

Elwin Koster\*

## XML-Coding technical accuracy and historical evidence in digital historical maps

*Keywords:* history of cartography; mark-up languages; historical methods; computing.

### Summary

More and more scholars start to use maps not only as illustrations of a past situation, but as primary sources for historical research. But, in order to use them as source material we need to do research on the reliability and accuracy of the maps. A reliable map in a technical sense is not necessarily the best source to describe a historical phenomenon and sometimes our choice of cartographical material is limited. Moreover differences in reliability between maps, or between different parts of the same map, are not only due to measurement techniques but also due to their original function and context. The project *Paper and Virtual Cities* financed by the Netherlands Organisation for Scientific Research (NWO) focuses on this use of the material. One of the goals of the project *Coding the Virtual City* is to create a mark-up language to annotate historical maps. This XML-based coding is based on already existing languages like GML, Heml, and MarcXML, but also has a number of extensions specifically made for the coding of historical maps. Besides the coding-language an application is made that offers the researcher a graphical interface to annotate the cartographic documents, and store the results in the new mark-up language. The proposed paper will focus on the necessary elements to annotate historical maps, both from the technical as well as from the historical domain. Secondly the paper will present a first glimpse of the application used to do the annotation work. Finally a number of possible end products are shown, including reliability maps, and the historical map ontology.

### Introduction

The research project *Paper and Virtual cities*, financed by the Netherlands Organisation for Scientific Research (NWO), aims to develop new methodologies in order to make explicit choices in the use of historical maps in virtual (re-) constructions for research and design. These maps can be both historical documents depicting a contemporary situation of the townscape, and digital artefacts like urban reconstructions created using Computer Aided Design (CAD). In the two sub-projects on the technical and historical reliability of maps the main goal is to find data and describe specific problems that come with the maps.<sup>1</sup> These result in a large number of points with corresponding values of reliability on the one hand, and historical descriptions with a rating on the other hand. One of the goals of the project *Coding the Virtual City* is to create a mark-up language plus a toolbox to annotate historical maps. The toolbox can be used to link the technical and historical reliability data to the map, the mark-up language is to store the data in such a way that it is easy to use by the researchers and others. This paper will focus on the necessary elements needed to annotate historical maps, both from the technical as well as from the historical domain, and the toolbox to annotate a map.

More and more historians start using Geographical Information Systems (GIS) in order to store data that is related to a specific geo-location. As a basis historical maps are used, in many cases

---

\* Researcher and lecturer: Rijksuniversiteit van Groningen RuG, Faculty of Arts, Humanities Computing department, Art- and Architectural History department [[e.a.koster@rug.nl](mailto:e.a.koster@rug.nl)]

<sup>1</sup> See Heuvel (2006: 113) and Benavides J. and E.Koster (2006: 194).

not only as a background for other information, but also as a primary source. All sorts of problems may arrive from this usage of maps. Many dated maps are known that show outdated objects, for example a missing steeple on the church tower where it had been rebuilt again. This is one example where the historian, using his map, might want to make a comment on the map. The location of the church might be very accurate, but someone writing the history of the city and not checking the date of the reconstruction of the steeple might make a mistake. In the other example the church might be depicted in a correct way, but its position might be wrong. Both these examples show the difference between what we in the project use as technical reliability and historical evidence. This is similar to what Harley (1968) proposed as the evidence on maps, the evidence about maps, and the evidence of maps.<sup>2</sup> However the historian who would like to make a short annotation on this map, to inform a future user about the error in this map, does not have a tool to do so.

### **Storing annotations with mark-up languages**

We could store this kind of annotations in a GIS; however this will limit the possibilities for exchange with others. The database to hold the annotations might be very complex with the result that not many researchers will invest in learning it. We would like to create a tool that enables anyone, independent of the GIS used, to annotate maps. The final annotations should be highly flexible and very easy to exchange with others preferably by Internet. In order to achieve this goal annotations are described in the extensible mark-up language (XML). XML is a flexible and very powerful documents publishing and data interchange format. It enables to create documents that contain structured data enclosed by elements. The flexibility of the language exists mainly in these elements, since there are no predefined element names. This means that everyone can create a specific language for a particular use; in short his own XML application. Although there are many different XML applications available there is not yet a specific one for historical maps. However, we can create XML applications by combining existing ones and adding elements that are missing this feature of XML uses the so-called namespaces. In order to make use of this sort of mixed documents a document model is needed. Such a document model contains modelling rules to define the elements that can be used, elements that are optional, the order of the elements, and possible attributes to the elements. This set of rules can be stored in a sort of configuration file: the Document Type Definition (DTD) or the XML Schema. What is needed is to create the XML schema to store the technical and the historical annotations. Based on the schema different applications can be created: for data input, manipulation, storage, and output.

An XML based mark-up language that is specifically created to store geographical data is the geography mark-up language (GML). This language “is expected to lead to the development of the geo Web, the seamless fabric of geographic data and services that may form the basic infrastructure for managing impact upon the world around us”.<sup>3</sup> The language is used to describe geographical or real-world objects like they are normally stored in a GIS. In GML these objects are called features. They are categorised into types, and described in terms of its geometrical and non-geometrical properties. In this way we can describe a bastion on the historical map by a set of properties describing the location and the form, and a set of properties describing things like its name, height, and date. If we would look carefully at the code created for this example we might

---

<sup>2</sup> See Heuvel (2006: 113); Harley (1968: 62-74).

<sup>3</sup> Lake, Burggraf et.al. (2004).

see that only part of it is using the GML namespace that is to say the elements describing the geometry. In many cases a feature will be an instance with a geometry property that defines the physical extent of the feature plus a number of elements from namespaces other than the GML namespace.

```

- <app:cthXML>
  - <app:location gml:id="L1">
    - <app:mapset gml:id="Groningen">
      - <gml:featureCollection>
        - <gml:featureMember>
          <gml:description>bastion</gml:description>
        - <hemi:Event>
          - <hemi:Chronology>
            <hemi:Date>1616</hemi:Date>
          </hemi:Chronology>
          - <hemi:Evidence>
            - <hemi:PhysicalSourceSet>
              <hemi:PhysicalSource>GA 474r 10x</hemi:PhysicalSource>
            </hemi:PhysicalSourceSet>
          </hemi:Evidence>
        </hemi:Event>
      - <gml:geometryProperty>
        - <gml:Polygon>
          - <gml:outerBoundaryIs>
            - <gml:LinearRing>
              - <gml:posList>
                1787 1134 1754 1151 1735 1117 1755 1007 1869 1056 1881 1079 1849 1103 1787 1134
              </gml:posList>
            </gml:LinearRing>
          </gml:outerBoundaryIs>
        </gml:Polygon>
      </gml:geometryProperty>
    </gml:featureMember>
  </gml:featureCollection>
</app:mapset>
</app:location>
</app:cthXML>

```

Figure 1. Code example of XML-application showing a bastion.

Comparable to the geography mark-up language other languages were created to store different types of data. For example in order to store data on historical events Bruce Robertson created an XML application called HEML.<sup>4</sup> HEML is the abbreviation for the Historical Event Markup and Linking project. The main goal is to provide “a means of coordinating and navigating disparate historical materials on the internet” which is done by providing XML schema and style sheets that creates documents based on this data. HEML offers us a good way to store historical events. Such an event can contain different kind of data, for example on actors involved, the location, the description of the event, any physical resources, and the starting and ending date.

The third existing mark-up language that is used is MarcXML. This language is the XML translation of the Marc21 library standard that is used in many library catalogues all over the world.<sup>5</sup> In

<sup>4</sup> <http://heml.mta.ca/samples/blocks/heml//description>

<sup>5</sup> <http://www.loc.gov/marc/>

the Marc21 standard all fields are represented by a numeric code, and in the XML translation these codes have become attributes of the datafield element. In reading the XML application the user needs to know what all these numbers stand for. Many librarians might know, but many others might get confused by this. Another translation, the so-called Metadata Object Description Schema (MODS) makes use of semantics, and although one might say that the XML is mainly for the computer to read the application becomes easier accessible by using MODS.<sup>6</sup>

With these three namespaces, GML, HEML, and MODS, we have the main building bricks for the XML application (figure 2). However it is far from complete yet since many things related to especially historical maps are missing in all three standards. How must we store for example the units in which the scale on the map is given? In the Marc21 this will probably appear somewhere in a comments field that in the conversion to MODS might even get lost.<sup>7</sup> On the other hand we do not want to make special fields for everything, so there must be some sort of standardised list of what we would like to add to a map.

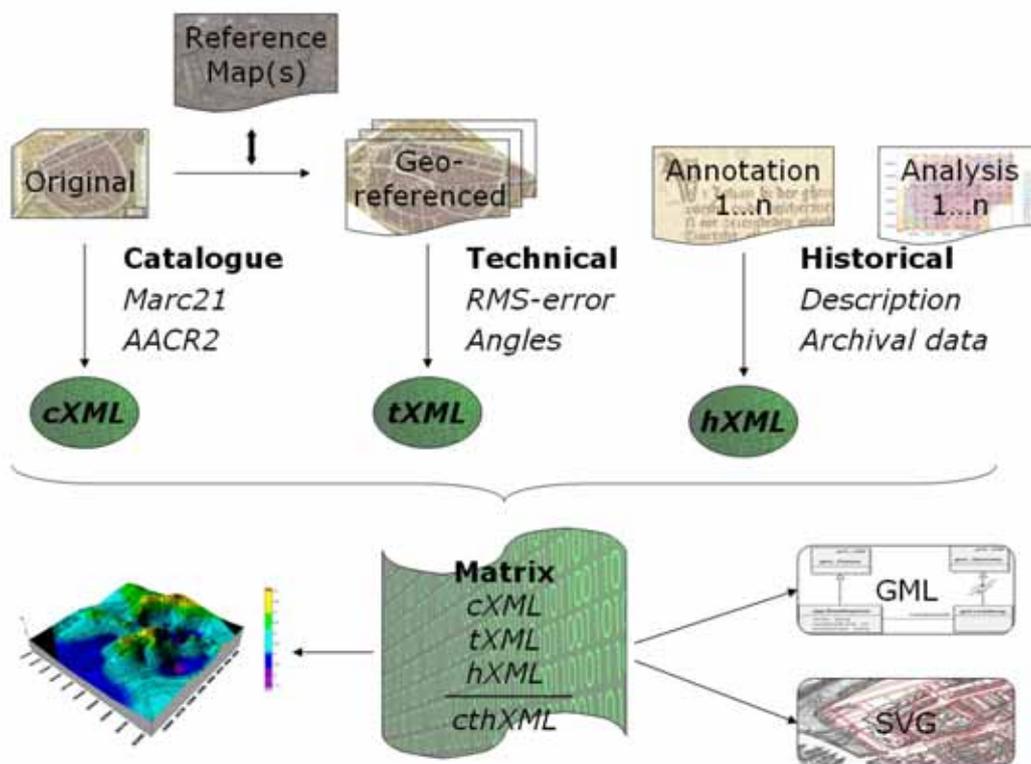


Figure 2. Schematic representation of the structure of the XML application.

Every new map that is created for example as a result of analysis like the layers that show the distribution of errors after geo-rectification should be treated as a new document. This means that for each of these documents a new XML file should be created. In this way data can be exchanged between different maps. In order to do so the map-set and the location are introduced as umbrella-like concepts. These concepts are derived from the way the open source GIS Grass is organised.<sup>8</sup> A location for example can contain all maps of one city. In this way the location will organise all

<sup>6</sup> <http://www.loc.gov/standards/mods/>

<sup>7</sup> Thanks to Geert Kloosterman (RUG, Humanities Computing) for his analysis of Marc21 and MODS.

<sup>8</sup> Neteler and Mitasova (2004: 31).

maps with the same geographical coordinates, or within certain range of coordinates. Very often the indication of location might be the name of the city or any other topographical name. The map-set is an underlying concept that might be used to group a set of maps, or to group a map and all its derivatives. Examples of map-sets are dates or the name of a mapmaker, but might be also a sub-location of a larger geographical area. With these two concepts the main structure of the XML application has been created. It is to say that we could choose to organise our data by putting every cartographical document inside a new XML file. However it is more useful to use the organising concept of maps within a map-set that is found within the location.

### **The XML application**

In the application the map is subdivided in three different groups. The first group deals with catalogue data. It will contain all sorts of fields that can be retrieved from standard cartobibliographies. The values of some of these fields are also standardised and follow the map cataloguing rules of the AACR2. In this way we can cover a majority of maps, although these are mainly the printed maps. Manuscripts maps have some peculiarities that come closer to drawing and because of that they can better be described using a different set of rules. This is where the Getty standard for the description of artworks called Categories for the Description of Works of Art (CDWA) comes into focus.<sup>9</sup> In many cases we will see that we can even compliment one standard with another one if we talk about historical maps.

The second group contains the listing of the technical reliability of a map as it is collected in the project "*measuring the historical city*". In this group the ground control points (GCP) are listed, but also the root mean square error that belongs to such a point. The coordinates of these points are written down as being GML geometries, that is to say like an exact position of the feature. One of the advantages is that in this way all the data used for a certain geo rectification will be stored with the image. In this way it will be possible to reproduce the process, something that is often forgotten by many GIS users. Even if different sets of geo reference points are tested we can store all of them, although one should realise that it will be necessary to indicate clearly why a certain set of points is preferred over another one. Every set of points should be documented. A second advantage of this way of working and of storing the data in GML geometries is that we can apply the same transformation to other files. By making use of the stylesheet language XSLT we can transform the dataset into any other format that can be used by the GIS.<sup>10</sup>

The third group will contain the historical context as well as historical annotations. In this group we can find the description of the map placed in its historical context for example that the map was made for specific purpose or in relation to a group of other maps. More specific examples of this will be shown later on. A second sub-group that we can find deals with what we can see on the map, for example a sketch of a citadel that was never executed. A third sub-group regards the context of the map such as other documents that explain what the map is about. In this tripartite structure we recognise the abovementioned structure of Harley. It is this third group that is the most complex mainly since it is never finished. Preferably this should be some sort of open structure. Since we have chosen to organise the data in such a way that it can be easily exchanged and processed by computers we need some levels of organisation. The article of Wallace and Van den

---

<sup>9</sup> [http://www.getty.edu/research/conducting\\_research/standards/cdwa/](http://www.getty.edu/research/conducting_research/standards/cdwa/)

<sup>10</sup> Some GIS can even import GML, but at the moment this is mainly to create vector files.

Heuvel offers a number of organizational structures that have been used in the past to organise historical evidence.<sup>11</sup>

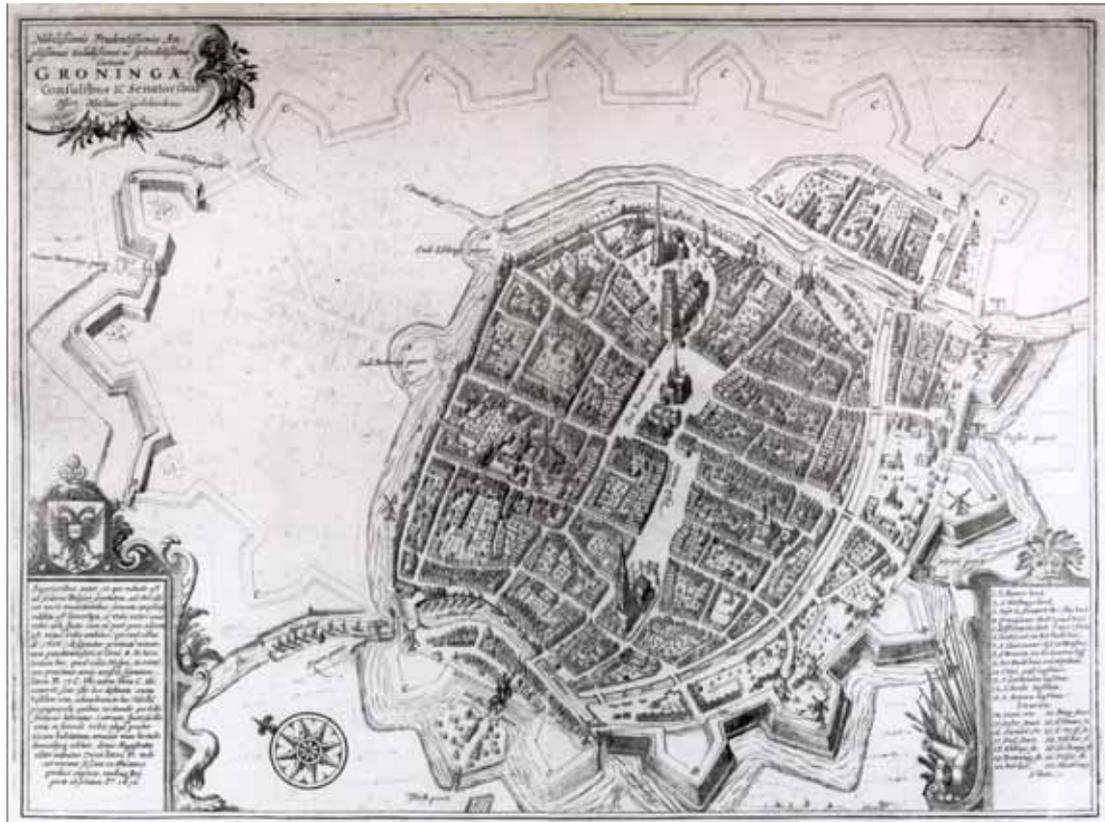


Figure 3. Nicolaas Geelkercken, Groningae, 1616. Groninger Archieven 2001.

#### Geelkerckens' map of Groningen (1616).

In the next paragraphs we will take the map of Nicolaas Geelkercken of the city of Groningen from 1616 as an example (figure 3).<sup>12</sup> We can retrieve a major part of the catalogue data from the different library catalogues where they have the map. In order not to type all the information again we can use a communication protocol like the Z39.50 protocol.<sup>13</sup> This protocol is able to connect to online catalogues and retrieve the data in such format that can be easily converted to Marc21 or to MODS. All standard fields for the carto bibliography are available although some of the elements depicted on the map could not be listed in the format. An example of this might be that the field 507, the scale note for graphic material, is missing in the MODS definition this will result in the mapping of this field to a comments field in which case any reference to the scale note will be gone. In the field 691, a field that is actually missing from the concise format, we find that the top of the map is oriented towards the North East.<sup>14</sup> But firstly there is no indication that a compass rose is depicted on the map and secondly that we know the orientation of the map not only because we know the city but mainly because of this compass rose. In the format there is an indica-

<sup>11</sup> Wallace and Heuvel (2005: 173-181).

<sup>12</sup> Geelkercken, N. van (1616). *Groningae*. University Library Groningen, the Netherlands, uklu 01-04-29.

<sup>13</sup> <http://www.loc.gov/z3950/agency/>

<sup>14</sup> <http://www.loc.gov/marc/bibliographic/ecbdhome.html>

tion of the legend listing the names of important objects like for example the city gates, the churches, and some other institutes. This refers to the inset on the lower right side of the map. However on the lower left we find another inset that is not listed anywhere in the Marc21. This text box describes why the map was created and what some of the markers on the map that are not listed in the legend stand for. The map was created as part of a book written by the local historian and scientist Ubbo Emmius, the first rector of the Groningen University.<sup>15</sup> The book on the history of the northern part of the Netherlands describes the changes that took place in the first decade of the 17th century. Since the city of Groningen was extended at that time the process has been depicted in the accompanying map. The main question is whether we should list the content of this text box in the catalogue data or that it should go into the historical section or the third option to have the data in both. In our view the second option of putting this kind of data in the historical section is best. Although there are a number of elements that are not in the catalogue data we can say that this map has been catalogued sufficiently.

When geo-referencing the map of Nicolaas Geelkercken we need to set a number of ground control points. Almost all GIS software and specific software for geo-referencing contain possibilities to store the XY coordinates of these points. In many cases this will be in an ASCII text file. When we reference the Geelkercken map by using the already geo-referenced cadastral plan of 1832 we need to set around 20 control points. On this map the city gates, a number of street corners, and some bridges are used, the distribution of the points is chosen in such a way that it is balanced.<sup>16</sup> If necessary we can indicate why a certain point was used or not. This again is directly related with the interpretation of the person, who is making references to the points, this can be important when others are making a similar geo-rectification. In the same way we can also indicate which points were not used in the process. The final result will be that we keep a list of the different interpretations plus the choices made in the geo-referencing process. In the research measuring the historical city that deals with these processes another type of control points are used: the so-called common points. These common points are not used to geo-reference the map but they are used as a reference in checking the distortion of the map after the transformation. These common points should also be stored in the XML application they do belong to a specific map but also they are directly related to a certain set of ground control points.

The historical context for this map should start with information on the mapmaker. From the repertory of Dutch map makers we can get the necessary historical data on Nicolaas Geelkercken.<sup>17</sup> We can use this data to fill in a number of elements in the HEML structure on the description of the person. We should set the person's name, the dates of his birth and his death, and other important dates from his career together with eventually references to external sources. We should fill in similar data for all other persons involved with this map in this case this might be on Ubbo Emmius since the map was a part of his book.<sup>18</sup> In other cases where for example the mapmaker, the engraver, and the publisher are different persons for each of them a new HEML block must be created. One of the advantages of HEML is that a unique identifier can be created for each person. After setting the data once we can refer to such a block by just giving the identifier. The only problem that can arise is that after an exchange with other researchers two persons are ending up

---

<sup>15</sup> Emmius (1616). Rinzema (1994: 49-62).

<sup>16</sup> Although in theory balancing should not be necessary my experience is that makes a difference the moment choices must be made by software.

<sup>17</sup> Donkersloot-de Vrij (2003).

<sup>18</sup> Emmius (1616).

with the same identifier. This situation is very similar to what we can find in many database management systems and the solution is the same. Another problem might be the different spelling varieties of the same name, or similar situation that appears very often when the name is suddenly given in Latin. But also for these occasions we can simply refer to the solutions as they can be found in databases for example by applying a thesaurus or authority files. HEML offers a very suitable XML application for this kind of situations.

The second block we should create is on the function of the map. In this case it is an illustration to a text on the history of the region. Unfortunately HEML does not offer the right structure to describe the function of a primary source. What it does offer is a so-called physical resources set that can contain one or more physical resources, but an important part of such physical resource has already been described in the catalogue data. The data we would like that add at this point is more on the context in which the map was produced, and the function of the map is part of that. In the case of the Geelkercken map there must be a reference here to the book of Ubbo Emmius. The second thing is the transcription and translation of the text in the box in the lower left corner of the map. So of the text here will need an interpretation by the researcher. Such an interpretation might occur at different levels: the explanation of a word, abbreviation, or phrase but also a comment referring to external data.

The third block is on things we can see in the map. From different archival deeds we know that one of the new city gates was finished in the second half of 1617 and that in 1618 pavours started with the street running to the gate.<sup>19</sup> On the map of 1616 we find the gate building already finished. This might be correct since we know that in the summer of 1615 a local bricklayer is working on the gate building. In one of the building specifications we find that the two bastions on either side of the new gate were completed. Also this is depicted very accurate in the map and one of the conclusions might be that the map is very accountable. From the examples we can see there are different sources and different types of data that describe the map. What is needed is a tool that enables us to create new entries, a tool that might be more specific for the maps we are dealing with in the project, but on the other hand a tool that can be easily adapted to new situations. However, let us make it clear that we do not aim to create a new library cataloguing tool. With a tool we should be able to add new data to the XML application, and to edit existing data. For example we can import data from library catalogues using the Z39.50 protocol as described before, and in the same way we can export the data to Marc21 or MODS to be used and imported by library catalogues. In the export functions all data that cannot be mapped directly to a field will be stored in a comments field. The main export functions however will be the own catalogue XML format developed in this project.

### **The software**

The software, DrawOverImage, will have several modules and all of them offer different possibilities for working on the main XML application groups as mentioned above. The main module of the application is a graphical interface to the map. In this module digital images of the map from different graphical image formats can be read in. For example we can read in a group of geo-referenced images from the same location. On top of this map we can start making annotations in a new layer. Such an annotation is everything that is geo-referenced or can be traced on the map. By using standard geometrical elements, the point, the line, and the polygon, areas can be created

---

<sup>19</sup> Koster (2001: 254 and 264).

on the map. Such an area will be the basis of any annotation. With the point we can indicate the place of the city gate and subsequently we can add the historical events we know from the building specification deeds. With the line, or the polygon, we can do the same thing for the bastions and any other object on the map.

A number of features can be added to the map by using one of special functions that are created to indicate the scale bar, compass rose, legends, and boundaries of the image and the paper (figure 4). When the user is adding a scale bar the software will ask a number of questions that have to do with the units and the measure length of the scale bar on the map. With this information the software can calculate distances on the map automatically. In order to do so a number of historical measurement units and its modern equivalents have been added to the system using a basic set of units from measure dictionaries.<sup>20</sup> Individual users can extend this list with their own units. If there is more than one scale bar on the map every object should be defined separately. The first one that is mentioned will be the one that is used for the calculations.

The software does contain an import module that makes it possible to read both in XY coordinates of points that have been used to register and rectify a map and points that were used to indicate common points like it has been described before. These points will be shown on the map and by clicking on them the user will be able to add extra annotations to them like the already mentioned considerations to include or to exclude a certain point. Also technical data like the RMS-error or the value of a certain point can be added in this way. In some cases it will be necessary to write a new import filter. This can be done using a separate XML file containing the rules of the filter.

Another import filter makes it possible to read in external XML data from the same software. In this way other users can use the data set of someone else. This is especially useful when team members want to exchange data on the same map or on the same set of maps. The data files that need to be exchanged are a very small especially when compared to sending raster layers from graphical software. If all team members would have the same geo-referenced maps on their hard disks they can simply overlay them with the XML data. Another use of this import is to compare at different map with different XML data. We can for example read the control points and common points of the map of Geelkercken and see how they fit on a different map of the same location.

All the results of the layer can be stored in different digital formats. Internally this format is stored in memory by using the Document Object Model (DOM) and from this a number of different formats can be written. The main file format is the XML application that has been described in this paper. Some of the other file formats can be created directly from the DOM an example of this are the so-called Scalable Vector Graphics (SVG). SVG is the new graphics file format that can be used to store vector information.<sup>21</sup> SVG can be used to quickly present the map in a browser window and in this way it can be used as an exchange format for a large group of users. In combination with a scripting language the maps can even be made interactive so that users can even connect to the data underneath the map. In this way data can be exchanged with people that do not have or do not want to install the software. Although it is possible to also exchange the raster images these can be very large and might slow down the process too much.

---

<sup>20</sup> These dictionaries are standard part of GML 3.0. They contain GML elements indicating the description and SI-value of a certain unit plus how to convert the unit to the so called preferred unit.

<sup>21</sup> In general: <http://www.w3.org/Graphics/SVG/> and specifically on mapping and SVG: <http://www.carto.net/>

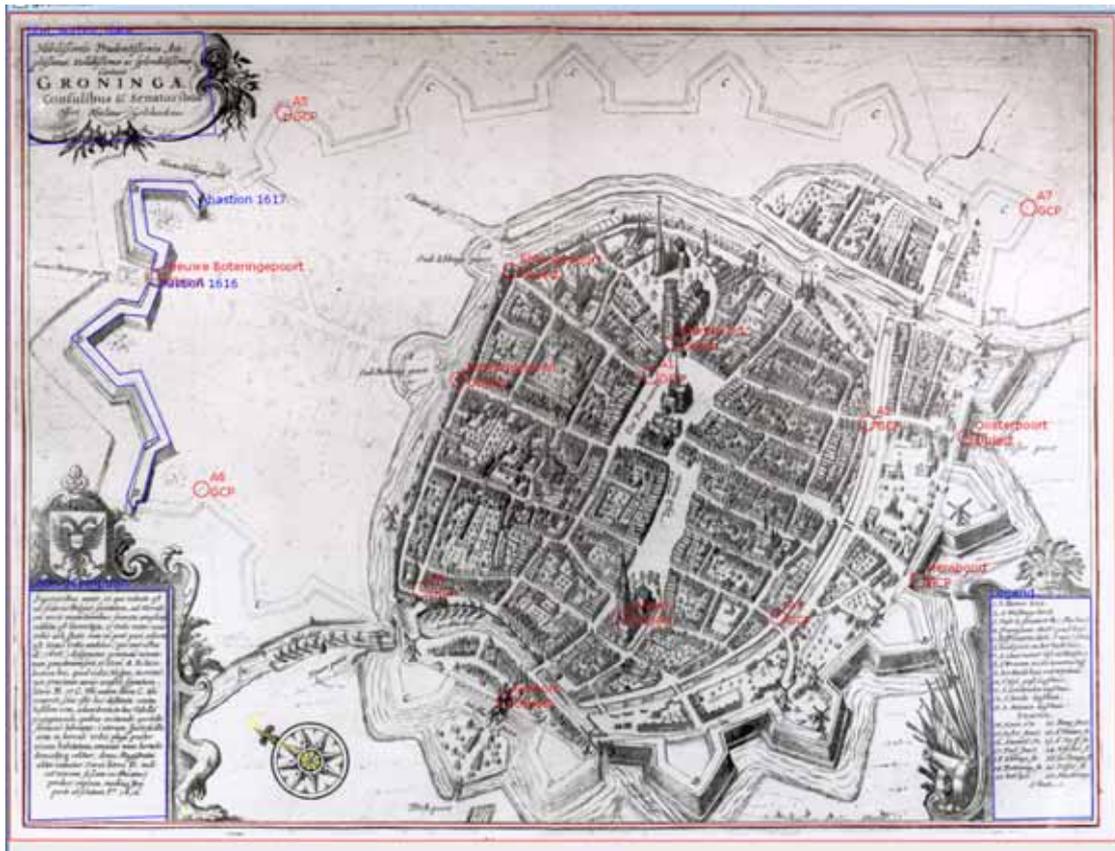


Figure 4. Example of the graphical editor DrawOverImage.

One of the other built-in export formats is GML. In this way it will be possible to import the data into several GIS programmes. The newest version of GML has been built using the object-property model. This means that the language is organised by separate objects that are stored as elements and that can have properties that contain the values.<sup>22</sup> Since the internal XML language of the software for a great part is organised in the same way, many objects already exist. GML can also be used in order to create geo-spatial Web services for example by using the OGC Web feature service (WFS). The Web server is a small piece of software that can receive messages from a browser and will respond by sending a response using a particular protocol. In this way the outputs of the programme can be presented on the web. The third type of application that is able to use the GML created by the software are relational databases. In this way the data can be retrieved by a number of other applications.

Except for the graphical module of the software there are two other components that are used to insert other types of elements. One piece of the software is able to create the historical annotations. New events or any other structure from HEML can be created by means of special forms. The other module does the same for the catalogue data. The forms must be flexible but on the other hand clear in the choices the user can make. In some of the fields the user input will be limited to a set of given choices. In the Marc21 field 006 we need to indicate the type of material and a given set of possible keys are given. In the research we consider architectural drawings of for example parts of the fortification is one of the key values. From the Marc 21 and the AACR2 this will not be a choice so it might be that this value is missing in the first layout of the form. The

<sup>22</sup> Lake (2004: 81).

software of this module is written in such a way that the forms are created based on a set of rules that are given in again an XML file. At start-up the software will read this XML file and create the accompanying form. In the choice of the format for the set of rules a language has been chosen that is based on the HTML rules for creating forms. This software can run independently from the graphical module so in the future it can have different other uses. The export from this program will be an XML application, the structure and the name of elements and attributes are given in the same XML rules file that contains the fields for the forms.

The final result of all this is a large XML data file that describes a map or a set of maps plus different types of annotations that belong to it. This data files are stored in such a way that the data from it can be retrieved smoothly and by a wide range of other future applications. To keep all possibilities open but on the other hand to have a good performance of the system an XML native database system is used. This open source database eXist has the capability of storing, indexing, and retrieving XML applications.<sup>23</sup> It uses a special language called XQuery to query files in the database but also full text searches can be done.

Based on the XML application other applications can be built in the future. One application that will be created in this project will be “a digital matrix that visualizes the reliability and veracity of historical town plans and virtual (re-) constructions (separately or in combination) in relation to function and context”.<sup>24</sup> This matrix is still under development. The matrix which is actually a piece of visualisation software can show in different ways the areas on the map with high and low technical reliability or historical evidence. On one hand this is done by offering the user pop-up windows that refer to the data from the annotations, on the other hand the values can be made visible by using isolines or contour maps.

### Conclusion

In this paper a sketch was drawn of a set of applications using the XML format as a structuring language for different types of data. The choice for a XML application offers the flexibility needed to describe different aspects of historical cartography. On the other hand the variety that can be found in coding the virtual city makes it necessary to develop a number of associated applications like graphical and textual editors. Historical maps are not more difficult to describe as their modern congeners. However we should keep in mind that describing the function and context in a way that is historically correct, makes it difficult to store the data in an unambiguous way.

### References

- Benavides J. and E. Koster (2006). Identifying surviving landmarks on historical maps, *e-Perimtron*, Vol. 1, No.3, 194-208 [<http://www.e-Perimtron.org>]
- Donkersloot-de Vrij, M. (2003). *Repertorium van Nederlandse Kaartmakers 1500-1900*. Utrecht. Available as pdf-file on: <http://www.maphist.nl> [12 April 2006].
- Emmius, U. (1616). *Rerum Frisicarum historiae decas sexta*. Leiden: Lvdovicvm Elzevirivm.

---

<sup>23</sup> <http://exist.sourceforge.net/>

<sup>24</sup> see the general text on the objectives of the project on projects website: <http://www.let.rug.nl/pvc>

- Harley, J. B. (1968). The evaluation of early maps: towards a methodology. *Imago Mundi* 22: 62-74.
- Heuvel C. van den (2006). Modelling historical evidence in digital maps. A preliminary sketch. *E-Perimtron*, Vol. 1, No. 2, 113-126 [<http://www.e-perimtron.org>]
- Koster, E.A. (2001). *Stadsmorfologie*, dissertation Groningen, Groningen.
- Lake, R., Burggraf D.S. et.al. (2004). *Geography Markup Language (GML)*. Chichester: Wiley .
- Neteler, M. and Mitasova, H. (2004). *Opensource GIS: a Grass GIS approach*. New York: Kluwer / Springer, p.31.
- Rinzema, A.J. (1994). Ubbo Emmius als historicus. In Koppers, W.J. *Ubbo Emmius. Een Oostfries geleerde in Groningen*. Groningen / Emden: Regio project, 49-62.
- Wallace, T.R. and Heuvel, C. van den (2005). Truth and Accountability in Geographic and Historical Visualization. *The Cartographic Journal* Vol42, No 2: 173-181.