A Lightweight Treatment of Inexact Dates*

Hai H. Nguyen[1], Stuart Taylor[1], Gemma Webster[1], Nophadol Jekjantuk[1], Chris Mellish[1], Jeff Z. Pan[1], Tristan ap Rheinallt[2], and Kate Byrne[3]

[1] dot.rural Digital Economy Hub, University of Aberdeen, Aberdeen AB24 5UA, UK
[2] Hebridean Connections, Ravenspoint, Kershader, Isle of Lewis HS2 9QA, UK
[3] School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, UK

Abstract This paper presents a *lightweight* approach to representing inexact dates on the semantic web, in that it imposes minimal ontological commitments on the ontology author and provides data that can be queried using standard approaches. The approach is presented in the context of a significant need to represent inexact dates but the heavyweight nature of existing proposals which can handle such information. Approaches to querying the represented information and an example user interface for creating such information are presented.

1 Introduction

There is as yet no standard approach for representing uncertain information in the semantic web. Existing proposals [1] suggest rather radical changes in representation, and the associated reasoning algorithms, in order to capture uncertainty. In this work, we seek solutions that can be combined more easily with standard practices but which can handle some common special cases. We focus in particular on the expression of inexact *dates*. Dates are very important in the semantic web, which is reflected in the existence of standard data representations (`xsd:date` and `xsd:dateTime`) for exact dates. However in many situations only partial information about a date is available. This arises particularly in applications of the semantic web to cultural heritage. When information about the past is represented, however, there is frequently uncertainty about when events happened, artefacts were made, people were born, etc.

In our own work, the University of Aberdeen and Hebridean Connections are working with historical societies based in the Western Isles of Scotland to produce a linked data resource documenting the people, places, events, boats, businesses etc. of their past. The current data consists of over 850,000 RDF triples, incorporated within a relatively simple OWL ontology. There are 13 different OWL properties that introduce dates for specific events (e.g. date of birth, date of origin for a photograph, date demolished for a building). Our ontology is the result of semi-automatic processing of data that originated from a database.In the database, dates were entered as free-form strings (13,470 of them). In all, information has been provided about 30,923 dates; these were entered by many different people over a significant time period. Figure 1 shows an analysis of the patterns found in this data. The "general class" is a rough indication of the precision of the information and the patterns indicate the rough syntactic forms used,

* The research described here is supported by the award made by the RCUK Digital Economy programme to the dot.rural Digital Economy Hub; award reference: EP/G066051/1. Many thanks to Panos Alexopoulos for useful conversations.

General Class	Pattern	Example	Frequency	Subtotal	Covered
Exact to the day	y-m-d	1780-06-13	12949		
	d-m-y	10/6/45	725		
	d-M-y	12 MAY 1780	272		
	M-d-y	May 12 1780	8		
				13954	yes
Exact to the month	y-m	1780-12	274		
	M-y	Aug 1780	443		
	m-y	03/1780	2		
				719	yes
Exact to the year	y	1978	10825	10825	yes
Exact to the decade	dec	IN 1860'S	1415	1415	yes
Exact to a range of years	y-y	1939-45	242		
	beforey	pre 1918	2		
	aftery	AFT 1890	3		
				247	yes
Exact to the century	cent	20th Century	4	4	yes
Vague within less than a month	mend	Aug/Sept 1972	26	26	yes (using a date range)
Vague within more than a month but less than a year	yend	1978/79	7	7	yes (using a date range)
Vague year	cy	C. 1932	566		yes
	moddec	early 1950s	86		(using a
				652	date range)
Vague around a decade	cdec	c 1950s	2		yes
	modcent	LATE 1600S	3		(using a
				5	date range)
Not directly interpretable as a date	unk	D.I.I.	3069	3069	no
GRAND TOTAL				30923	

Figure 1. Analysis of Date Forms in the Corpus

with "y", "m" and "d" indicating numbers for years, months and dates, "M" indicating a month expressed as a string, "-" indicating a separator, such as "-", "/", "." or a space and some patterns just having vaguely mnemonic names. Interestingly, only the first general pattern (total frequency 13,954, about 45% of the data) represents exact dates: all the rest are inexact to some extent. Amongst the inexact dates, we have distinguished those which are exact within a specified range from those that are *vague*, in that there is scope for argument about the intended boundaries. Since all of the relevant properties are intended to indicate specific dates, not periods of time, this inexactness must be due to uncertainty on the part of the person entering the information. The inexact dates in our collection represent a substantial amount of information - a significant semantic resource that would be lost if we left the information as free-form strings.

Our example is likely to be typical of other projects seeking to exploit cultural heritage data. What is needed is a general mechanism capable of representing (most of) the types of inexactness encountered here in a way that supports some semantic reasoning. This mechanism needs to be *lightweight*, in that:

1. Expressing information about inexact dates should involve as few changes as possible to the ontological decisions already made in the original data;
2. It should be possible easily to query and update information about inexact dates using standard languages such as SPARQL 1.1. In particular, given that the underlying data is uncertain, it should be possible to represent queries that are: **high precision** - in that results returned definitely satisfy the search criteria; and **high recall** - in that all results that might possibly satisfy the criteria are returned.

where criteria for the information seached for can be expressed either as an inexact date or as a specific date.

2 Previous Work

Naturally the first question to ask about inexact dates is whether they can be represented by standard XML Scheme datatypes; for this, the two possibilities are `xsd:date` and `xsd:dateTime`. Unfortunately, SPARQL 1.1 has no support for `xsd:date`. Although `xsd:dateTime` is supported by SPARQL 1.1, "dateTime uses the date/time SevenPropertyModel, with no properties except -timezoneOffset- permitted to be absent" [2]. Therefore this property cannot be used on its own to represent inexact dates.

There has been a significant amount of work on ontologies for representing time (and hence dates). In the cultural heritage domain, CIDOC CRM [3] defines a class of entities `E2 Temporal Entity` which describes "objects characterised by a certain condition over a time-span". `E2 Temporal Entity`s include `E4 Period`s, which include examples such as "Jurassic". Although the philosophical position of the OWL-Time ontology [4] is somewhat different to that of CIDOC CRM (e.g., time instants are believed to exist), the treatment of dates is rather similar to the above. In this case, the analogue of `E2 Temporal Entity` is `owltime:TemporalEntity`, which has subclasses `owltime:Instant` and `owltime:Interval`. As above, the actual `TemporalEntity` has to be distinguished from the information stated about it – the latter is included within the former. The latter can be directly associated with the `owltime:Instant`s which are its beginning and end. These can in turn be associated with `xsd:dateTime`s.

When it comes to the representation of relatively simple inexact dates these proposals probably satisfy our criteria about access via standard query mechanisms. However, all of them involve significant ontological commitments that may not be made by the rest of the ontology. Whereas for many applications "date of birth" is thought of as a simple property of a person, akin to "name" or "gender", CIDOC CRM requires the existence of 2 extra individuals and OWL-Time 4 extra individuals to express the information, apart from the basic `xsd:dateTime` values involved. Although both are highly principled and in the end very flexible ways of incorporating inexact dates, CIDOC CRM or OWL-Time would significantly break this way of viewing the world, and we suspect they would be similarly disruptive in terms of other ontologies. The question is whether there is a lighter weight alternative available.

3 The Proposal

Our proposal is a very simple one that caters for many kinds of inexact dates and complicates the ontology only minimally. The proposal involves the introduction of one

new class, `hc:dateRange`[4], on which the `xsd:dateTime`-valued datatype properties `hc:dateFrom` and `hc:dateTo` are defined. An `hc:dateRange` represents a specific date/time which is within the range between the `hc:dateFrom` and the `hc:dateTo`, *not* the period of time between those values. Figure 2 shows how this

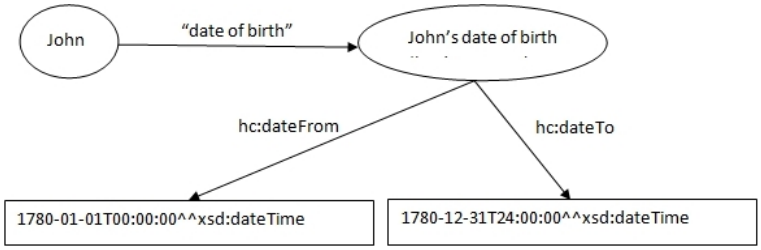

Figure 2. "John was born in 1780" in Hebridean Connections

looks for John's birthday. Expressing the information this way commits the ontology author to John being born at a specific time; however, it does not require them to specify that time precisely. Notice the inclusion of time information in the representation – this is forced by the requirements for the completeness of `xsd:dateTime` instances mentioned above. In terms of the dates in our corpus, this proposal provides a way of directly encoding all "Exact to ..." patterns (13,954 exact dates and 13,210 inexact dates, in total about 88% of the corpus). With an appropriate user interface to apply or elicit the relevant cultural norms (see Section 3.2 below), we believe that it can also provide some coverage of the vague patterns in our corpus as well.

3.1 Queries involving Inexact Dates

An `hc:dateRange` represents the start and end of an interval of time. A query to the dataset (e.g. "find all people born in 1780") also involves an interval of time (even a specific day involves two different times for its start and end). In general, then, in a querying situation there is an interval i provided by the query and an interval j provided by data that has been retrieved and is being considered as a possible solution. Two possible types of query then involve particular required relationships between i and j – we name these using the relations defined by [5]:

high recall query - this is a query guaranteed to return any results which might satisfy the query. The required relationship is $Overlaps(i, j)$, i.e. the lower bound of i is less than or equal to the upper bound of j, and the upper bound of i is greater than or equal to the lower bound of j.

high precision query - this is a query guaranteed only to return results which definitely satisfy the query. The required relationship is $Contains(i, j)$, i.e. the lower bound of j is greater than or equal to that of i, and the upper bound of j is less than or equal to that of i.

```
SELECT ?id ?name ?from ?to WHERE
{   ?id hc:born ?dr .
     ?dr hc:dateFrom ?from .
     ?dr hc:dateTo ?to .
     FILTER (?to >= "1780-01-01T00:00:00"^^xsd:dateTime &&
        ?from <= "1780-12-31T24:00:00"^^xsd:dateTime)  .
      ?id hc:englishName ?name}
SELECT ?id ?name ?from ?to WHERE
{   ?id hc:born ?dr .
     ?dr hc:dateFrom ?from .
     ?dr hc:dateTo ?to .
     FILTER (?from >= "1780-01-01T00:00:00"^^xsd:dateTime &&
        ?to <= "1780-12-31T24:00:00"^^xsd:dateTime)  .
      ?id hc:englishName ?name}
```

Figure 3. High Recall and High Precision SPARQL queries for "Find people born 1780"

Figure 3 shows example SPARQL queries of these two types, both giving answers to "find people born in 1780". In these examples, `hc:born` is the property that relates a person to their date of birth and `hc:englishName` is the property that provides the English name of a person.

3.2 User Interface Issues

Our simple proposal is still unable to handle directly the *vague* dates found in our corpus (those without clear boundaries). In fact, the actual usage of vague date expressions in English can be very specific to particular communities. For instance, for one of our partners, plus or minus 4 years is a good interpretation for circa dates from 1850 on, but prior to 1850 they were more vague and plus or minus 10 years would be more realistic. However in the linked data world we have to represent information in such a way that anybody can access and understand it. We therefore believe that it is the role of user interfaces to intervene between specific cultural communities and the unbiased data that they create and access. In particular, the mapping between a phrase such as *c.1987* and a particular `hc:DateRange` has to be managed by an interface aware of the characteristics of the particular user community it is serving.

We are building a new website based on the Hebridean Connections data (with the intention of replacing the earlier database-based website), using generic software for the semi-automatic construction of a Drupal website which supports the management and presentation of information from a semantic web ontology [6]. A part of this is a user interface for this community, to be used for the acquisition of inexact date information. The user can decide to enter an "exact" date (in which case they are directed to a standard calendar widget and end up creating an `hc:dateRange` where the two endpoints correspond to the start and end of the specified day) or a "circa" date (which here means "inexact"). On selecting the latter, they are then presented with five possible patterns (Figure 4). "Covered" column in Figure 1 shows the classes of dates in the corpus

[4] We abbreviate the namespace used in the Hebridean Connections ontology, `http://www.hebrideanconnections.com/hebridean.owl#`, by `hc:`.

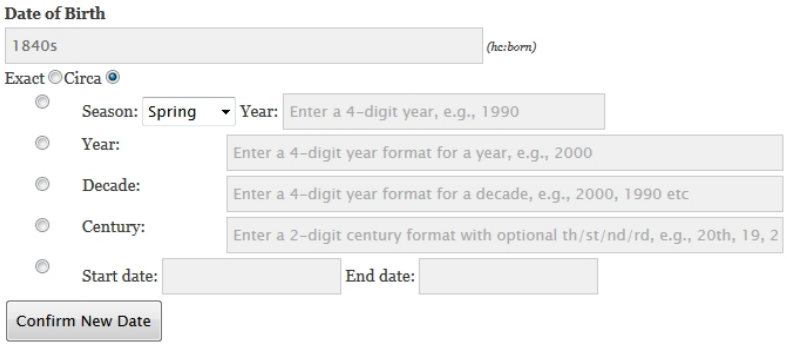

Figure 4. Subsequent interface for entering an inexact date

can be covered by this interface. Although the first pattern cannot be found anywhere in our corpus (though it is somewhat similar to the pattern `yend`), it was suggested by our project partners as being particularly appropriate when one is attempting to date a photograph. In this case, the partners proposed a particular interpretation of the seasons in terms of dates. The next three patterns correspond to three "exact to ..." classes that we noted above. The remaining pattern allows for the entry of an arbitrary date range where we aim to cover dates in the "vague ..." classes. We imagine that this is a last resort as mostly one of the previous patterns will apply.

4 Future Work

Although our lightweight representation of inexact dates covers very well the inexact date data we have collected so far, further work is needed to see how well it copes with the entry and use of new data in the future. In particular, we may find it useful to add further input patterns to the user interface where a clear convention emerges.

References

1. Lukasiewiwc, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the semantic web. Web Semantics **6**(4) (2008) 291–308
2. Peterson, D., Gao, S., Malhotra, A., Sperberg-McQueen, C.M., Thompson, H.S.: W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes. `http://www.w3.org/TR/xmlschema11-2/` (April 2012)
3. Boeuf, P.L., Doerr, M., Ore, C.E., Stead, S.: Definition of the CIDOC Conceptual Reference Model . `http://www.cidoc-crm.org/docs/cidoc_crm_version_5.1-draft-2013May.pdf` (May 2013)
4. Hobbs, J., Pan, F.: Time Ontology in OWL . `http://www.w3.org/TR/owl-time/` (September 2006)
5. Allen, J.F., Ferguson, G.: Actions and events in interval temporal logic. J Logic Computation **4**(5) (1994) 531–579
6. Taylor, S., Jekjantuk, N., Mellish, C., Pan, J.Z.: Reasoning driven configuration of linked data content management systems. In: JIST. (2013)