Issues in Historical Geography

O. Signore, R. Bartoli, G. Fresta, A. Marchetti, CNUC, Institute of CNR, Italy

Abstract

In the management of art history data, the spatial and temporal context has an extensive relevance, as every information is meaningful only if it can be related to their spatial and historical context. However, the scholars are used to specify dates in many "free text" formats, that have a clear semantics for a human mind, but are unsuitable for computer storage and management. The proposed solution allows the specification of any possible date with a clear, simple and natural "external syntax", keeping all the semantics like uncertainty or fuzziness, while keeping an internal format that allows management and comparison of such temporal data. For this purpose, a simple temporal algebra and related operators have been implemented. Data stored according to the proposed approach can be used to extract the items needed to produce historical maps, using conventional GIS software.

Keywords

Temporal algebra; Historical geography; Temporal databases

Introduction

Any scholar involved in historical research is often faced with the need to refer to a historical geography that differs sharply from present-day geography. This is because historical geography is the cumulation of historical phenomena which have followed one another in an uninterrupted sequence of events since time immemorial. However, we must note that historical geography is the "everyday geography" as atlases published just some years ago are completely out of date, due to rapid changes in political boundaries. As it happens with every type of historical development, in historical geography we meet long range structural patterns, or else we meet circumstances that are intimately bound up with clean-cut, individual events. This is an issue that leads to the need of a historical atlas that is at once both comprehensive and dynamic, in the sense that it could be constantly updated and capable of producing the correct maps for any arbitrarily chosen time interval.

In this sense, the dynamic archive is more than just a dictionary of ancient and modern geographical names with illustrations of how such names have evolved lexically. It must be as well a system of correlations between:

- geographical names, understood in territorial and topographical terms;
- historical events, also understood in territorial and topographical terms;
- historical events, such as those circumstances which have served to gather together various geographical locations into political, administrative, linguistic and social "units" or else those circumstances which have been responsible for creating flows of migration, spheres of economic influence and so forth at particular points in time for which we have been able to provide precise dates, or at least the most plausible time frames.

All these considerations lead to state that the probably most relevant issue is an appropriate way of representing and managing time dependent correlations between information elements, and therefore dates themselves.

The related issues have been widely discussed in [Papaldo89], [Signore89], [Signore90a,b]. In this paper we will describe a refinement of some ideas expressed there for supporting imprecise dates in conventional data processing environment.

1. The Proposed Approach

As we have pointed out, every information is meaningful only if it can be put in a spatial and historical context. In short, the problem of historical geography can be reduced to:

a) storing time-dependent information;

b) displaying maps containing time-consistent data.

The conventional approaches, like thesauri or Geographic Information Systems (GIS), show some limitations and are unsatisfactory.

In brief, we can reduce the limitations of the traditional thesaurus approach to the excessive number of relationships that should be defined to represent all the possible associations between entries. Just as trivial example, think to the number of relationships that would be necessary to define to state the correct political affiliations of Alsace during last centuries. In addition, standard thesaurus relators do not permit to represent time dependent relationships.

We must stress that conventional Geographic Information Systems (GIS) do not adequately manage the time coordinate. If a line (a river, a boundary, a communication route, etc.) changes over time, even for a little detail, it will entirely be stored as many times as a single variation occurs, in many different and fixed information "layers". For a more detailed description of the time-related issues in Geographic Information Systems, see [Langran93].

An additional complexity factor arises from the fact that very often dates are only approximately known. Therefore scholarly tradition uses some special expressions (century, before, after, circa, etc.) to express this uncertainty, even when the date itself corresponds to a well defined interval (e.g. *probably in 1619, but certainly after 1618 and before 1622*).

However, even if dates expressed in this traditional way are meaningful to the scholars and easy to specify, nevertheless they present several disadvantages.

Firstly, computer programs can hardly manage them, and joining of different information on the basis of the time span they apply is almost impossible. No DBMS can effectively handle them, and the ISO SQL standard only considers characters and numbers as data types. Even the ISO SQL2 "date" and "datetime" data types are applicable to recent, precise dates only, and do not permit to maintain the semantic richness of the traditional way of expressing dates.

Secondly, there are some linguistic differences that should be taken in account, especially in the light of worldwide access to databases on the web.

Thirdly, dates can be expressed with a different level of approximation and according to different calendars (Jewish, French revolution, Mussulman). Therefore, it comes that we need a single, absolute, computer processable representation of dates, and a temporal algebra, to support all the user needs. Quite obviously, we must map this representation on available tools, like relational database management systems, and mask to the user any complexity.

The solution proposed in this paper, and tested in several environments, consists in the definition of a formal "external syntax" useful for representing the dates in a compact, semantically meaningful format. The uncertain dates can be represented, with all their fuzziness, as a fork, leaving to the scholar the possibility of identifying a "probable date", which can differ from the simple arithmetic mean of the two edges. Dates expressed in this format are easily stored in the underlying database, and all meaningful operations on the time intervals (overlapping intervals, excluding intervals, contained intervals, etc.) can be translated into the appropriate relational language constructs.

We must remember that in formalising and storing the information, attention must be paid to the identification of time-dependent relationships and/or properties.

In implementing a historical geographical database, we can rely upon a database that stores the time-varying relationships among the places, their past names, political entities, and so on. As shown in Figure 1, an appropriate intermediary (technically a parser) will be in charge of interpreting the time operators in the user's query and translating them into appropriate relational queries. A conventional GIS, or a simpler and cheaper tool, can afterwards display on a map data gathered from the database.

We stress that the user will interact using just an "external syntax", that has been kept as natural as possible.

2. THE EXTERNAL SYNTAX

The principal aim in defining the external syntax has been to preserve the semantic richness of the traditional way of expressing them, maintaining a high degree of flexibility without introducing useless or unnatural formalisms.

It has been decided to express any date in five possible ways:

- *Exact date* this form is used to express dates known as certain, therefore they are expressed as year, month, day.
- *Numeric dates* can be used when the scholar is unable to determine a precise date, but can specify at least the year and possibly the month.
- *Roman dates* can be used when the date must be expressed as a century or a fraction of it, like: "XVI century" or "beginning X century", etc.
- *Interval* is one of the most frequently used formats, that specifies a lower and upper limit, both as exact or as approximate.
- *Probable dates* sometimes, given an interval, it is possible to determine, as the most probable date, a date contained in the interval, but not exactly the arithmetic mean.

More formally, we defined a MY_DATE datatype, as shown in Figure 2:

We must take into consideration:

- The MY_DATE data type includes exact dates, too, for events exactly known in terms of year, month, day. This is because RDBMSs support exact dates, but with lower bounds unacceptable for the application field.
- d_type can be used to indicate that the date is only approximately known.
- d_reliab can be set to "?" to indicate that date is merely hypothetical, and is specified just to avoid the system returning inconsistent data.

According to this syntax, valid expressions are:

- 4 September 476;
- October 732;
- (1235 - November 1238);
- 323 B.C. (327 B.C. - 321 B.C.);
- 1358;
- beginning IV B.C.

It is easily seen that the basic element of the temporal domain is a point on the temporal axis, corresponding to an exact date. Consequently every date

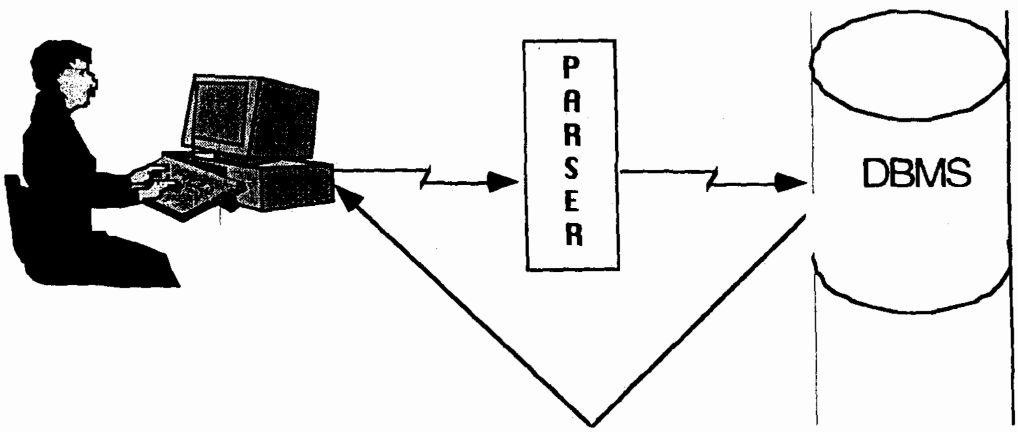

Figure 1 - The architecture

```
<MY_DATE>              ::=   <exact_date> | <imprecise_date>

<exact_date>           ::=   <day>"/"<month>"/"<year><sign>
<sign>                 ::=   a.C. | d.C. | B.C. | A.D.
<year>                 ::=   0 | ..| 2999
<month>                ::=   1 | ..| 12
<day>                  ::=   1 | ..| 31

<imprecise_date>       ::=   <d_type><numeric_date><d_reliab>
                             |<d_type><roman_date><d_reliab>
|<interval>
                             |<probable_date>

<numeric_date>         ::=   <opt_month><year><sign>
<opt_month>            ::=   ¯ | <month>"/"

<roman_date>           ::=
<century> <roman> <sign>
<century>              ::=
<fraction> century
<fraction>             ::=   ¯| beginning | middle| end
<roman>                ::=   I | II | .. | XXX

<interval>             ::=   "("<boundary > , < boundary>")"
<boundary>             ::=   <exact_date>
                             | <d_type><numeric_date><d_reliab>
                             | <d_type><roman_date><d_reliab>

<probable_date>        ::=   <prob_date> <interval>
<prob_date>            ::=   <exact_date>
                             | <d_type><numeric_date><d_reliab>
                             | <d_type><roman_date><d_reliab>

<d_type>               ::=   ¯ | "circa"
<d_reliab>             ::=   ¯ | "?"
```

Figure 2: The formal Expression of the MY-DATE Datatype

corresponds to an interval, possibly with coincident upper and lower bound. The system automatically adjusts upper and lower bound of the interval according to the degree of accuracy the date has (month, year, century fraction, century).

3. THE TEMPORAL ALGEBRA

Ordering between intervals is defined by ordering their boundaries. To state order relations between imprecise dates we defined a set of boolean operators: After, Before_of, Falls_in, Is_coincident_with, Overlaps, Outside.

As every imprecise date is managed as an interval whose extremes are precise dates, the operators model an order relation between imprecise dates as an order relation between exact dates corresponding to the interval extremes. Let us take a MY_DATE D_j, corresponding to an I_j interval, Figure 2 explains the semantics of the temporal algebra operators.

4. IMPLEMENTATION ISSUES

The whole system has been implemented in UNIX environment (namely Sun/OS) for the Sybase relational system, using the usual tools Lex and Yacc. However, the idea is so general that a porting to other hardware and software platforms should be easy.

We remember that the parser acts both at the Data Definition and Data Manipulation level.

In the first case, when the database administrator defines the database structure, it reads the extended DDL and produces a standard DDL. In this phase, some information is added to a catalog that is "parallel" to the system catalog.

At the query level the parser gets information from this parallel catalog, and transforms the extended DML into a standard DML.

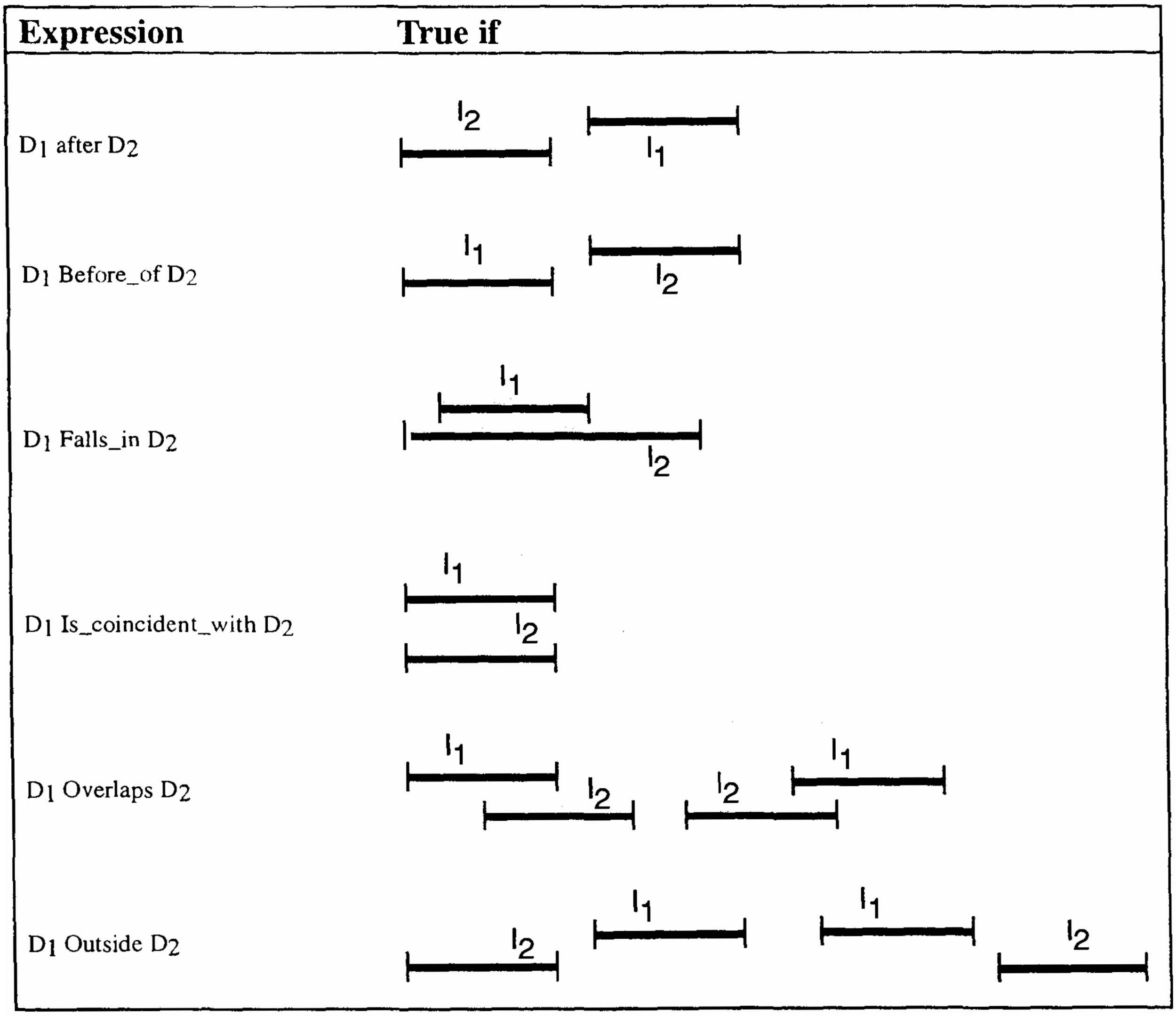

Figure 2 - The semantics of the temporal algebra operators

5. CONCLUSION

We defined a syntax to codify imprecise dates and a set of operators to operate on them. We implemented a parser to translate the newly defined data type and temporal algebra operators in standard SQL statements.

The users will interact with the system using the "external syntax" only, and do not have to worry about the internal format. The implementation does not support operations on dates, like difference, but it seemed irrelevant in the specific context.

REFERENCES

[Langran93] Langran G.: *Time in Geographic Information Systems*, Taylor & Francis (1993), ISBN 0-7484-0059-6

[Papaldo89] Papaldo S., Signore O.: *Un approccio metodologico per la realizzazione di una banca dati storico-geografica (A methodological approach to producing a historical/geographical databank)*, Multigrafica Editrice, Roma (1989), pp. 573 ISBN 88-7597-105-6

[Signore89] Signore O., Bartoli R.: *Managing Art History Fuzzy dates: An Application in Historico-Geographical Authority*, Historical Social Research, Vol. 14, n. 3, (1989) pp.98-104 (Special Issue - Computer Applications in the Historical Sciences: Selected Contributions to the Cologne Computer Conference 1988)

[Signore90a] Signore O., Bartoli R.: *A case study for historico-geographical authority*, in Terminology for Museums, Proceedings of an International Conference held in Cambridge, England, 21-24 September 1988, Maney and Son Limited (1990), ISBN 0-905963-62-8,pp.150-158

[Signore90] Signore O., Bartoli R.: *Implementation of a historical/geographical database with support of imprecise dates*, Database and Expert Systems Application, Proceedings of the International Conference in Vienna, Austria, 1990, (Tjoa A.M., Wagner R., Eds.) Springer Verlag Wien New York, ISBN 3-211-82234-8, pp. 271-274

ACKNOWLEDGEMENTS

We owe our thanks to Daniela Di Pasquale, who implemented the parsers.