fhiso

CFPS 4
(Call for Papers Submission number 4)

Modeling Research, not Conclusions

Submitted by: Tychonievich, Luther

Created: 2013-03-10

URL: Most recent version: http://fhiso.org/files/cfp/cfps4.pdf
This version: http://fhiso.org/files/cfp/cfps4_v1-0.pdf

Description: An data model architecture for storing each step in the research process, simplifying collaboration.

Keywords: Data model, architecture, research

Modelling Research, not Conclusions

Luther A. Tychonievich

10 March 2013

Abstract: *Because researchers often disagree on the details of research, any model based on sharing conclusions will result in edit wars. By sharing instead each action and decision during research, collaboration may proceed between researchers who disagree on subsets of one another's research.*

The goal of this data model is to store, as much as possible, the atomic steps in the research process. The conclusions embodied in those steps can then be automatically derived as reports. This ideal allows collaboration without full agreement: disagreeing researchers can share new steps in their research freely and chose whether or not to include other researchers' decisions on a case-by-case basis. It also allows collaboration between tools implementing different custom data features.

In order to match every step of research closely it is important to be able to model the actual assertions of a source without extrapolation. For example, "X is Y's mother" is a distinct kind of assertion to "Y gave birth to X"—the later implies participation in a birth event, while the former is a more general assertion that might be made of adoptive relationships as well. The data model proposed herein is designed to allow both kinds of assertions to coexist freely.

In order to express each step of research, it is necessary to be able to express under-constrained and self-contradictory beliefs, as these often arise during research.

1 Data Model

A data file consists of the following elements.

Digital Source Sources may, but do not need to be, present inside a data file. Included source may include transcripts, images, research log entries, etc. A researcher is sometimes a source (for annotations, explanations, etc) and may be described in a source element. Personal knowledge is better modelled by documenting that knowledge and citing the document.

Sources have provenance and citations. I assume the details of these are discussed by other proposals.

Thing A thing node has a type label and a single citation. It express the assertion of a source that some noun exists, be it a person, place, personal event, historical event, etc. Details of that extant thing are stored in properties and connections (see below).

Connection A connection node has a type label, two references to thing nodes, and a single citation. Common connections include "participant in," "happened at," "daughter of," etc. Participation connections typically include role (e.g., "as witness").

Property A property node has a single reference to another node of any type and a single citation. It also includes additional data that describes the referenced element. This data may be a standardized key-value pair (e.g., "name: Jane Smith") or more free-form text (e.g., "I suspect this is a pseudonym").

Match A match node has two references to nodes of the same type. It asserts "these nodes refer to the same thing or idea."

A match between two or more thing nodes of type X may be treated in every way as a thing node of type X inheriting every property, connection, and source of each of the nodes it matches.

I am unsure of the value of matching connections and properties. All use cases I have imagined either add no meaning (when nodes differ only in source) or fail to provide enough meaning and ought to be handled by inferences instead (when nodes differ in asserted content).

Inference Elements supported by the direct evidence of a source cite that source. Elements supported by indirect evidence cite instead an inference node describing that indirect reasoning.

An inference node contains a description of the rule or reasoning behind the inference and references to the nodes that contain the antecedents to the inference. The references should not imply a match or connection not explicit in the data in the data. Contextual information like "is not on the census" should be in the data, not in the description of the inference.

Inferences have the most potential for being handled differently between tools. I was unable to discover an extant pattern for handling for indirect evidence in the community. See my paper "Inference Rules" for one possible proposal that, among other things, normalizes inferences.

Belief A belief node refers to a set of elements with the property that no node in the set inherits (via a match) from another node in the set. The closure of a belief also contains all nodes referenced by elements of the belief set. Belief nodes allow researchers to store and share elements of research that they do not currently accept as true.

Inheritance must be acyclic.

Each element is immutable (they are values, not entities) meaning that "editing" a node is actually introducing a new node.

Properties and connections may be "negative" in meaning; for example, a research log source failing to find a person on a census might create a "not resident of" connection. Properties may also be used to negate nodes, expressing the belief that the information expressed by a node is not correct.

2 Files and Collaboration

Data files contain a set of nodes, in no particular order, each with an internal identifier (unique within the file) which is used for in-file references.

In addition to the components listed above, each node in a file also contains a set of all of the nodes that share its citation and refer to it. This addition is necessary to uniquely identify nodes in the case where citations are course enough that two elements might have identical citation; where precise citations are available, these extra sets are all empty.

When merging data files, first the identifiers must be made unique across each file. Then nodes are merged (with identifiers and references updated) in strongly connected component groups. A group of strongly-connected nodes is identical to another group of strongly-connected nodes if there exists a reassignment of the identities of the nodes of one group that can make each node in that group identical to a node in the other group. Such graph-matching algorithms are generally computationally expensive for large unstructured graphs, but for the kinds of small labelled graphs anticipated in this data model they are fairly efficient.

Two kinds of data files exist. Stand-alone data files contain the closure of a belief. All references are to other nodes within the file and may be implemented using any arbitrary internally-consistent identifiers. Partial data files have some references external to the file itself. External references may be hashes (as is done in DVCSs like git and mercurial) or URIs. Partial files may be useful for small deltas between close collaborators or for protecting intellectual property or private information.

3 Minor Extensions and Open Issues

Many ideas can be expressed either as a property or as a thing and connection pair. For example, are places and names things, or are there "location: X" and "name: X" properties?

Immutability is important for collaboration, but sometimes typos and other errors must be corrected. This can be handled by adding **update-of** nodes to the data model.

It can be useful to who thought what when. This kind of social research history can be accommodated with various properties such as "created by", "added to belief of", "removed from belief of", etc.

See also my proposal "Inference Rules" which is designed to dovetail with this proposal by adding structure to the inference nodes.

4 Example

Consider the following research process:

1. A birth record is found for Dana Doe with parents John and Jane Doe.
2. A marriage of John and Jane Doe is inferred.
3. A marriage record is found for Dana Doe, with an age.
4. A birth event is inferred from Dana Doe's age.
5. The two Dana Does are inferred to be the same person.

6. The birth events of the two Dana Does are inferred to be the same event.
7. The date from the birth record is inferred to be more accurate than the inferred date.
8. The age is inferred to be incorrect.

The resulting data model and conclusion view are shown in Figure 1.

5 Comments

There is no need for the user of an application using this data model to see anything more than the conclusion defined by the current belief.

As the nodes of a file are value- not identity-based and have no inherent order, they might easily be stored in databases the way that sources and extractions are currently.

The "wisdom of the crowds" might easily be tracked by recording how many beliefs include each node.

Automated suggestions and computerized research can also be safely added generated nodes are available to, but do not step on the toes of, humans.

Verification of others' research is simplified by allowing each decision to be individually inspected and either accepted or not.

Creating a single "global family tree" using this model requires only that every element from every user's belief is combined into a single large belief, and then some kind of automated arbitration (via inference nodes representing some heuristic, majority rule, or the like) be used to remove conflicts from that belief. Individual researchers need not be constrained by nor believe that arbitration.

New research findings can result in radical changes to the structure of family trees without any unusual effort because the data is just a set of facts; the tree structure exists only in the derivative view.

Current-day properties assigned to deceased individuals (e.g., LDS temple ordinances, membership in military recognition groups, tribal information, etc.) can be readily handled as properties associated with nodes in this data model. Questions such as "is this the same John Smith we already have" can be answered by seeing if the new John Smith is connected by matches to the John Smith to whom the existing property refers.

The number of possible ways to match n elements is super-exponential, but the potential state explosion is controlled because humans will only create so many matches. However, since $((x = y) = z)$ and $(x = (y = z))$ are perceptually identical there is a potential for redundant matches to be added when files merge. I suggest automatically including matches with no perceptual effect into beliefs so that any annotations of those matches may be brought in as well.

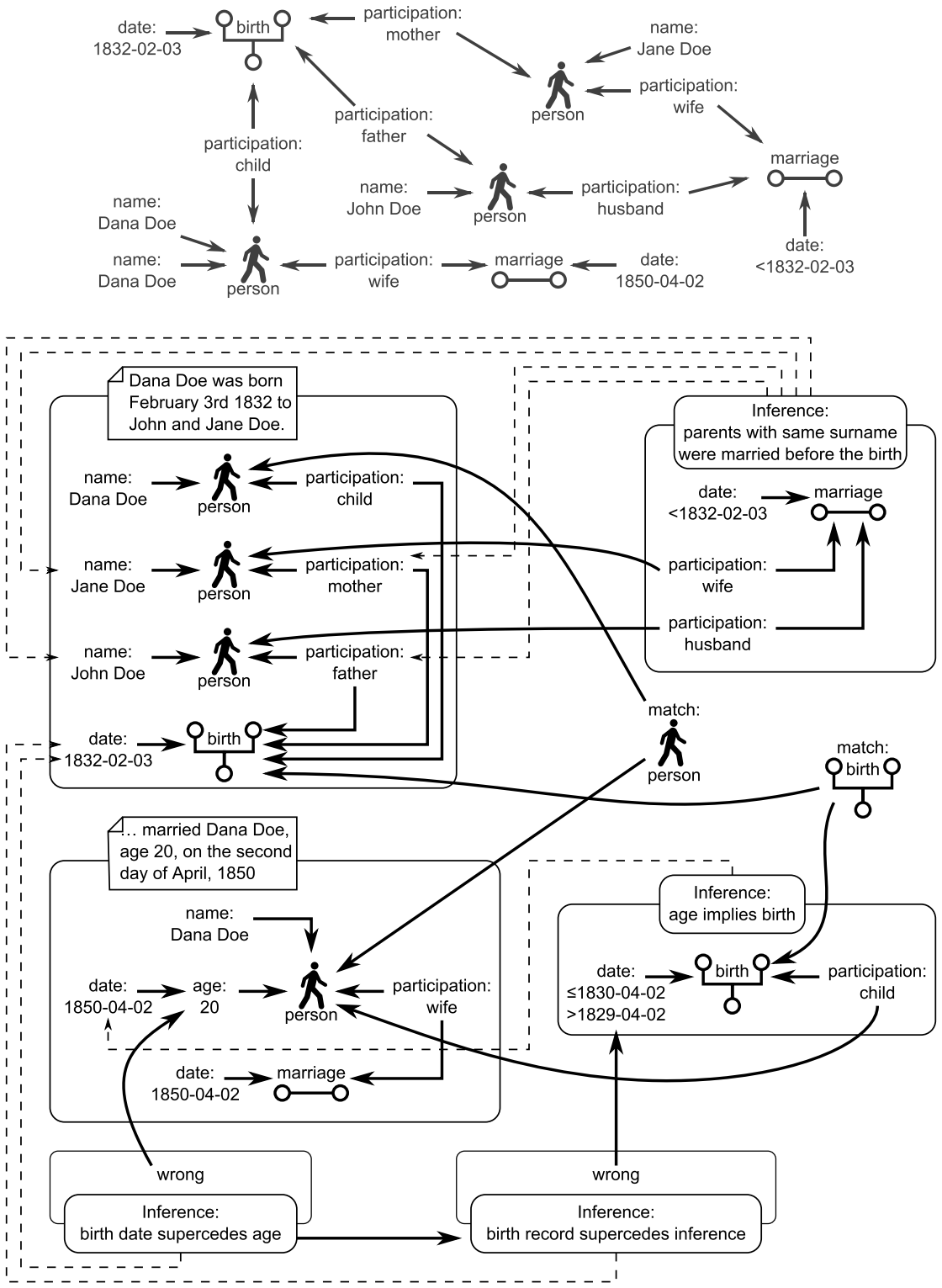

Figure 1: Worked example from Section 4. The greyed diagram at the top is a view of the conclusion represented by the data below it. The doubled name properties in the conclusion represent two sources both attesting the same name.