Integration of Fuzzy ERD Modeling to the Management of Global Contextual Data

Gregory Vert and S.S. Iyengar

Abstract. This chapter introduces the idiosyncrasies of managing the new paradigm of global contextual data, sets of context data and super sets of context data. It introduces some of the basic idea's behind contexts and then develops a model for management of aggregated sets of contextual data and proposes methods for dealing with the selection and retrieval of context data that is inherently ambiguous about what to retrieve for a given query. Because contexts are characterized by four dimensions, those of time, space, impact and similarity they are inherently complicated to manage.

This work builds on previous work and extends that work to incorporate contexts. The original model for spatial-temporal management is presented and then analyzed to determine much coverage it can provide to the new context paradigm.

Introduction to the Idea of Context

The concept of context has existed in computer science for many years especially in the area of artificial intelligence. The goal of research in this area has been to link the environment a machine exists in to how the machine may process information. An example typically given is that a cell phone will sense that its owner is in a meeting and send incoming calls to voicemail as a result. Application of this idea has been applied to robotics and to business process management [1].

Some preliminary work has been done in the mid 90's. Schilit was one of the first researchers to coin the term context-awareness [2,3]. Dey extended the notion of a context with that of the idea that information could be used to characterize a situation and thus could be responded to [4]. In the recent past more powerful models of contextual processing have been developed in which users are more involved [5]. Most current and previous research has still largely been focused on development of models for sensing devices [6] and not contexts for information processing.

Gregory Vert and S.S. Iyengar
Center For Secure Cyber Security
Louisiana State University
Baton Rouge, LA 70803
e-mail: gvert12@csc.lsu.edu

J. Kacprzyk et al. (Eds.): Uncer. Appro. for Spatial Data Model. and Process., SCI 271, pp. 155–173.
springerlink.com

Little work has been done on the application of contexts to that of how information is processed. The model that we have developed is that of creating meta-data describing information events and thus giving them a context. This context then can be used to control the processing and dissemination of such information in a hyper distributed global fashion. The next section will provide a very general overview of the newly developed model and how contexts are defined. The following section will give an overview of the fuzzy ERD model that previously developed could be used for management of contextual information. Finally, the model is evaluated to determine what level of coverage it may provide as it is for management of global contexts data.

Global Contextual Processing

To understand the issues connected with security models for contexts we introduce some details about the newly developing model for contextual processing.

Contextual processing is based on the idea that information can be collected about natural or abstract events and that meta information about the event can then be used to control how the information is processed and disseminated on a global scale. In its simplest form, a context is composed of a feature vector

$$F_n<a_1,..a_n>$$

where the attributes of the vector can be of any data type describing the event. This means that the vector can be composed of images, audio, alpha-numeric etc. Feature vectors can be aggregated via similarity analysis methods into super contexts. The methods that might be applied for similarity reasoning can be statistical, probabilistic (e.g. Baysian), possibilistic (e.g fuzzy sets) or machine learning and data mining based (e.g. decision trees). Aggregation into super sets is done to mitigate collection of missing or imperfect information and to minimize computational overhead when processing contexts.

definition: A context is a collection of attributes aggregated into a feature vector describing a natural or abstract event.

A super context is described as a triple denoted by:

$$S_n = (C_n, R_n, S_n)$$

where C is the context data of multiple feature vectors, R is the meta-data processing rules derived from the event and contexts data and S is controls security processing. S is defined to be a feature vector in this model that holds information about security levels elements or including overall security level requirements.

definition: A super context is a collection of contexts with a feature vector describing the processing of the super context and a security vector that contains security level and other types of security information.

Data Management of Contexts

Having examined contexts, what they contain and how they can be analyzed, it becomes clear that the data management issues of contexts are not readily solved by traditional approaches. Data management consists primarily of the simple storage of information in a way that the relationships among the entities is preserved. Due to the fact that a context can really be composed of any type of data ranging from binary to images, to narratives and audio there is a need for a new model for storage of context data that can handle widely different types of data. Additionally, data management involves the issues of correlations between related types of data. As an example, context C1 may be very similar to context C13-C21 for a given event, thus they should be included in the process of analysis and knowledge creation operations. Related to this idea is that similarity in contexts also is the driving force in the how and what of which contexts are retrieved for a given query.

With the above in mind, there is a need to examine how contextual data might be managed in a previously defined fuzzy data model developed by this author [12]. This model presents an architectural overview of how an original model was developed using fuzzy set theory to manage storage and ambiguous retrieval of information. The elements of the model are presented and how it functions is described. The first part of the next section presents an argument for a new type of paradigm of how data should be thought of, that of the Set model. Problems with this new way of thinking about data organization are then discussed as a beginning for discussion of solutions to the problems of Sets. The section then continues on to discuss a new method of modeling sets that gives the the model an ability to store, manage and retrieve any type of data currently existing and any type of data that may be created in the future. Finally the section presents concepts about how the overlap problems with Sets that create ambiguity in retrieval can now be addressed with new operators based on fuzzy set theory that can identify similarities in data based on time, space and contextual similarity and retrieve the best candidates to satisfy a given query.

Overview of Spatial Data and its Management

Spatial information science is a relatively new and rapidly evolving field. Because global contextual models are highly spatial in many aspects of their operation, including the dimensions of space and time, it is appropriate to look at the issues of context based data management in terms of how spatial data is managed.

Spatial data management systems are an integration of software and hardware tools for the input, analysis, display, and output of spatial data and associated attributes. These systems are being used across a broad range of disciplines for analysis, modeling, prediction, and simulation of spatial phenomena and processes. Applications of spatial data are diverse: natural resource management, traffic control and road building, economic suitability, geophysical exploration, and global climate modeling, to name just a few. In the case of contextual data

management, spatial data systems need to extended for a purposes of managing wide types of information such sensed images (i.e., aerial photos, satellite images) and to store data in a number of different raster and vector data structures. A contextual management system based on spatial data management principles may contain digital images, tabular survey data, and text among many other possibilities.

Current spatial data management systems have limitations. One of these is an inability to retrieve and present all the data relevant to a problem to the system user in an orderly and helpful manner. When, for example, a user wants to access information about a particular geospatial region or type of geospatial feature (e.g., tsunami distance and travel information), he or she selects what appears to be an appropriate data entity. This process can be frustrating and error-prone, because, typically, many data entities (maps, photos, etc.) contain information of that type and choosing among them, or even being able to view them all, is very difficult. Furthermore, there may be several data entities (e.g. maps, photos, sensor information) for the same area that have been collected and/or modified at different times for different purposes, and the scales of these maps may differ. Additionally, not all data related to a region may be stored in a database. Some of the data may be in files, scattered across computer systems hard disks.

As an example, a Tsunami has just occurred in the Indian Ocean where map of the coast lines indicate that the area is prone to tsunamis, it has been sensed by NASA from outer space, a ship's captain has noticed a telltale raise in the ocean around his boat and radioed this information to his shipping company and beach vacationers have noticed that the tide has receded dramatically. All of this information is stored somewhere and individually may not have a lot of comprehensive meaning to disaster relief personal in the countries surrounding the Indian Ocean. However as a whole, they clearly indicate a natural disaster with subsequent responses.

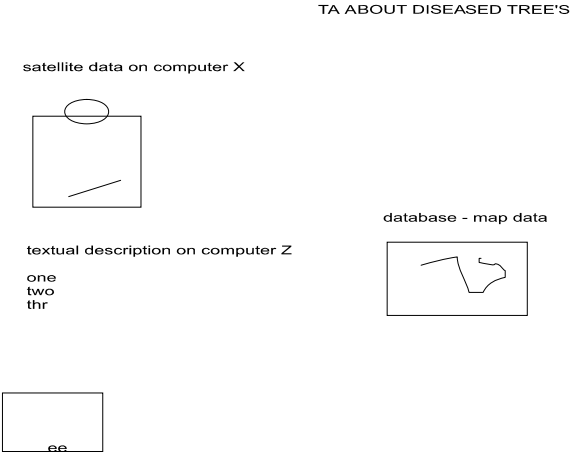

Figure 1. Distributed locations of information that collectively could be stored about a tsunami.

In the above example, a user may know to go to a database and retrieve data about one type of information about tsunamis. However, he or she may be unaware that other types of data not in the database are available that may provide useful and potentially critical information. Even if users are aware of other data, they may not be able to locate.

A goal then of context data management is to develop a way to manage all the type of data that can be found in a context. This can be done by aggregation of context data objects and related contexts into *sets* of data, rather than individual files describing one aspect of the event, in this case a tsunami. This approach can logically associate all related data for a type of event and select the appropriate contextual set based on user-supplied criteria.

The rest of this chapter describes provides an overview of previous work in this area and then a discussion of how contextual information might be stored in the previously defined model.

Context Oriented Data Set Management

Current approaches to data management assume that each individual piece of data, or data file must be managed. For example, in a relational database, as a mountain object would be stored in one row of the table containing mountains. Attributes of the mountain might be stored in a different table. In object-oriented methods, a mountain and its attributes might be stored in a single object. This approach works well in a homogeneous environment where all the data being managed are owned by the application that is managing it. Specifically, the application knows the format of the data it owns and thus manages each and every piece directly. However, this is not practical nor in most cases feasible in a heterogeneous environment that includes a multitude of different applications data. Applications cannot generally read and interpret each other's data. Nevertheless, while data may be for different applications and in different formats, it can still apply to the same geospatial region or problem. When this occurs, there is a basis for the creation of information about relationships among the data, but no mechanism to build the relation because of the differing formats.

This problem can be solved by a shift in the approach to how data is logically thought about and organized. Instead of attempting to manage individual pieces of data, e.g. mountain and attributes of mountains, which may be impossible in a heterogeneous data format environment, one can make the approach less specific, less granular. The key is to manage contextual data on the thematic attributes describing contexts, those of time, space and similarity. In this model specific data objects in a contexts feature vector are not managed they are organized into sets where the set is the lowest level of context data management.

The shift to set management for contextual information produces benefits that address other problems with managing data found in a context feature vector. Specifically, sets can be copies of a base contextual set. These sets can thus be lineages/versions of the base set. Once versions of context sets are established each set can become a particular view of the data included in a context. When views and versions become possible as a result of this approach, then so do

multiple information consuming entities with their own lineage trees and domains of control for the sets they define and own. Extending this concept, it is possible to see that multiple views serving multiple users does a very thorough job of addressing the previously user data coupling which is defined to be users modifying their own data and often working on overlapping spatial or temporal themes. Thus the benefits of this approach can have large a impact on a variety of problems. Because the set paradigm is data-format-independent, this is robust approach. The addition of new formats of data that can be described in a context as they are developed, will not cause the new approach to degrade as would current approaches. Instead, one simply adds the new-format data file to the set without any impact to the management and retrieval of such data.

Finally, a set management paradigm can introduce the problem of having multiple members in a set that covers the same geospatial region. This is referred to as ambiguity and was addressed through the application of fuzzy set theory to the metadata that manages the set abstraction. Fuzzy set theory can be used to make a generalized comment about the degree of possible membership a particular data file might have in a set covering a specific geographic region.

Contextual Set Ambiguity

Dataset ambiguity in contexts refers to the fact that for a given query or selection it may be impossible to select and exact match for the query because multiple sets of context may satisfy the query fully or partially. For example, if a query is interested in all data about an event located at a geographic point on the ground, multiple sets may have overlapping boundaries that the point can fall inside, thus the question becomes which set to return for a query. Another example can be found when one considers the spatial data in a context where the boundaries of objects are approximately known but not precisely known. For example if one is mapping the extent of the spreading wave of a tsunami, the edge of the wave and thus its boundary may be one meter wide or it may be considered to be hundreds of meters wide. The selection of information about the edge of the tsunami wave then becomes an ambigious problem. Because multiple contextual sets about a given tsunamis boundary may exist, perhaps one defines the edge to be one meter and the other for the same geographic location defines the edge to be 100 meters the question is which context sets data should be retrieved for a query. Because context have multiple dimensions, this problem can also exist for the temporal dimension of contextual sets and the similarity dimension. It also may exist for the impact dimension.

Ambiguity in contextual data sets impacts their use in a fairly significant fashion and has been studied for spatial data but not the additional dimensions of context set data. Contextual data sets (CDS) could be organized into large databases. Users of the database could then create spatial or geographic queries to the database to retrieve CDS data that is of interest to them based on geographic extent. This process of doing this is sometimes referred to as geographic information retrieval if the queries are for spatial data (GIR) [19]. GIR seeks to deal with spatial

uncertainty and approximation in the methods by which traditional spatial data is indexed and retrieved.

A key shift in the new model for CDS management is towards being less granular in the management of CDS data. Instead of managing geographic entities such as one might find in a GIR database, or for that matter the dimensional entities of temporality and similarity, the smallest unit of management in this approach is a single covering logical device that of a set for CDS data. This shift to being less granular has a variety of benefits, but it can introduce further ambiguity in selecting and defining sets with multiple overlapping coverage's. In this sense a coverage is the data found in a CDS that describes the dimension of space, time, similarity and impact. With this in mind, the new model defines ambiguity as condition where multiple tracts of CDS data may satisfy a given query for a particular geographic location, point in time, type of similarity or type of impact. When this is the case, the question becomes which set of data should be returned for a spatial query to retrieve the correct coverage?

To illustrate this point, consider the case where two contextual datasets have a coverage that contains the origin of a tsunami. One dataset contains is satellite imagery and the other is sensor information from the ocean. This is a "point in polygon" type of ambiguity problem. The center of the Tsunami is the point and a polygon is the rectangular bounding polygon of each spatial coverage. The expected ambiguity between trying to select between these coverage can be seen in Figure 1.

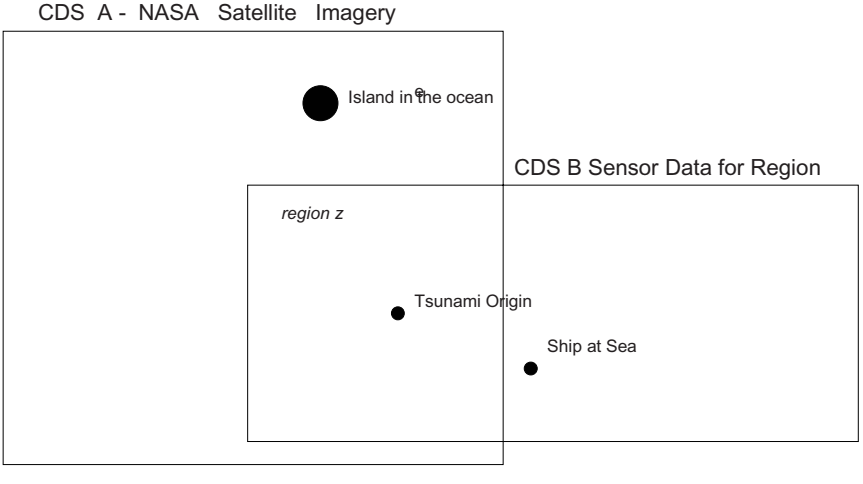

Fig. 1. Example of data set ambiguities for spatial coverage of the origin of a tsunami.

In the above example, the question is whether one wants CDS B or CDS A for a query about tsunami CDS data at time T_0. This is an example of ambiguous spatial data, and a point in polygon ambiguous problem.

Ultimately, the choice of which set to choose in an ambiguous problem should be left up to the analyst, or the person using the data. However, application of

fuzzy set theory and computational geometry can be applied to presenting potential datasets in ways that might solve the problem shown above. The solution involves a stepwise algorithm in finding a solution.

First must be identified the CDS datasets that potentially might solve the query for data about the tsunami. Step one would then be to do a simple range check to see if the location of the tsunamis' y coordinates are within the y extent of any dataset known to the system. An example of how this could be accomplished is illustrated in Figure 2.

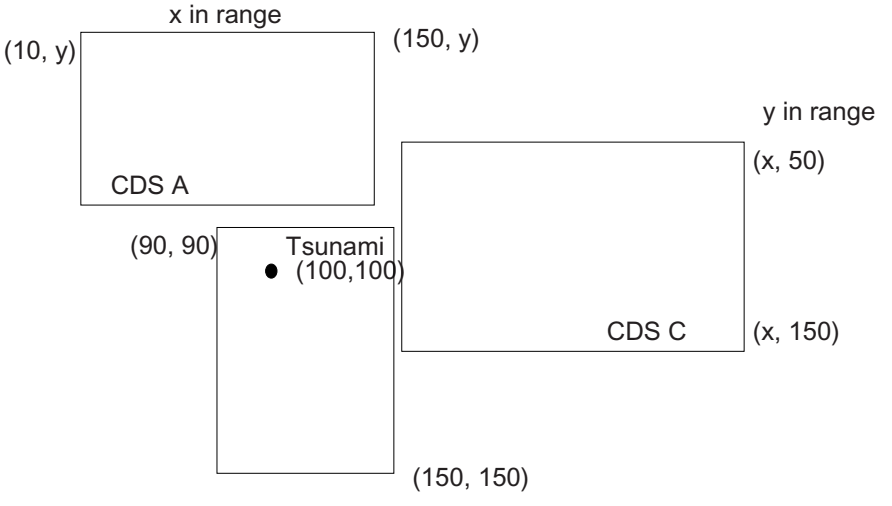

Fig. 2. Initial range check to determine inclusion of datasets.

After some examination of the coordinate pairs, it is clear a simple range check on CDS A and CDS C eliminates them from inclusion as a candidate dataset. This is because their x or y coordinate extents do not intersect those of the dataset enclosing the tsunami point.

Using this technique coupled with vector cross product techniques it is possible to establish that a spatial coverage for a CDS does include the spatial point in question. Without much modification this technique can also be made to work for regions delimited by polygons that are entirely or partially contained by a dataset's bounding polygon. Once a coverage has been selected as a potential solution using this method, the next step is to apply fuzzy set theory to rank the relevance[22] of the coverage to the point of origin of the tsunami.

The approach used in this model is to do this in one of several fashions. The first of these would be to calculate the distance between the tsunami's point of origin point and the centroid of a bounding polygon. The distance value could become a component in the return value for the fuzzy membership function for the spatial aspects of CDS data, MSpatial() which is discussed later. In this case, the smaller the distance, the more centrally located the point representing the tsunami's point of origin is to the coverage being considered. This scenario is shown in figure 4.

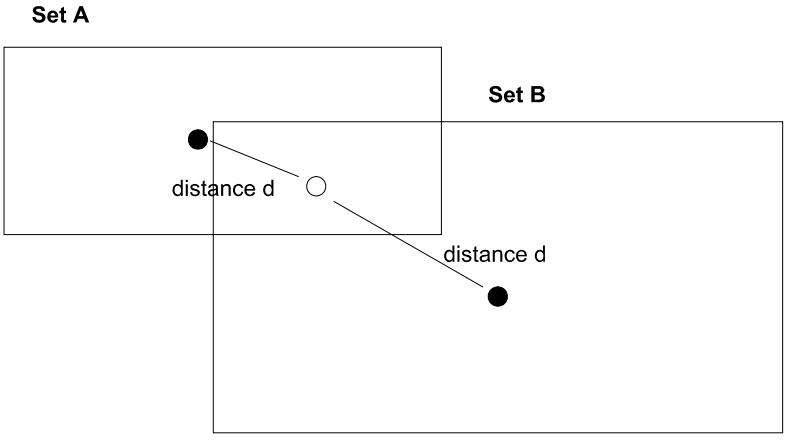

Fig. 3. Application of distance as a return value for a fuzzy function.

The value of the fuzzy membership function MSpatial() to select for Set B may be .9, whereas the value of the fuzzy function MSpatial() for Set A might be .3. Higher values of MSpatial() would reflect the fact that the distance to the nearest centroid was smaller. This would then suggest that because the point represented by the tsunami's origin is more centrally located in Set A, that this set is a much better set to use if one wants to examine data for Pullman and the surrounding area.

While this approach makes sense, it is simplistic. Therefore, weighting factors in conjunction with attributes of the data sets themselves might also be considered. An example of this might be that both sets have metadata for "data accuracy", and the "time last data collection". In a CDS model one might argue that the data accuracy is very important (.9) because improved accuracy would be expected to reflect current reality better. Following this logic, a scheme then might weight "time of last data collection" very heavily. We might give lesser weight to the accuracy weight (.3).

A weighting scheme to use in retrieval might develop in the CDS model a function to assist in resolving data ambiguity in this case might become

$$\mathbf{C_{weight}() = distance * (.9 * days\ since\ last\ edit + .3 * time\ of\ day\ edited)}$$

In above scheme, geometric properties can be combined with attribute properties for a CDS set to solve an ambiguous selection and retrieval problems.

Rationale For Fuzzy ERD's to Manage Contextual Data

Chen [8] defines an Extended Entity Relation (EER) model to consist of the triple

$$\boldsymbol{M = (E, R, A)}$$

where M represents model, E represents entities, R represents relationships and A represents relationships. E, R, A are defined to have fuzzy membership functions. In particular:

$$R = \{U_r(R)/R \mid \text{where R is a relationship involving entities in Domain(E) and } Ur(R) \in [0,1]$$

In this case, ***Ur()*** is a fuzzy membership function on the relationship between two entities in a data model. Chen defines fuzzy membership functions on attributes and entities as well. Because of the above, it is possible to have fuzzy relations on relations, without built in dependencies on other types of fuzzy objects in a model. Based on this work, our research now extends our data model ERD to defining notations that describe the application of fuzzy theory to relations.

The next section we will examine how contextual data can be managed in a previously developed [12] model for management of fuzzy spatial temporal information. The previously defined operators will be briefly presented and then a discussion will be made about how the model supports or does not support that of contextual data management.

A Fuzzy ERD Model for Context Management

The data model in figure 2 provides an initial foundation to address problems inherent with management of context data. The data model was developed to model spatial and temporal information which are two key dimensions of contextual data. One of the problems with contexts, as with spatial and temporal data, is that of ambiguity in the selection and retrieval of data. These problems can be addressed by the application of fuzzy set theory. For example, several overlapping coverages for tsunami information could exist based on time and space. In this sense overlapping coverage is defined to be multiple contexts with information fully or partially about the same event, e.g. the origin of the tsunami. Keep in mind there is a tendency to think of such information as geo-spatial but it may also include images and textual descriptions. The key concept in retrieval is to find the most "appropriate" coverage for a given query. Appropriate is a term that can only be defined for the consumer of the information, the user. The logical question becomes which context data set to select and use for a given purpose.

Overlapping contextual spatial coverages are a type of ambiguity. We have also identified that there can be overlapping contextual temporal locations. We can also have overlap in the similarity of contexts and their impact dimension which are not addressed in this chapter. When considering the problem of overlap in selection and retrieval of information the types of overlap that can be present must also be considered. Overlap can be partial or complete overlap with different descriptive characteristics to coverage such as different projections, scale and data types. These can also become complications to ambiguity of selection. Considering this situation, it is clear that ambiguity on an attribute of spatial data can compound with other ambiguities about the same data. This can have the potential of leading to much larger ambiguities.

To date, a lot of work has been done in the development of fuzzy set theory, and techniques for decision making such as using Open Weighted Operators [4]. Little of this work has been applied to the management of contextual data. In particular, theoretical discussions needs some form of implementation to solve real world problems. What is needed is the application of theory and a representational notation. The application could then be used to solve real world problems such as data ambiguities. The figure 2 data model was extended with new types of fuzzy operators that address ambiguous selection problems to create a more powerful model that can deal with the problem of ambiguous data.

Contextual Subsets

The first new notational convention is the context subset symbol. The *Subset* symbol defines a new type of relationship on an entity, that is it borrows from object oriented constructs, that of the "bag". An entity with the subset symbol defined on one of its relations is a non-unique entity, unlike most entities in an ERD model. The rationale for its existence is that multiple copies a *Subset* containing the same elements can exist for different overlapping temporal, spatial, impact and similarity coverages for a given event, e.g. the tsunami. This circumstance can occur as a result of various versions of the same *Subset*, or normal editing operations. The symbol is defined as:

$$\subseteq$$

By its nature of being a non-unique entity, a relationship with the *Subset* definition, also is a fuzzy relationship. This is due to the fact that when one desires to view a *Subset*, the question becomes which one should be selected. Because *Subset*s are discrete, the *Subset* symbol occurs in our model with the symbol for fuzzy relation M() which is defined next.

Fuzzy Relation M()

Fuzzy theory literature [7] defines a membership function that operates on discrete objects. This function is defined as M() and has the following property:

$$\text{Similiar}(a) = \begin{cases} \{\ 1 \quad | \text{ if } a \in \text{domain}(A)\ \} \\ \{\ 0 \quad | \text{ if } a \notin \text{domain}(A)\ \} \\ \{[0,1] \mid \text{if } a \text{ is a partial member of domain}(A)\} \end{cases}$$

This function is particularly useful in contexts where overlapping coverages of the same event space may exist, but some coverage for a variety of reasons may be more relevant to a particular concept such as a desire to perform editing of surrounding regions. The actual definition of how partial membership if calculated has been the subject of much research including the application of Open Weighted Operators (OWA) [3,10] and the calculation of relevance to a concept [9].

The data model developed for contexts model in this chapter seeks to provide alternative view support for overlapping geospatial coverages. Because of the ambiguities induced by this, we introduce a notation that represents the fuzzy relation resolved by the definition of the function M(). This symbol is referred to as the fuzzy relation M() symbol and may be displayed in and ERD model along the relations between entities. It has the following notation:

This symbol makes no comment about the nature or calculation of M() per se, but does suggest that the M() function is evaluated when a query on the relationship in the ERD is generated. The query returns a ranked set of items with a similarity value in the range of [0,,1]

Another property of the function M() is that it reflects the fuzzy degree of relation that entities have with other entities.

Fuzzy Directionality

Fuzziness in the context data model is not bi-directional on a given relationship. Therefore there needs to be some indication of the direction fuzziness applies. This is denoted by the inclusion of the following arrow symbols on the fuzzy relationship. These arrows are found to the left of the fuzzy symbol and point in the direction that the fuzzy function M() or MSpatial() applies. If a fuzzy relationship is defined in both directions, which implies a type of m:n relation, the symbol is a double headed arrow.

Directional fuzziness for the M() or MSpatial() function, points in the direction the function is applied for selection and is denoted by the following symbols:

$$\leftarrow, \uparrow, \rightarrow, \downarrow$$

Bi-directional application, is a member of the class of m:n relations and is denoted by:

$$\leftrightarrow$$

Discretinizing Function D()

Because of the data ambiguities mentioned previously, the new context based data model uses time as an attribute in describing data. However, this leads to temporal ambiguities in the selection of a *Subset* of data because the *Subset* can exist at many points in time. However, there are certain points in time where the relevance of data to a concept or operation, e.g. a selection, query is more relevant. Therefore the relation of *Subset* to *Temporal Location* entity can have fuzzy logic applied. Time is not a discrete value, it is continuous, and therefore it is referred to as a continuous field. Some attempts have been made to discretinize continuous

temporal data by Shekar [9] using the *discretized by* relation. But no known attempts have been made to deal with this in a fuzzy fashion This leads to the need to define a new function D() that can be used to calculate discrete fuzzy membership value over continuous fields.

In the new contextual model for data management, the inclusion of continuous field data is useful. This is due to the fact that sets of data not only cover a geographic extent, but they also cover this particular extent for a period of time and then can be replaced by another set, perhaps not of the same geographic coverage but at a different point in time.

If time is non-discrete and a function must be developed, the question becomes how to represent continuous data in a fashion that a function can make computations on the data and return a discrete value representing membership. Upon examination of this issue, non-discrete data can be defined as a bounded range [m,n] where the beginning of the continuous temporal data starts at time m and terminates at time n. This representation then makes it possible to develop the function D() and its behavior over continuous data.

In this function one wants to think of a window of time that a set of context data was created at time $t_{m,}$ spanning to a point in time where the data is no longer modified, t_n above equation, the range $[t_m,t_n]$ is referred to as the "window" because it is a sliding window on the continuous field that one seeks to determine the degree of membership of selection point of time t to be. A function that can then be used to retrieve relevant sets of contexts can be defined as:

$$\text{INRANGE}([t_m,t_n], t) = \{ \text{ ABS } [(t - t_m) / (t_n - t_m)] \}$$

where ABS() is simply the absolute value of the calculation.

The effect of D() is to make a discrete statement about non-discrete data, which makes it possible to make assertions about fuzziness and possibilities. The statement is of course relative to the bounded range [m,n] and therefore D() should be formally denoted as:

$$\text{MSpatial()}_{[m,,n]}$$

when referring to value returned for particular calculation of the function MSpatial()

The application of MSpatial() is found in the dataset management model on the fuzzy notation denoted by the symbol :

This symbol is displayed on the model oriented such that relation lines intersect the vertical lines of the symbol. This notation means that the relation is fuzzy and is determined by the discretinizing function MSpatial() as defined above. For the purposes of the data management model, discretinizing functions are applied to temporal entities that define a *Subset* of sets of contextual data by a temporal location. They can however, be applied to any type of continuous field data.

Fuzzy Relation MSpatial()

The function M() is not a complete function for the solution to selection problems in the developed ERD model. This is because it does not consider the centrality of a point P composed of an x,y and perhaps z component that one wishes to retrieve context data about. This lead to the creation of of a function referred to as MSpatial(). The characteristic function MSpatial() needs to contain a function that measures distance, a new term, d, that can be derived in the following manner:

$$d = \min\left(\sqrt{(centroidx - po\text{int}\, x)^2 + (centroidy - po\text{int}\, y)^2}\right)$$

$$centroidx = .5*(s1x2 - s1x1)$$

$$centroidy = .5*(s2y2 - s2y1)$$

The above equation refers to a rectangular bounding hull created around a geographic coverage. Centroid_n is the centroid of the bounding hull found by finding the mid point of of side one for centroidx and the mid point of side 2 in y for centroidy.

Pointx and pointy are the coordinates of a spatial entity or center of a region of interest that one is seeking the most centrally located coverage for. The d value in the a characteristic function for MSpatial() then becomes a measure of the minimum distance of a coverage's centroid to a spatial entity. The effect of the equation is to find a contexts coverage that is most central to the spatial entity of interest. The goal is to weight the characteristic functions values with a measured degree of centralization to a spatial center of interest when selecting fuzzy data for a particular problem.

The application of MSpatial() is found in the dataset management model on the fuzzy notation denoted by the symbol :

This symbol is displayed on the model oriented such that relation lines intersect the vertical lines of the symbol. This notation means that the relation is fuzzy and is determined by the MSpatial() function

Extended Data Model for The Storage of Context Data Sets

With an understanding of the issues found in retrieval of context data sets and some new fuzzy characteristic functions and notations a new data model can be presented for management of sets of context data. This model considers the vagaries of ambiguity in time and space selection which are dimensions of contexts but does not support the contextual dimensions of similarity and impact. The model is presented in figure 2.

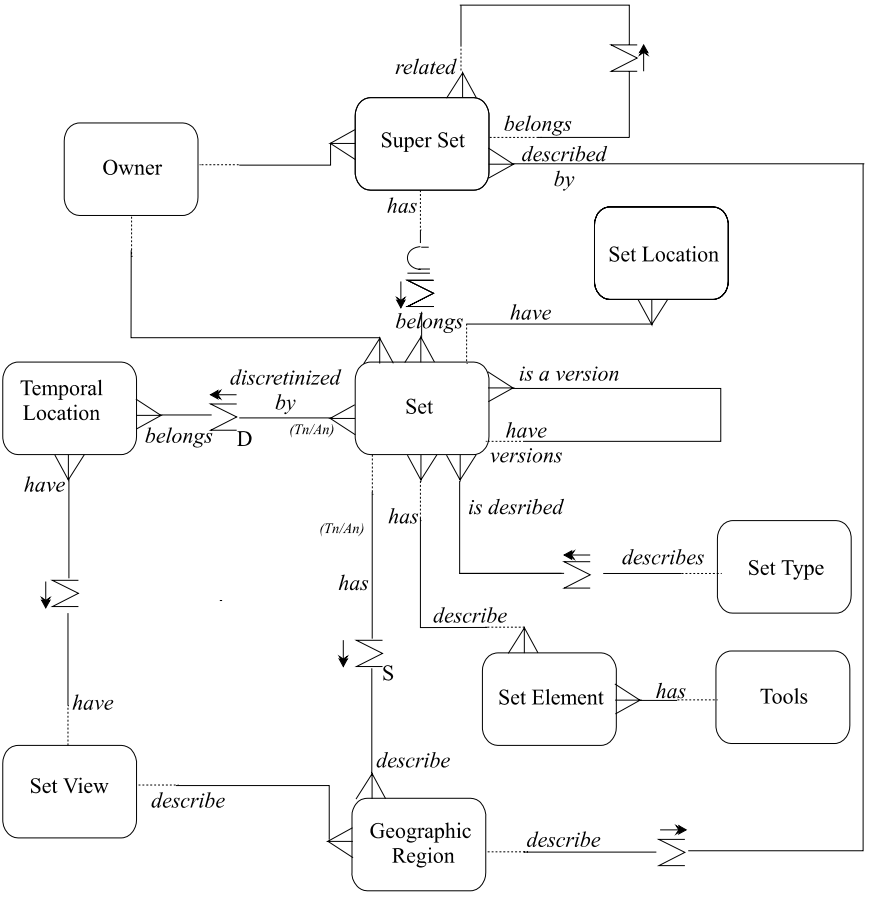

Figure 2. Extended fuzzy model that can be applied to context set management.

We now examine some characteristics of context data for how the above model for management of spatial data will support these. Contexts have the following properties:

- four dimensions that uniquely characterize them, time, space, impact and similarity
- do not have an owner because they stream from their sources
- do not have a specific location they reside because they float around the internet
- do not have a particular type associated with them because contexts can be composed of many different data types
- might have specific tools that process them depending on the consumer of the information
- do not have views of the data that are currently defined
- super contexts are composed of contexts which are composed of elements

With these in mind, we discuss the components of the fuzzy model and see how they might be supported by the existing fuzzy model.

In figure 2 the original fuzzy model entities are defined to be the following:

The *SuperSet* entity defines collections of contextual *Subset*s. It has a recursive relationship to other *SuperSet* instances. The relationship *related* shows the fact that multiple supersets may be related to each other. For example, supersets may cover the same dimensions of a multiple contexts.

A *SuperSet* is composed of multiple *Subset*s. Using the example of the tsunami, a superset may be composed of all the CDS sets of tsunami data that has been created over the a given period of time.. The relationship of supersets to subsets introduces the new extension of fuzzy subsets. An instance of a subset entity is a single logical, meta description of a component in a superset. It also has a recursive relationship to itself that allows the user to implement versions of the subset and thus versions of the sets, a lineage tree.

The relationship between *SuperSet* and *Set* has a subset notation. Subsets by definition are not unique entities. When considered with the entities with which they have relations with, they can become unique. The existence of the subset symbol and fuzzy relations to other entities dictates that this relationship have a MSpatial() relation on it.

A subset of context data has an unusual property in that this entity is not unique in itself. It becomes unique when the fuzzy relations around it are considered. It also has an ISA type of relationship with Superset in that it inherits attributes from the superset. There can be multiple physical files containing context data and rules that a subset may represent.

This relation can be characterized by the existence of multiple *Set*s of heterogeneous data formats that cover the same area but may be of different scale or perspective. Each one of the *Set*s may cover a minute area part of the spatial coverage a superset has and is therefore a subset. Additionally, the *Set* coverage's may not be crisply defined in the spatial sense. They may also cover other areas defined as part of other partitions in the superset. This leads to the property of sometimes being unique and sometimes not being unique.

The *Set* and *Super Set* entities and their subset relationship support the concepts of contexts. Specially, they support the idea that super contexts are composed of contexts and a feature vectors data in a given context is composed of elements also referred to as attributes in the context paradigm.

The *Temporal Location* entity represents a locational time definition for a data set. It has temporal attribute values and geospatial coordinates that collectively create identifiers for a particular data set.

The D() relation between *Temporal Location* relation and *Set* can be used to select context *Set*s that cover a given range in time. These can occur due to the existence of long editing transactions on the data. A *Set* is not instantly updated during a long transaction. Certain points in the existence of the *Set* may be more of interest when selecting a set to view, but all are valid descriptions of the subset. The D() function exists as a sliding window of possibility for selecting a *Set* that has existed and was updated over a period of time. This allows one to select the *Set* in a specific time range.

The *Temporal Location* entity exists because there is a need for given data sets to map to various locations in time and spatial coverages. The relationship "discretized" was originally defined to map a value to a continuous field. In this case the discretized relation has been extended to represent a discretized function where the continuous field is time. Because this function can relate a data set to various points in time and coverage of several different spaces, this function is a fuzzy function, D() that selects on time and spatial definitions for a *Set*.

In this case *Temporal Location* supports the concepts that contexts have a dimension of time that describes them. The aggregation of this fact as it relates to specific contexts then defines a super contexts window of temporal existence embodied in the *Super Set* Entity. The D() operator reflects the fact that contexts stream data as they are created, thus there is a need to select data that may span ambiguous moments in time.

The *Geographic Region* entity locates a *Subset* of data by the type of spatial coverage it has. This entity works in conjunction with the *Temporal Location* entity to locate a set of data in time and space. The rationale is that for a given spatial area, there may be multiple coverages generated over time. Therefore, the problem becomes one of locating spatial data in 2D space.

The *Geographic Region* entity has a M() relationship with *Set*. The rationale is that because context *Set*s of data can be overlapping in spatial coverage for a given point in space, selection of a subset becomes an ambiguous problem. The MSpatial() symbol then implies that selection of *Set*s covering a geospatial point needs to done using some type of fuzzy selection.

The *Geographic Region* entity is not clearly defined in the context model at the present because the regions that contexts are created for is assumed to be fixed and thus is not ambiguous. However, it can be logically argued that contexts may not be registered exactly over the same geographic point. This could be the subject of future investigation.

The *Set Type* entity describes the type of data a *Subset* may contain. An example of the expected types where "image", "raster", etc. This entity was also a candidate for fuzzy notation extension, following this section. The *Set Type* entity is not defined in the context model because a set maps to a feature vector and a feature vector is composed of multiple types of disparate data that are stored at the media location described by *Set Location*.

The *Set View* entity in the model provides a repository for information about various views of data that may exist for a given *Subset*. *Set View* makes it possible to have multiple views of the same data set. Such views would differ by such things as a datasets perspective, scale or projection. The *Set View* entity is not supported in the context model and therefore there is no current mapping or application of its functionality.

The *Set Location* entity describes the physical location of the *Subset* and thus a contexts data. This entity is required because of the need for a model where data can be distributed around a computer network or around the internet. This entity provides the potential to support distributed repository mechanisms because parts of the database are not in the same physical data space at all times. It also provides a way to have alternative views of data that are not centrally located. This

entity is very highly supported in the context model where contexts are hyper distributed around the internet. The context model will probably spend considerably more time developing the concepts behind *Set Location*.

Analysis of Coverage Support of the Fuzzy ERD for Contextual Data Sets

Having established that the Fuzzy ERD model does support management of Contextual data sets, it is useful to determine how much of the model is extra and could be trimmed to refine the model.

The above section finds that the entities that are not utilized when mapping the context model onto the fuzzy storage model are *Owner, Set View*, and *Set Type*. This is due to the characteristics of contextual data discussed previously that differ from geo-spatial data. Entities that are marginally supported are *Tools* and *Geographic Location.* If we determine a coverage ratio for the context models mapping onto the fuzzy set management model it come to the following:

2 entities partially supported which arbitrarily are counted as 1
3 entities that do not map to the context model
6 entities that fully support the model

This produces a coverage ratio of 1+ 6 / 11 = 63%. This means 37% of the previously developed fuzzy ERD model is not utilized and potential for elimination in a new tailored model. This ratio could be increased if the ERD entities that are partially supported by the mapping were honed to be fully functional in the support of the context model. If this is done, the ratio of utilized entities to non utilized entities become 8/11 or 72%. This means that 28% of the existing model does not really support contexts and is a candidate for removal.

Future Research

This research merges the concepts of global contexts with that of an existing fuzzy data model for management of spatial and temporal information. What is discovered in the process is that the dimensional elements of contexts, those of time and space lend themselves well to management by such a model. The dimensions of similarity and impact are not supported in such a model and thus a subject of future research. The final results of the analysis of coverage suggest that a new type of data model should be developed that is more tuned to support of the contextual model. Additionally, the performance impact of the fuzzy operators should be also evaluated to see how sharply they affect retrieval of information. Mechanisms should also be examined that can help the fuzzy selection functions adapt to their own performance in such a way that feedback can improve their performance. Much work remains to be done in this newly emerging area of a new paradigm for information sharing on a global scale.

References

[1] Rosemann, M., Recker, J.: Context-aware process design: Exploring the extrinsic drivers for process flexibility. In: Latour, T., Petit, M. (eds.) 18th international conference on advanced information systems engineering. Proceedings of workshops and doctoral consortium, pp. 149–158. Namur University Press, Luxembourg (2006)

[2] Schilit, B.N.A., Want, R.: Context-aware computing applications" (PDF). In: IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 1994), Santa Cruz, CA, US, pp. 89–101 (1994)

[3] Schilit, B.N., Theimer, M.M.: Disseminating Active Map Information to Mobile Hosts. IEEE Network 8(5), 22–32 (1994)

[4] Dey, A.K.: Understanding and Using Context. Personal Ubiquitous Computing 5(1), 4–7 (2001)

[5] Bolchini, C., Curino, C.A., Quintarelli, E., Schreiber, F.A., Tanca, L.: A data-oriented survey of context models (PDF). SIGMOD Rec. (ACM) 36(4), 19–26 (2007), http://carlo.curino.us/documents/curino-context2007-survey.pdf

[6] Schmidt, A., Aidoo, K.A., Takaluoma, A., Tuomela, U., Van Laerhoven, K., Van de Velde, W.: Advanced Interaction in Context (PDF). In: 1th International Symposium (1999)

[7] Burrough, P.: Natural Objects With Indeterminate Boundaries. Geographic Objects with Indeterminate Boundaries. Taylor and Francis, Abington (1996)

[8] Chen, G., Kerre, E.: Extending ER/EER Concepts Towards Fuzzy Conceptual Data Modeling, MIS Division, School of Economics & Management. Tsinghua University, Bejing, China

[9] Morris, A., Petry, F., Cobb, M.: Incorporating Spatial Data into the Fuzzy Object Oriented Data Model. In: Proceedings Seventh International Conference IPMU (1998)

[10] Shekar, S., Coyle, M., Goyal, B., Liu, D., Sarkar, S.: Data Models in Geographic Information Systems. Communications of the ACM 40 (April 1997)

[11] Yager, R., Kacprzyk, J.: The Weighted Averaging Operators, Theory and Applications. Kluwer Academic Publishers, Boston (1997)

[12] Vert, G., Stock, M., Morris, A.: Extending ERD modeling notation to fuzzy management of GIS datasets. Data and Knowledge Engineering 40, 163–169 (2002)

[13] Larson, R.: Geographic Information Retrieval and Spatial Browsing. School of Library and Information Sciences, University of California, Berkely (1999)

[14] Morris, A., Petry, F., Cobb, M.: Incorporating Spatial Data into the Fuzzy Object Oriented Data Model. In: Proceedings of Seventh International Conference IPMU (1998)