From TEI to a CIDOC-CRM Conforming Model

Towards a Better Integration Between Text Collections and Other Sources of Cultural Historical Documentation

Øyvind Eide and Christian-Emil Ore, Unit for Digital Documentation, University of Oslo

Mapping from TEI to CIDOC-CRM

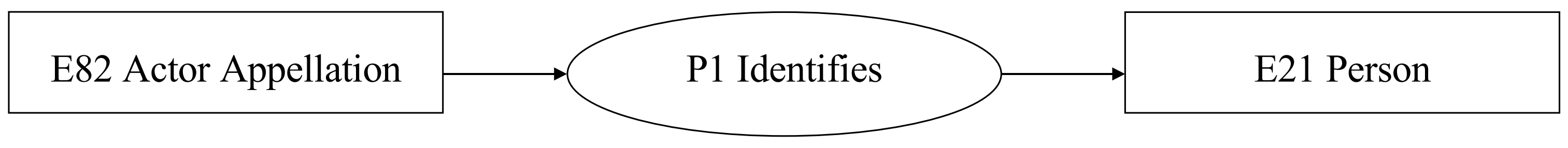
Some of the elements in a TEI encoded text can be mapped directly to entities in a CIDOC-CRM compliant model. Examples of such mappings are shown to the right. A more complete draft mapping can be found at http://www.edd.uio.no/artiklar/tekstkoding.html

TEI element	CIDOC-CRM class
Name	E41 Appellation
Person	E21 Person
persEvent	E5 Event
Origin [of a manuscript]	E65 Creation event

In many TEI documents, the elements can be used in a more specific way. Then specific CIDOC-CRM elements may be used: A TEI name element with the attribute type set to person should be mapped to an instance of the CRM class E82 Actor Appellation.

In other cases it is correct to choose another CRM class. I for example the tagUsage element in a TEI documents states that all occurrences of Person in the document is to mark organisations such as museums and libraries, then Person elements should be mapped to instances of E40 Legal Body.

A document mapping is more than just creating one CIDOC-CRM entity for each TEI element. Many facts may be deduced generally from the existence of TEI elements.



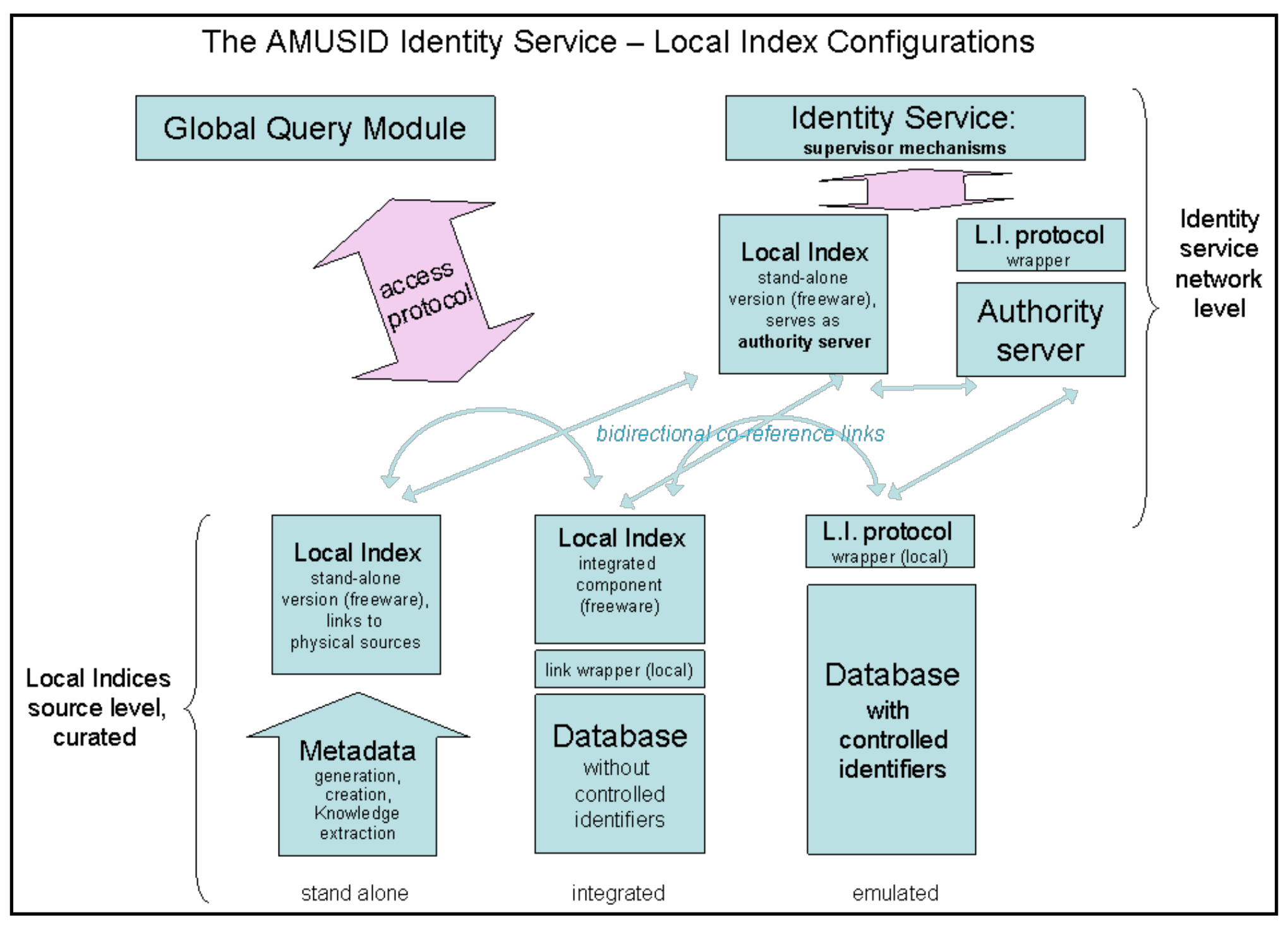
If a name of a person is used in a TEI document, we may deduce the existence of a person. This gives us the model to the left based on a TEI persName element.

TEI-CRM mappings and a semantic web application

In the proposed AMUSID project (Adaptive MUSeological IDentity-service) CIDOC-CRM is the chosen as the standard for data interchange .The illustration to the right is from the project proposal.

To be able to identify identities between references in databases and texts at a network level, we have to express the references to objects in all sources in a structured way. In our approach, this will be based on the use of CIDOC-CRM compliant models.

A TEI document collection will often be a stand alone local index, as shown in the illustration to the right. In some cases, information in TEI documents will be stored in databases. In those cases, it is important to have pointers back to the elements in the documents.


The AMUSID Identity Service – Local Index Configurations
Global Query Module
Identity Service: supervisor mechanisms
access protocol
Local Index stand-alone version (freeware), serves as authority server
L.I. protocol wrapper
Authority server
Identity service network level
bidirectional co-reference links
Local Indices source level, curated
Local Index stand-alone version (freeware), links to physical sources
Metadata generation, creation, Knowledge extraction
Local Index integrated component (freeware)
link wrapper (local)
Database without controlled identifiers
L.I. protocol wrapper (local)
Database with controlled identifiers
stand alone
integrated
emulated

The AMUSID approach have two distinct parts:

•To make the information in documents such as the TEI example available in a structured format through a mapping to the CIDOC CRM

•To assist the scholar in the creation of the structures in TEI documents.

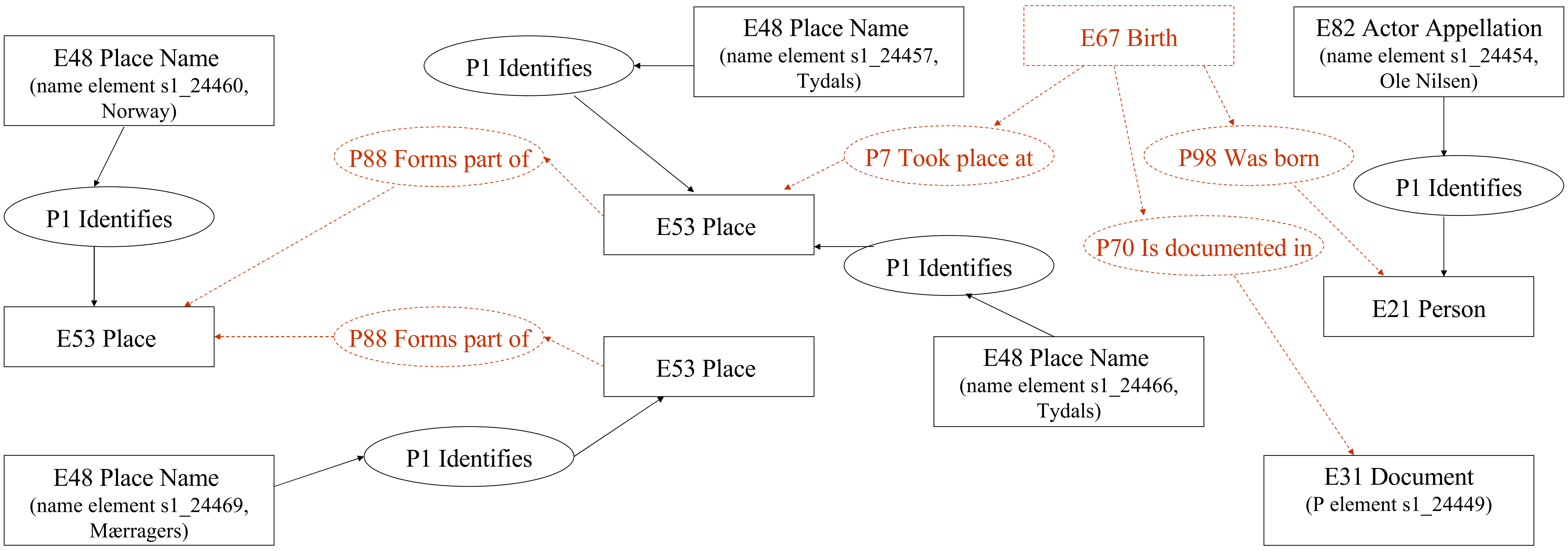
The idea is that a long term build-up of structured information available in a decentralised network will make it possible to improve the quality of the automated methods, as well as increase the speed of the manual methods.

Mapping a TEI document fragment

To demonstrate this approach, we will show a mapping of the text to the right. A mapping will never includes all the information that can be read from a text. The mapping comprises a selection of some parts in order to give a clearer picture of the chosen part of the content.

The black parts of the diagram below can be created automatically from the markup by the use of simple algorithms. The dashed red parts are based on a human's reading of the text and cannot easily be mapped correctly by automated tools.

```
<p xml:id="s1_24449">Answ: Named
<name type="person" xml:id="s1_24454">Ole Nilsen</name>, is born in
<name type="place" xml:id="s1_24457">Tydals</name> mountains, Which is in
<name type="place" xml:id="s1_24460">Norway</name>, of Sami parents, is 50 years old,
married, and having one child; has mostly dwelled in
<name type="place" xml:id="s1_24466">Tydals</name> mountains, and now dwelling in the Norwegian
<name type="place" xml:id="s1_24469">Mærragers</name> mountains. </p>
```


E48 Place Name (name element s1_24460, Norway)
P1 Identifies
E53 Place
P88 Forms part of
P1 Identifies
E48 Place Name (name element s1_24457, Tydals)
E53 Place
P88 Forms part of
E53 Place
E48 Place Name (name element s1_24469, Mærragers)
P1 Identifies
E67 Birth
P7 Took place at
P1 Identifies
E48 Place Name (name element s1_24466, Tydals)
P98 Was born
P70 Is documented in
E82 Actor Appellation (name element s1_24454, Ole Nilsen)
P1 Identifies
E21 Person
E31 Document (P element s1_24449)

Automatic, manual or semi-automatic

The problem

Our goal is a CIDOC-CRM compliant model with persons, places, dates. This model should also include the events linking the objects together. Thus, we want to include the red dotted parts of the model as well as the black solid parts.

What we have, and this is common to many text encoding projects, is TEI documents with place names, person names, and dates tagged.

We lack explicit mark-up of the following categories of information:

•Events and processes, such as birth, death, "has mostly dwelled".

•References to persons and places other than names, e.g. pronouns referring to persons and descriptions referring to places.

•Relations, e.g. "Which is in Norway" giving us a "Forms part of" property.

Possible solutions

The most straight-forward solution is to go through the text manually and add elements for the information we would like to mark-up. If some of the elements are lacking from TEI, there are well defined methods for adding them.

But this is very time consuming. Therefore, we try to find automated methods for extracting this information. Such methods exist, developed in the Natural Language Processing community for many years. But they are time-consuming to implement for a new language (no such system exist for 18th century Danish). And even an advanced implementation will not give correct answers in all cases.

Based on general rules, it is possible to infer automatically that Tydal in the two name elements are pointing to the same place.

The algorithm to deduce automatically that both Tydal and Mærragers is in Norway would not be very complicated, but it depends on place identifications at a gazetteer level, e.g. adding UTM coordinates to all places.

It is very hard to deduce automatically the relationship between the person Ole Nilsen and the places. He was born in one of them, Tydals, and lives in another, Mærragers.

Conclusion

The best solution seems to be a combination of the automatic and manual methods. Fast algorithms should create draft mappings to be refined and corrected by humans.

It is important to document who is responsible for which part of the added information, both at document and element level. This will help the user evaluating the quality of the information, and is necessary for scholarly reproducibility.

A document that has been automatically processed but is waiting for the human proof-reading, may still be used, but it is important that the user know the status of the document.

We add the extra information by adding elements to the TEI document, but it may also be done by stand-off markup in a database,

http://www.edd.uio.no/
oyvind.eide@edd.uio.no
c.e.s.ore@edd.uio.no