STEMMA® Data Model

Tony Proctor, 19 Apr 2017

1 Introduction 2
2 History 2
3 Goals 2
3.1 Balance between Human versus Machine 3
3.2 Structured Narrative 4
3.3 Events 5
3.4 Places 5
3.5 Persons 6
3.6 Information and Timelines 6
3.7 Inheritance 7
4 Data Model Case Studies 7
4.1 Timelines 7
4.2 Multi-Source Events 11
4.3 Personal Events 13
4.4 Dual Dates 14
4.5 Multi-Role Event 16
4.6 Complex Citation 16
4.7 Multiple Births Spanning Midnight 19
4.8 Recording Oral History 23
4.9 Evidence, Reasoning, and Conclusion 27
4.10 Transcription Anomalies 31
4.11 Family Units 33
4.12 Census Roles 35
4.13 Nature and Nurture 46

1 Introduction

This paper discusses the main ideas behind the STEMMA® Data Model. This includes the major influences, why it was structured differently to other models, and the advantages of its approach.

Various familiar problems are examined to show how STEMMA copes with them. These are expressed using XML syntax, and sometimes using pictures, in order to give a clear and unambiguous explanation.

Note: This text capitalises data entity names like Person and Place to distinguish them from the common English usage of those terms.

2 History

STEMMA was primarily a research project with the intention of creating a data format for the preservation of my own family history data. This data was as much micro-history as family history and so required more than simply Person entities and lineage. I wanted this to accommodate my data in a natural way, without having to bend any rules, and in a format that could easily transport to other locales.

My situation was unusual in that I did not use any commercial software products. Although I was aware of GEDCOM and GedXML, I hadn’t read a detailed definition of either. To a large extent, this gave me a free rein to design the data model as I wanted, and to not follow any precedent or industry norm. I believe that has been fruitful in finding some elegant and far-reaching ways of structuring the data, and of solving common genealogical issues.

Until this point, my data was actually kept in word-processor documents; one per surname and entries number according to the d'Aboville System. These documents used a semi-formal style to represent biological lineage (using indented headings), source citations, properties derived from sources, events, etc. They also made copious use of narrative for historical and biographical information, as well as for the reasoning when forming conclusions.

3 Goals

There were a number of design goals that STEMMA had to achieve. My background as a software architect, and my ground-up participation in software globalisation and the writing of locale systems, meant I already knew a lot of the basic design requirements. Best practices and the relevance of computer standards might have been less obvious to someone with a different background.

® STEMMA is a registered trademark of Tony Proctor.

The desire to embrace multiple cultures through being *culturally neutral* was more an exercise in good design. My own data did not require this to any great depth.

The remainder of this section briefly discusses some of the thoughts and desires influencing my design.

3.1 Balance between Human versus Machine

This is a subtle issue, and one that's hard for some people to accept. It involves the balance between how much is for (and by) the computer, and how much is for (and by) humans. Even computer professionals have been known to get this wrong, and consider things too much from the human perspective, or too much from the machine perspective.

A classic example concerns the data that is stored in a file. If it is to be transportable between people of different locales then it must be expressed in a locale-independent way — one designed for the computer to recognise rather than humans. A good illustration in the field of date values is the ISO 8601 standard. All issues of how data is presented to the user, including the formatting of dates, descriptions of types for persons/places/events, and citation styles, should be under the control of your software product. It has the responsibility for converting its locale-independent computer format into something with the correct cultural and regional formatting for the current end-user.

There are related issues beyond this, though. We should recognise that there are other things that will be the responsibility of a software product and that do not need to be accommodated in the data format. As well as formatting according to a user's regional settings or personal preferences, there is data entry. The way that a date is entered and assimilated, or a personal name, or a place name, is something your software does. What is written to the data file is the computer-readable version, usually supplemented by a transcription of the original data or an image of it.

A different issue involves narrative: narrative is difficult for software to store as an integral part of the genealogy data — especially if using a relational database — and so it is often relegated to either simple text fields or external word-processor files. Neither of these is adequate, but there is also a frame of mind that fears narrative because it is understood primarily by the researcher, and cannot be formalised for complete assimilation by software.

Another issue is the process involved in researching, recording, and analysing micro-history data. There are documented standards such as the Genealogical Proof Standard but they should be the choice of the human, or the software product, and should not be a fundamental part of the data's representation.

3.2 Structured Narrative

There is a tendency for genealogy to be too formal and data-centric when considering computer representations. For instance, adding specific record types for every conceivable facet, including level-by-level reasoning to represent a proof-argument, or relying on a controlled vocabulary of “genealogical variables”. However, much is lost if this approach is followed to its conclusion. Consider a published biography, for instance. This can express all manner of information about a person, their family, and the places where they lived or worked. Most existing data formats would find it virtually impossible to represent such a volume of work in a usable way (TEI being a possible exception). On the other hand, even with an index and full use of citations, a typical biography could not be adequately integrated with your other computer data.

STEMMA must be capable of including copious amounts of narrative but it needs to be ‘structured’. This is far more than a simple NOTE or COMMENT feature. It means having marked-up text segments integrated with, and cross-linked with, the overall data schema. The mark-up has to facilitate semantics tagging, structure and content (as opposed to presentation), transcription anomalies, distinguishing information from conclusion, the generation of reference-note citations, and the generation of annotation-style or discursive notes. It could be argued that this is essential to make the jump from *genealogical data* to *micro-history data*.

There are two very important functional categories of text that have to be supported:

- Transcription (including transcribed extracts) — requires support for textual anomalies (uncertain characters, marginalia, footnotes, interlinear/intralinear notes), audio anomalies (noises, gestures, pauses), indications of alternative spellings/pronunciation/meanings, indications of different contributors, different styles or emphasis, and semantic mark-up for references to persons, places, groups, animals, events, and dates. The latter semantic mark-up also needs to clearly distinguish objective information (e.g. that a reference is to a person) from subjective information (e.g. a conclusion as to whom that person is).
- Narrative work — requires support for layout and presentation. Descriptive mark-up captures the content and structure in a way that provides visualisation software with the ultimate control over its rendering It needs to be able to generate references to known persons, places, and dates that result in a similar mark-up to that for transcriptions. The difference here is that a textual reference is being generated from the ID of a Person entity, say, as opposed to marking an existing textual reference and possibly linking it to a Person with a given ID. Also needs to be capable of generating reference-note citations and general discursive notes.

Too often, source information is considered to be something external to your data (e.g. in an external document or online) that needs no representation within your data.

Identifying references to people is a good example. If the text contains an element that identifies a reference to a Person entity, defined elsewhere in your data, then:

- Software can decide what name to replace that reference with. Remember that a person may have multiple names, or there may be several people with the same name and so you may want to see it annotated (e.g. with their year of birth).
- Software can decide on the style of its presentation. I absolutely detest the Stone-Age convention of storing surnames in uppercase, or even between slashes. Uppercasing a surname does not transport well to other cultures, and is effectively a corruption of the name, but it is also totally unnecessary at the data level. Once software knows a piece of text is a Person reference then it can be presented in whatever way the current end-user would like.
- Software can automatically convert such references into hyperlinks when the data is visualised.

3.3 Events

Events (e.g. something that happened on a particular date) are extremely important in micro-history data. They allow timelines to be created in order to present a chronological history. More than mere dates could alone, Events can provide a single focal point that links multiple Persons/Groups/Animals and a Place. They also provide a single point at which the relevant sources can be cited. I therefore wanted to represent Events as top-level entities with their own identifiers.

From an historical point of view, events are rarely constrained to a single date. Most have a significant duration, and some can be subdivided into smaller events to give a more detailed depiction. STEMMA therefore had to support lengthy, or protracted, Events as well as hierarchical Events if it was going to model real-life events.

A deeper discussion of STEMMA’s approach to Events may be found at: Eventful Genealogy and Eventful Genealogy - Part II.

3.4 Places

In just the same way that we may have biographical narrative and pictures associated with a person, then we may have historical narrative and pictures associated with a place. Places tend to undervalued in family history but they share many aspects with persons (e.g. variable names, parentage, association with events) so I needed them as top-level entities.

Treating Place in an analogous fashion to Person enhances the possibility that STEMMA could be applied to One-Place Studies.

I wanted to take full advantage of correlating place references and, for instance, finding ancestors who lived nearby, or even next door. To this end, I needed a scheme that provided unique references to every place, right down to the level of a household, and provided hierarchical information about where every place was located in terms of bigger places.

3.5 *Persons*

I wanted to capitalise on the many similarities between the requirements of place names and personal names. I wanted to generalise the requirements so that a single mechanism could be used.

STEMMA doesn't provide any specific entity to represent a family, or even a marriage. The only universal concepts for every person are their birth, their two biological parents, and their death. All other aspects are subject to cultural differences (e.g. what constitutes a marriage?) and inference (what constitutes a family unit?). Instead, STEMMA needed a general event concept that, in conjunction with a system of roles, could model any type of union. It also needed a general Group concept that could be used to model any notion of a family.

The Group concept should be applicable to simple Sets of Persons and Animals, as well as modelling real-world entities such as organisations, regiments, classes, etc. Groups should take advantage of the similar support offered for Person and Place in terms of alternative names, events, sources & evidence, narrative, etc.

I wanted the option of having my Person entities disjoint, meaning that they might not constitute a single tree or network. There might be isolated groups or individuals, including non-relatives who happened to play an important part in a family's history. It also enhances the possibility that STEMMA may be applied to One-Name Studies.

3.6 *Information and Timelines*

I needed a way of recording "pockets of information" against each Person, Animal, Group, or Place. This is not unusual in itself but over 95% of source information is of a temporal nature, and related to something that happened on a particular date, or range of dates. Since I wanted to be able to create timelines, I needed the ability to associate extracted and summarised items of information with an Event rather than directly with the source. The Event is effectively an intermediary to the source of the data.

As an example, consider a birth registration. There would be a single Event for the birth, and several Persons would be associated with it — including the child (or children) being born. The properties relating each Person, such as their residence and occupation, would be associated with each Event-to-

Person linkage, not directly with either the shared Event or the source description.

3.7 Inheritance

In order to avoid duplication, and reduce the risk of errors, I wanted an inheritance mechanism where details of an event, source citation, or a resource could be taken from a prior instance.

For example, if several pages of a book were cited then it could factor out all the common details, such as title and author, and then each derived citation need only specify the information that differs, such as a chapter or page. A similar situation occurs with an event such as a census night. Details such as the date, and some parts of the census reference, could be inherited from a single definition.

4 Data Model Case Studies

This section will take a closer look at some important parts of the STEMMA Data Model. Their importance will be illustrated with specific use cases based on familiar genealogical problems.

4.1 Timelines

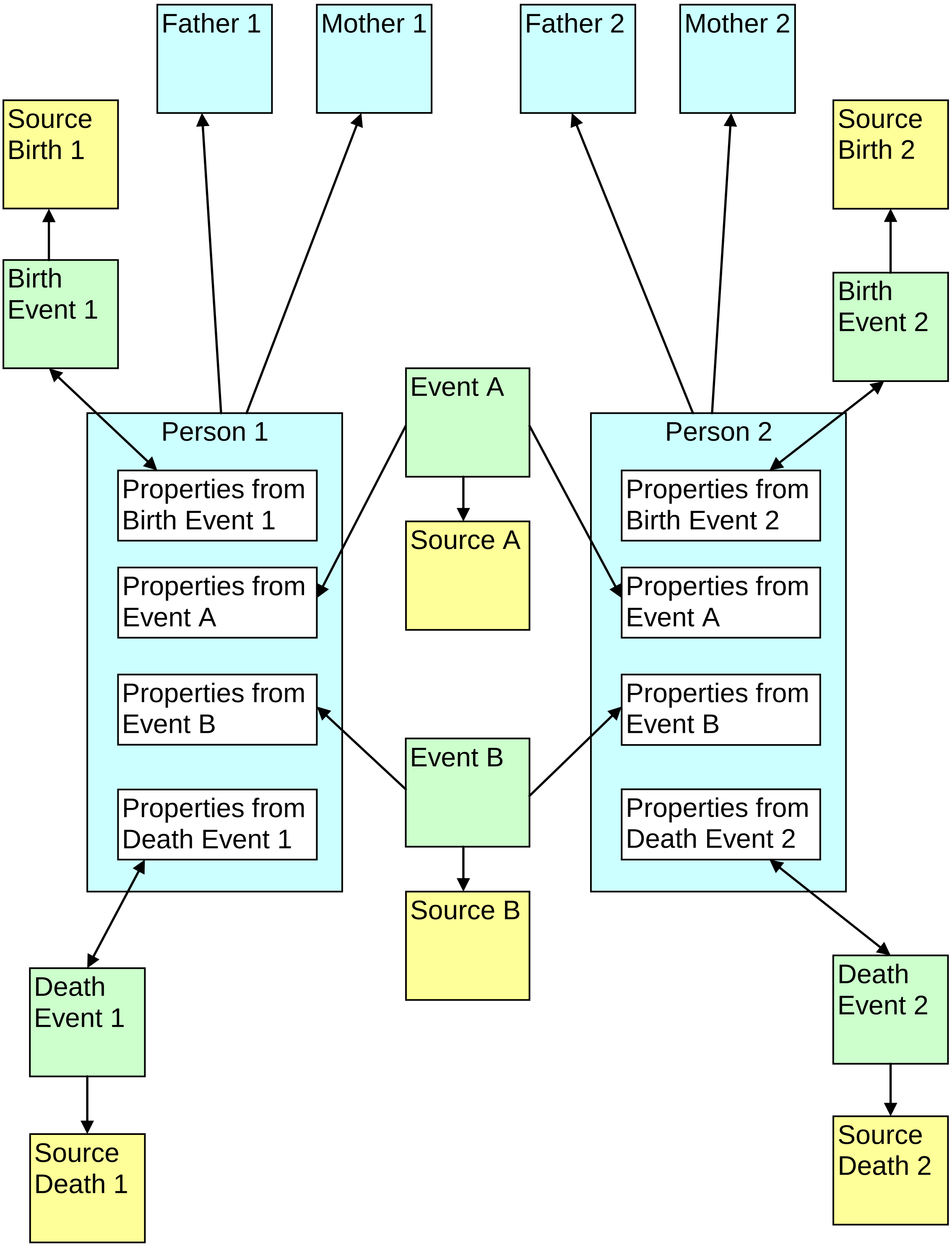
In this section, I want to illustrate how STEMMA deals with time-dependent information for a subject, such as a person, and how it incorporates that information into its conclusional sub-model.

The following diagram shows entities for two people (Person 1 and Person 2). They have distinct parents but have two shared Events that they were both present at (e.g. a census).

Every STEMMA Person can have a direct link to their biological parents (if known), and to a single birth Event and possible death Event. In this diagram, the mothers, and probably the fathers, should also have shared the respective birth Events with their children but that was omitted for clarity.

The importance of this mechanism concerns the Properties provided for each Person (or other subject entity, such as Place/Group) by the underlying source data (e.g. an age, or occupation). Such Properties are specific to a subject entity and to the supporting source, but are generally specific to a given date too. Rather than associate them directly with the subject entity, or the supporting source, or even the Event, STEMMA associates them with each Event-to-entity link. Since the Event will reference the source already then this is the natural way to attach the Properties. Doing the same for multiple Events provides a valuable timeline for a Person’s life.

Note that these Properties are sets of named values constituting extracted and summarised information from the supporting source(s). A Person’s full name(s), sex, birth event, death event, and parentage are obviously part of the *conclusional sub-model* but so are Properties as they are associated with conclusion entities and may reference other conclusion entities. The details from those Properties may be consolidated to form, for instance, a conclusion birth Event — the Person does not have multiple birth Events (i.e. one for each source).


Father 1
Mother 1
Father 2
Mother 2
Source Birth 1
Source Birth 2
Birth Event 1
Birth Event 2
Event A
Person 1
Person 2
Properties from Birth Event 1
Properties from Birth Event 2
Source A
Properties from Event A
Properties from Event A
Properties from Event B
Properties from Event B
Event B
Properties from Death Event 1
Properties from Death Event 2
Source B
Death Event 1
Death Event 2
Source Death 1
Source Death 2

Let’s examine the syntax used for this mechanism. The following STEMMA fragment indicates that EventA occurred on 1861-04-07 and is supported by one cited source. EventB occurred on 1864-11-17 and is also supported by For the purpose of illustration, only one Property (Age) is derived for each Person from each of the sources, below. Although superfluous to the illustration, we’ve also used custom Event-types in this fragment.

```
<Dataset Name=’Timeline’ xmlns:ev=”http://vocab.company.org/events”>

<Person Key=’pPerson1’>
        <FatherPersonLnk Key=’pFather1’/>
        <MotherPersonLnk Key=’pMother1’/>

        <Birth><EventLnk Key=’eBirth1’/></Birth>
        <Death><EventLnk Key=’eDeath1’/></Death>
</Person>

<Person Key=’pPerson2’>
        <FatherPersonLnk Key=’pFather2’/>
        <MotherPersonLnk Key=’pMother2’/>

        <Birth><EventLnk Key=’eBirth2’/></Birth>
        <Death><EventLnk Key=’eDeath2’/></Death>
</Person>

<Event Key=’eEventA’>
        <When Value=’1861-04-07’/>
        <Type> ev:FamilyMeeting </Type>
        <SourceLnk Key=’sSourceA’>
                <PersonLnk Key=’pPerson1’>
                        <Property Name=’Age’>26</Property>
                </PersonLnk>
                <PersonLnk Key=’pPerson2’>
                        <Property Name=’Age’>10</Property>
                </PersonLnk>
        </SourceLnk>
</Event>

<Event Key=’eEventB’>
        <When Value=’1864-11-17’/>
        <Type> ev:FamilyMeeting </Type>
        <SourceLnk Key=’sSourceB’>
                <PersonLnk Key=’pPerson1’>
                        <Property Name=’Age’>29</Property>
                </PersonLnk>
                <PersonLnk Key=’pPerson2’>
                        <Property Name=’Age’>13</Property>
                </PersonLnk>
        </SourceLnk>
```

```
</Event>

</Dataset>
```

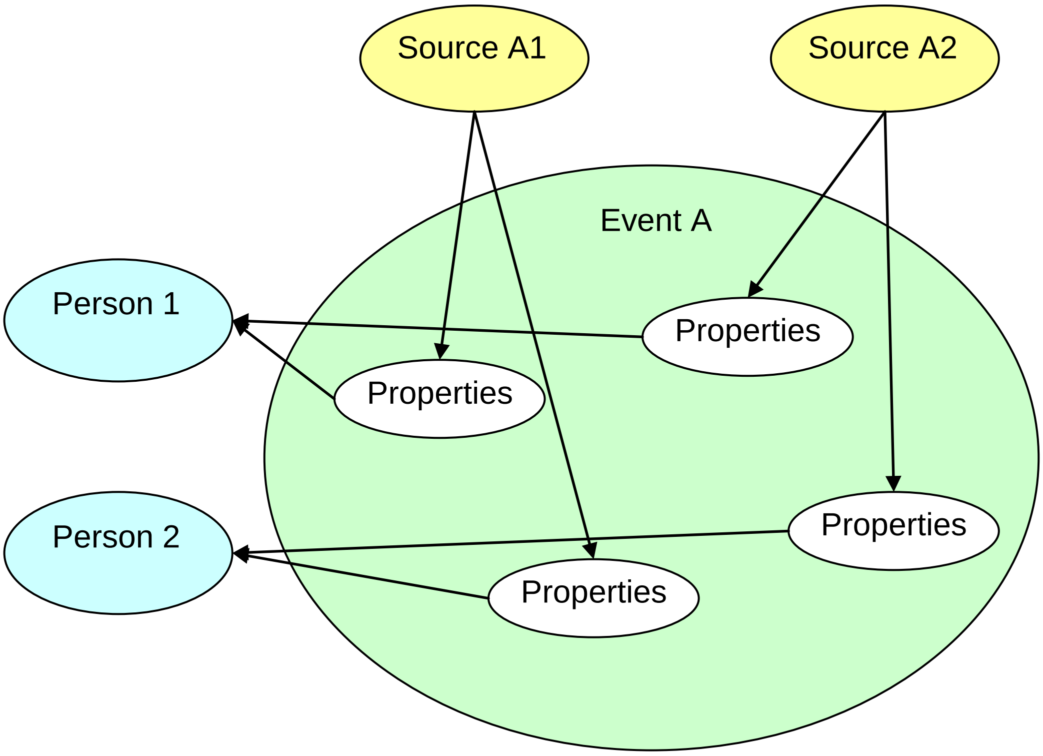
From this, we might conclude that Person1 was born between 1834-11-18 and 1835-04-07, and Person2 was born between 1850-11-18 and1851-04-07. We can therefore construct a birth Event for each Person and indicate the dates are within those ranges, even without a birth certificate or other birth-related source. <Text> elements can indicate how those figures were derived. Our birth Events, in this case, would then be conclusion entities rather than directly citing a birth-related source. The Person's biological parentage, birth sex, and even their name(s), are also conclusions. In contrast, the <SourceLnk> element provides extracted and summarised data from a given source.

When a Person entity is loaded, the identification of the birth/death Events, their name, their sex, and the links to biological parents supports the concept of traditional family trees and pedigree charts. However, the collected Events connected to that Person can be used to create a timeline for their life.

Relationships between people, and particularly unions such as marriage, may be inferred from shared Events, and the Roles or Relationships those people had in those shared Events.

4.2 Multi-Source Events

OK, you might be about to ask 'what happens if you have multiple sources supporting those Events, and the Properties from each are not identical?' Considering just EventA, for a second, this situation might be represented as follows:



In other words, SourceA1 and SourceA2 both support EventA, and are both cited from EventA using different <SourceLnk> elements, but they yield slightly different Properties or Property values. A real-life example might be a marriage supported by both a certificate and a newspaper notice. How does the Event reflect this situation for each Person?

This is quite easy in STEMMA because each Event can have multiple <SourceLnk> elements, and these can each cite a different source. Let's look at a modified version of the above fragment:

```
<Person Key='pPerson1'>
        ... etc ...
</Person>

<Person Key='pPerson2'>
        ... etc ...
</Person>

<Event Key='eEventA'>
        <When Value='1861-04-07'/>
        <SourceLnk Key='sSourceA1'>
                <PersonLnk Key='pPerson1'>
                        <Property Name='Age'>26</Property>
                </PersonLnk>
                <PersonLnk Key='pPerson2'>
                        <Property Name='Age'>10</Property>
        </PersonLnk>
    </SourceLnk>
    <SourceLnk Key='sSourceA2'>
        <PersonLnk Key='pPerson1'>
            <Property Name='Age'>27</Property>
        </PersonLnk>
        <PersonLnk Key='pPerson2'>
            <Property Name='Age'>10</Property>
            <Property Name='Name'>Jack</Property>
        </PersonLnk>
    </SourceLnk>
</Event>
```

4.3 Personal Events

In many cases, an Event in a Person's timeline will not affect anyone else in the data. We still need to provide a date, event-type, source citations, and other similar information, but it should not be necessary to create a top-level Event entity with a unique name, and give that Person an associated role.

Purely for convenience, STEMMA provides a localised Event concept called an Eventlet. Note that this only really presents *personal events* when it is employed in a Person entity, but the Eventlet element is also valid within Place, Animal, and Group entities.

Let's return to the previous example in Timelines. If Person 2 is the only one relevant to Events eEventA and eEventB then they can both be replaced by Eventlets as follows:

```
<Person Key='pPerson2'>
    ... etc ...

    <Eventlet>
        <When Value='1861-04-07'/>
        <Type> ev:FamilyMeeting </Type>
        <SourceLnk Key='sSourceA'>
            <PersonLnk>
                <Property Name='Age'>10</Property>
            </PersonLnk>
        </SourceLnk>
    </Eventlet>

    <Eventlet>
        <When Value='1864-11-17'/>
        <Type> ev:FamilyMeeting </Type>
        <SourceLnk Key='sSourceB'>
            <PersonLnk>
                <Property Name='Age'>13</Property>
            </PersonLnk>
        </SourceLnk>
    </Eventlet>
</Person>
```

Notice that the <PersonLnk> element is not allowed to have an explicit Key in an Eventlet.

Moving on to the extended version of this example in Multi-Source Events, Person 2 now has two information sources supporting the first of those events.

```
<Person Key=’pPerson2’>
    ... etc ...

    <Eventlet>
        <When Value=’1861-04-07’/>
        <Type> ev:FamilyMeeting </Type>
        <SourceLnk Key=’sSourceA1’>
            <PersonLnk>
                <Property Name=’Age’>10</Property>
            </PersonLnk>
        </SourceLnk>
        <SourceLnk Key=’sSourceA2’>
            <PersonLnk>
                <Property Name=’Age’>10</Property>
                <Property Name=’Name’>Jack</Property>
            </PersonLnk>
        </SourceLnk>
    </Eventlet>

    ... etc ...
</Person>
```

4.4 Dual Dates

The concept of Dual Dates (aka Double Dates) is discussed in the research notes under Dates and Calendars. This example will represent the date “12/23 Feb 1750/1751” as a STEMMA Date entity. Note that this example is a full-blown *dual date* rather than a simple *double year*. It actually represents 12 February 1750 in the (Old Style) Julian calendar, and 23 February 1751 in the (New Style) Gregorian calendar.

```
<Date>
    <Text> Example dual date </Text>
    <Value Calendar=’Gregorian’> 1751-02-23 </Value>
    <Value Calendar=’Julian’> 1750-02-12 </Value>
</Date>
```

In the syntax of the <Date> element, the date values (i.e. std-date, using the STEMMA terminology) can specify explicit calendar prefixes. In this instance, the calendar indication is not required in the date values since it is provided by the Calendar attribute, although other contexts may not have that available. Unfortunately, there is no available encoding standard for other calendars at the time of writing, and this is discussed further in the aforementioned research notes.

On looking at the code sample above, you might be asking where the original transcription of the date is. Well, this date entity could be used in a conclusion context, e.g. as a date in an Event definition, or in a source information context, e.g. associated with a transcription.

The following is an example use within an extract from some source.

```
<Text>
        The example dual date is <DateRef> 12/23 Feb 1750/1751
                <Date>
                        <Value Calendar='Gregorian'> 1751-02-23 </Value>
                        <Value Calendar='Julian'> 1750-02-12 </Value>
                </Date>
        </DateRef>
</Text>
```

The next example is for the value of a date Property. Remember that a STEMMA Property is just an extracted item of information from a source.

```
<Property Name='DateOfReg'> 12/23 Feb 1750/1751
        <Date>
                <Value Calendar='Gregorian'> 1751-02-23 </Value>
                <Value Calendar='Julian'> 1750-02-12 </Value>
        </Date>
</Property>
```

In both cases, the original text is included verbatim, and supplemented by a date entity representing the computer-readable dates. If the computer-readable date were simpler, such as a normal Gregorian date, then the date entity could be replaced with a STEMMA date value string. For instance:

```
<Property Name='DateOfReg' Value='17-03-1903'> St Patrick's Day, 1903
</Property>
```

Also, any transcription anomalies, such as uncertain characters, struck-out text, or alternative spellings, can be marked-up in the original value. For instance, the following would highlight a typographical error:

```
<Property Name='DateOfReg' Value='17-03-1903'>
        St Patrick's <Alt Value='Day'>Dat</Alt>, 1903
</Property>
```

4.5 Multi-Role Event

This example looks at an Event where a Person may have multiple roles. It will use a club social event but to make it more interesting it introduces some custom role types and a custom Property.

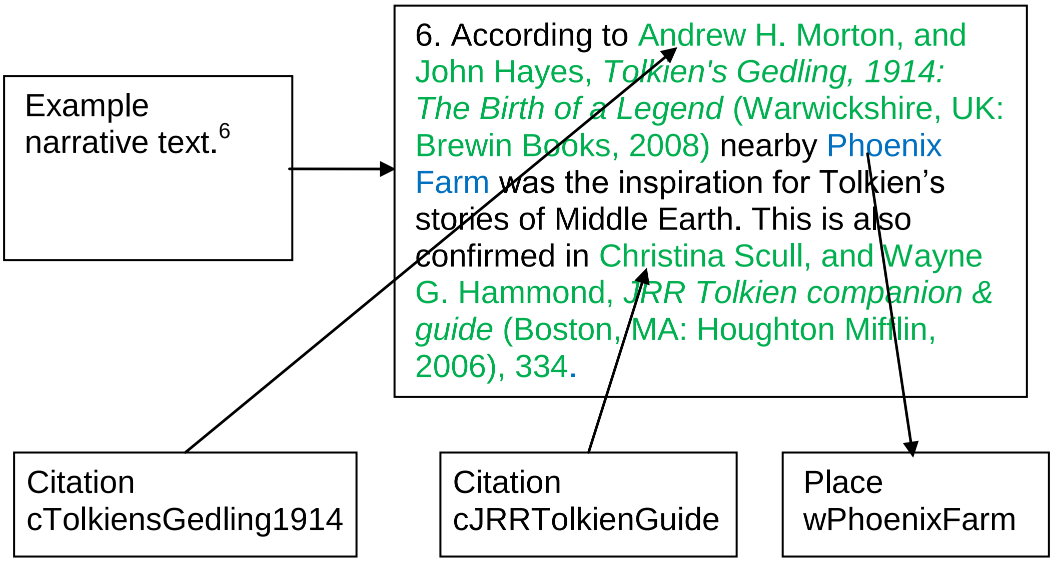
```
<Dataset Name=’Multi_Role_Example’
        xmlns:roles=’http://mydomain.com/roles’
        xmlns:props=’mailto:name@emaildomain.com?subject=properties’
        >

<ExtendedProperties>
        <PersonProperties>
                <PropertyDef Name=’props:MemberID’  Type=’Integer’/>
        </PersonProperties>
</ExtendedProperties>

<Event Key=’eClubSocial’>
        <When Value=’1960-06-09’/>
        <Type> Social </Type>
</Event>

<Person Key=’pGordonBennet’>
        …details of Gordon Bennet…
</Person>

<Event Key=’eClubSocial’>
        <SourceLnk Key=’sClubSocial’>
                <PersonLnk Key=’pGordonBennet’>
                        <Property Name=’Role’>
                                <Item Value=’roles:Photographer’/>
                                <Item Value=’roles:Host’/>
                        </Property>
                        <Property Name=’props:MemberID’>2314</Property>
                </PersonLnk>
        </SourceLnk>
</Event>

</Dataset>
```

In other words, Gordon Bennet was both the host and the photographer at the club meeting, and his membership ID was 2314.

4.6 Complex Citation

This illustration involves a complex citation, i.e. a reference note that includes multiple simple citations and discursive notes. A reference note normally uses analytical notes to comment on the quality or credibility of the associated source. This example is really commentary since it is making a general point of interest, although this does involve references to specific sources.

The scenario is broken down using a general footnote that includes multiple inline simple citations.



Example narrative text.[6]

6. According to Andrew H. Morton, and John Hayes, *Tolkien's Gedling, 1914: The Birth of a Legend* (Warwickshire, UK: Brewin Books, 2008) nearby Phoenix Farm was the inspiration for Tolkien’s stories of Middle Earth. This is also confirmed in Christina Scull, and Wayne G. Hammond, *JRR Tolkien companion & guide* (Boston, MA: Houghton Mifflin, 2006), 334.

Citation cTolkiensGedling1914

Citation cJRRTolkienGuide

Place wPhoenixFarm

Again, let’s examine this as it might be expressed in STEMMA.

```
<Dataset Name=’Example’ xmlns:DC=’http://purl.org/dc/terms/’>

<Citation Key=’cBookMultiAuthor’ Abstract=’1’>
        <Title> Generic citation for published multi-author books </Title>
        <URI> http://stemma.parallaxview.co/source-
        type/book/multiauthor  </URI>
        <Params>
                <Param Name=’Authors’ SemType=’DC:creator’ ItemList=’1’/>
                <Param Name=’Title’ SemType=’DC:title’/>
                <Param Name=’Publisher’ SemType=’DC:publisher’/>
                <Param Name=’Date’ Type=’Date’ SemType=’DC:date’/>
                <Param Name=’Pages’  Optional=’1’ ItemList=’1’/>
        </Params>
</Citation>

<Citation Key=’cTolkiensGedling1914’>
        <Title> Tolkien’s Gedling </Title>
        <BaseCitationLnk Key=’cBookMultiAuthor’/>
        <Params>
                <Param Name=’Authors’>
                        <Item> Andrew H. Morton </Item>
                        <Item> John Hayes </Item>
                </Param>
                <Param Name=’Title’> Tolkien's Gedling, 1914: The Birth of a
                Legend </Param>
                <Param Name='Publisher'> Brewin Books </Param>
                <Param name='PublisherAddr'> Warwickshire, UK </Param>
                <Param Name='Date'>2008</Param>
        </Params>

        <Text>
        In late September 1914, J.R.R. Tolkien, his life in crisis, visited his
        Aunt Jane's Phoenix Farm in Gedling near Nottingham. The poem he
        wrote there on September 24th, "The Voyage of Earendel the
        Evening Star", was the spark that ignited the whole of his later
        mythology. Focusing on this single event, the authors set out to
        discover more about Phoenix Farm, Jane Neave and the poem.
        (information from www.nottinghambooks.co.uk)
        </Text>
</Citation>

<Citation Key='cJRRTolkienGuide'>
        <Title> Tolkien Companion Guide </Title>
        <BaseCitationLnk Key='cBookMultiAuthor'/>
        <Params>
                <Param Name='Authors'>
                        <Item> Christina Scull </Item>
                        <Item> Wayne G. Hammond </Item>
                </Param>
                <Param Name='Title'> JRR Tolkien companion & guide
                </Param>
                <Param Name='Publisher'> Houghton Mifflin </Param>
                <Param name='PublisherAddr'> Boston, MA </Param>
                <Param Name='Date'>2006</Param>
                <Param Name='Pages'>334</Param>
        </Params>
</Citation>

<Place Key='wNotts'>
        <Title> Nottinghamshire </Title>
        <Type> County </Type>
        <PlaceName> Nottinghamshire </PlaceName>
</Place>

<Place Key='wGedling'>
        <Title> Gedling </Title>
        <Type> Village </Type>
        <PlaceName> Gedling </PlaceName>
        <ParentPlaceLnk Key='wNotts'/>
</Place>

<Place Key='wPhoenixFarm'>
        <Title> Phoenix Farm </Title>
        <Type> Building </Type>
```

```
        <PlaceName> Phoenix Farm </PlaceName>
        <ParentPlaceLnk Key='wGedling'/>
</Place>

<Place Key='wGrangeCrescent'>
        <Title> Grange Crescent </Title>
        <Type> Street </Type>
        <PlaceName> Grange Crescent </PlaceName>
        <ParentPlaceLnk Key='wGedling'/>

        <Text Key='tPhoenixFarm'>
                According to <CitationRef Key='cTolkiensGedling1914'
                Mode='Inline'/>, nearby <PlaceRef Key='wPhoenixFarm'/>
                was the inspiration for Tolkien's stories of Middle Earth. This
                is also confirmed in <CitationRef Key='cJRRTolkienGuide'
                Mode='Inline'/>.
        </Text>

        <!-- A reference to this note could be generated as a footnote as
        follows -->

        <Text>
        Example narrative text.<NoteRef Mode='Footnote'>
        <FromText Key='tPhoenixFarm'>
        </NoteRef>
        </Text>
</Place>
```

This example illustrates the inheritance mechanism by having both book citations derive common details from a base citation, analogous to a base class in software programming. Although both the book citations are self-contained, it is possible to specify explicit parameter values in the respective <CitationRef> elements. A good example of this feature is for specifying alternative page references from the same book.

The importance of this example is not so much in that it references two books in some commentary, but that there is sufficient flexibility in the mechanisms to achieve this. STEMMA handles the more conventional issues, such as adding analytical notes or having real multi-source citations, using this same capability of allowing inline citation strings embedded in a general footnote/endnote. A more readable introduction to this feature may be found at: Cite seeing.

4.7 Multiple Births Spanning Midnight

There are few real-life instances of this scenario but it is often wheeled out as a difficult problem: how do you represent two twins who are born either side of midnight in their local time?

The essence of the problem is that they have different dates of birth, and yet they are twins and effectively share the same birth event. If you try to ignore that sharing, and treat each birth independently, then you lose the fact that they are twins.

STEMMA has a couple of possible ways of representing this. The easiest way is to simply treat the Event as having a duration so it begins before midnight and ends after midnight. This is not really satisfactory, though, because you cannot give distinct birth dates to each twin, and you do not know which was born first.

The recommended way is to use a hierarchical Event. Each twin would have their own distinct birth Event, but there would be a higher-level Event representing the ‘birth experience’, for want of another term, that would span the two births. That higher-level Event would nominate each of the separate birth Events as its start and end as appropriate. The roles of the parents would be associated with the shared Event whereas those of the children would be in their respective birth Events.

A real-life case may be found at: Twins born in different years. We’ll use these details as reported in the newspapers as the basis for a local example, and we’ll assume that relevant birth certificates would not substantially disagree if we had them.

```
<Person Key=’pFather’>
        <Sex> 1 </Sex>
        <PersonName> Thomas Rosputni </PersonName>
</Person>

<Person Key=’pMother’>
        <Sex> 0 </Sex>
        <PersonName> Brighid Maura O’Brien Rosputni </PersonName>
</Person>

<Person Key=’pRonan’>
        <Sex> 1 </Sex>
        <FatherPersonLnk Key=’pFather’/>
        <MotherPersonLnk Key=’pMother’/>

        <Birth><EventLnk Key=’eBirthRonan’/></Birth>
</Person>

<Person Key=’pRory’>
        <Sex> 1 </Sex>
        <FatherPersonLnk Key=’pFather’/>
        <MotherPersonLnk Key=’pMother’/>

        <Birth><EventLnk Key=’eBirthRory’/></Birth>
</Person>
```

```
<Event Key=’eBirthExperience’>
        <Type> Birth </Event>
        <SourceLnk Key=’sIrishCentral’>
                <PersonLnk Key=’pMother’>
                        <Property Name=’Name’> Brighid Maura O’Brien
                        Rosputni </Property>
                        <Property Name=’Role’> Mother </Property>
                </PersonLnk>
                <PersonLnk Key=’pFather’>
                        <Property Name=’Name’> Thomas Rosputni
                        </Property>
                        <Property Name=’Role’> Father </Property>
                </PersonLnk>
        </SourceLnk>
</Event>

<Event Key=’eBirthRonan’>
        <Type> Birth </Event>
        <When Value=’2011-12-31’/>
        <ParentEvent Key=’eBirthExperience’/>
        <PlaceLnk Key=’wBuffalo’/>
        <SourceLnk Key=’sIrishCentral’>
                <PersonLnk Key=’pRonan’>
                        <Property Name=’Name’> Ronan Rosputni </Property>
                        <Property Name=’Role’> Child </Property>
                </PersonLnk>
        </SourceLnk>
</Event>

<Event Key=’eBirthRory’>
        <Type> Birth </Event>
        <When Value=’2012-01-01’/>
        <ParentEvent Key=’eBirthExperience’/>
        <PlaceLnk Key=’wBuffalo’/>
        <SourceLnk Key=’sIrishCentral’>
                <PersonLnk Key=’pRory’>
                        <Property Name=’Name’> Rory Rosputni </Property>
                        <Property Name=’Role’> Child </Property>
                </PersonLnk>
        </SourceLnk>
</Event>

<Citation Key=’cIrishCentral’>
        <URI> http://stemma.parallaxview.co/source-type/web-media
        </URI>
        <Params>
                <Param Name=’Author’> Bernie Malone </Param>
                <Param Name=’Title’> Irish American twins born in different
                years in historic first </Param>
                <Param Name='Publisher'> Irish Central </Param>
                <Param name='PublisherAddr'> New York, NY </Param>
                <Param Name='Date'> 2012-01-03 </Param>
                <Param Name='URL'>
                http://www.irishcentral.com/news/Irish-American-twins-
                born-in-different-years-in-historic-first-136582258.html
                </Param>
                <Param Name='Accessed'> 2012-07-09 </Param>
        </Params>
</Citation>

<Source Key='sIrishCentral'>
        <Frame>
                <CitationLnk Key='cIrishCentral'/>
        </Frame>
</Source>
```

There's obviously more information that we could have represented here, such as Brighid's maiden name, her father's details, and her grandparent's details, but this was skipped for clarity.

The parent Event automatically spans the discrete birth dates — which effectively define its start and end dates — and automatically uses the common wBuffalo Place of its child Events as its own effective Place.

The same principle can be used to model the births of triplets, etc., since the span of the parent Event will be from the earliest to the latest. Each discrete birth could even have a different PlaceLnk associated with it (see Twins Born in Different Countries), and the effective Place of the parent Event will be the largest common Place of the birth Events. Although less useful for a birth Event, this feature can be more useful for an emigration Event where embarkation and disembarkation occur in different places and on different dates. By contrast, Twins Born to Different Fathers is actually an easier variation to represent in most models, although software designers must be careful to avoid any presumption about the legality of such data.

This is all very well, you might say, but what about the simple case of twins born on the same day; how could you indicate their birth order. Well, you would use Event-constraints, as in the following example:

```
<Person Key='pAlan'>
        <Sex> 1 </Sex>
        <FatherPersonLnk Key='pFather'/>
        <MotherPersonLnk Key='pMother'/>

        <Birth><EventLnk Key='eBirthAlan'/></Birth>
</Person>

<Person Key='pDavid'>
        <Sex> 1 </Sex>
```

```
    <FatherPersonLnk Key=’pFather’/>
    <MotherPersonLnk Key=’pMother’/>

    <Birth><EventLnk Key=’eBirthDavid’/></Birth>
</Person>

<Event Key=’eBirthAlan’>
    <Type> Birth </Event>
    <When Value=’1988-03-16’>
        <Constraints>
            <Constraint AfterEvent=’eBirthDavid’/>
        </Constraints>
    </When>
</Event>

<Event Key=’eBirthDavid’>
    <Type> Birth </Event>
    <When Value=’1988-03-16’/>
</Event>
```

4.8 Recording Oral History

For this example, we consider a number of conversations with a given family member, each recounting family events, and which were recorded over a period of time.

Let’s assume that three conversations took place on the 1st June, 12th July, and 5th August 2008. These were recorded electronically but notes were also taken.

```
<Person Key=’pGenghis’>
    <Sex> 1 </Sex>
    <Names>
        <Sequences>
            <Canonical> Genghis Khan </Canonical>
            <Sequence>
                <Tokens>
                    <Token> Genghis </Token>
                    <Token> Chinghiz </Token>
                    <Token> Chinghis </Token>
                    <Token> Chingiz </Token>
                    <Token> Temujin </Token>
                </Tokens>
                <Tokens>
                    <Token> Khan </Token>
                </Tokens>
            </Sequence>
        </Sequences>
    </Names>
```

```
        <Eventlet>
                <PlaceLnk Key=’wHospital’/>
                <When Value=’2008-06-01’/>
                <SourceLnk Key=’sConv1’/>
        </Eventlet>

        <Eventlet>
                <PlaceLnk Key=’wHospital’/>
                <When Value=’2008-07-12’/>
                <SourceLnk Key=’sConv2’/>
        </Eventlet>

        <Eventlet>
                <PlaceLnk Key=’wHospital’/>
                <When Value=’2008-08-05’/>
                <SourceLnk Key=’sConv3’/>
        </Eventlet>
</Person>

<Place Key=’wHospital’>
        <PlaceName> Home for the Terminally Bewildered </PlaceName>
</Place>

<Citation Key=’cConv1’>
        <BaseCitationLnk Key=’cConversation’/>
        <Params>
                <Param Name=’Source’ Key=’pGenghis’/>
                <Param Name=’Date’> 2008-06-01 </Param>
        </Params>
</Citation>

<Citation Key=’cConv2’>
        <BaseCitationLnk Key=’cConversation’/>
        <Params>
                <Param Name=’Source’ Key=’pGenghis’/>
                <Param Name=’Date’> 2008-07-12 </Param>
        </Params>
</Citation>

<Citation Key=’cConv3’>
        <BaseCitationLnk Key=’cConversation’/>
        <Params>
                <Param Name=’Source’ Key=’pGenghis’/>
                <Param Name=’Date’> 2008-08-05 </Param>
        </Params>
</Citation>

<Resource Key=’rConv1’>
```

```
        <BaseResourceLnk Key=’rRecordings’/>
        <Params>
                <Param Name=’File’> Conv_2008_06_01 </Param>
        </Params>
        <Text><voice>
                My ally Jamukha also wants to be a ruler of Mongol tribes.
        </voice></Text>
</Resource>

<Resource Key=’rConv2’>
        <BaseResourceLnk Key=’rRecordings’/>
        <Params>
                <Param Name=’File’> Conv_2008_07_12 </Param>
        </Params>
        <Text><voice>
                The shaman is trying to drive a wedge between me and my
                loyal brother, Khasar.
        </voice></Text>
</Resource>

<Resource Key=’rConv3’>
        <BaseResourceLnk Key=’rRecordings’/>
        <Params>
                <Param Name=’File’> Conv_2008_08_05 </Param>
        </Params>
        <Text><voice>
                A council of Mongol chiefs at Khuruldai acknowledged me as
                "Khan" of the consolidated tribes and gave me the new title
                Genghis Khan.
        </voice></Text>
</Resource>

<Source Key=’sConv1’>
        <Frame>
                <CitationLnk Key=’cConv1’/>
                <ResourceLnk Key=’rConv1’/>
                <Credibility> Questionable </Credibility>
        </Frame>
</Source>

<Source Key=’sConv2’>
        <Frame>
                <CitationLnk Key=’cConv2’/>
                <ResourceLnk Key=’rConv2’/>
                <Credibility> Questionable </Credibility>
        </Frame>
</Source>

<Source Key=’sConv3’>
```

```
        <Frame>
                <CitationLnk Key=’cConv3’/>
                <ResourceLnk Key=’rConv3’/>
                <Credibility> Questionable </Credibility>
        </Frame>
</Source>

<!-- Base Entities -->

<Citation Key=’cConversation’ Abstract=’1’>
        <URI> http://stemma.parallaxview.co/source-type/testimony-1
        </URI>
        <Params>
                <Param Name=’Source’ Type=’PersonRef’/>
                <Param Name=’Date’ Type=’Date’/>
        </Params>
</Citation>

<Resource Key=’rRecordings’ Abstract=’1’>
        <Title> Voice recording: ${File} </Title>
        <Type> Recording </Type>
        <URL ContentType=’audio/mpeg’>
        file:myrecordings/sound/{$File}.mp3 </URL>
        <Params>
                <Param Name=’File’/>
        </Params>
</Resource>
```

So, we have a Person entity that is associated with three events; one for each conversation. Since these are non-shared events, and so only relate to the one Person entity, their details are embedded using Eventlet elements rather than the Person being connected to Event entities via <SourceLnk> elements. Although the events all took place at the same Hospital, the use of Eventlet precludes the example from using inheritance to take a Place reference from a generic base-Event.

We have three separate Citations (cConv1-3) but these inherit data such as the URI, source parameters, and information credibility, from a generic base-Citation.

We have three separate recordings (rConv1-3) and these inherit data such as the base URL, parameter names & types, and resource-type, from a generic base-Resource. Each Resource also includes a transcribed abstract of the associated recording.

The example illustrates how reuse can be achieved through inheritance and parameterisation. See Dialogue Transcription for an example of transcribing multi-person conversations, including gestures, pauses, and intonation.

4.9 Evidence, Reasoning, and Conclusion

The notes under Evidence and Conclusion make a case for not only distinguishing *source information* and *conclusion* (which most people would expect), but also *reasoning* (which implies *evidence* too). The rationale partly being that the level of sharing expected in collaborative trees will differ for each of them. BMD information is expected for constructing family trees and pedigree charts but this data is effectively part of a conclusion model. Someone will have had to research the available evidence and infer such things as lineage. The more effort expended in that research then the less inclined the researcher will be to give it away freely.

This example focuses on the birth of a particular person and suggests how STEMMA might represent conflicting evidence, the reasoning, and the associated conclusions. The idea is that this approach can be extrapolated to more complex cases.

The example is actually another version of the data presented for William Elliott in the STEMMA Example section. The birth date for William is a conclusion as we have no birth certificate, but we do have an indication of his age from four census returns and one marriage certificate. Unfortunately, those ages are not all in agreement. Note firstly that there was a single birth date and a single birth event, but we have separate items of evidence to work with; what we do not do is record multiple birth events!

```
<Person Key=’pWilliamElliott’>
        <PersonName> William Elliott </PersonName>
        <Sex> 1 </Sex>

        <Birth><EventLnk Key=’eBirthWilliamElliott’/></Birth>
</Person>

<Event Key=’eBirthWilliamElliott’>
        <Title>Birth of William Elliott </Title>
        <Type> Birth </Type> <SubType> Birth </SubType>
        <When DetLnk=’mWilliamElliott’><Date><Range>
                <Min> 1840-03-01 </Min> <Max> 1841-04-01 </Max>
        <Range></Date></When>
        <PlaceLnk Key=’wUttoxeter’/>
</Event>

<Event Key=’eCensusElliott1851’>
        <When Value=’1851-03-30’/>
        <Title>1851 census for William Elliott</Title>
        <Type> Survey </Type> <SubType> Census </SubType>
        <PlaceLnk Key=’wTinkersLane’/>
        <SourceLnk Key=’sCensusElliott1851’>
                <PersonLnk Key='pWilliamElliott'>
                        <Property Name=’Age’> 10 </Property>
                        <Property Name=’Occupation’> Scholar </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Event Key=’eCensusElliott1861’>
    <When Value=’1861-04-07’/>
    <Title>1861 census for William Elliott</Title>
    <Type> Survey </Type> <SubType> Census </SubType>
    <PlaceLnk Key=’wRussellStreet’/>
    <SourceLnk Key=’sCensusElliott1861’>
        <PersonLnk Key='pWilliamElliott'>
            <Property Name=’Age’> 20 </Property>
            <Property Name=’Occupation’> Labourer </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Event Key=’eCensusElliott1871’>
    <When Value=’1871-04-02’/>
    <Title>1871 census for William Elliott</Title>
    <Type> Survey </Type> <SubType> Census </SubType>
    <PlaceLnk Key=’wSiddalsLane62’/>
    <SourceLnk Key=’sCensusElliott1871’>
        <PersonLnk Key='pWilliamElliott'>
            <Property Name=’Age’> 31 </Property>
            <Property Name=’Occupation’> Labourer in Iron works
            </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Event Key=’eCensusElliott1881’>
    <When Value=’1881-04-03’/>
    <Title>1881 census for William Elliott</Title>
    <Type> Survey </Type> <SubType> Census </SubType>
    <PlaceLnk Key=’wCarringtonSq14’/>
    <SourceLnk Key=’sCensusElliott1881’>
        <PersonLnk Key='pWilliamElliott'>
            <Property Name=’Age’ Surety=’10%’> 35 </Property>
            <Property Name=’Occupation’> Striker Iron Foundry
            </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Event Key=’eMarriageElliott1862’>
    <When Value=’1862-03-12’/>
    <Title>Marriage of William Elliott and Sarah Wildgoose</Title>
    <Type> Union </Type> <SubType> Marriage </SubType>
```

```
        <PlaceLnk Key=’wDerbyRegOffice’/>
        <SourceLnk Key=’sMarriageElliott1862’>
            <PersonLnk Key='pWilliamElliott'>
                <Property Name=’Age’> 21 </Property>
                <Property Name=’Occupation’> Hammersman
                </Property>
            </PersonLnk>
        </SourceLnk>
    </Event>

    <Source Key=’sCensusElliott1851’>
        <Frame>
            <When Value=’1851-03-30’/>
            <CitationLnk Key=’eCensusElliott1851’/>
            <ResourceLnk Key=’rCensusElliott1851’/>
        </Frame>
        <ProtoPerson Key='dpWilliamElliott1851'>
            <Link DetLnk=’dsAge1851’ Value=’10’>
                <Text>age</Text>
            </Link>
        </ProtoPerson>
    </Source>

    <Source Key=’sCensusElliott1861’>
        <Frame>
            <When Value=’1861-04-07’/>
            <CitationLnk Key=’eCensusElliott1861’/>
            <ResourceLnk Key=’rCensusElliott1861’/>
        </Frame>
        <ProtoPerson Key='dpWilliamElliott1861'>
            <Link DetLnk=’dsAge1861’ Value=’20’>
                <Text>age</Text>
            </Link>
        </ProtoPerson>
    </Source>

    <Source Key=’sCensusElliott1871’>
        <Frame>
            <When Value=’1871-04-02’/>
            <CitationLnk Key=’eCensusElliott1871’/>
            <ResourceLnk Key=’rCensusElliott1871’/>
        </Frame>
        <ProtoPerson Key='dpWilliamElliott1871'>
            <Link DetLnk=’dsAge1871’ Value=’31’>
                <Text>age</Text>
            </Link>
        </ProtoPerson>
    </Source>
```

```
<Source Key=’sCensusElliott1881’>
        <Frame>
                <When Value=’1881-04-03’/>
                <CitationLnk Key=’eCensusElliott1881’/>
                <ResourceLnk Key=’rCensusElliott1881’/>
        </Frame>
        <ProtoPerson Key='dpWilliamElliott1881'>
                <Link DetLnk=’dsAge1881’ Value=’35’>
                        <Text>age</Text>
                </Link>
        </ProtoPerson>
</Source>

<Source Key=’sMarriageElliott1862’>
        <Frame>
                <When Value=’1862-03-12’/>
                <CitationLnk Key=’eMarriageElliott1862’/>
                <ResourceLnk Key=’rMarriageElliott1862’/>
        </Frame>
        <ProtoPerson Key='dpWilliamElliott1862'>
                <Link DetLnk=’dsAge1862’ Value=’21’>
                        <Text>age</Text>
                </Link>
        </ProtoPerson>
</Source>

<Matrix Key=’mWilliamElliott’>
        <Frame>
                <SourceLnk Key=’sCensusElliott1851’/>
                <SourceLnk Key=’sCensusElliott1861’/>
                <SourceLnk Key=’sCensusElliott1871’/>
                <SourceLnk Key=’sCensusElliott1881’/>
                <SourceLnk Key=’sMarriageElliott1862’/>
        </Frame>
        <ProtoPerson Key=’pWilliamElliott’>
                <Link DetLnk='dpCensusElliott1851'/>
                <Link DetLnk='dpCensusElliott1861'/>
                <Link DetLnk='dpCensusElliott1871'/>
                <Link DetLnk='dpCensusElliott1881'/>
                <Link DetLnk='dpWilliamElliott1862'/>

                <Link Value=’1840 to 1841’ Type=’Inference’>
                        <Text>birthdate</Text>
                </Link>

                <Text Inference=’1’>
                William’s date of birth here is derived from his age in various
                census returns, and at the time of his first marriage. All but
                the 1881 census put his date of birth around 1840 to 1841. His
                age in the 1881 census is an outlier due to his age probably
                being estimated at 35 by the proprietor of <PlaceRef
                Key='wCarringtonSq14'/> where he and his wife were lodging
                on <DateRef Value='1881-04-03'/>.
                </Text>
        </ProtoPerson>
</Matrix>
```

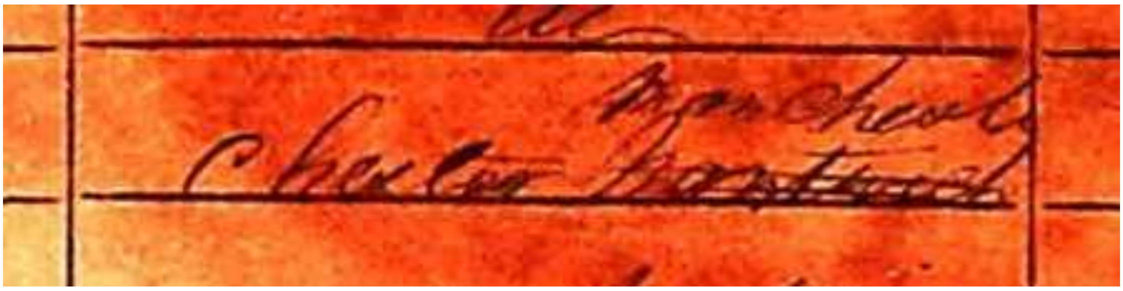
What this is showing is that the Event entity representing the birth has a single approximate date, based on assessment of evidence. There is a link from there to the Matrix entity that correlates evidence from different sources, and links from there to those different sources. Each of the respective Source entities implies a different date-of-birth by subtracting the age from the date that the source information was recorded — possibly using a tool such as a Birth-date Calculator. Each Source entity also links the corresponding age to the original source fragment (usually in a transcription).

4.10 Transcription Anomalies

The STEMMA model strives to represent source information, including transcription anomalies such as uncertain characters or annotation, in conjunction with other data. It is insufficient for any data model to only focus on conclusions — with or without the reasoning used to achieve them; source information is part of a data collection rather than simply being something in an external document or online.

Whereas the example at Evidence, Reasoning, and Conclusion shows a source-upwards approach to handling conflicting evidence, this example will employ STEMMA's other mechanism for associating information with subject entities: Properties. Although they do not have the same flexibility for detailed analysis, they do capture both the transcribed information and a normalised version of it. It is therefore a more convenient mechanism for database-orientated products.

The image below shows the birth place for an Alfred Campbell in the 1851 census of England and Wales. This particular entry had a correction applied by the enumerator since it originally said "Chester Nantwich" but Nantwich was crossed-out and replaced with something similar to "Manchester".



Manchester
Chester ~~Nantwich~~

This introduces a few issues: Chester is a city rather than a county, but it is the “county town” of Cheshire. The original town of Nantwich is within Cheshire, but the new town of Manchester is not.

```
<Person Key='pAlfredCampbell'>
    <Title> Alfred Campbell </Title>
    <Sex> 1 </Sex>
</Person>

<Event Key='eCensusCampbell1851'>
    <SourceLnk Key=’sCensusCampbell1851’>
        <PersonLnk Key=’pAlfredCampbell’>
            <Property Name='Name'> Alfred Campbell </Property>
            <Property Name='Age'> 11 </Property>
            <Property Name='Occupation'> Calico Printer
            (Journeyman) Child </Property>
            <Property Name='BirthPlace' Key='wManchester'>
            Chester <s>Nantwich</s> <Alt
            Value=’Manchester’>Manchest</Alt>
            </Property>
            <Property Name='Role'> Lodger </Property>
            <Property Name='Status'> Unmarried </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Place Key=’wManchester’>
    <Title> Manchester </Title>
    <Type> Town </Type>
    <PlaceName> Manchester </PlaceName>
    <ParentPlaceLnk Key=’wLancs’/>
</Place>

<Place Key=’wLancs’>
    <Title> Lancashire </Title>
    <Type> County </Type>
    <PlaceName> Lancashire </PlaceName>
</Place>
```

What this STEMMA transcription has done is to record each of the three visible words: “Chester”, “Nantwich”, and “Manchest” for the Property called ‘BirthPlace’. It records that “Nantwich” was struck-out, and it also associates an alternative spelling with “Manchest” to equate it with Manchester. However, the Key attribute on the Property is a conclusion that Alfred’s birthplace was, indeed, Manchester in the county of Lancashire.

This example keeps the details of the original within the Property definition, and it doesn’t indicate that the text was in manuscript (rather than typescript) form, or that “Manchest” was written above the line, implying a correction.

STEMMA does have more comprehensive mechanisms for doing all this within a transcription, and a corresponding example can be found at: Transcription Anomalies.

4.11 Family Units

STEMMA doesn’t include a specific Family element because the concept of a family-unit is too subjective. See under Family Units. However, its generic Group entity does have some predefined types that can model most types of family unit.

The following illustration uses an example of a John Smith who marries a Jane Doe. They and their unmarried children constitute a traditional Nuclear family (also called a Conjugal family) from the time of their marriage. Their children are part of this unit from their birth and until their own marriages.

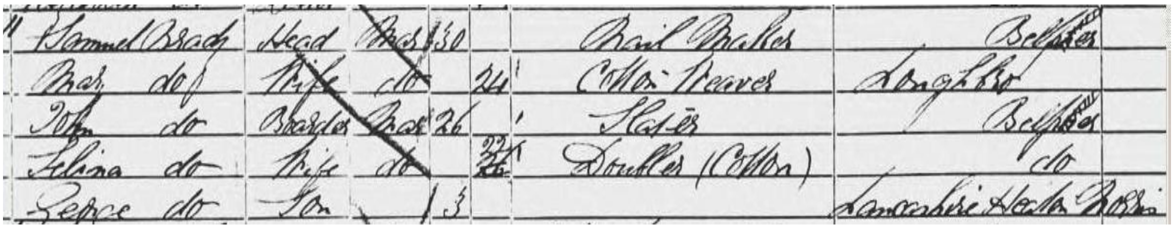
```
<Person Key=’pJohnSmith’>
    <Title> John Smith </Title>
    <Sex> 1 </Sex>
    <MemberOf Key=’gJohnJaneSmith’
        FromEvent=’eMarriageJohnJane’/>
</Person>

<Person Key=’pJaneDoe’>
    <Title> Jane Smith née Doe </Title>
    <Sex> 0 </Sex>
    <MemberOf Key=’gJohnJaneSmith’
        FromEvent=’eMarriageJohnJane’/>

    <Names>
        <Sequences BeforeEvent=’eMarriageJohnJane’>
            <Canonical> Jane Doe </Canonical>
            <Sequence>
                <Tokens>
                    <Token> Jane </Token>
                </Tokens>
                <Tokens>
                    <Token> Doe </Token>
                </Tokens>
            </Sequence>
        </Sequences>
        <Sequences FromEvent=’eMarriageJohnJane’ Type=’Married’>
            <Canonical> Jane Smith </Canonical>
            <Sequence>
                <Tokens>
                    <Token> Jane </Token>
                </Tokens>
                <Tokens>
                    <Token> Smith </Token>
                </Tokens>
            </Sequence>
        </Sequences>
    </Names>
</Person>

<Person Key=’pTomSmith’>
    <Title> Thomas Smith </Title>
    <Sex> 1 </Sex>
    <FatherPersonLnk Key=’pJohnSmith’/>
    <MotherPersonLnk Key=’pJaneDoe’/>
    <MemberOf Key=’gJohnJaneSmith’
        UntilEvent=’eMarriageTom’/>
</Person>

<Person Key=’pSarahSmith’>
    <Title> Sarah Smith </Title>
    <Sex> 0 </Sex>
    <FatherPersonLnk Key=’pJohnSmith’/>
    <MotherPersonLnk Key=’pJaneDoe’/>
    <MemberOf Key=’gJohnJaneSmith’
        UntilEvent=’eMarriageSarah’/>
</Person>

<Event Key=’eMarriageJohnJane’>
    <When Value=’1920-01-30’/>
    <Type> Union </Type> <SubType> Marriage </SubType>
    <PlaceLnk Key=’wStElsewhere’/>
</Event>

<Event Key=’eMarriageSarah’>
    <When Value=’1946-08-20’/>
    <Type> Union </Type> <SubType> Marriage </SubType>
    <PlaceLnk Key=’wStElsewhere’/>
</Event>

<Event Key=’eMarriageTom’>
    <When Value=’1942-05-20’/>
    <Type> Union </Type> <SubType> Marriage </SubType>
    <PlaceLnk Key=’wStElsewhere’/>
</Event>

<Group Key=’gJohnJaneSmith’>
    <Title> John and Jane Smith’s family </Title>
    <Type> Family </Type> <SubType> Nuclear </SubType>
</Group>
```

This example also shows how dates implied by Events, as opposed to explicit dates, can be used to control both Group association and alternative personal names.

The example is idealised insomuch as it assumes marriage always precedes childbirth but the general approach should be clear.

4.12 Census Roles

This next example picks a page from the 1861 census of England & Wales (Piece: 2560, Folio: 23, Page: 6). It contains a household for the following family of five residing at 8 Homleys Court, Heaton Norris, Stockport, Cheshire.



Samuel Brady	Head	Mar	30		Nail Maker	Belper
Mary do	Wife	do		24	Cotton Weaver	Loughbro
John do	Boarder	Mar	26		Slater	Belper
Selina do	Wife	do		~~26~~ 22	Doubler (Cotton)	do
George do	Son		3			Lancashire Heaton Norris

Many census returns have inaccuracies. However, this one is interesting because the recorded relationships are incorrect and can result in the wrong associations being recorded. In the census column headed ‘Relation to Head of Family’ there are two women with a relationship of ‘Wife’. For products that insist on recording — and indexing on — the data verbatim, this results in illegal spousal relationships and misdirected searches.

The details of this family should have been:

Name	Relation	Condition	Sex	Age	Birth Year	Occupation	Birth Place
Samuel Bradley	Head	Married	M	30	1831	Nail Maker	Belper, Derbyshire
Mary Bradley	Wife	Married	F	24	1837	Cotton Weaver	Lougborough, Leicestershire
John Bradley	Boarder	Married	M	26	1835	Slater	Belper, Derbyshire
Selina Bradley	Boarder’s Wife	Married	F	22	1839	Doubler (Cotton)	Belper, Derbyshire
George Bradley	Boarder’s Son	-	M	3	1858	-	Heaton Norris, Lancashire

There were several problems here though:

- The relationship of Selina should have been wife to John (the boarder), not wife to Samuel (the head).
- The surname appears to be recorded as ‘Brady’ but the family can be identified as ‘Bradley’ using other data.
- Selina’s age was initially recorded as 26, but then corrected to 22.

- Mary’s birth place was recorded as either ‘Longhbro’ or ‘Loughbro’ which caused some transcribers to interpret it as ‘Longborough’.

So how do we record the information and the conclusions, and justify how one relates to the other? The following sample code illustrates the combination of the two mechanisms that STEMMA has for recording information and associating it with relevant entities: Property values (part of the conclusional sub-model) and Source entities (part of the informational sub-model). Property values are extracted and summarised items of information recorded according to a software taxonomy, and the actual values may include the original transcription and a normalised version of it. The Source entity allows a network of information and inference to be constructed from source fragments upwards, before making associations with conclusion entities. Both are presented here for comparison.

```
<!-- Conclusion entities -->

<Person Key=’pSamuelBradley’>
        <PersonName> Samuel Bradley </PersonName>
        <Sex> 1 </Sex>
</Person>

<Person Key=’pMaryBradley’>
        <PersonName> Mary Bradley </PersonName>
        <Sex> 0 </Sex>
</Person>

<Person Key=’pJohnBradley’>
        <PersonName> John Bradley </PersonName>
        <Sex> 1 </Sex>
</Person>

<Person Key=’pSelinaBradley’>
        <PersonName> Selina Bradley </PersonName>
        <Sex> 0 </Sex>
</Person>

<Person Key=’pGeorge Bradley’>
        <PersonName> George Bradley </PersonName>
        <Sex> 1 </Sex>
</Person>

<Place Key=’w8HomleysCourt’>
        <Type> Number </Type>
        <PlaceName> 8 </PlaceName>
        <ParentPlaceLnk Key='wHomleysCourt'/>
</Place>

<Place Key=’wHomleysCourt’>
```

```
        <Type> Street </Type>
        <PlaceName> Homleys Court </PlaceName>
        <ParentPlaceLnk Key='wStockport'/>
</Place>

<Place Key=’wBelper’>
        <Type> Town </Type>
        <PlaceName> Belper </PlaceName>
        <ParentPlaceLnk Key='wDerbyshire'/>
</Place>

<Event Key=’eCensusBradley1861’>
        <When Value=’1861-04-07’/>
        <Title> 1861 census for Bradley family </Title>
        <Type> Survey </Type>
        <SubType> Census </SubType>
        <PlaceLnk Key=’w8HomleysCourt’/>

        <SourceLnk Key=’sCensusBradley’>
                <PersonLnk Key=‘pSamuelBradley’>
                        <Property Name=’Name’ Value=’Samuel Bradley’>
                                <Alt>Samuel Brady
                                <FromText Key=‘tBradySurname’>
                                </Alt>
                        </Property>

                        <Property Name=’ResidencePlace
                        Key=’w8HomleysCourt’/>
                        <Property Name=’Age’> 30 </Property>
                        <Property Name=’Occupation’>
                                Nail maker
                        </Property>
                        <Property Name=’BirthPlace‘ Key='wBelper'/>
                        <Property Name=’Role’> Head </Property>
                        <Property Name=’Status’> Married </Status>
                </PersonLnk>

                <PersonLnk Key=‘pMaryBradley’>
                        <Property Name=’Name’ Value=’Mary Bradley’>
                                <Alt>Mary Brady
                                <FromText Key=‘tBradySurname’>
                                </Alt>
                        </Property>

                        <Property Name=’ResidencePlace
                        Key=’w8HomleysCourt’/>
                        <Property Name=’Age’> 24 </Property>
                        <Property Name=’Occupation’> Cotton Weaver
                        </Property>
```

```
    <Property Name=’BirthPlace‘ Key='wLoughborough'/>
    <Property Name=’Relationship’
    Key=’pSamuelBradley’>
        Wife
    </Property>
    <Property Name=’Status’> Married </Status>
</PersonLnk>

<PersonLnk Key=‘pJohnBradley’>
    <Property Name=’Name’ Value=’John Bradley’>
        <Alt>John Brady
        <FromText Key=‘tBradySurname’>
        </Alt>
    </Property>

    <Property Name=’ResidencePlace
    Key=’w8HomleysCourt’/>
    <Property Name=’Age’> 26 </Property>
    <Property Name=’Occupation’> Slater </Property>
    <Property Name=’BirthPlace‘ Key='wBelper'/>
    <Property Name=’Role’> Boarder </Property>
    <Property Name=’Status’> Married </Status>
</PersonLnk>

<PersonLnk Key=‘pSelinaBradley’>
    <Property Name=’Name’ Value=’Selina Bradley’>
        <Alt>Selina Brady
        <FromText Key=‘tBradySurname’>
        </Alt>
    </Property>

    <Property Name=’ResidencePlace
    Key=’w8HomleysCourt’/>
    <Property Name=’Age’ Value=’22’>
        <s>26</s> 22
    </Property>
    <Property Name=’Occupation’>
        Doubler (Cotton)
    </Property>
    <Property Name=’BirthPlace‘ Key='wBelper'/>
    <Property Name=’Relationship’ Key=’pJohnBradley’>
        <Alt>Wife
        <FromText Key=‘tSelinaRole’/>
        </Alt>
    </Property>
    <Property Name=’Status’> Married </Status>
</PersonLnk>

<PersonLnk Key=‘pGeorgeBradley’>
```

```
            <Property Name=’Name’ Value=’George Bradley’>
                <Alt>George Brady
                <FromText Key=‘tBradySurname’>
                </Alt>
            </Property>

            <Property Name=’ResidencePlace
            Key=’w8HomleysCourt’/>
            <Property Name=’Age’> 3 </Property>
            <Property Name=’BirthPlace‘ Key='wHeatonNorris'>
                Lancashire Heaton Norris
            </Property>
            <Property Name=’Relationship’ Key=’pJohnBradley’>
                <Alt>Son
                <FromText Key=’tGeorgeRole’/>
                </Alt>
            </Property>
        </PersonLnk>

        <PlaceLnk DetKey=’wBelper’>
            <Property Name=’Name’> Belper </Property>
        </PlaceLnk>

        <PlaceLnk Key='wLoughborough'>
            <Property Name=’Name‘>
                <Alt>Lo<Ucf>[nu]</Ucf>ghbro
                <FromText Key=‘tLonghbro’>
                </Alt>
            </Property>
        </PlaceLnk>

        <PlaceLnk Key=’w8HomleysCourt’>
            <Property Name=’Name’>
                8 Homleys Court
            </Property>
        </PlaceLnk>

        <Property Name=’Where’ Key=’w8HomleysCourt’/>
    </SourceLnk>
</Event>
```

The Property values can provide both an extracted copy of the relevant information and a normalised copy of the value. For instance, a copy of a date as-written and a representation of it using STEMMA’s date syntax. There is also some limited scope for inserting explanation into the interpretations, such as the identification of the place Loughborough, or the interpretation of the family’s surname. The Property values are dynamic and so capable of giving a potted history of some subject through the different Events in which they were mentioned. However, this mechanism is really aimed at database orientated genealogy where named fields are the norm. It deals with conclusions and so a relationship — such as that of Mary to Samuel — has to be expressed between their respective conclusion entities. If the case where two person references had not been given conclusion entities (see Incidental People) then the relationship could not be expressed easily.

Let’s look at the other mechanism:

```
<!-- Source analysis -->

<Source Key=’sCensusBradley’>
        <Frame>
                <CitationLnk Key=’cCensusEngWales’>
                        <Param Name=’Series’>RG09</Param>
                        <Param Name=’Piece’>2560</Param>
                        <Param Name=’Folio’>23</Param>
                        <Param Name=’Page’>6</Param>
                </CitationLnk>
                <Where DetLnk=’dwHomleysCourt’/>
                <When Value=’1861-04-07’/>
        </Frame>

        <Commentary DetKey=’dcBradySurname’>
                <Link DetLnk=’dsBradySurname’ Type=’Source’>
                        <Text> surname</Text>
                </Link>
                <Link Value=’Bradley’ Type=’Inference’>
                        <Text> surname</Text>
                </Link>
                <FromText Key=‘tBradySurname’ />
        </Commentary>

        <ProtoPerson Key=’pSamuelBradley’>
                <Link DetLnk=’dwSamuel’/>
                <FromText Key=‘tSamuelBradley’>
        </ProtoPerson>

        <ProtoPerson DetKey=‘dpSamuel’>
                <Link DetLnk=’dsSamuel’ Type=’Source’>
                        <Text>record</Text>
                </Link>

                <Link Value=’Samuel Brady’ Type=’Reading’>
                        <Text>name</Text>
                </Link>
                <Link DetLnk=’dcBradySurname’>
                        <Text>surname</Text>
                </Link>
                <Link Value=’Samuel Bradley’ Type=’Inference’>
                        <Text>name</Text>
                </Link>

                <Link DetLnk=’dwHomleysCourt’ Type=’Reading’>
                        <Text>residence</Text>
                </Link>
                <Link DetLnk=’dwBelper’ Type=’Reading’>
                        <Text>birth place</Text>
                </Link>
                <Link Value=’30’ Type=’Reading’>
                        <Text>age</Text>
                </Link>
                <Link Value=’Nail maker’ Type=’Reading’>
                        <Text>occupation</Text>
                </Link>
                <Link Value=’Head’ Type=’Reading’>
                        <Text>role</Text>
                </Link>
                <Link Value=’Married’ Type=’Reading’>
                        <Text>status</Text>
                </Link>
        </ProtoPerson>

        <ProtoPerson DetKey=‘dpMary’ Key=’pMaryBradley’>
                <Link DetLnk=’dsMary’ Type=’Source’>
                        <Text>record</Text>
                </Link>

                <Link Value=’Mary Brady’ Type=’Reading’>
                        <Text>name</Text>
                </Link>
                <Link DetLnk=’dcBradySurname’>
                        <Text>surname</Text>
                </Link>
                <Link Value=’Mary Bradley’ Type=’Inference’>
                        <Text>name</Text>
                </Link>

                <Link DetLnk=’dwHomleysCourt’ Type=’Reading’>
                        <Text>residence</Text>
                </Link>
                <Link DetLnk=’dwLonghbro’ Type=’Reading’>
                        <Text>birth place</Text>
                </Link>
                <Link Value=’24’ Type=’Reading’>
                        <Text>age</Text>
                </Link>
                <Link Value=’Cotton weaver’ Type=’Reading’>
                        <Text>occupation</Text>
                        </Link>
                        <Link Value=’Wife’ Type=’Reading’>
                                <Text>role</Text>
                        </Link>
                        <Link DetLnk=’dwSamuel’ Type=’Inference’>
                                <Text>spouse</Text>
                        </Link>
                        <Link Value=’Married’ Type=’Reading’>
                                <Text>status</Text>
                        </Link>
        </ProtoPerson>

        <ProtoPerson DetKey=’dpJohn’>
                        <Link DetLnk=’dsJohn’ Type=’Source’>
                                <Text>record</Text>
                        </Link>

                        <Link Value=’John Brady’ Type=’Reading’>
                                <Text>name</Text>
                        </Link>
                        <Link DetLnk=’dcBradySurname’>
                                <Text>surname</Text>
                        </Link>
                        <Link Value=’John Bradley’ Type=’Inference’>
                                <Text>name</Text>
                        </Link>

                        <Link DetLnk=’dwHomleysCourt’ Type=’Reading’>
                                <Text>residence</Text>
                        </Link>
                        <Link DetLnk=’dwBelper’ Type=’Reading’>
                                <Text>birth place</Text>
                        </Link>
                        <Link Value=’26’ Type=’Reading’>
                                <Text>age</Text>
                        </Link>
                        <Link Value=’Slater’ Type=’Reading’>
                                <Text>occupation</Text>
                        </Link>
                        <Link Value=’Boarder’ Type=’Reading’>
                                <Text>role</Text>
                        </Link>
                        <Link Value=’Married’ Type=’Reading’>
                                <Text>status</Text>
                        </Link>
        </ProtoPerson>

        <ProtoPerson Key=‘pSelinaBradley’>
                        <Link DetLnk=’dsSelina’ Type=’Source’>
```

```
            <Text>record</Text>
        </Link>

        <Link Value=’Selina Brady’ Type=’Reading’>
            <Text>name</Text>
        </Link>
        <Link DetLnk=’dcBradySurname’>
            <Text>surname</Text>
        </Link>
        <Link Value=’Selina Bradley’ Type=’Inference’>
            <Text>name</Text>
        </Link>

        <Link DetLnk=’dwHomleysCourt’ Type=’Reading’>
            <Text>residence</Text>
        </Link>
        <Link DetLnk=’dwBelper’ Type=’Reading’>
            <Text>birth place</Text>
        </Link>
        <Link Value=’22’ Type=’Reading’>
            <Text>age</Text>
        </Link>
        <Text>
        Age originally written as 26 and then corrected to 22.
        </Text>
        <Link Value=’ Doubler (Cotton)’ Type=’Reading’>
            <Text>occupation</Text>
        </Link>
        <Link Value=’Wife’ Type=’Reading’>
            <Text>role</Text>
        </Link>
        <Link DetLnk=’dwJohn’ Type=’Inference’>
            spouse
        </Link>
        <FromText Key=’tSelinaRole’/>
        <Link Value=’Married’ Type=’Reading’>
            <Text>status</Text>
        </Link>
    </ProtoPerson>

    <ProtoPerson Key=‘pGeorgeBradley’>
        <Link DetLnk=’dsGeorge’ Type=’Source’>
            <Text>record</Text>
        </Link>

        <Link Value=’George Brady’ Type=’Reading’>
            <Text>name</Text>
        </Link>
        <Link DetLnk=’dcBradySurname’>
```

```
            <Text>surname</Text>
        </Link>
        <Link Value=’George Bradley’ Type=’Inference’>
            <Text>name</Text>
        </Link>

        <Link DetLnk=’dwHomleysCourt’ Type=’Reading’>
            <Text>residence</Text>
        </Link>
        <Link Value=’Lancashire Heaton Norris’ Type=’Reading’>
            <Text>birth place</Text>
        </Link>
        <Link Value=’3’ Type=’Reading’>
            <Text>age</Text>
        </Link>
        <Link Value=’Son’ Type=’Reading’>
            <Text>role</Text>
        </Link>
        <Link DetLnk=’dwJohn’ Type=’Inference’>
            son of
        </Link>
        <FromText Key=’tGeorgeRole’/>
    </ProtoPerson>

    <ProtoPlace DetKey=’dwBelper’ Key='wBelper'>
        <Link DetLnk=’dsBelper’ Type=’Source’>
            <Text>name</Text>
        </Link>
    </ProtoPlace>

    <ProtoPlace DetLnk=’dwLonghbro’ Key='wLoughborough'>
        <Link DetLnk=’dsLonghbro’ Type=’Source’>
            <Text>name</Text>
        </Link>
        <Link Value=’Loughborough’ Type=’Inference’>
            <Text>name</Text>
        </Link>
        <FromText Key=‘tLonghbro’>
    </ProtoPlace>

    <ProtoPlace DetKey=’dwHomleysCourt’ Key=’w8HomleysCourt’>
        <Link DetLnk=’dsHomleysCt’ Value=’ 8 Homleys Court’
        Type=’Source’>
            <Text>name</Text>
        </Link>
    </ProtoPlace>
</Source>
```

Note that the identification of the place Loughborough and the Bradley surname have their own profiles, and these are used as inputs to other profiles. Furthermore, though, the tentative identification of Samuel as John’s half-brother takes place in another step. That would leave the option of not making that association, but of still being able to record that Mary was Samuel’s wife, even though neither had a corresponding conclusion entity.

```
<!-- Reusable pieces of text that explain inferences -->

<Narrative Key=’nForReuse’>
        <Text Key=‘tBradySurname’ Inference=’1’>
        <Title> Analysis of recorded surname </Title>
        The enumerator recorded the family surname as Brady rather than
        Bradley. However, John and Selina can easily be identified since John
        married Selena Shepherd (b. c1835 in Belper, Derbys) on
        16/12/1855 at Duffield, Derbys [1855/Q4/7b/764]. Their son,
        George, was b. 1858 in Stockport, Lancs. Registered in Heaton Norris
        sub-district. Local ref: [HEA/25/83];
        </Text>

        <Text Key=‘tSamuelBradley’>
        <Title> Who is Samuel Bradley? </Title>
        John Bradley had a half-brother called Samuel who was born in
        Belper in c1825 and was a ‘Nailer’. The birth year doesn’t quite
        match though. In 1851, this Samuel was unmarried and in Melton
        Mowbray, Leicestershire (Piece: 2091, Folio: 184, Page: 8), and this
        may further support the Loughborough interpretation for his wife’s
        birthplace.
        </Text>

        <Text Key=‘tLonghbro’ Inference=’1’>
        <Title> Where is Longhbro? </Title>
        The enumerator wrote either ‘Longhbro’ or ‘Loughbro’. Although
        some content providers have interpreted this as Longborough (a
        town in Gloucestershire) this means ignoring the ‘h’. If the ‘n’ is
        actually a ‘u’, though, then it becomes an abbreviation for
        Loughborough (a town in Leicestershire) which is closer.
        </Text>

        <Text Key=‘tSelinaRole’ Inference=’1’>
        The enumerator meant that Selina was the wife of the person above
        (John), not of the Head of the household.
        </Text>

        <Text Key=’tGeorgeRole’ Inference=’1’>
        The enumerator meant that George was the son of the couple above,
        not of the Head of the household.
        </Text>
</Narrative>
```

The application of this analysis technique to a census household is slightly different to that of narrative text. The density of the transcribed data — which would probably be in tabular form — allows this example to specify a whole record as the source input for each prototype subject. When profiling narrative text then the salient points to work with — which could even be just words or phrases — may be more isolated, and each may have a corresponding low-level profile.

I have taken the liberty of simply alluding to other sources in the identification of the Bradley surname for the sake of brevity. Ideally, that identification would be done in a Matrix entity (as in Evidence, Reasoning, and Conclusion) by assimilating the profiles for multiple sources. The STEMMA example compares the two mechanisms that could be used to represent the corrected information from a source — in this case, a census page. The choice depends on the depth of the representation and the future functionality that is expected of it, including being able to “drill down” on conclusions. These two mechanisms can be used together since a Property value can take a DetLnk attribute that connects it to a relevant profile in a more-detailed analysis.

4.13 Nature and Nurture

STEMMA emphasises that biological lineage is different to the concept of a family-unit, and that union-type Events may not be involved in either. There are still prevailing views, though, that may confuse these issues, or even maintain that they’re directly related. See Happy Families for discussion.

This illustration uses an example of a John Smith who conceives children with two women: Jane Doe and Ann Other. He subsequently marries the second partner after a short period of living with her and her child. His first child is part of an independent family unit despite there being no registered union.

```
<Person Key=’pJohnSmith’>
        <Title> John Smith </Title>
        <Sex> 1 </Sex>
        <MemberOf Key=’gJohnAnnOther’
                FromEvent=’eBirthSarahOther’/>
</Person>

<Person Key=’pJaneDoe’>
        <Title> Jane Doe </Title>
        <Sex> 0 </Sex>
        <MemberOf Key=’gJaneDoe’
                FromEvent=’eBirthTomDoe’/>
</Person>

<Person Key=’pAnnOther’>
        <Title> Ann Smith née Other </Title>
        <Sex> 0 </Sex>
```

```
    <MemberOf Key=’gJohnAnnOther’
        FromEvent=’eBirthSarahOther’/>
    <Names>
        <Sequences BeforeEvent=’eMarriageAnnOther’>
            <Canonical> Ann Other </Canonical>
        </Sequences>
        <Sequences FromEvent=’eMarriageAnnOther’
            Type=’Married’>
            <Canonical> Ann Smith </Canonical>
        </Sequences>
    </Names>
</Person>

<Person Key=’pTomDoe’>
    <Title> Thomas Doe </Title>
    <Sex> 1 </Sex>
    <MemberOf Key=’gJaneDoe’
        FromEvent=’eBirthTomDoe’/>

    <FatherPersonLnk Key=’pJohnSmith’/>
    <MotherPersonLnk Key=’pJaneDoe’/>
    <Birth><EventLnk Key=’eBirthTomDoe’/></Birth>
</Person>

<Person Key=’pSarahOther’>
    <Title> Sarah Other </Title>
    <Sex> 0 </Sex>
    <MemberOf Key=’gJohnAnnOther’
        FromEvent=’eBirthSarahOther’/>

    <Names>
        <Sequences BeforeEvent=’eMarriageAnnOther’>
            <Canonical> Sarah Other </Canonical>
        </Sequences>
        <Sequences FromEvent=’eMarriageAnnOther’
            Type=’Married’>
            <Canonical> Sarah Smith </Canonical>
        </Sequences>
    </Names>
    <FatherPersonLnk Key=’pJohnSmith’/>
    <MotherPersonLnk Key=’pAnnOther’/>
    <Birth><EventLnk Key=’eBirthSarahOther’/></Birth>
</Person>

<Event Key=’eBirthTomDoe’>
    <When Value=’1919-12-25’/>
    <Type> Birth </Type>
    <SubType> Birth </SubType>
    <PlaceLnk Key=’wSmallVille’/>
    <SourceLnk Key=’sBirthTomDoe’>
        <PersonLnk Key=’pJaneDoe’>
            <Property Name=’Role’> Mother </Property>
        </PersonLnk>
        <PersonLnk Key=’pJohnSmith’>
            <Property Name=’Role’> Father </Property>
        </PersonLnk>
        <PersonLnk Key=’pTomDoe’>
            <Property Name=’Role’> Child </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Event Key=’eBirthSarahOther’>
    <When Value=’1920-01-07’/>
    <Type> Birth </Type>
    <SubType> Birth </SubType>
    <PlaceLnk Key=’wSmallVille’/>
    <SourceLnk Key=’sBirthSarahOther’>
        <PersonLnk Key=’pAnnOther’>
            <Property Name=’Role’> Mother </Property>
        </PersonLnk>
        <PersonLnk Key=’pJohnSmith’>
            <Property Name=’Role’> Father </Property>
        </PersonLnk>
        <PersonLnk Key=’pSarahOther’>
            <Property Name=’Role’> Child </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Event Key=’eMarriageAnnOther’>
    <When Value=’1920-01-30’/>
    <Type> Union </Type>
    <SubType> Marriage </SubType>
    <PlaceLnk Key=’wStElsewhere’/>
    <SourceLnk Key=’sMarriageAnnOther’>
        <PersonLnk Key=’pAnnOther’>
            <Property Name=’Role’> Bride </Property>
        </PersonLnk>
        <PersonLnk Key=’pJohnSmith’>
            <Property Name=’Role’> Groom </Property>
        </PersonLnk>
    </SourceLnk>
</Event>

<Group Key=’gJaneDoe’>
    <Title> Jane Doe’s family </Title>
    <Type> Family </Type> <SubType> Matrilocal </subType>
```

```
</Group>

<Group Key=’gJohnAnnOther’>
        <Title> John and Ann Other’s family </Title>
        <Type> Family </Type> <SubType> Nuclear </SubType>
</Group>
```

The most important part of this illustration is that it demonstrates how STEMMA has independent mechanisms for representing biological lineage, events & roles, and family units.