Geospatial Annotations for 3D Environments and their WFS-based Implementation

Jan Klimke, Jürgen Döllner

Hasso-Plattner-Institute, University of Potsdam,
Prof.-Dr.-Helmert-Strasse 2-3,
14482 Potsdam, Germany,
{jan.klimke, juergen.doellner}@hpi.uni-potsdam.de

Abstract. Collaborative geovisualization provides effective means to communicate spatial information among a group of users. Annotations as one key element of collaborative geovisualization systems enable comprehension of collaboration processes and support time-shifted communication. By *annotations* we refer to user-generated information such as remarks, comments, findings and any other information related to the 3D environment. They have to be efficiently modeled, stored and visualized while precisely retaining their spatial reference and creation context. Existing models for annotations generally do not fully support spatial references and, therefore, do not fully take advantage of the spatial relationships associated with annotations. This paper presents a GML-based data model for geospatial annotations that explicitly incorporates spatial references and allows different types of annotations to be stored together with their context of creation. With this approach annotations can be represented as first-class spatial features. Consequently, annotations can be seamlessly integrated into their 3D environment and the author's original intention and message can be better expressed and understood. An OGC Web Feature Service is used as standardized interface for storage and retrieval of annotations, which assures data interoperability with existing geodata infrastructures. We have identified three types of *annotation subjects*, namely geographic features, geometry, and scene views, represented by their corresponding 2D/3D geometry. The model also defines a point-based approximation for complex geometry, such that annotations can also be used by client application with limited abilities regarding display size, band-

width or geometry handling. Furthermore we extended our model by annotations that can contain 3D geometry besides textual information. In this way the expressiveness of annotations can be further enhanced for communicating spatial relationships such as distances or arrangements of geographic features.

1 Introduction

Collaborative geovisualization provides effective means to communicate spatial information among a group of users for sharing knowledge and information. This kind of communication occurs in a variety of applications such as public participation in planning projects, city management (i.e., complaint management), security monitoring or disaster management. To enable a comprehensible, potentially time-shifted communication of spatial information a user should be able to create, store, display and analyze *geospatial annotations* as pieces of information that are connected to geospatial objects, structures or regions. These annotations can represent, e.g., opinions, remarks, hints, explanations, or questions regarding a spatial subject. Contents and spatial references of annotations should be as flexible as possible to allow users to precisely, directly, and efficiently express their thoughts. Beside textual and multimedia contents, we propose free-hand sketches as expressive type of annotation for visually communicating fuzzy, sketchy or vague information. Using sketches, for example, feature arrangements or change proposals in planning scenarios can be effectively communicated.

To provide a common understanding of geospatial annotations, a model is required that is general enough to serve as basis for data integration into heterogeneous service-based software systems and applications. Especially the definition of a model for an annotation's *spatial reference* is important to prevent loss of information concerning the annotation's spatial subject. Such spatial references are typically specified explicitly using tools provided by an annotation authoring system to avoid non-georeferenced, purely textual descriptions of spatial subjects that may lead to ambiguities. The comprehension of such descriptions depends on a user's context like skills or current tasks (Cai et al., 2003). Explicit specification using georeferenced geometry obviates the use of specialized language to draw a reader's attention to an annotation's spatial subject (Hopfer and MacEachren, 2007).

Our annotation model is designed for 3D geovirtual environments (3D GeoVE) such as 3D virtual city and landscape models. In this paper we assume in the following an urban area as the scope of a collaboration. Simple 2D geometries are not fully sufficient for describing an annotation's subject geometry due to the nature of features in such areas. For example, underground structures or indoor references for certain parts of a building cannot be expressed unambiguously using 2D geometry as spatial reference. Our annotation definition and implementation uses 3D georeferenced geometries for spatial reference specification. The unambiguously specified spatial reference geometry is particularly important to enable automated analysis of larger amounts of annotations using spatial parameters. Using our annotation model, for example, to gather and afterwards manage and visualize annotations in a public participation scenario, such analysis can help to improve the process of evaluation and processing of issues expressed by annotations.

Besides supporting a clear and flexible specification of an annotation's spatial reference, our model supports capturing the *creation context* of an annotation. The collaboration context includes metadata such as creation time and author information but also the author's 3D view on model data visualization. This view bears information that helps a later reader to comprehend the meaning of an annotation.

The purpose and applicability of geospatial annotations is widespread. They may be used, for example, to collect information concerning urban planning scenarios for public participation purposes or for persisting agreements on problem solving during remote or local meetings using a virtual 3D city model. Such annotations can afterwards help to review findings and therefore help to recall key aspects of a collaborative work process (Shrinivasan and van Wijk, 2009). Annotation data created during collaboration processes must be widely usable in heterogeneous software environments. When using the same open and standardized data encoding and service interface that is used for geodata itself, annotation functionality can be embedded into a variety of applications that are already capable of dealing with such data.

In this paper we introduce an object oriented model of geospatial annotations in connection with its implementation using the *Geography Markup Language* (GML) (Portele, 2007) as data exchange format between a transactional *Web Feature Service* (WFS-T (Vretanos, 2005)) and clients creating and visualizing annotation data in 3D geovirtual environments. For this purpose an annotation's spatial references are modeled as distinct objects describing 3D geometries. By doing so, those reference objects can be shared throughout annotation objects to explicitly share spatial references.

The rest of this paper is organized as follows: Section 2 provides a short overview of related work. Section 3 introduces our model of geospatial annotations. The design and implementation of the collaborative annotation system is presented in Section 4. A short discussion including the limitations of our approach is given in Section 5. Section 6 summarizes the paper and proposes some additional research directions to take.

2 Related Work

Schill et al. (2008) introduce in the context of the Virtual Environment Planning System project (VEPs) a model of geospatial comments for public participation in urban planning projects, using GML for data encoding and an OGC Web Feature Service for storage and retrieval. Text is used as annotation contents, and object URLs for each annotation can be stored to reference multimedia objects. An annotation's spatial reference is modeled as point, which is interpreted differently depending on the type of the annotation. An identifier of a parent annotation can be set to create annotation chains as discussions. The approach is limited regarding the definition of multiple objects, for example feature groups, or more complex geometries as spatial reference for annotations.

An interactive geocollaboration framework supporting geographic annotations is introduced by Mittlböck et al. (2006), which combines data from heterogeneous sources for presentation and analysis. A user is able to vote and to comment on geospatial subjects visualized by maps. Annotations are georeferenced using 2D coordinates. As real-time visualization component Google Earth[1] is used. Unlike our implementation, a separate service combines data from different sources (for example WFS and WMS) for generating output of annotation data in KML (Wilson, 2008) format.

Several researchers worked on supporting geo collaboration using maps. Yu and Cai (2009) propose *GeoAnnotator* as a service-oriented system for map based public participation. They outline requirements of such a system to provide necessary features for annotation of geospatial objects as well as for encouraging people to provide their opinions. A many-to-many relation between annotations and spatial references is considered to be important to support, e.g., comparison arguments as annotated information. Further they outline the need for multi-modal multimedia annotations to support sharing geographical information more easily. Rinner (2001, 2005) introduced Argumentation Maps to support discussions on planning activities by connecting discussion contributions to geographic features or geometries. This object-based model is used to store discussion information in

[1] http://earth.google.com

databases. In contrast to Argumentation Maps, our model aims at a more general approach for annotation of geographic areas and features, which can be used in many application domains.

Hopfer and MacEachren (2007) investigate the use of geospatial annotation for collaboration using map-based displays, analyzing how annotations facilitate decision making in groups. They recommend to avoid introducing knowledge that is already known to each participant (shared knowledge) into decision making processes and outline the importance of flexible annotation systems (query, analysis and access possibilities).

Text and sketch annotations in 3D virtual environments for architectural design are presented by Jung et al. (2002). They outline the demand for non text annotations in an earlier user study Jung et al. (2002a).

Tohidi et al. (2006) report on the usage of user created sketches during user interface design processes. They state the advantage of providing a user with communication means to propose own ideas or proposals beside, e.g., textual comments or questionnaires. Sketches are also used frequently for describing intentions in the field of human-computer-interaction, e.g., for navigation (Igarashi, et al., 1998, Hagedorn and Döllner, 2009) or 3D modeling (Karpenko and Hughes, 2006). We are using a sketch-based approach for communicating visual information to provide equally expressive communication tools, which allow more useful annotations, i.e., to express alternate approaches or change requests.

Heer et al. (2009) deal with asynchronous (time shifted) collaboration on data visualization using annotations. They provide tools for diagram annotation. Additionally they conducted a user study to analyze the usage of these tools. Drawing sketches on top of the visualization is seen as expressive means especially for pointing: It turned out that 88.6 % of all sketch annotations involved pointing. In contrast to our drawing approach more tools are provided for drawing complex shapes like arrows or boxes, while our client implementation does exclusively support free-hand sketching.

Isenberg et al. (2009) conducted a user study on usability of a collaboratively retrofitted information visualization system. They introduced collaborative interaction and did changes concerning the data representation to enable collocated collaborative work. One improvement requested by several participating groups was to integrate explicit ways to ensure that decisions would not get lost in the collaboration process, which is also motivation for annotation in collaborative processes in 3D GeoVEs.

3 Modeling Geospatial Annotations

This section presents a model for geospatial annotations that concentrates on precise, creation context aware storage of information concerning a spatial subject. Annotations are used as means to make knowledge or information persistent and are intended for later access and analysis. We distinguish three types of annotations by their contents: textual information, multimedia contents (e.g., images, videos, or audio records), and additional geometry visually communicating a concept or proposal.

3.1 Spatial References

Spatial references define the location and extent of an annotation's subject in 3D space. To ease sharing of those between annotation objects, we model spatial references as separate first-class features, which does also allow us to define groups of spatial references to be the subject of an annotation. By marking an annotation's spatial subject area using our model of spatial reference, specialized language to communicate the spatial reference in annotation contents can be obviated (Hopfer and MacEachren, 2007).

Our model for spatial references is partitioned into two parts (Fig. 1): `SpatialReference` and specialized reference types. The `SpatialReference` class defines basic parameters, which every reference type must have. A point as location indicator facilitates using an annotation's spatial references for clients that have very limited capabilities concerning computational power, display size, bandwidth or geometry handling. This especially eases the implementation of web-based clients for annotation exploration and creation, without having to implement the full support for GML geometry needed for precise and complete handling of complex reference geometry. The `modelId` attribute identifies the model data set in the database. This data set describes parameters of the city model that is used to create the `SpatialReference` object. The information about the used model can be retrieved from the WFS if information about the overall spatial extend or additional information like access parameters for model data are required.

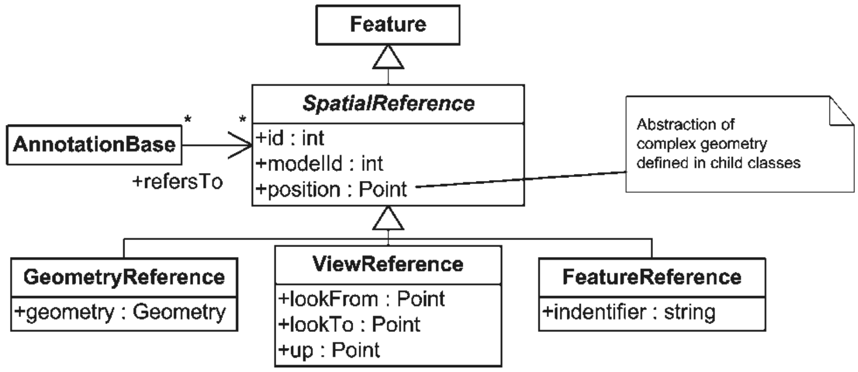

Fig. 1. Reference types as UML class diagram. Every geographic reference is a unique feature which can be referred

We define the following three types of spatial reference objects (Fig. 1):

- **Geometry**: This is the most explicit type of spatial reference. It contains geometry (e.g., point, polygon or box) defined in real-world coordinates. It is encoded using the GML 3 geometry model, which supports 3D geometries. The point geometry defining the approximate location is set depending on the type of geometry the references holds. If the geometry is a point, it is set as position property equally. For lines or line strings the center of the line, defined by the client creating the reference, is used as position marker. For areas or volumes the center of the bounding box is used to provide the value of the position property.
- **Scene Views:** A scene view is the second type of an annotation's spatial subject. A large amount of information is included in a user's current view of the scene through many perceptional impressions like the current line-of-sight or visible parts of certain structures are view dependent. A `ViewReference` instance is specified by three point properties: look-from position, look-to position and up-position. The up-position determines in conjunction with the look-from position camera's up direction. The look-from position also defines the position property of this type of spatial preference.
- **Geographic Features:** In contrast to references containing explicitly defined complex geometries, a reference to a model object is connected to a geographic feature (e.g., building, square, or street). This provides possibilities to use topological, hierarchical and other relations defined by the city model for computation like positioning calculation for annotation visualization elements or further analysis of larger numbers of annotations. Large amounts of annotations can occur, e.g., in public participation scenarios or planning activities. The indirection of those

references allows us to follow changes in the feature geometry providing the possibility to, for example, annotate features that do not have a fixed location. A `FeatureReference`, therefore, defines a link to a feature data set included in a city model. The `identifier` string in connection with the `modelId` identifier must enable a client to retrieve the complex geometry from the data source defined in the model description. An example for such an identifier is a URI used as gml:id attribute value in a GML-based city model. At least the client creating this type of spatial reference must be capable of getting feature data from the model to calculate the position property. Other clients can use the precomputed position property instead. By default the position property is set to be the center of the referenced feature's bounding box.

•

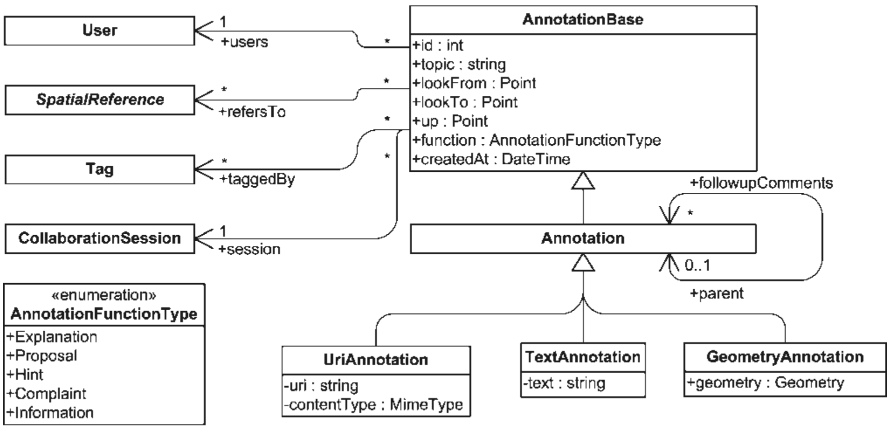

Fig. 2. UML class diagram for geospatial annotations. Metadata like geospatial annotation subjects or the author of an annotation is defined at the base class. The Annotation class adds the possibility to define annotation chains for discussions

3.2 Annotation Contents

An annotation's content defines the information associated to the spatial reference. To provide the users with a wide range of possibilities to express their opinions, remarks, or proposals we define three types of annotations according to their type of contents (Fig. 2):

- **Text:** A user can state opinions or other information by giving textual descriptions. Because of the well defined spatial references, users may refer to those objects easily. Although being quite expressive, text is not the optimal means for communicating information that refers to spatial relations.

- **References to Multimedia Contents:** This annotation type enables a user to connect multimedia contents to a spatial reference. The contents themselves are not stored together with the annotation data but are referenced using an URL. To support clients to handle the playback or display of the linked contents, information about the type of the referenced media is stored (see `contentType` property). Through using this quite flexible form of annotation contents, a wide range of media (e.g., audio recordings, videos, or images) can be associated with spatial references.
- **Geometry:** The third type of annotation contents is either 2D or 3D geometry that is used to annotate the city model by using direction indicators (i.e., arrows), measurement indicators, sketches, or extensions to existing object geometries like, e.g., lines as proposal for routes (Strobl, 2007) (Fig. 3). Those geometries are means to communicate spatial information like object arrangement, object size or design ideas. The communication of such visual forms of information through non visual (verbal, textual) means involves a loss of information due to the necessary mental translation effort (Yao, et al., 2005). To help to avoid such a translation loss, we allow the creation of free-hand sketches as special case of geometry annotation. A sketch is an intuitive and efficient way for communicating information or concepts (Stefik, et al., 1987). The user is free to express its own concepts or proposals. Due to the creative freedom a resulting sketch serves as a basis for later analysis and interpretation (Tohidi, et al., 2006), which may help to improve planning.

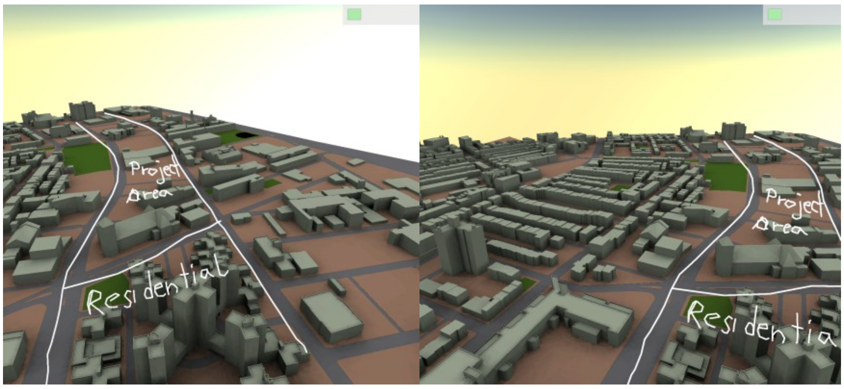

Fig.3. An example of view-plane sketches for communicating proposals, ideas, or spatial relations. The sketches are connected to one viewpoint, but the camera orientation can be changed while the sketch's position is maintained

3.3 Expressing Uncertainty for Spatial References

If there is no precisely definable subject geometry for an issue, further means for expressing spatial vagueness are needed. Imprecisely known subject geometry can be necessary, e.g., when assumptions or guesses concerning spatial issues shall be made. Our concept of an annotation's uncertainty extent provides means for specifying further spatial attributes than the spatial reference as annotation subject only. Annotation's contents may refer to this geometry to express an alternative concerning a spatial extent. An extent geometry can be defined in two ways:

- Indirectly by using an offset given in meters which enlarges the spatial reference geometry
- Directly through defining a separate explicit extent geometry

By taking the geometry specified by the annotation extent into account for search and analysis, the scope of a search request can be broadened to include annotations that are possibly related to the geometry defined as search parameter.

3.4 Annotation Metadata

Basic annotation attributes describe metadata concerning the annotation's contents. They can help to comprehend the author's original intention and message when annotations are explored. The following 6 items are stored alongside with every annotation for that purpose:

- **Scene View Specification:** The parameters describing an author's current scene view are stored together with annotation data providing the reader with information about the creation context. When the annotation was created using an interactive 3D client, we assume the author chose a viewpoint in such a way that objects that are important for understanding the spatial situation are visible and properly aligned concerning the message that is intended to be communicated.
- **Annotation Function:** As shown in Fig. 2 each annotation can have a function assigned that describes what the author has intended to express. The categorization, which is possible through this function attribute, can be used for annotation visualization and analysis. E.g., where and how many complaints or proposals have been given as annotations to identify problematic areas. By now, the annotations functions have been defined exemplarily for the use case of public participation in urban planning or city management scenarios. They may have to be adapted or extended to

serve for other application areas. The function is also a good criterion for grouping of annotations especially for visualization purposes.

- **Session:** Annotations can be grouped by sessions that describe the occasion for annotation creation, i.e., a team meeting or a planning project. An annotation's session does also describe the geographical extent of the overall area of interest using a bounding box. A model description associated with a session holds information about the model that is used for annotation authoring. A session provides a short description of the overall topic (e.g., project name or activity description). By assigning a session id to an annotation, they are assigned to a session dataset.
- **Tags:** Keywords (*Tags*) can be assigned by a user to briefly describe what an annotation is about. Through using tags for annotation description groups are created, each containing annotations that hold the same tag. A user is free to assign arbitrary unstructured keywords to his annotations. The keywords may define a broad range of annotation attributes like, e.g., contents or intended function. The meta-information provided by such a keyword set per annotation can be used for searching or filtering of annotation objects (Xu et al., 2006).
- **Author:** An annotation holds information (e.g., id, name, color) about its author to enable to tracing of annotations created by a certain user or a group of users which matches given parameters.
- **Discussion:** Parent-child (`followupAnnotations`) relations introduced by the Annotation class definition in Fig. 2 allow to create tree structures of annotations. Those structures can be used to model discussion threads. The spatial reference of a parent annotation is implicitly inherited by child annotations, which do not have to have a `SpatialReference` object assigned. This also provides the possibility to define further `SpatialReference` objects to broaden the spatial scope of a follow-up annotation. Since we can model discussion threads using our geospatial annotation model it can be applied for issue-based information systems (Kunz and Rittel, 1970) for applications like map-based argumentations (Rinner, 2005).

4 Implementation

This section provides an overview of our prototypic implementation of a system that supports authoring, storage and visualization geospatial annotations using 3D GeoVEs.

4.1 System Architecture

Our system architecture consists of three major parts (Fig. 4): A geo-enabled database management system, a WFS implementation, and client applications. The data back end is provided by a PostGIS[2] spatially enabled PostgreSQL database which is encapsulated using a transactional WFS-implementation. This supports usage of the annotation data by a variety of client applications through offering an open, standardized interface for data retrieval, insertion and update. OGC Filter (Vretanos 2005), as query language for WFS requests, allows to define queries using spatial and non spatial predicates for restricting search requests in a standardized way.

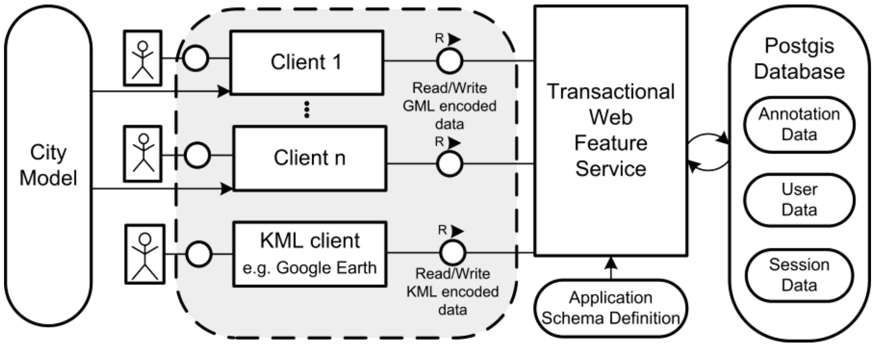

Fig. 4. Architectural overview of the annotation system. The client is responsible for user interaction, visualization and conversion to the respective GML-based data encoding while the WFS provides the interface for data storage using the PostGIS database. Through the standard WFS interface and its variable output format different types of clients are possible

Several types of clients may be used together with our data back end. We have implemented a C++ application that can create and visualize annotations for virtual 3D city models. It uses GML for encoding of annotation data. For annotation exploration, a KML enabled client like Google Earth can be used to visualize annotations by using the KML encoded output generated by the WFS implementation through transforming the GML encoded output to KML using XSL transformations.

The city model must be shared throughout all clients creating annotations to be able to create and resolve `FeatureReferences`, e.g., through distributing the model data file throughout all client instances.

[2] http://postgis.refractions.net

Another possibility for data distribution is a service-based access to a CityGML (Gröger, et al., 2008) encoded model also using a WFS.

4.2 Data Storage

Data retrieval, creation and update is encapsulated by a transactional WFS. Due to our requirements regarding support for 3D geometries and transactional service functionality (insert, update and delete features), we use a service implementation that supports the WFS specification version 1.1., which defines GML version 3.1.1 as mandatory data exchange format and allows encoding of 3D geometries. Since annotation data creation and update of data properties are inevitable functionalities for our purposes, a WFS implementation is used which supports the *Transaction* operation, defined by the WFS specification to be optional.

We selected the WFS implementation of the deegree project[3] version 2.2 to be used with a PostGIS spatially enabled database. The WFS is configured using a GML application schema that defines XML elements and XML Schema types according to our model presented in section 3. In the schema definition we made extensive use of complex typed child elements to implement our annotation model. Each complex typed element is mapped onto a database table in our relational database model. Inheritance relations (e.g., the one between `SpatialReference` and `GeometryReference` depicted in Fig. 1) are defined in both, the XML schema definition using the type extension mechanism and the relational database model using the table inheritance feature of PostgreSQL. All XML schema types that extend other types are mapped to an own database table that inherits from the table of the extension's base type. This eases a consistent handling of child types regarding id generation, feature update, or deletion behavior down to database level.

Data Export as KML

A system for annotations should enable the largest possible public to use it (Strobl, 2007). Virtual globe tools like Google Earth, Microsoft's Bing Maps 3D[4] or NASA World Wind[5] are very popular and many web users are familiar with using such virtual globes. All those applications mentioned support KML as input or output format. We defined a XSL trans-

[3] http://www.deegree.org
[4] http://www.bing.com/maps
[5] http://worldwind.nasa.gov

formation that is used as output Filter for our WFS instance to enable annotation exploration via afore mentioned KML enabled clients (Fig. 4). It translates annotation data encoded in GML according to our application schema into KML placemarks. Therefore the position property held by `SpatialReference` instances provides a placemark's location. Equally the definition of an input transformation from KML to our GML dialect would be possible to allow creation of annotation features using KML-encoded input data.

4.3 Interactive 3D Client

For annotation authoring and visualization we have implemented a C++ client application that uses the GML-encoded data provided by the WFS. It provides a user interface for exploring of a 3D city model. Users are enabled to define spatial references visually by selecting features or defining reference geometries. Annotation metadata is captured implicitly on annotation creation.

Annotations are visualized using 3D display elements, which are embedded into the scene and additionally through icons on a mini map. Annotations are grouped by their spatial references to limit the amount of 3D annotation items being displayed. Different interaction and visualization strategies have to be applied depending on annotation types. Sketches concerning scene views can be created as `GeometryAnnotation` using the mouse or a tangible display. To view such sketches the user takes the author's camera position and sketch geometry is displayed front of the user's viewpoint. This creates the impression of the sketch being projected into the scene (see Fig. 3).

The process for annotation creation has to assure data integrity, which also includes avoidance of double entries, especially for `SpatialReference` instances, where possible. Partly this is ensured by the deegree WFS implementation by checking input data for validity according to the application schema. It does also check for doublets using predefined equality criteria for feature types. Unfortunately, the check for duplicates does not take complex typed child properties and geometry properties into account. So this has to be implemented in the client application. We avoided this missing feature by creating the `SpatialReference` features independently before creating the annotations objects.

5 Discussion

Currently the client annotation-system is implemented to prove the applicability of our annotation model. No larger user tests have been done right now. In the following we will discuss the current client/server-system and model.

Screen overlay sketches are currently defined using a set of curves given in 3D real world coordinates which have been calculated depending on near clipping plane of the client's current camera projection settings. For it, screen coordinates of each point of the sketch are unprojected to 3D coordinates situated on the camera's view plane. By saving those coordinates in that way, we enable a camera look around while maintaining the alignment of the sketch in connection to the city model scene (see Fig. 3). While this is sufficient for a fixed viewpoint in connection with sketches, it is not possible to display sketches that are associated with objects or certain areas independently from viewpoints. Sin et al. (2006, 2006a) provide possibilities to use object surfaces as sketch canvas. They adjust the sketch display depending on the orientation and viewpoint dependence of the information contained in such sketches.

Our representation of overlay sketches and camera viewpoints defined for annotation features are based on GML geometries. From a semantic point of view the geographical bounding box of those features should be the bounding box of their spatial reference. Unfortunately this is not possible at the moment because the WFS takes every geometry property of a feature into account for bounding box calculations. This falsifies the bounding box of such features. The bounding box definition is also a problem for view references and other types of annotations because of the definition of a camera position using three georeferenced points. Those points are also contributing to the bounding box calculation of the deegree WFS.

The client introduced in this paper provides basic functionality to explore a city model, to select (also multiple) spatial references and to visualize annotations in 3D scenes. The methods for alignment and display of such metadata elements are not subject of this paper. But there are some restrictions concerning the usage of our annotation model's expressiveness. For example the ability to define a `GeometryReferences` is limited for simplicity to points, boxes, and 2D polygons. Also the client is not yet able to create or visualize annotation chains (discussions).At the moment KML output is restricted to a placemark for each annotation which are localized using the position property of the `SpatialReference` type. Also `GeometryAnnotation` objects are not covered by the XSL transformation by now. Complex reference geometries could be translated into KML geometries using more sophisticated transformations. Those would have to include additional functionalities implemented in Java classes providing for example coordinate transformations or other computations. This would also enable geometry processing for `GeometryAnnotations` and enable sketch display in KML enabled 3D clients through KML-encoded complex geometries.

The types of annotation content are defined exemplarily. All metainformation that is needed for annotation handling are defined independently from the type of the information contained in an annotation. This way the model can be extended towards, e.g., composite types of annotation contents. For each new type of annotation content client applications have to be adapted to enable authoring and visualization of those new content types.

6 Conclusions and Future Work

In this paper we have shown a model for geospatial annotations that pays special attention on modeling the data structures for referencing geospatial objects or geometries using GML-features. A transactional WFS provides a standardized interface for data access and storage, which allows embedding annotation functionality into a variety of GML compatible applications. Metadata supports comprehension of annotation meanings especially with regard to 3D GeoVEs. We further propose an annotation type containing geometry as intuitive communication tool. We have outlined the possible value of sketches for communication of proposals or ideas.

Concerning geospatial annotations some further work may be applying geospatial ontologies for defining a more precise and semantically valuable georeferencing. The additional information provided by such an ontology may be used for a more subject specific visualization (arrangement, appearance) when interactive 3D client applications are used. Further, additional annotation types with regard to their contents could be defined. For example a type for representing questionnaires for guided data acquisition by users could be evaluated in connection with virtual tours through the area of interest.

We do not use a definition of what equality of spatial references mean, except for equal-valued object attributes. Defining variable criteria concerning the degree of containment of a reference's geometry in other reference geometries could help to visualize and analyze larger amounts of annotations.

A web based approach for interactive creation and visualization of annotations would lower barriers that are posed through installation requirements and dependencies of the current client system. Therefore other OGC web services may be used to provide map-based (Web Map Service - WMS (de la Beaujardiere, 2006)) or 3D visualization and interaction. For the three dimensional case, a *Web Perspective View Service* (WPVS) could be used to generate images of a virtual city model. Using such images for presentation, very thin client applications are possible without losing the possibility to give an impression of the author's context when creating the annotation (Hagedorn, et al., 2009). When additional interactivity is introduced using image data delivered by a WPVS, users would be able to define all necessary attributes for annotation creation using a web browser. Such a technique could enable the usage of annotation with precise geospatial references for a far broader audience and therefore enable using our annotation model in connection with a virtual 3D city or landscape model for large scale public participation applications.

An important next step is the acquisition of a larger amount of user-generated data to find requirements for annotation analysis; user tests should show how users can handle annotation tools and which kind of information they want to store as annotation.

References

de la Beaujardiere, J. (2006) OpenGIS® Web Map Service (WMS) Implementation Specification 1.3. *Opengeospatial Consortium.*. http://www.opengeospatial.org/standards/wms (accessed 11 05, 2009).

Cai, G., Wang, H., and Maceachren, A. M. (2003) Communicating Vague Spatial Concepts in Human-GIS Interactions: A Collaborative Dialogue Approach. Proceedings of the Conference on Spatial Information Theory, COSIT 2003, Springer. 287-300.

Gröger, G., Kolbe, T. H., Czerwinski A. and Nagel. C. (2008) OpenGIS® City Geography Markup Language (CityGML) Encoding Standard. *Opengeospatial Consortium*. http://www.opengeospatial.org/standards/citygml (accessed 11 05, 2009).

Hagedorn, B. and Döllner, J. (2009) Sketch-Based Navigation in 3D Virtual Environments. *it - Information Technology 03/2009*, 163-170.

Hagedorn, B., Hildebrandt, D. and Döllner, J. (2009) Towards Advanced and Interactive Web Perspective View Services. Developments in 3D Geo-Information Sciences. Springer: 33-51

Heer, J., Viégas, F. B. and Wattenberg, M. (2009) Voyagers and voyeurs: Supporting asynchronous collaborative visualization. *Communications of the ACM* (ACM) 52 (2009): 87-97.

Hopfer, S., and MacEachren, A. M. (2007) Leveraging the potential of geospatial annotations for collaboration: a communication theory perspective. *Int. J. Geogr. Inf. Sci.* (Taylor & Francis Inc.) 21 : 921-934.

Igarashi, T., Kadobayashi R., Mase, K., and Tanaka, H. (1998) Path drawing for 3D walkthrough. *UIST '98: Proceedings of the 11th annual ACM symposium on User interface software and technology.* ACM. 173-174.

Isenberg, P., Carpendale S., Bezerianos, A., Fekete, N., and Henry, J. (2009) CoCoNutTrix: CollaboartiveRetrofitting for Information Visualization. *IEEE Computer Graphics and Applications*: 44-57.

Jung, T., Gross, M. D., and Do, E. Y. (2002) Annotating and Sketching on 3D Web Models. *IUI '02: Proceedings of the 7th international conference on Intelligent user interfaces.* New York: ACM. 95-102.

Jung, T., Gross, M. D., and Do, E. Y. (2002a) Sketching Annotations in a 3D Web Environment. *CHI '02: CHI '02 extended abstracts on Human factors in computing systems.* New York: ACM. 618-619.

Karpenko, O. A. and Hughes, J. F. (2006) SmoothSketch: 3D free-form shapes from complex sketches. *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers.* ACM, 589-598.

Kunz, W. and Rittel, H. W. J. (1970) Issues as Elements of Information Systems, Working Paper No. 131, Studiengruppe fur Systemforschung, Heidelberg (July 1970)

Mittlböck, M., Resch, B., and Eibl, C. (2006) geOpinion: Interaktives geo-Collaboration Framework - 3D-Visualisierung in Google Earth mit OGC WMS- und WFS- Diensten. *Angewandte Geoinformatik 2006. Beiträge zum 18. AGIT-Symposium.* Salzburg: Herbert Wichman Verlag, Heidelberg 464-469.

Portele, C. (2007) OpenGIS® Geography Markup Language (GML) Encoding Standard. *Opengeospatial Consortium.* http://www.opengeospatial.org/standards/gml (accessed 11 05, 2009).

Rinner, C. (2001) Argumentation maps: GIS-based discussion support for on-line planning, *Environment and Planning B: Planning and Design*, vol. 28, 2001, 847-863.

Rinner, C. (2005) Computer Support for Discussions in Spatial Planning**.** *GIS for Sustainable Development*. Taylor & Francis, 167-180

Schill, C., Koch, B., Bogdahn, J., and Coors, V. (2008) Public Participation Comment Markup Language and WFS 1.1. *Urban and Regional Data Management.* Taylor & Francis. 85-92.

Shrinivasan, Y. B., and van Wijk, J. J. (2009) Supporting Exploration Awareness in Information Visualization. *IEEE Computer Graphics and Applications*: 34-43.

Sin, E., Choy, Y., and Lim, S. (2006) A Study on Sketch Input Technique by Surface of 3D Object for Collaboration. *Proceedings of the 2006 International Conference on Hybrid Information Technology.* IEEE Computer Society: 623-628.

Sin E., Choy Y., and Lim, S. (2006a) Content based sketch annotations for collaboration. *SIGGRAPH '06: ACM SIGGRAPH 2006 Research posters.* ACM. 21.

Stefik, M., Bobrow, D. G., Foster, G., Lanning, S., and Tatar, D. (1987) WYSIWIS Revised: Early Experiences with Multiuser Interfaces. *ACM Trans. Inf. Syst.* (ACM) 5 : 147-167.

Strobl, J. (2007) Visual Interaction: Enhancing Public Participation ? Edited by Strobl Buhmann, Ervin. *Trends in Knowledge-Based Landscape Modeling Proceedings at Anhalt University of Applied Sciences 2006.*.

Tohidi, M., Buxton, W., Baecker, R., and Sellen, A. (2006) User sketches: a quick, inexpensive, and effective way to elicit more reflective user feedback. *NordiCHI '06: Proceedings of the 4th Nordic conference on Human-computer interaction.* ACM. 105-114.

VEPs. VEPS project website. http://veps3d.org (accessed 11 05, 2009).

Vretanos, P. A. (2005) OpenGIS® Web Feature Service (WFS) Implementation Specification. Vers. 1.1. *Opengeospatial Consortium*. http://www.opengeospatial.org/standards/wfs (accessed 11 05, 2009).

Wilson, T. (2008) OGC® KML. *Opengeospatial Consortium.* http://www.opengeospatial.org/standards/kml (accessed 11 05, 2009).

Xu, Z., Fu, Y., Mao, J., and Su, D. (2006) Towards the Semantic Web: Collaborative Tag Suggestions. *Proceedings of Collaborative Web Tagging Workshop at 15th International World Wide Web Conference*.

Yao, J., Fernando, T., Tawfik, H., Armtiage, R., and Billing, I. (2005) A VR-centred Workspace for Supporting Collaborative Urban Planning. *Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design.* 2005. 564-569.

Yu, B., and Cai, G. (2009) Facilitating participatory decision-making in local communities through map-based online discussion. *Proceedings of the 4th international Conference on Communities and Technologies.* ACM, 2009. 215-224.