

# DYNETML: INTERCHANGE FORMAT FOR RICH SOCIAL NETWORK DATA

MAKSIM TSVETOVAT, JEFF REMINGA, AND KATHLEEN M. CARLEY

## 1. INTRODUCTION

Current state of the art in social network data representation presents a fairly bleak picture. Each of analysis and simulation packages uses its own, proprietary and incompatible data format. Some of the file formats do not even have a specification document, making the files unreadable without the software that produced them.

Data formats that were designed for interoperability (such as DL) are rarely expressive enough to fully represent the datasets.

As a result, most researchers are forced to deal with data interchange in a makeshift fashion, at best increasing the workload and at worst resulting in loss of data integrity.

To improve cooperation between researchers and to promote interoperability of software, the community needs to agree on a common data interchange language. In an informal meeting at CASOS 2002, a number of prominent developers and users of social network analysis tools have agreed on cooperating with development of such interchange language and supporting it when it is available.

This paper presents a proposal for an XML-derived language that addresses requirements for expressivity and compatibility. We proceed to outline our vision for development of social network analysis toolchains which will exponentially increase the analysis power at the fingertips of researchers.

## 2. PITFALLS OF THE EXISTING TOOLS

As we mentioned above, the current social network data formats have a number of deficiencies:

**Binary files** are very difficult to read if exact specification of the file format is not provided. Significant extra efforts are required to keep compatibility with other tools or between versions of the same tool.

**Multiple files** used for specification of rich data or saving analysis output present a number of problems. First of all, there is a significant potential for data loss due to misplaced or corrupted files (for example, while sent through email). Secondly, a consistent naming scheme for all files and a file catalogue are required

---

This work was supported in part by the National Science Foundation under the IGERT program for training and research in CASOS, and the NSF KDI 00-142 1-5-31737, NSF ITR 1040059 and the Office of Naval Research N00014-02-1-0973. Additional support was provided by CASOS - the center for Computational Analysis of Social and Organizational Systems at Carnegie Mellon University. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Research, the National Science Foundation, or the U.S. government.

to prevent data loss - which requires a certain amount of discipline on the part of the researcher (as these features are not included in the analysis software)

**Raw Data** file such as binary matrices or edge lists lack the expressiveness required to represent multiple relations between nodes or evolution of social networks over time.

**Human-Readable Data** in text files or spreadsheets solves the expressivity problem but requires extensive post-processing by hand or with post-processing scripts. However, these programs often represent the weakest link in the software chain (due to hasty design and dependence on outside tools such as Perl or Awk).

### 3. REQUIREMENTS FOR DATA INTERCHANGE

In light of the problems outlined above, we proceed to define requirements for a universal data interchange format that would make much easier the task of exchanging rich social network data and improving compatibility of analysis and visualization tools.

- (1) The data interchange format shall be contained in human-readable text files that are in the same time easily parsable by computers.
- (2) The data interchange format shall allow an entire dataset, complete with all computed measurements, to be stored in one file
- (3) The data interchange format shall provide maximum expressive power to its users, allowing:
  - Typed nodes (types may include "person", "resource", "organization", "knowledge", etc)
  - Multiple sets of nodes of the same type (to express multiple units within the company, etc)
  - Multiple typed attributes per node
  - Typed edges
  - Multiple typed attributes per edge
  - Multiple graphs (sets of edges) expressed within the same file
  - Dynamic network data expressed in a single file
- (4) The data interchange format shall allow developers to extend it in a fashion that will not break existing software
- (5) The data interchange format shall be flexible enough to be used as both input and output of analysis tools.

### 4. DYNETML: RICH SOCIAL NETWORK DATA INTERCHANGE LANGUAGE

Faced with a pressing need for tool interoperability within our laboratory, we have developed DyNetML - and XML derivative language that addresses the above requirements.

Figure 1 shows the hierarchical structure of the DyNetML files. The **MetaMatrix** element encompasses data related to a single time period within the dataset. If only one time period is present, MetaMatrix can be used as a *root element* of the file. Otherwise, a top level element **DynamicNetwork** is used to unite all time periods in a single dataset.

Each **MetaMatrix** consists of a number of **Nodeset** elements that encompass node data, followed by a set of networks that specify edge data for a set of graphs involving the nodes.

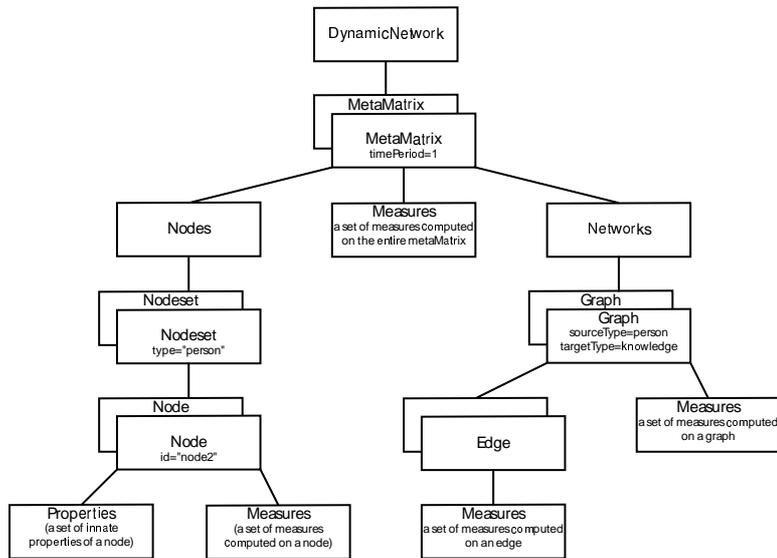


FIGURE 1. Structure of DyNetML

Nodesets store nodes or actors in the dataset in collections ordered by type. In the PCANS model, the nodesets are People, Knowledge, Tasks, and Resources. However, it is important to note that DyNetML does not impose the PCANS model structure and allows the users and developers of tools to freely configure their nodesets.

Each **Node** within a **Nodeset** has to be supplied with a unique ID and can contain an arbitrary number of innate **Properties** or computed **Measures**. This allows the data collectors to specify arbitrarily complex data about nodes while separating collected data from results of analysis.

The **Graph** nodes within **Networks** are specified with a unique ID and IDs of the source and target nodesets. This allows the user to specify an arbitrary number of networks involving the same (e.g. friendship and advice networks) or different types of actors (e.g. communication and resource distribution networks). Each graph and edge can also be followed by a set of innate **Properties** and computed **Measures**.

The DyNetML language thus builds an extremely flexible and expressive, yet structured way of presenting rich social network data and is suitable for becoming a data interchange format between social network gathering, analysis and visualization tools, as well as a way to distribute datasets throughout the research community.

**4.1. Support of DyNetML.** DyNetML is currently supported through a C and Java libraries that are a part of the CASOS software suite. Since XML parsers exist for practically all platforms and languages, integration of DyNetML into existing tools can be completed in one day or less.

## 5. ANALYSIS TOOLCHAINS: A VISION OF THE FUTURE

While the research community has developed a number of very powerful data gathering, analysis and visualization tools, the tools rarely operate well with each other. While file import/export options make it possible to use multiple analysis tools within a single project, a lack of automation and scripting features does not allow for batch-processing of data and report generation, thus vastly increasing labour requirements for analysis of complex datasets.

In our vision, the future of social network analysis lies in creation of a seamless toolchain, allowing researchers to mix and match data gathering, analysis and visualization tools and create analysis scripts of batch-mode processing of large datasets or repetition of the same analysis on different datasets. Publishing analysis scripts would allow for greater ability to verify findings through making experiments repeatable by different researchers.

Each of the tools on the toolchain shall:

- Take the accepted data interchange format (such as DyNetML) as input *and* produce it as output (with the exception of conversion tools)
- Analysis tools shall integrate results of their computation within the dataset, using accepted measure identifiers
- Each tool that modifies the dataset shall mark its modifications with tool name or ID.
- Each tool shall provide a command-line interface that allows full access to its features via scripting language
- A C-like scripting language shall be developed for integration of tools within the toolchain. Alternatively, existing scripting languages such as Java, Perl or Python can be used.
- Visual analysis builder tools shall be developed to allow creation of analysis scripts by non-programmers

## 6. CONCLUSION

An integrated toolchain such as the one outlined above can be only created through cooperation of members of the research community through an open-source development process, but the first step must involve creation of uniform data interchange language. In this paper, we presented a proposal for such language in form of DyNetML, an XML-derived language for specification of rich social network data.

It is important to note that since DyNetML is intended as a service to the social network analysis and simulation community, comments and requests for revisions are welcome at any time. Once the project has considerable community support, we shall establish a revision process that will allow requirements of the community to be taken care of while maintaining backward compatibility with existing software.

CARNEGIE MELLON UNIVERSITY

*E-mail address:* maksim@cs.cmu.edu and jreminga@cs.cmu.edu and kcarley@ece.cmu.edu