Longman.

Faloutsos, C., McCurley, K. S., Tomkins, A., 2004. Fast discovery of connection subgraphs. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 118–127.

Freeman, L. C., 2004. The Development of Social Network Analysis. Empirical

Press.

Freeman, L. C., Romney, A. K., Douglas R. White, e., 1992. Research Methods in Social Network Analysis. Transaction Publishers.

Grahne, G., Thomo, A., 2001. Algebraic rewritings for optimizing regular path queries. In: International Conference on Database Theory proceedings. pp. 301–315.

Gray, J., Liu, D. T., Nieto-Santisteban, M. A., Szalay, A., DeWitt, D. J., Heber, G., 2005. Scientific data management in the coming decade. SIGMOD Record 34 (4), 34–41.

Hanneman, R. A., Riddle, M., 1994. Introduction to social network methods. University of California, Riverside.

Hidders, J., 2003. Typing graph-manipulation operations. In: International Conference on Database Theory proceedings. pp. 394–409.

Hopcroft, J., Khan, O., Kulis, B., Selman, B., 2003. Natural communities in large linked networks. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 541–546.

Horváth, T., Gärtner, T., Wrobel, S., 2004. Cyclic pattern kernels for predictive graph mining. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 158–147.

Huisman, M., van Duijn, M., March 2005. Software for social network analysis. Models and Methods in Social Network Analysis.

Jagadish, H. V., Olken, F. (Eds.), November 2003a. Data Management for the Biosciences: Report of the NSF/NLM Workshop on Data Management for Molecular and Cell Biology at the National Library of Medicine February 2-3, 2003.

Jagadish, H. V., Olken, F., 2003b. Database Management for Life Science Research: Summary Report of the Workshop on Data Management for Molecular and Cell Biology at the National Library of Medicine, Bethesda, Maryland, February 2-3, 2003. OMICS 7 (1), 131–137.

Jeh, G., Widom, J., 2004. Mining the space of graph properties. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 187–196.

Jin, E. M., Girvan, M., Newman, M., June 1998. Collective dynamics of 'small-world' networks. Nature 393, 440–442.

Katriel, I., Meyer, U., 2002. Elementary graph algorithms in external memory. In: Algorithms for Memory Hierarchies. pp. 62–84.

Kaushik, R., Bohannon, P., Naughton, J. F., Korth, H. F., 2002. Covering indexes for branching path queries. In: ACM SIG on Management of Data proceedings. pp. 133–144.

Ley, M., 2002. The DBLP computer science bibliography: Evolution, research issues, perspectives. In: Laender, A. H. F., Oliveira, A. L. (Eds.), SPIRE. Vol. 2476 of Lecture Notes in Computer Science. Springer, pp. 1–10.

Milo, T., Suciu, D., 1999. Index structures for path expressions. In: International Conference on Database Theory proceedings. pp. 277–295.

Navathe, S. B., September 1992. Evolution of data modeling for databases. Communications of the ACM 35 (9), 112–123.

Newman, M., 2003. The structure and function of complex networks. SIAM Review 45 (2), 167–256.

Newman, M., Park, J., May 2003. Why social networks are different from other types of networks. arXiv 393.

Noble, C. C., Cook, D. J., 2003. Graph-based anomaly detection. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 631–636.

Pagh, R., 2002. Basic external memory data structures. In: Algorithms for Memory Hierarchies. pp. 14–35.

Sanders, P., 2002. Memory hierarchies - models and lower bounds. In: Algorithms for Memory Hierarchies. pp. 1–13.

Scott, J., 2000. Social Network Analysis, 2nd Edition. SAGE Publications.

Silberschatz, A., Korth, H. F., Sudarshan, S., 2001. Database System Concepts, 4th Edition. McGraw-Hill.

Suciu, D., December 1998. An overview of semistructured data. ACM SIGACT News 29 (4), 28–38.

Topaloglou, T., Davidson, S. B., Jagadish, H. V., Markowitz, V. M., Steeg, E. W., Tyers, M., 2004. Biological data management: Research, practice and opportunities. In: VLDB. pp. 1233–1236.

Tsichritzis, D. C., Lochovsky, F. H., 1982. Data Models. Prentice-Hall Inc.

Tsvetovat, M., Diesner, J., Carley, K. M., March 2005. Netintel: A database for manipulation of rich social network data. CMU-ISRI-04-135.

Tsvetovat, M., Reminga, J., Carley, K. M., 2004. Dynetml: Interchange format for rich social network data. CMU-ISRI-04-105.

Wang, C., Wang, W., Pei, J., Zhu, Y., Shi, B., 2004. Scalable mining of large disk-based graph databases. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 316–325.

Wang, W., Wang, C., Zhu, Y., Shi, B., Pei, J., Yan, X., Han, J., 2005. Graph indexing: A frequent structure-based approach. In: ACM SIG on Management of Data proceedings. pp. 879–882.

Wasserman, S., Faust, K., 1994. Social Network Analysis: Methods and Applications, 1st Edition. Structural Analysis in the Social Sciences. Cambridge University Press.

White, S., Smyth, P., August 2003. Algorithms for estimating relative importance in networks. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 266–275.

Wu, A. Y., Garland, M., Han, J., 2004. Mining scale-free networks using geodesic clustering. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 719–724.

Yan, X., Han, J., 2003. CloseGraph: Mining closed frequent graph patterns. In: ACM SIG Knowledge Discovery and Data Mining proceedings. pp. 286–295.

Yan, X., Yu, P. S., Han, J., 2004. Graph indexing: A frequent structure-based approach. In: ACM SIG on Management of Data proceedings. pp. 335–346.

Yan, X., Yu, P. S., Han, J., 2005. Substructure similarity search in graph databases. In: ACM SIG on Management of Data proceedings. pp. 766–777.