A Framework to Annotate the Uncertainty for Geospatial Data

Zhao Yang
zyang1@uno.edu

Follow this and additional works at: https://scholarworks.uno.edu/td

Part of the Software Engineering Commons

Recommended Citation
Yang, Zhao, "A Framework to Annotate the Uncertainty for Geospatial Data" (2012). University of New Orleans Theses and Dissertations. 1528.
https://scholarworks.uno.edu<td>/1528

This Thesis is brought to you for free and open access by the Dissertations and Theses at ScholarWorks@UNO. It has been accepted for inclusion in University of New Orleans Theses and Dissertations by an authorized administrator of ScholarWorks@UNO. The author is solely responsible for ensuring compliance with copyright. For more information, please contact scholarworks@uno.edu.
A Framework to Annotate the Uncertainty for Geospatial Data

A Thesis

Submitted to the Graduate Faculty of the
University of New Orleans
in partial fulfillment of the
requirements for the degree of

Master of Science
in
the Department of Computer Science

by

Zhao Yang

Northwestern Polytechnical University, China, 2001

August 2012
ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to my major professor Dr. Mahdi Abdelguefi for his belief in me and the support extended by him throughout my work. It has been a wonderful experience to work under his guidance. Dr. Elias Ioup also gave me help on my research.

I would like to thank Dr. Shengru Tu and Dr. Golden Richard III for being a part of my thesis committee.

Lastly, I would like to thank my friends and family for their love and support throughout.
# Table of Contents

ABSTRACT ............................................................................................................................................ v  
CHAPTER 1 INTRODUCTION ............................................................................................................ 1  
CHAPTER 2 BACKGROUND .............................................................................................................. 4  
2.1 Geographic information system ........................................................................................................ 5  
   2.1.1 Geography Markup Language ................................................................................ 5  
   2.1.2 GML application schema ............................................................................... 7  
   2.1.3 GML Grid schema .......................................................................................... 8  
   2.1.4 Domain set definition .................................................................................. 11  
2.2 Overview of Uncertainty schema .................................................................................................... 13  
2.3 OGC standards and OGC Web Services ......................................................................................... 15  
2.4 GIS uncertainties ............................................................................................................................. 16  
2.5 Previous work ................................................................................................................................. 18  
CHAPTER 3 THE FRAMEWORK ...................................................................................................... 20  
3.1 Framework Overview ..................................................................................................................... 20  
3.2 Framework Architecture ................................................................................................................. 20  
3.3 Framework Components ................................................................................................................. 22  
   3.3.1 GML application schema ...................................................................... 23  
   3.3.2 GML ......................................................................................................... 24  
   3.3.3 Grids schema ............................................................................................ 25  
   3.3.4 Uncertainty Schema .............................................................................. 26  
   3.3.5 Geoserver ............................................................................................... 29  
   3.3.6 Gaia ........................................................................................................ 30  
CHAPTER 4 FRAMEWORK IMPLEMENTATION ........................................................................... 32  
4.1 Uncertainty data loading process .................................................................................................... 32  
4.2 Define the coverage schema ............................................................................................................ 33  
4.3 Define the grid ................................................................................................................................. 36  
4.4 Specify the distribution information ............................................................................................... 38  
4.5 User defined distribution ................................................................................................................. 41  
4.6 Making data with user-defined distribution .................................................................................... 43  
4.7 Visualize data in Gaia ...................................................................................................................... 46  
CHAPTER 5 APPLICATIONS OF THE FRAMEWORK ................................................................... 50  
5.1 Depth sounding error measurement ................................................................................................ 50  
5.2 Advanced route planning ............................................................................................................... 55  
5.3 Circular error for shore line ............................................................................................................ 65  
CHAPTER 6 CONCLUSION AND FUTURE WORK ........................................................................ 75  
Reference .............................................................................................................................................. 76  
VITA ..................................................................................................................................................... 77
Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Sample Grid definition[3]</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>RectifiedGrid definition format[3]</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>RectifiedGrid Geometry[3]</td>
<td>10</td>
</tr>
<tr>
<td>2.4</td>
<td>Sample RectifiedGrid definition[3]</td>
<td>11</td>
</tr>
<tr>
<td>2.5</td>
<td>Sample AbstractCoverageType definition[3]</td>
<td>12</td>
</tr>
<tr>
<td>2.6</td>
<td>Sample DomainSetType definition[3]</td>
<td>13</td>
</tr>
<tr>
<td>3.1</td>
<td>Framework Architecture</td>
<td>21</td>
</tr>
<tr>
<td>3.2</td>
<td>Framework Components Diagram</td>
<td>22</td>
</tr>
<tr>
<td>3.3</td>
<td>Import GML application schema[3]</td>
<td>23</td>
</tr>
<tr>
<td>3.4</td>
<td>GML data file format</td>
<td>25</td>
</tr>
<tr>
<td>3.5</td>
<td>XML schema of Confidence Interval [4]</td>
<td>27</td>
</tr>
<tr>
<td>3.6</td>
<td>XML example of Confidence Interval [4]</td>
<td>28</td>
</tr>
<tr>
<td>3.8</td>
<td>XML example of Binomial Distribution [4]</td>
<td>29</td>
</tr>
<tr>
<td>4.1</td>
<td>GML data file format</td>
<td>33</td>
</tr>
<tr>
<td>4.2</td>
<td>GML AbstractCoverageType definition[3]</td>
<td>34</td>
</tr>
<tr>
<td>4.3</td>
<td>GML DiscreteCoverageType definition[3]</td>
<td>35</td>
</tr>
<tr>
<td>4.4</td>
<td>UncertaintyCoverage element definition in application schema</td>
<td>35</td>
</tr>
<tr>
<td>4.5</td>
<td>Schema information in the GML data file</td>
<td>36</td>
</tr>
<tr>
<td>4.6</td>
<td>Example instance of a gml:RectifiedGrid[3]</td>
<td>37</td>
</tr>
<tr>
<td>4.7</td>
<td>Example instance of a gml:RectifiedGrid</td>
<td>38</td>
</tr>
<tr>
<td>4.8</td>
<td>Distribution type definition in Uncertainty schema</td>
<td>40</td>
</tr>
<tr>
<td>4.9</td>
<td>Uncertainty data in GML using uncertainty schema</td>
<td>41</td>
</tr>
<tr>
<td>4.11</td>
<td>Application schema definition for CEP</td>
<td>42</td>
</tr>
<tr>
<td>4.12</td>
<td>CEP usage sample</td>
<td>43</td>
</tr>
<tr>
<td>4.13</td>
<td>Sample application schema for Louisiana highway case</td>
<td>45</td>
</tr>
<tr>
<td>4.14</td>
<td>Sample GML for Louisiana highway case</td>
<td>46</td>
</tr>
<tr>
<td>4.15</td>
<td>Load GML data file to Gaia</td>
<td>47</td>
</tr>
<tr>
<td>4.16</td>
<td>Grids information showed in Gaia</td>
<td>48</td>
</tr>
<tr>
<td>4.17</td>
<td>Uncertainty information showed in Gaia</td>
<td>49</td>
</tr>
<tr>
<td>5.1</td>
<td>Schema for depth sounding with error notations</td>
<td>51</td>
</tr>
<tr>
<td>5.2</td>
<td>Hierarchical nature of XML</td>
<td>52</td>
</tr>
<tr>
<td>5.3</td>
<td>Sample application schema for depth sounding case</td>
<td>53</td>
</tr>
<tr>
<td>5.4</td>
<td>Sample GML for depth sounding case</td>
<td>55</td>
</tr>
</tbody>
</table>
Figure 5. 5 The advanced route planning algorithm ................................................................. 56
Figure 5. 6 Data source and error for route planning ............................................................... 57
Figure 5. 7 Different data source with different standard deviation ........................................... 58
Figure 5. 8 Sample application schema for Louisiana roads case ........................................... 61
Figure 5. 9 Sample GML for Louisiana highway case .............................................................. 63
Figure 5. 10 Sample GML for tl_roads case ............................................................................ 65
Figure 5. 11 Map with 1:250,000 scale level ........................................................................... 67
Figure 5. 12 Map with 1:12,000,000 scale level ...................................................................... 67
Figure 5. 13 Sample application schema for Shore Line case .................................................. 69
Figure 5. 14 Sample GML for 1:250,000 map scale level ......................................................... 71
Figure 5. 15 Sample GML for 1:12,000,000 map scale level ..................................................... 72
Figure 5. 16 Map with circular error of 500m ......................................................................... 73
Figure 5. 17 Map with circular error of 24km ........................................................................ 74
ABSTRACT

We have developed a new approach to annotate the uncertainty information of geospatial data. This framework is composed of a geospatial platform and the data with uncertainty. The framework supports geospatial sources such as Geography Markup Language (GML) with uncertainty information. The purpose of this framework is to integrate the uncertainty information of data from the application users and thereby ease the development of processing uncertainty information of geospatial data. Having well organized data and using this framework, the end-users can store the uncertainty information on the current geospatial data structure. For example, a GIS user can share the error information for environmental and geospatial data to others. We also report on the enhanced geographic information system functionality.

Keywords

Geospatial Information System,
Geospatial Markup Language,
Uncertainty,
Framework,
Grid
CHAPTER 1 INTRODUCTION

The Geography Markup Language (GML) is the XML grammar defined by the Open Geospatial Consortium (OGC) to express geographical features. GML serves as a modeling language for geographic systems as well as an open interchange format for geographic transactions on the Internet [1]. However there is no general language format for sharing error information of geospatial data. The ability to provide a general language for sharing uncertainty information for environmental and geospatial data is a requirement for many Geospatial Information System (GIS) users [2].

Current techniques for transmitting uncertainty information are overly specialized or ad hoc. The techniques are not supported by data sharing standards. They are incapable of reflecting the origins of error in data sets. Data sharing and automation are also features required by GIS users. GIS users are expected to use unfamiliar datasets. The implicit error information is insufficient. The automated sharing is limited without error information. In the field of ocean and acoustic modeling, there is still limited use of scientific data without error measures. The new models are incorporating statistical distributions as inputs and as outputs.

The goal of the framework is to create a language for marking data with error information. The language is an XML schema that formally defines the vocabulary and syntax for indicating error estimates. The following factors need be considered. First the error distributions come in many forms. The language must be able to represent all forms of
error distributions, for instance, continuous distribution or discrete distribution, systematic distribution or random distribution, multi-variable distribution or non-independent distribution. Second, the language must support user-defined error analysis. The automatic error calculations for complex processes are not possible. Lastly, the error information can be cumulative. The language must support complete traceability of error terms from end-to-end.

In this project, we have developed a framework that helps users process uncertainty information of Geospatial data. This framework is composed of GML application schema [3], Uncertainty schema, GML Grids schema, Geoserver and Gaia software that allow the uncertainty information to be visualized by software. With a well-defined GML application schema and this framework, the uncertainty information is easy to access by Web services. The whole framework provides a general solution for marking, propagation and transmission of error information for Environmental and Geospatial data.

Compared to UncertML [4], our framework supports GML instead of XML. It is convenient to transform current GML data to the format of GML with error information. The framework is being constructed on an open source software platform. However, the framework does not require comprehensive programming skills; the uncertainty information is easy to access and visualized.

This thesis is organized as follows: Chapter 2 deals with the background knowledge of the project. The concepts and definitions of Geography Information System, Geospatial Markup Language and Uncertainty schema are presented. Chapter 3 presents the
framework introduction; the architecture and advantages of the framework have been discussed. The design of various components of the framework has been presented in detail. Chapter 4 describes the implementation details of the framework. The implementation of all the framework components will be discussed in detail. Chapter 5 presents examples on how to use the framework to build uncertainty information based GML schema. Three such applications are presented. Chapter 6 discusses the conclusions and future directions for our work.
CHAPTER 2 BACKGROUND

The Geography Markup Language (GML) is the XML grammar defined by the Open Geospatial Consortium (OGC) to express geographical features. In many circumstances, user have a requirement to provide a general language for sharing error information for environmental and geospatial data. Previous research automates the use of existing standards for data sharing.

- GML: Geospatial Markup Language
- KML: Keyhole Markup Language (Google Earth)
- WFS: Web Feature Service
- WCS: Web Coverage Service
- WPS: Web Processing Service

Data providers are mandated to use these standards. Standards all support sophisticated geospatial and environmental constructs by default such as datums, projections, topology, grids, etc. However none of these standards support uncertainty information. To accomplish this we need to build a framework into the GML standard. This can be done using the framework described by this paper.
2.1 Geographic information system

A geographic information system is a system designed to capture, store, manipulate, analyze, manage, and present all types of geographical data. The acronym GIS is sometimes used for geographical information science or geospatial information studies to refer to the academic discipline or career of working with geographic information systems. In the simplest terms, GIS is the merging of cartography, statistical analysis, and database technology.

A GIS can be thought of as a system—it digitally creates and "manipulates" spatial areas that may be jurisdictional, purpose, or application-oriented. Generally, a GIS is custom-designed for an organization. What goes beyond a GIS is a spatial data infrastructure (SDI), a concept that has no such restrictive boundaries. In a general sense, the term describes any information system that integrates stores, edits, analyzes, shares, and displays geographic information for informing decision making. GIS applications are tools that allow users to create interactive queries (user-created searches), analyze spatial information, edit data in maps, and present the results of all these operations [2].

2.1.1 Geography Markup Language

Geography Markup Language is an XML grammar written in XML Schema for use in the description of application schemas as well as the transport and storage of geographic information. The key concepts used by Geography Markup Language (GML) to model
the world are drawn from the ISO 19100 series of International Standards and the

Geographic features with geometry are those with properties that may be
gometry-valued. A feature collection is a collection of features that may itself be
regarded as a feature; as a consequence a feature collection has a feature type and thus
may have distinct properties of its own, in addition to the features it contains.

Predefined types of geographic feature in GML include coverage and simple observations.
We will use these types in our framework to define uncertainty information.

Coverage is a subtype of feature that has a coverage function with a spatiotemporal
domain and a value set range of homogeneous 1- to n-dimensional tuples. Coverage may
represent one feature or a collection of features — to model and make visible spatial
relationships between, and the spatial distribution of, Earth phenomena and a
coverage —acts as a function to return values from its range for any direct position within
its spatiotemporal domain (ISO 19123). An observation is considered to be a GML
feature with a time at which the observation took place, and with a value for the
observation.

A reference system provides a scale of measurement for assigning values to a position,
time or other descriptive quantity or quality.

A coordinate reference system consists of a set of coordinate system axes that is related to
the Earth through a datum that defines the size and shape of the Earth.

A temporal reference system provides standard units for measuring time and describing
temporal length or duration.

A reference system dictionary provides definitions of reference systems used in spatial or temporal geometries [3].

2.1.2 GML application schema

GML application schema has been used to define the data structure of our framework. Designers of GML application schemas may extend or restrict the types defined in the GML schema to define appropriate types for an application domain. Non-abstract elements, attributes and types from the GML schema may be used directly in an application schema, if no changes are required.

Following ISO 19109, the feature types of an application or application domain are specified in an application schema. A GML application schema shall be specified in XML Schema and import the GML schema. It may be constructed in one of two different ways:

- By adhering to the rules for GML application schemas specified for creating a GML application schema directly in XML Schema.
- By adhering to the rules specified in ISO 19109 for application schemas in UML, and conforming to both the constraints on such schemas and the rules for mapping them to GML application schemas. The mapping from an ISO 19109 conformant Application Schema in UML to the corresponding GML application schema is based on a set of encoding rules. These encoding rules conform to the rules for
GML application schemas and ISO 19118.

Both ways are valid approaches to construct GML application schemas. All application schemas shall be modeled in accordance with the General Feature Model specified in ISO 19109. Within the ISO 19100 series, UML is the preferred language to describe conceptual schemas.

GML application schemas conformant with this International Standard shall use all of the applicable GML schema components, either directly or by specialization, and are valid in accordance with the rules for XML Schema [3]. The detail of application schema will be represented in following chapter.

2.1.3 GML Grid schema

In this framework we use GML Grid to display sets of uncertainty information. This clause provides grid geometries that are used in the description of gridded coverage and other applications.

In GML two grid structures are defined, namely gml:Grid and gml:RectifiedGrid.

gml:Grid implements ISO 19123 CV_Grid (see D.2.11 and ISO 19123:2005, 8.3) and is defined as follows:

```
<complexType name="GridType">
  <complexContent>
    <extension base="gml:AbstractGeometryType">
      <sequence>

Figure 2. 1 Sample Grid definition[3]
The gml:Grid implicitly defines an unrectified grid, which is a network composed of two or more sets of curves in which the members of each set intersect the members of the other sets in an algorithmic way. The region of interest within the grid is given in terms of its gml:limits, being the grid coordinates of diagonally opposed corners of a rectangular region. gml:axisLabels is provided with a list of labels of the axes of the grid (gml:axisName has been deprecated). gml:dimension specifies the dimension of the grid.

A rectified grid is a grid for which there is an affine transformation between the grid coordinates and the coordinates of an external coordinate reference system. It is defined by specifying the position (in some geometric space) of the grid —origin and of the vectors that specify the post locations [3].

gml:RectifiedGrid implements ISO 19123 CV_RectifiedGrid (see D.2.11 and ISO 19123:2005, 8.9) and is declared as follows:
Note that the grid limits (post indexes) and axis name properties are inherited from gml:GridType and that gml:RectifiedGrid adds a gml:origin property (contains or references a gml:Point) and a list of gml:offsetVector properties (specified using gml:VectorType as its data type as described in 10.1.4.5).
EXAMPLE 2 An example instance of a gml:RectifiedGrid is as follows:

```xml
<gml:RectifiedGrid dimension="2">
  <gml:limits>
    <gml:GridEnvelope>
      <gml:low>1 1</gml:low>
      <gml:high>4 4</gml:high>
    </gml:GridEnvelope>
  </gml:limits>
  <gml:axisLabels>u v</gml:axisLabels>
  <gml:origin>
    <gml:Point gml:id="palindrome"
      srsName="urn:x-ogc:def:crs:EPSG:6.6:4326">
      <gml:pos>3 1.1</gml:pos>
    </gml:Point>
  </gml:origin>
  <gml:offsetVector
    srsName="urn:x-ogc:def:crs:EPSG:6.6:4329">
    -0.2 1.25
  </gml:offsetVector>
  <gml:offsetVector
    srsName="urn:x-ogc:def:crs:EPSG:6.6:4329">
    1.3 0.2
  </gml:offsetVector>
</gml:RectifiedGrid>
```

Figure 2. 4 Sample RectifiedGrid definition[3]

The detail of implementation will be discussed in following chapter.

2.1.4 Domain set definition

The base type for coverages is gml:AbstractCoverageType, defined in the schema as follows:
The basic elements of a coverage can be seen in this content model: the coverage contains gml:domainSet and gml:rangeSet properties. The gml:domainSet property describes the domain of the coverage and the gml:rangeSet property describes the range of the coverage.

The abstract element gml:AbstractCoverage implements ISO 19123 CV_Coverage (see D.2.11 and ISO 19123:2005, 5.3) and is declared as follows:

```xml
<element name="AbstractCoverage" type="gml:AbstractCoverageType" abstract="true"
substitutionGroup="gml:AbstractFeature"/>
```

This element serves as the head of a substitution group which may contain any coverage whose type is derived from gml:AbstractCoverageType. It may act as a variable in the definition of content models where it is required to permit any coverage to be valid.

The gml:domainSet property element describes the spatio-temporal region of interest, within which the coverage is defined. Its content model is given by gml:DomainSetType which is defined as follows:
The value of the domain is thus a choice between a gml:AbstractGeometry and a gml:AbstractTimeObject. In the instance these abstract elements will normally be substituted by a geometry complex or temporal complex, to represent spatial coverage and time-series, respectively.

The presence of the gml:AssociationAttributeGroup means that domainSet follows the usual GML property model and may use the xlink:href attribute to point to the domain, as an alternative to describing the domain inline. Ownership semantics may be provided using the gml:OwnershipAttributeGroup [3]. The detail of implementation will be discussed in following chapter.

2.2 Overview of Uncertainty schema
In this framework we use Uncertainty schema to define the data structure of error information. UncertML is a conceptual model and XML encoding designed for encapsulating probabilistic uncertainties. UncertML is a conceptual model, with accompanying XML schema that may be used to quantify and exchange complex uncertainties in data. The interoperable model can be used to describe uncertainty in a variety of ways including:

- Samples
- Statistics including mean, variance, standard deviation and quantile
- Probability distributions including marginal and joint distributions and mixture models

Utilizing the XML schema provides an interoperable framework for exchanging uncertainties. This allows uncertainty to be propagated through processing chains.

Uncertainty can be quantified in several different ways within UncertML. There are several methods for describing uncertainty including the common elements.

In some situation a user may provide a sample of the data which allows the uncertainties to be described implicitly. Unfortunately, a sufficiently large sample of data is required for calculating the uncertainties, introducing the issue of encapsulating large amounts of data efficiently.

There is an extensive range of options available in UncertML for describing 'summary statistics'. Such statistics are used to provide a summary of a variable ranging from measures of location (mean, mode, median etc) to measures of dispersion (range,
standard deviation, variance etc). While certain statistics do not provide any information about uncertainty they are often used in conjunction with other statistics to provide a concise but detailed summary [4]. The detail of implementation will be discussed in following chapter.

2.3 OGC standards and OGC Web Services

OGC(R) standards are technical documents that detail interfaces or encodings. Software developers use these documents to build open interfaces and encodings into their products and services. These standards are the main "products" of the Open Geospatial Consortium and have been developed by the membership to address specific interoperability challenges.

OGC standards and supporting documents are available to the public at no cost. OGC Web Services (OWS) are OGC standards created for use in World Wide Web applications. Any Schemas (xsd, xslt, etc) that support an approved (that is, approved by the OGC membership) OGC standard can be found in the official OGC Schema Repository [5].

The generality of W3C Services are a problem when users desire a standard set of functionality from Web services. In the geospatial domain, there are several common classes of services. If these services each had standardized functionality, they would become easier to integrate into systems and thus, more useful. The Open Geospatial Consortium (OGC) defines a number of geospatial services which have both a
standardized interface and standardized functionality. By standardizing functionality, the clients for these services may be generic. Unlike for W3C Services, there is no need to create a client for each specific service. As long as a service follows the OGC standard, the client who also follows the standard will work with it.

The OGC has created a large number of geospatial Web service standards. However, three are the most commonly used. The Web Mapping Service (WMS) is used to transfer georeferenced images from the server to the client. The Web Feature Service (WFS) is used to transfer vector data (points, lines, polygons, etc.) encoded using Geography Markup Language (GML), a geospatial specific XML subset. The Web Coverage Service (WCS) is used to transfer geospatial multidimensional raster data. As opposed to the WMS standard, the focus of WCS is on data encoded in formats not supported by Web browsers. WCS originally only supported grid data formats such as GeoTIFF or NetCDF; however, current versions of the standard allow any encoding format for data transfers [6]. The detail of implementation will be discussed in following chapter.

### 2.4 GIS uncertainties

GIS accuracy depends upon source data, and how it is encoded to be data referenced. Land surveyors have been able to provide a high level of positional accuracy utilizing the GPS derived positions. The high-resolution digital terrain and aerial imagery, the powerful computers, Web technology, are changing the quality, utility, and expectations of GIS to
serve society on a grand scale, but nevertheless there are other source data that has an impact on the overall GIS accuracy like: paper maps that are not found to be very suitable to achieve the desired accuracy since the aging of maps affects their dimensional stability.

In developing a digital topographic data base for a GIS, topographical maps are the main source of data. Aerial photography and satellite images are extra sources for collecting data and identifying attributes which can be mapped in layers over a location facsimile of scale. The scale of a map and geographical rendering area representation type are very important aspects since the information content depends mainly on the scale set and resulting locatability of the map's representations. In order to digitize a map, the map has to be checked within theoretical dimensions, and then scanned into a raster format, and resulting raster data has to be given a theoretical dimension by a rubber sheeting/warping technology process.

A quantitative analysis of maps brings accuracy issues into focus. The electronic and other equipment used to make measurements for GIS is far more precise than the machines of conventional map analysis. All geographical data are inherently inaccurate, and these inaccuracies will propagate through GIS operations in ways that are difficult to predict.

A GIS can also convert existing digital information, which may not yet be in map form, into forms it can recognize, employ for its data analysis processes, and use in forming mapping output. For example, digital satellite images generated through remote sensing can be analyzed to produce a map-like layer of digital information about vegetative
covers on land locations. Another fairly recently developed resource for naming GIS location objects is the Getty Thesaurus of Geographic Names (GTGN), which is a structured vocabulary containing about 1,000,000 names and other information about places. Likewise, researched census or hydrological tabular data can be displayed in map-like form, serving as layers of thematic information for forming a GIS map [2].

2.5 Previous work

For processing uncertainty information, there must be a framework to organize and parse the uncertainty data format. The solutions used in existing research are to create XML schema descriptions.

In the previous INTAMAP approach of Aston University of UK, they have developed a framework to support information-rich transfer of complex and uncertain predictions. They developed schema to represent probabilistic results in a GML3.1 (object-property) style. The system will also offer more easily accessible Web Map Service and Web Coverage Service interfaces to allow users to access the system at the level of complexity they require for their specific application. Such a system will offer a very valuable contribution to the next generation of Environmental Information Systems in the context of real time mapping for monitoring and security, particularly for systems that employ service oriented architecture [7].

In the UncertML approach the author present UncertML, an XML schema which
provides a framework for describing uncertainty as it propagates through many applications, including online risk management chains. This uncertainty description ranges from simple summary statistics (e.g., mean and variance) to complex representations such as parametric, multivariate distributions at each point of a regular grid. The philosophy adopted in UncertML is that all data values are inherently uncertain, (i.e., they are random variables, rather than values with defined quality metadata) [8].

The paper “Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know” talked about identifying seven key research challenges in visualizing information uncertainty, particularly as it applies to decision making and analysis. An important part of a broad strategy is to enable analysts, decision makers, and others to cope with uncertain information. A wide variety of strategies has been proposed for uncertainty visualization, and there is a growing body of empirical research that is providing insights concerning which method share effective in different use contexts and for which types of tasks [9]. Still, the author has only scratched the surface of the problem. For example, we cannot yet say definitively whether decisions are better if uncertainty is visualized or suppressed, or under what conditions they are better; nor do we understand the impact of uncertainty visualization on the process of analysis or decision making.
3.1 Framework Overview

There are no standards supporting uncertainties of geospatial data processing. The data providers are mandated to use the existing OGC standards. Current standards all support sophisticated geospatial and environmental constructs by default, for example, Datums, Projections, Topology, Grids, etc. However none of these standards support uncertainty information. In this project, we have developed a framework that helps the users who need integrate the uncertainty information to the geospatial data. The framework will provide the following functions:

- Formal Standard as an XML Schema that defines the vocabulary and syntax for indicating error estimates.
- Tier based classification system for qualities of certainty reporting.
- Guidance for integration of tiered framework into existing processes.
- Guidance and tools for discretization of tiered certainty information.
- Best practice recommendations and tools for certainty provenance analysis.

3.2 Framework Architecture

The framework design has been based on Geospatial Markup Language and GML application schema. The structure of uncertainty information is defined as element of GML application schema. The user imports the uncertainty GML application schema to
parse the uncertainty information in GML data. The GML data file is readable by geospatial software such as Gaia.

The flow control of the framework is simple: The uncertainty information is defined as GML application schema. The GML files import that application schema. The coordinate information also is defined as a rectified grid object. The uncertainty information is defined as <gml:rangeSet> object of the GML grid. Each POI (Points of Interest) has its own uncertainty information. The framework is flexible and scalable for various types of uncertainty information. It is easy to customize and extend the framework for other user defined data set.

![Figure 3.1 Framework Architecture](image)
3.3 Framework Components

The framework consists of several major components as shown in Figure 3.2 in which the component GML application schema represents the “Schema”. The GML data file component represents the “GML”. The distribution type definition schema represents the “uncertainty Schema” to import. The application schema and uncertainty schema are specified in the head of GML data file. The grid definition of GML data file specifies the GML grid definition. The user data will be loaded into range definition of GML data file. The components are discussed in the following sub sections.

Figure 3.2 Framework Components Diagram
3.3.1 GML application schema

A schema is an abstract collection of metadata, consisting of a set of schema components: chiefly element and attribute declarations and complex and simple type definitions. These components are usually created by processing a collection of schema documents, which contain the source language definitions of these components. In popular usage, however, a schema document is often referred to as a schema. The application schema has been written to conform to the GML standard. The application schema must import several GML namespaces.

A GML application schema shall import the full GML schema. It may identify GML profiles that include all of the components from GML that it directly or indirectly uses to define its vocabulary as specified.

The required import of the GML schema components may be provided indirectly via the import of another schema in the namespace of GML that includes the required GML schema documents.

EXAMPLE 1 The import of gml.xsd from Annex C would satisfy any of these schema import requirements.

```xml
<import namespace="http://www.opengis.net/gml/3.2"
schemaLocation="../gml.xsd"/>
```

Figure 3.3 Import GML application schema[3]

The `<import>` element specifies that the components described in the imported GML schema document are associated with the GML namespace.
http://www.opengis.net/gml/3.2. This namespace identifier shall match the target namespace specified in the schema being imported in order to ensure XML Schema validity.

The path (schemaLocation) to the imported GML schema document shall be provided and may point to a local copy of the document, or may point to a URI reference to a copy of the schema document in some remote repository [3].

In order to create a GML application schema user also need import the Uncertainty schema as namespace. It consists of an XSD element for type of Grids, which contain the <gml:CovarianceType> type. The structure of application schema is flexible and easy to extend for more types of features. Users can define new type based on current sample format. Therefore, user can organize his data by organizing the sequence of the XSD element. Conforming to the GML, various methods have been defined in the GML application schema for parsing the internal structures of GML files.

3.3.2 GML

The GML file contains the uncertainty information appended at the current geospatial information. Designers of GML application schemas may extend or restrict the types defined in the GML schema to define appropriate types for an application domain. Non-abstract elements, attributes and types from the GML schema may be used directly in an application schema, if no changes are required. In this framework, the GML file
contains two different parts: Grid definition and Range definition. The Grid definition is used to define positions of grid points. The Range definition is used to define the values and the uncertainty information of the grid points.

We use EPSG:6.6:4326 as our coordinates system. User should specify the application schema to define the structure of this GML file.

![Imported GML application schema and Uncertainty schema](image)

![Define GML Grids schema](image)
Specify the coordinate system, the origin and the offset

![Define the range set](image)
Specify the uncertainty data format and location

Figure 3. 4 GML data file format

### 3.3.3 Grids schema

A grid is simply a two-dimensional array. A grid has an integer width and height. The structure allows you to set and retrieve the value of cells in the grid by giving the index of it (which starts with 0 in both the x- and the y-direction). We can set the value in regions, add values, and retrieve the sum, max, min, and mean value over a region. The structure
is useful to represent e.g. an acoustic field. Even though all functionality can also be
achieved using two-dimensional arrays, the operations on regions are a lot faster. In the
current case the coordinates of the POIs are continuous. So a Grid is an easy way to
visualize the data and uncertainty information. For the correctness of the data, users
should specify the appropriate value of the following parameters: coordinate system,

3.3.4 Uncertainty Schema

UncertML is a conceptual model, realized as an XML schema, that allows uncertainty to
be quantified in a variety of ways i.e. realizations, statistics and probability distributions.
UncertML is based upon a soft-typed XML schema design that provides a generic
framework from which any statistic or distribution may be created. Making extensive use
of Geography Markup Language (GML) dictionaries, UncertML provides a collection of
definitions for common uncertainty types. Containing both written descriptions and
mathematical functions, encoded as MathML, the definitions within these dictionaries
provide a robust mechanism for defining any statistic or distribution and can be easily
extended. Universal Resource Identifiers (URIs) are used to introduce semantics to the
soft-typed elements by linking to these dictionary definitions [10].

UncertML is divided into three distinct packages. Each package is tailored toward
describing uncertainty using a specific mechanism; either through realizations, statistics
or probability distributions. In this framework, we only use distribution types. The Gaussian distribution is the most common distribution in this case.

Here is an example about “Confidence Interval” of UncertML. Confidence Interval is a concept to describe statistics information in UncertML.

```xml
<xs:element name="ConfidenceInterval" substitutionGroup="un:AbstractSummaryStatistic">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="un:ConfidenceIntervalType"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

<xs:complexType name="ConfidenceIntervalType">
  <xs:complexContent>
    <xs:extension base="un:AbstractSummaryStatisticType">
      <xs:sequence>
        <xs:element name="lower" type="un:QuantileType"/>
        <xs:element name="upper" type="un:QuantileType"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Figure 3. 5 XML schema of Confidence Interval [4]

```xml
<un:ConfidenceInterval xmlns:un="http://www.uncertml.org/2.0">
  <un:lower level="0.05">
    <un:values>3.14</un:values>
  </un:lower>
  <un:upper level="0.95">
    <un:values>6.28</un:values>
  </un:upper>
</un:ConfidenceInterval>
```

Figure 3. 6 XML example of Confidence Interval [4]
Here is an example about “Binomial Distribution” of UncertML. Binomial Distribution is a concept to describe distribution information in UncertML.
3.3.5 Geoserver

GeoServer is an open source software server written in Java that allows users to share and edit geospatial data. Designed for interoperability, it publishes data from any major spatial data source using open standards. Being a community-driven project, GeoServer is developed, tested, and supported by a diverse group of individuals and organizations from around the world.

GeoServer is the reference implementation of the Open Geospatial Consortium (OGC) Web Feature Service (WFS) and Web Coverage Service (WCS) standards, as well as a high performance certified compliant Web Map Service (WMS). GeoServer forms a core component of the Geospatial Web.

GeoServer allows you to display your spatial information to the world. Implementing the Web Map Service (WMS) standard, GeoServer can create maps in a variety of output...
formats. OpenLayers, a free mapping library, is integrated into GeoServer, making map
generation quick and easy. GeoServer is built on Geotools, an open source Java GIS
toolkit.

GeoServer can display data on any of the popular mapping applications such as Google
Maps, Google Earth, Yahoo Maps, and Microsoft Virtual Earth. In addition, GeoServer
can connect with traditional GIS architectures such as ESRI ArcGIS [11].

The primary purpose of the GeoServer in this framework is to publish the GML
application schema and Uncertainty schema on web. That make the schema is accessible
for the GML file. The GML file should use that GML application schema and
Uncertainty schema to parse the GML data.

3.3.6 Gaia

Gaia is a platform designed for advanced geospatial network and SDI needs. The primary
purpose of the GeoServer in this framework is to publish the GML application schema
and Uncertainty schema on the web. That make the schema accessible for the GML file.
The GML file should use that GML application schema and Uncertainty schema to parse
the GML data [12].

The Gaia software has the ability to visualize the GML data. The GML file should be
loaded as a layer. The user also can import and load the Geospatial Session Files (GSF).
The uncertainty information is marked with POI. That information is visualized and easy to access by Gaia.
CHAPTER 4 FRAMEWORK IMPLEMENTATION

The Framework provides a general solution for marking, propagation and transmission of error information for Environmental and Geospatial data.

The Framework supports storing and processing uncertainty information of geospatial data with the four capabilities listed below:

1. Create methods for reporting uncertainty associated with geospatial and environmental data

2. Establish data provenance procedures for traceability of the origins of uncertainty

3. Establish standard procedures for discretization of distributions

4. Use example studies to test and validate the framework

4.1 Uncertainty data loading process

This function is implemented in the GML application schema and GML data. At first, the user should write a specific application schema to define the GML coverage type. In the GML file, user should import this application schema and the uncertainty schema. User should specify the distribution type of the data. Then the data will be stored as <gml:rangeSet>. So the uncertainty information will automatically parsed by the GML file reader.

Figure 4.1 shows the raw data format. The uncertainty data contains the coordinates information and distribution information. The following is the example:
<table>
<thead>
<tr>
<th>Longitude</th>
<th>latitude</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>-130.0000</td>
<td>49.0000</td>
<td>2698</td>
<td>26.98</td>
</tr>
<tr>
<td>-129.9917</td>
<td>49.0000</td>
<td>2678</td>
<td>26.78</td>
</tr>
<tr>
<td>-129.9833</td>
<td>49.0000</td>
<td>2657</td>
<td>26.57</td>
</tr>
<tr>
<td>-129.9750</td>
<td>49.0000</td>
<td>2637</td>
<td>26.37</td>
</tr>
<tr>
<td>-129.9667</td>
<td>49.0000</td>
<td>2639</td>
<td>26.39</td>
</tr>
<tr>
<td>-129.9583</td>
<td>49.0000</td>
<td>2651</td>
<td>26.51</td>
</tr>
<tr>
<td>-129.9500</td>
<td>49.0000</td>
<td>2650</td>
<td>26.50</td>
</tr>
<tr>
<td>-129.9417</td>
<td>49.0000</td>
<td>2650</td>
<td>26.50</td>
</tr>
<tr>
<td>-129.9333</td>
<td>49.0000</td>
<td>2635</td>
<td>26.35</td>
</tr>
<tr>
<td>-129.9250</td>
<td>49.0000</td>
<td>2613</td>
<td>26.13</td>
</tr>
<tr>
<td>-129.9167</td>
<td>49.0000</td>
<td>2603</td>
<td>26.03</td>
</tr>
<tr>
<td>-129.9083</td>
<td>49.0000</td>
<td>2599</td>
<td>25.99</td>
</tr>
<tr>
<td>-129.9000</td>
<td>49.0000</td>
<td>2591</td>
<td>25.91</td>
</tr>
<tr>
<td>-129.8917</td>
<td>49.0000</td>
<td>2581</td>
<td>25.81</td>
</tr>
<tr>
<td>-129.8833</td>
<td>49.0000</td>
<td>2569</td>
<td>25.69</td>
</tr>
<tr>
<td>-129.8750</td>
<td>49.0000</td>
<td>2535</td>
<td>25.35</td>
</tr>
<tr>
<td>-129.8667</td>
<td>49.0000</td>
<td>2547</td>
<td>25.47</td>
</tr>
<tr>
<td>-129.8583</td>
<td>49.0000</td>
<td>2539</td>
<td>25.39</td>
</tr>
<tr>
<td>-129.8500</td>
<td>49.0000</td>
<td>2544</td>
<td>25.44</td>
</tr>
<tr>
<td>-129.8417</td>
<td>49.0000</td>
<td>2542</td>
<td>25.42</td>
</tr>
<tr>
<td>-129.8333</td>
<td>49.0000</td>
<td>2548</td>
<td>25.48</td>
</tr>
<tr>
<td>-129.8250</td>
<td>49.0000</td>
<td>2540</td>
<td>25.40</td>
</tr>
<tr>
<td>-129.8167</td>
<td>49.0000</td>
<td>2533</td>
<td>25.33</td>
</tr>
<tr>
<td>-129.8083</td>
<td>49.0000</td>
<td>2528</td>
<td>25.28</td>
</tr>
<tr>
<td>-129.8000</td>
<td>49.0000</td>
<td>2519</td>
<td>25.19</td>
</tr>
<tr>
<td>-129.7917</td>
<td>49.0000</td>
<td>2490</td>
<td>24.90</td>
</tr>
<tr>
<td>-129.7833</td>
<td>49.0000</td>
<td>2481</td>
<td>24.81</td>
</tr>
<tr>
<td>-129.7750</td>
<td>49.0000</td>
<td>2458</td>
<td>24.58</td>
</tr>
</tbody>
</table>

Figure 4. 1 GML data file format

From this sample data, we get the data boundary from \([-130.0000, 49.0000]\) to \([-129.0000, 50.0000]\). The origin point is \([-130.0000, 49.0000]\). The offset vectors in this raw data are \([0.083, 0],[0, 0.083]\).

4.2 Define the coverage schema
The base type for coverages is gml:AbstractCoverageType, defined in the schema as follows:

```xml
<complexType name="AbstractCoverageType" abstract="true">
  <complexContent>
    <extension base="gml:AbstractFeatureType">
      <sequence>
        <element ref="gml:domainSet"/>
        <element ref="gml:rangeSet"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

Figure 4. 2 GML AbstractCoverageType definition[3]

The basic elements of a coverage can be seen in this content model: the coverage contains gml:domainSet and gml:rangeSet properties. The gml:domainSet property describes the domain of the coverage and the gml:rangeSet property describes the range of the coverage.

A discrete coverage consists of a domain set, range set and optionally a coverage function. The domain set consists of either spatial or temporal geometry objects, finite in number.

The range set is comprised of a finite number of attribute values each of which is associated to every direct position within any single spatiotemporal object in the domain.

In other words, the range values are constant on each spatiotemporal object in the domain.

This coverage function maps each element from the coverage domain to an element in its range. This definition conforms to ISO 19123. The base type for discrete coverage is DiscreteCoverageType, defined in the schema as follows:
In our case, the uncertainty information should be stored as a DiscreteCoverageType. The uncertainty schema is defined as a namespace un

```
<complexType name="DiscreteCoverageType">
  <complexContent>
    <extension base="gml:AbstractCoverageType">
      <sequence>
        <element ref="gml:coverageFunction" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

Figure 4. 3 GML DiscreteCoverageType definition[3]

In the head of the GML data file, we specify the application schema location as

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:un="http://www.uncertml.org/2.0"
  xmlns:cite="http://www.opengeospatial.net/cite"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:it.geosolutions="http://www.geo-solutions.it"
  xmlns:nurc="http://www.nurc.nato.int"
  xmlns:sde="http://geoserver.sf.net"
  xmlns:sf="http://www.openplans.org/spearfish"
  xmlns:test="http://137.30.122.154/test"
  xmlns:un="http://www.uncertml.org/2.0"
  xmlns:topp="http://www.openplans.org/topp" elementFormDefault="qualified"
  targetNamespace="http://137.30.122.154/test">
  <xsd:import namespace="http://www.opengis.net/gml"
    schemaLocation="http://127.0.0.1:8080/geoserver/schemas/gml/3.1.1/base/gml.xsd"/>
  <xsd:element name="UncertaintyCoverage" type="DiscreteCoverageType"/>
</xsd:schema>
```

Figure 4. 4 UncertaintyCoverage element definition in application schema
We also specify the uncertainty schema as namespace un: “xmlns:un=http://www.uncertml.org/2.0“. So the GML data file can use the distribution data type. The following the GML data file example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<test:UncertaintyCoverage
 xmlns:test="http://137.30.122.154/test"
 xmlns:un="http://www.uncertml.org/2.0"
 ...
```

Figure 4. 5 Schema information in the GML data file

4.3 Define the grid

This function is implemented in the GML data. A rectified grid is a grid for which there is an affine transformation between the grid coordinates and the coordinates of an external coordinate reference system. It is defined by specifying the position (in some geometric space) of the grid — origin and of the vectors that specify the post locations.

```xml
<gml:RectifiedGrid dimension="2">
    <gml:limits>
        <gml:GridEnvelope>
            <gml:low>1 1</gml:low>
        </gml:GridEnvelope>
    </gml:limits>
</gml:RectifiedGrid>
```

Figure 4. 6 Example instance of a gml:RectifiedGrid[3]
In our user case, the boundary of POIs are \([-130.00, 49.00]\) and \([-129.00, 50.00]\). The origin of grid is \([-130.00, 49.00]\). The offset vector are \([0.0083, 0.0000]\) and \([0.0000, 0.0083]\).

The grid definition is defined at GML data file as

```xml
<!--
==================================================================================
============
Grid definition
==================================================================================
============= -->
<gml:domainSet>
  <gml:RectifiedGrid gml:id="RG001_C001" srsName="urn:x-ogc:def:crs:EPSG:6.6:4326" axisLabels="Long Lat" uomLabels="deg deg" dimension="2">
    <gml:limits>
      <gml:GridEnvelope>
        <gml:low>-130.0000 49.0000</gml:low>
        <gml:high>-129.0000 50.0000</gml:high>
      </gml:GridEnvelope>
    </gml:limits>
    <gml:axisLabels>x y</gml:axisLabels>
  </gml:RectifiedGrid>
</gml:domainSet>
```

Figure 4.7 Example instance of a gml:RectifiedGrid
4.4 Specify the distribution information

UncertML is a conceptual model, with accompanying XML schema that may be used to quantify and exchange complex uncertainties in data. The interoperable model can be used to describe uncertainty in a variety of ways including:

- Samples
- Statistics including mean, variance, standard deviation and quantile
- Probability distributions including marginal and joint distributions and mixture models

When the uncertainties of your data are more clearly understood it may be desirable to describe them through the use of probability distributions. The elements listed below are specifically designed to allow a concise encapsulation of many distributions without sacrificing the simplicity of UncertML.

- BernoulliDistribution
- BetaDistribution
- BinomialDistribution
- CauchyDistribution
- ChiSquareDistribution
- DirichletDistribution
- ExponentialDistribution
- FDistribution
- GammaDistribution
- GeometricDistribution
- HypergeometricDistribution
- InverseGammaDistribution
- LaplaceDistribution
- LogNormalDistribution
- LogisticDistribution
- MixtureModel
- MultinomialDistribution
- MultivariateNormalDistribution
- MultivariateStudentTDistribution
- NegativeBinomialDistribution
- NormalDistribution
- NormalInverseGammaDistribution
- ParetoDistribution
- PoissonDistribution
- StudentTDistribution
- UniformDistribution
- WeibullDistribution
- WishartDistribution

In our user case, the uncertainty information is normally distributed. Therefore, we choose NormalDistributionType as the data type.

```xml
<xs:element name="NormalDistribution" substitutionGroup="un:AbstractDistribution">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="un:NormalDistributionType"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

Figure 4. Distribution type definition in Uncertainty schema
The <gml:rangeSet> contains the uncertainty data type and user data. In our case we choose <un:NormalDistribution> as distribution type. The uncertainty information is stored as <gml:doubleOrNilReasonTupleList>. It consists of a list of gml:doubleOrNilReason values, each separated by whitespace. The gml:doubleOrNilReason values are grouped into tuples where the dimension of each tuple in the list is equal to the number of range parameters.
4.5 User defined distribution

UncertML does not cover all the types of distribution. Sometimes users can define their own distribution of data to extend UncertML. For example, a user may need to implement the concept of CEP (circular error probability) [13] in GML. In the military science of ballistics, CEP is an intuitive measure of a weapon system's precision. It is defined as the radius of a circle, centered about the mean, whose boundary is expected to include 50% of the population within it.

However, there is no definition of circular error type in UncertML. Since the data structure of quantile in UncertML is similar to CEP we use quantile as base type to represent the data of CEP.
In our application schema, users can define an extended xs:attribute to implement the concept of CEP.

Figure 4. 10 XML schema for quantile in UncertML [4]

Figure 4. 11 Application schema definition for CEP
User can represent CEP information which is defined by this application schema in the GML. The level shows the circular error probability. The value shows the radius of the circle.

```xml
<?xml version="1.0" encoding="utf-8" ?>
<ogr:FeatureCollection
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://137.30.122.154:8080/geoserver/schemas/gml/3.1.1/gaia/shore_line120620.xsd"
    xmlns:ogr="http://ogr.maptools.org/"
    xmlns:un="http://www.uncertml.org/2.0"
    xmlns:gml="http://www.opengis.net/gml">
    <ogr:CircularErrorProbability level="0.90">
        <un:values>500</un:values>
    </ogr:CircularErrorProbability>
</ogr:FeatureCollection>
```

Figure 4. 12 CEP usage sample

### 4.6 Making data with user-defined distribution

The format of GML data must conform to the format of corresponding user-defined schema. For example, there is already a defined schema for Louisiana highway data.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://ogr.maptools.org/"
    xmlns:ogr="http://ogr.maptools.org/"
    xmlns:un="http://www.uncertml.org/2.0"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:gml="http://www.opengis.net/gml"
    elementFormDefault="qualified" version="1.0">
    <xs:import namespace="http://www.opengis.net/gml"
        schemaLocation="http://schemas.opengeospatial.net/gml/2.1.2/feature.xsd"/>
    <xs:element name="FeatureCollection" type="ogr:FeatureCollectionType"
        schemaLocation="http://schemas.opengeospatial.net/gml/2.1.2/feature.xsd">
</xs:element>
```

Figure 4. 13 Sample application schema for Louisiana highway case
Figure 4. 13 Sample application schema for Louisiana highway case
Figure 4. 13 Sample application schema for Louisiana highway case

In the sample schema, we have defined an element <xs:attribute> as StandardDeviation for all of the data and an element <xs:element> as StandardDeviation for single road segment. To import the schema, a user should specify the imported schema location and the StandardDeviation element. If a user defines the value of <xs:attribute> element, each coordinate has same value of standard deviation. If a user defines the value as <xs:element> each coordinate has its own value of standard deviation.

<?xml version="1.0" encoding="utf-8" ?>
<ogr:FeatureCollection
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://137.30.122.154:8080/geoserver/schemas/gml/3.1.1/gaia/louisiana_road120620.xsd"
   xmlns:ogr="http://ogr.maptools.org/"
   xmlns:un="http://www.uncertml.org/2.0"
   xmlns:gml="http://www.opengis.net/gml">
   <gml:boundedBy>
      <gml:Box>
      </gml:Box>
   </gml:boundedBy>
   .......
   <gml:featureMember>
      <ogr:louisiana_highway fid="F913">
         <ogr:geometryProperty><gml:LineString srsName="EPSG:4326"><gml:coordinates>-90.037174014541918,30.032596722361284,121.06703112367</gml:coordinates>
      </gml:LineString></ogr:geometryProperty>
   </ogr:louisiana_highway>
   .......
</ogr:FeatureCollection>

Figure 4. 14 Sample GML for Louisiana highway case
4.7 Visualize data in Gaia

Gaia is a platform designed for advanced geospatial network and SDI needs. Based on the CarbonTools PRO open-geospatial development toolkit, this viewer can access an array of geospatial sources such as the Open Geospatial Consortium (OGC) Web Mapping Service (WMS), Web Map Tile Service (WMTS), Web Coverage Service (WCS), Web Feature Service (WFS), Filter Encoding (FE). It also supports services such as Microsoft Bing Maps, Yahoo! Maps and OpenStreetMap (OSM), as well as file formats such as ESRI Shapefiles, Google Earth KML/KMZ, DXF, MIF, Geography Markup Language (GML) and GML Simple Features (GMLsf).

With Gaia you can use geospatial content from different sources and overlay them into a
single map view, with each layer individually configured and styled. The Gaia multi-layer view allows seamless use of multiple layers of different types. Panning, zooming and other mapping tools provide a fast and convenient tool for browsing the map. Gaia uses dynamic caching of content to memory, providing enhanced mapping performance [9].

In Gaia, choose File-> New to create a new project session file. Then we should choose “Add layers to Map” option to add the GML files to the map. The GML version should be specified as GML3.

Figure 4. 15 Load GML data file to Gaia
Users can click the “information” button in the menu bar, and then select the specific POI point in the map. A window will popup to show the uncertainty information of this POI. The following chart shows that uncertainty information has successfully been loaded and visualized by Gaia.

![Figure 4. 16 Grids information showed in Gaia](image-url)
Figure 4. 17 Uncertainty information showed in Gaia
CHAPTER 5 APPLICATIONS OF THE FRAMEWORK

The framework effectively shields the complexity of uncertainty information from the user data and eases the development of Geospatial applications in various fields. Applications in different domains will be discussed in this chapter to demonstrate how effective applications can be developed using the proposed framework.

The framework makes it possible to represent error information in a structured way and thus expresses the distribution of user data clearly. Application experts can create the GML file with uncertainty; the result is more accurate compared to current GML and KML (Keyhole Markup Language) standards.

5.1 Depth sounding error measurement

Given the set of tiers any combination of uncertainty measures can be represented in a hierarchical fashion. The first example is depth sounding. The depth is around fifty meters. The time is at 2008-05-16 T05:45:01.23. The coordinates position is -130.38342, 9.76323. The datum is WGS84. The depth of the error values is confidence interval of 1.2m. The time of the error values implicit confidence interval of 0.2 seconds. The position of the error values joint distribution over x and y with

\[ N(0,0,\sigma_x,\sigma_y,\rho_{xy}) \]
We apply the framework representation of the depth sounding with error notations. The formal schema allows for standardized transmission and computer automated reasoning on error information. The hierarchical nature of XML allows complex relationships to be modeled.

Figure 5.1 Schema for depth sounding with error notations
Figure 5.2 Hierarchical nature of XML

Data providers must classify their error into one of the 5 tiers. Once a tier is selected, the framework standard tells the user how to encode their error data. Data consumers, those designing models or processes, decide what type of error analysis to perform.

- If all inputs are normal, end users can perform analytical error calculations
- If all inputs are discrete, end users can perform numerical error calculations

The framework does not mandate which tier providers use or how and if consumers perform error analysis.

This is the sample application schema for this user case.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"

Figure 5.3 Sample application schema for depth sounding case
This is the sample GML data file for this user case.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<test:UncertaintyCoverage
 xsi:schemaLocation="
 http://137.30.122.154:8080/geoserver/schemas/gml/3.1.1/gaia/schema120302.xsd"
 xmlns:ogc="http://www.opengis.net/ogc"
 xmlns:wfs="http://www.opengis.net/wfs"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:ows="http://www.opengis.net/ows"
 xmlns:un="http://www.uncertml.org/2.0"
 xmlns:xlink="http://www.w3.org/1999/xlink">

<!--
=================================================================================================================
============
Boundary definition
=================================================================================================================
-->

Figure 5. 4 Sample GML for depth sounding case
```
Figure 5. 4 Sample GML for depth sounding case
5.2 Advanced route planning

The second example covers complicated orchestration of processes and data types. The example provides variety of ways to define distribution of user data. The following diagram shows the design of an advanced route planning algorithm.
Each data source or process is a source of error. The example is for four road databases with vastly different certainty models. All are currently treated as having equivalent quality. The model is flexible because the current data sharing methods do not support error making.
User can track the origin of the error information. It also helps the user to estimate the error information.

There are two shape file datasets that cover the New Orleans region. The first is from Open Street map, which is a free worldwide map [15]. Open Street map is a community contributed map. The second is from Tiger2010 map which is provided by US Census Bureau. TIGER/Line®Shapefiles are spatial extracts from the Census Bureau's MAF/TIGER database, containing features such as roads, railroads, rivers, as well as legal and statistical geographic areas. Tiger is a commercial standard map. They are made available to the public at no charge and are typically used to provide the digital map base for a Geographic Information System or for mapping software [16].
Figure 5. 7 Different data source with different standard deviation

The road segments in figure 5.7 are from different data sources. The road segments in black are from Open Street map. The road segments in blue are from Tiger map. Since these two datasets are in ESRI shapefile format, user should use specific tools to convert it to GML. Geodata converter is an online website to do the conversion between ESRI shapefile and GML [17]. Several road segments have been chosen as a sample from the full data set. The first was annotated with standard deviation randomly distributed between 0.001 and 0.01 for each road segment. The second was annotated with standard deviation of 0.005 for each road segment.

This is the sample schema for Louisiana Roads XSD. The standard deviation is defined as `<xs:attribute>` element. If a user defines the value of `<xs:attribute>` element, each
coordinate has same value of standard deviation. If a user defines the value as
<xs:element> each coordinate has its own value of standard deviation.

Figure 5.8 Sample application schema for Louisiana roads case
Figure 5. 8 Sample application schema for Louisiana roads case
This is the sample GML data file for Louisiana Highway. In this file, each road segment has the same standard deviation of 0.05.
Figure 5. 9 Sample GML for Louisiana highway case
This is the sample GML data file for the tl_roads data. In this file, each road segment has a standard deviation which is randomly distributed between 0.001 and 0.01.
Figure 5. 10 Sample GML for tl_roads case
5.3 Circular error for shore line

The third use case deals with the accurate usage of multi-scale vector data. All vector data is targeted to specific fixed map scales. The map scale that a vector product is tied to leads directly to some estimate of its underlying error. Usually, neither the map scale nor the error is ever transmitted with the product. As a result, the product is used
incorrectly. Often if the vector data is drawn on a map then it will be drawn as if it were more accurate than it truly is. For example, that the accuracy could change when that island is drawn over some satellite imagery. By attaching the circular error to vector data user can visualize the data properly so that it indicates the underlying uncertainty. This way, these products can be used for things like navigation where not having the uncertainty can cause accidents.

There are two shape file datasets from General Bathymetric Chart of the Oceans (GEBCO) [18]. The GEBCO Digital Atlas (GDA) is maintained by BODC on behalf of the International Hydrographic Organization (IHO) and the Intergovernmental Oceanographic Commission (IOC) of UNESCO. This section presents a collection of many different coastline data files. In this case, the first product is at 1:250,000 map scale level. The second product is at 1:12,000,000 map scale level.
Figure 5. 11 Map with 1:250,000 scale level

Figure 5. 12 Map with 1:12,000,000 scale level
Since these two datasets are in ESRI shapefile format, user should use the method described in sub-clause 5.2 to convert them to GML. Several shoreline segments have been chosen as sample from the full data set.

This is the sample schema for shore line XSD. The standard deviation is defined as <xs:attribute> element. If a user specifies the value of <xs:attribute> element, each coordinate has the same value of circular error.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://ogr.maptools.org/
xmlns:ogr="http://ogr.maptools.org/
xmlns:un="http://www uncertml.org/2.0/
xmlns:xsl="http://www.w3.org/2001/XMLSchema
xmlns:gml="http://www.opengis.net/gml" elementFormDefault="qualified" version="1.0">
<xs:element name="FeatureCollection" type="ogr:FeatureCollectionType" substitutionGroup="gml:_FeatureCollection"/>
<xs:complexType name="FeatureCollectionType">
  <xs:complexContent>
    <xs:extension base="gml:AbstractFeatureCollectionType">
      <xs:attribute name="lockId" type="xs:string" use="optional"/>
      <xs:attribute name="scope" type="xs:string" use="optional"/>
      <xs:attribute name="CircularErrorProbability" type="un:Quantile" use="optional"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="ShoreLine" type="ogr:ShoreLine_Type" substitutionGroup="gml:_Feature"/>
<xs:complexType name="ShoreLine_Type">
  <xs:complexContent>
    <xs:extension base="gml:AbstractFeatureType">
      <xs:sequence>
        <xs:element name="geometryProperty" type="gml:GeometryPropertyType" nillable="true" minOccurs="1"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Figure 5.13 Sample application schema for Shore Line case
This is the sample GML data file at 1:250,000 map scale level. In this file the circular error is 500m with a probability of 90 percent.
Figure 5. 14 Sample GML for 1:250,000 map scale level
Figure 5. 14 Sample GML for 1:250,000 map scale level

This is the sample GML data file at 1:12,000,000 map scale level. In this file the circular error is 24km with a probability of 90 percent.

<?xml version="1.0" encoding="utf-8" ?>
<ogr:FeatureCollection
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://137.30.122.154:8080/geoserver/schemas/gml/3.1.1/gaia/shore_line120620.xsd"
 xmlns:ogr="http://ogr.maptools.org/"
 xmlns:un="http://www.uncertml.org/2.0"
 xmlns:gml="http://www.opengis.net/gml">

<ogr:CircularErrorProbability level="0.90">
 <un:values>24000</un:values>
</ogr:CircularErrorProbability>

<gml:boundedBy>
 <gml:Box>
 </gml:Box>
</gml:boundedBy>
</gml:featureMember>
</ogr:FeatureCollection>

Figure 5. 15 Sample GML for 1:12,000,000 map scale level
The circular error information has been added to the GML data. The first map was
annotated with a circular error of 500m with probability 90 percent.

Figure 5. 16 Map with circular error of 500m

The second map was annotated with a circular error of 24km with probability 90 percent.
Figure 5. Map with circular error of 24km
CHAPTER 6 CONCLUSION AND FUTURE WORK

A flexible framework that enables the utilization of uncertainty information by users of Geospatial Information Systems has been outlined. The proposed framework provides a convenient way to process the uncertainty information with GML data. This framework fills an important gap as there is currently no commercial product to provide the function of integrating and processing the uncertainty information with GML. In this project, we have developed a framework to enable GIS system users and Web users to process uncertainty information and visualize it using Gaia and other software visualization tools. We have presented three examples demonstrating the applications of our framework. Future work includes the extension of the current framework with better automation of product creation and data sharing. The future version will add a method to read, trace and analyze the uncertainty information from GML data. This will make the extended framework a more useful tool for informed environmental decision making and better oceanographic and acoustic modeling.
References

1. Geography Markup Language:  
2. Geographic information system:  
   http://en.wikipedia.org/wiki/Geographic_information_system
5. OGC Standards: http://www.opengeospatial.org/standards
9. Alan M. MacEachren, Anthony Robinson, Susan Hopper, Steven Gardner, Robert Murray, Mark Gahegan, and Elisabeth Hetzler: Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know
14. Modeling of Floating Debris, Environmental Modeling Center National Centers for Environmental Prediction, NOAA
15. OpenStreetMap: http://www.openstreetmap.org/
16. TIGER/Line Shapefiles and TIGER/Line Files:  
   http://www.census.gov/geo/www/tiger/shp.html
18. GEBCO:  
   http://www.badc.ac.uk/projects/international/gebco/gebco_digital_atlas/
VITA

Zhao Yang was born in Shanxi, China. He earned a Bachelor Degree in Computer Science at Northwestern Polytechnical University, Xi’an China. He was enrolled in the graduate program in Computer Science at the University of New Orleans in August 2010.