시간 공간 통합 본원적 데이터 모델링 및 그 구현에 관한 연구

이 우 기*

Modeling and Implementation for Generic Spatio-Temporal Incorporated Information

Wookey Lee*

Abstract

An architectural framework is developed for integrating geospatial and temporal data with relational information from which a spatio-temporal data warehouse (STDW) system is built. In order to implement the STDW, a generic conceptual model was designed that accommodated six dimensions : spatial (map object), temporal (time), agent (contractor), management (e.g. planting) and tree species (specific species) that addressed the "where", "when", "who", "what", "why" and "how" (5W1H) of the STDW information, respectively. A formal algebraic notation was developed based on a triplet schema that corresponded with spatial, temporal, and relational data type objects. Spatial object structures and spatial operators (spatial selection, spatial projection, and spatial join) were defined and incorporated along with other database operators having interfaces via the generic model.

Keywords : Spatio-Temporal Modeling, Data Warehouse, ST Algebra, ST-ERD

논문접수일 : 2004년 8월 14일　　논문게재확정일 : 2004년 11월 6일
※ *This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Advanced Information Technology Research Center (AITrc).*
* Computer Science, Sungkyul University, Korea

1. Introduction

Data warehousing provides an enterprise solution for organizations that have accumulated a considerable amount of operational data over the years and need to develop a methodology to turn that data into useful information. A data warehouse is a sophisticatedtechnology infrastructure that stores and analyzes data to aid decision support. It is the next level above a database. Data warehousing is a process in which operational data is re-processed, aggregated and stored in base tables. Pratt and Adamski [2000] summed up a data warehouse as an architecture containing read-only views of extracted, highly aggregated, and summarized data from multiple internal and external sources that are refreshed periodically.

One of the fundamental difficulties in designing a general spatio-temporal database is its data model. Incorporating both time and space in data models increases the complexity of the data structure and is a challenging task [Raza and Kainsz, 1999]. Various approaches have been adopted to capture the essence of time and space. Fernndez and Rusinkiewicz [1993] have used the Entity-Relationship (ER) and its extensions to capture spatial semantics in geospatial modeling, using a model which was originally proposed for business applications. Erwig and Schneider [1997] applied fuzzy logic to spatial information, which forced a fuzzy function between a base table and a corresponding related table. Belussi et al. [2000] extended the ER model based on the research of one or more spatial characteristics.

Objectives

The objective of this paper was to develop a framework for integrating geospatial and temporal data with relational information. In order to achieve full integration, data should flow bi-directionally from source to destination and vice versa between spatial and temporal/relational domains. The goal of this project is to develop a mature and guaranteed high performing technology that would provide students and researchers with enhanced access not only to current data but also to critical historical data that would enable them make informed management decisions based on reliable data.

2. Related Work

The logical design of the Geographic Relational Data Model (GRDM) by Hadzilacos and Tryfona [1996] and Tryfona and Hadzilacos [1998] was a follow up on the conceptual modeling designs in geographic applications. The GRDM was an extension of the relational model and it was based on the spatial and temporal requirements of the design. It provided a language for the definition of : (a) relations, used for non-spatial entities and relationships, (b) layers, which represented space varying attributes, and (c) object classes. To capture spatio-temporal information, the logical concepts were transformed into specific constructs of semantic models as an extended relational model to accommodate the needs of the spatial and temporal domain. These research efforts focused on developing a database model as opposed to a data warehouse model.

Object-oriented (OO) methodologies were thought of as having evolved from programming languages and the ensuing engineering model diagrams [Graham, 1994]. OO approaches to spatial data modeling did not advance further and have remained as research and development case studies. The major thrust of this research was in developing formal diagrams and their interaction models. Examples include the Object Modeling Technique (OMT) [Rumbaugh et al., 1991] and the Unified Modeling Language (UML) [Booch et al., 1999].

UML as an integrated method for modeling, has been used to model OMT (Rumbaugh et al., 1991), object-oriented software engineering (OO SE) [Jacobsen et al., 1992], and object-oriented analysis/object-oriented design (OOA/OOD). UML comprises a number of diagrams used to describe particular aspects of software artifacts. Depending on the intended use, the diagrams describe the structural or behavioral aspects of the phenomena. UML diagram's abstract syntax, and the well-formed formulae are devised in a special expression language called object constraint language (OCL), and the semantics are described primarily in precise natural language. Many claims have been made about the flexibility of the OO paradigm in regard to modeling the behaviors of objects (encapsulation of data and process), the ease of modification and extensibility of OO models, and the data abstraction facilities (inheritance and polymorphism). However, the main drawback of the OO model is its failure to exploit large volumes of data [Dawson, 2001].

It is common in GIS to make a distinction between space-based and feature-based models [Takayama and Couclelis, 1997]. The former represents regions within the space where attributes (such as height, temperature, etc.) are associated with these spaces. The latter represents geographic entities that are the primary objects with which spatial attributes are associated. Recently these two approaches were integrated [Hadzilacos and Tryfona, 1997]. The integrated approach has been adopted in this paper. Prior to integrating information, we derived a fundamental atomic data unit, called a triplet, by which the abstract information of an object and its inheritance are separated.

Zhou et al. [1999] were the first to use a framework for spatial data online analytical processing (OLAP). They extended the star schema to cube dimensions in both spatial and non-spatial domains in which the measures were regions in space, in addition to numerical data. They suggested a method for selecting spatial objects for materialization. They proposed a way of merging spatial objects using efficient input/output (IO) measures and applied spatial procedures to aggregate data. Coincidentally, Papadias and Arkoumanis [2002] applied the terms spatial data warehouse for their spatial index model that appeared to represent a star-schema. Their focus was on developing an index structure for a spatial object.

In order to generate a common framework for geospatial information, a number of researchers proposed approaches that appeared to represent a generic data model. Alternatives included an Extended Entity Relationship (EER) model [Hjelsvold and Midtstraum, 1994 ; Shekhar et al., 1997 ; Sudarsky and Hjelsvold, 2002], an annotation approach [Cassidy and Bird, 2002 ; Marshall, 1998], and a data warehouse approach [Tryfona et al., 1999].

Shekhar et al., [1997] applied the EER model that included continuous fields associated with discretization and interpolation models. There was however no perfect interpolation model that matched identical continuous field features exactly. Hence, the EER model inevitably allowed approximation errors that were due to proximate interpolations. The annotation approach mostly focused on video retrieval and the granularity issue in order not to take the whole video as a binary large object (BLOB), but to divide it into smaller segments to which annotations were made. The annotation approach performed poorly when applied to relational database information [Cassidy and Bird, 2002]. Many conventional conceptual models, such as the ER model, Extended Entity Relationship model (EER) [Hjelsvold and Midtstraum, 1994 ; Shekhar et al., 1997 ; Sudarsky and Hjelsvold, 2002], and starER [Tryfona, 1999] could potentially serve as a basis for representing a generic geospatial DW model.

The problem of integrating spatial objects with temporal and/or relational objects is still largely unresolved. Other approaches such as the Non-First Normal Form (NFNF) [Ng et al., 1999], annotation approaches [Jones and Edens, 2002], spatio-temporal representations [Erwig and Schneider, 1997], and conceptual modeling approaches [Shekhar et al., 1997 ; Vert et al., 2003] appeared to integrate multimedia information (e.g., movies) to database technology. The NFNF (also known as NF^2) tried to enhance hierarchical characteristics of the video retrieval system to relational representation, such that a tuple of a relational table could include another table or object, which would in effect take the relational database out of first normal form. In this format of representing spatial objects, a relational table could have a map as a tuple. This approach had the serious limitation of not being able to merge a spatial object as a relational object.

Recently, an XML based data integration method was suggested using NFNF [Chen et al., 2003]. Another approach was to give tags or descriptions to multimedia objects, including Digital Library [Marshall, 1998], spatial objects and Web views [Vasudevan and Palmer, 1999], and Video content Indexes/Annotations [Jones and Edens, 2002]. Similar approaches to spatio-temporal representations were applied [Raza and Kainsz, 1999]. It was found that the annotation worked as a contact point between spatial (multimedia) objects and data model or database systems ; however, most were built on a specific domain of a database system, and were not extended to the data warehouse model.

The starER [Tryfona et al., 1999] was considered as a viable conceptual model for a data warehouse, but has several limitations as follows. The interface of spatial dimension with the fact table is represented by just an object ID (e.g., real estate 'number') to which a hierarchical region and a city are linked without a schema. This example fitted the description of a small data mart and did not embrace the essence of a spatial data warehouse. Tryfona et al., [1999] in this model did not show a generic spatial schema but maintained the word 'star' while sacrificing the true meaning of data representation by including cyclic relationships or alternative relationships between the corresponding objects. There is neither a schema of spatial objects nor spatial operators accessing a spatial point corresponding to relational objects and vice versa. Therefore, in summary, their representation is that of a relational star schema, with just a dimension that references spatial names (e.g., city, region, county).

3. Geospatial Data Warehouse Model

3.1 Generic Modeling

A database system managing geospatial information should provide database support for a diverse range of applications. The same database should also be available for several different uses and purposes. In the forestry context for example, this would include carrying out simultaneous management activities at various locations, such as planting, harvesting, surveying, using specific agents (personnel). To facilitate sharing of geospatial information across the board, a common data model needs to be adopted for each application. The common data model is known as a 'least common denominator,' and can be referred to as a 'generic model,' which can be applied as a consistent framework for our objects. See [Lim & Kang, 2004].

3.2 DW Model Representations

3.2.1 Spatial Dimension

Spatial Objects Representation represents a dimension table for the DW, we have considered a spatial object as an ADT (abstract data type) that will be described later in detail. The highest abstract level of geometry in this paper is a map. The map is a collection of spatial objects, for instance a Forest Cover Polygon (FCP), and this spatial object is derived from the real world forest. A FCP is defined in this paper as a stand of trees with similar attributes at a particular point in time. In designing a dimension for DW, there are two extreme choices : (1) The highest abstract level the object is engaged in, or (2) the detailed object instances the object is engaged in. The second one was introduced by Tryfona et al., [1999], as representing the interface between the spatial object and the other objects (e.g., relational objects). And that individual objects or instances had to be engaged as dimensions. Their model was too complex and vague for a detailed fact table and associated dimension tables. In this paper, however, the first approach is applied, because, the map is as a whole (not as individual objects) engaged in the GSDW. The dimension should be simple, and the details can be maintained in the Map object instance table.

We define a thematic map as a collection of spatial objects as follows. A Spatial object is derived from the real world forest. A spatial object may consist of other spatial objects, such as a stream composed of branches. The

MapObject

MapObjectID	Geometry	ClassType	MapID
935	polygon	vector	1
936	polygon	vector	1
937	polygon	vector	1
938	polygon	vector	1

Map

MapID	MapName	Description
1	ForestCover	forestinventory
2	Cutblocks	Forest openings
3	Strata	Intensive management units
4	BECZone	Bio-ecological zones

Fact table

TimeID	TreeSpeciesID	OrganizationID	ActivityID	MapID	MapObjectID
8-Sep-2003	HW	UBC	harvesting	2	2003-U001-01
17-Sep-2003	CW	UBC	planting	3	2003-U001-01-1A
19-Sep-2003	FD	UBC	planting	3	2003-U001-01-1B

〈Figure 3-1〉 Map database of the Geospatial model

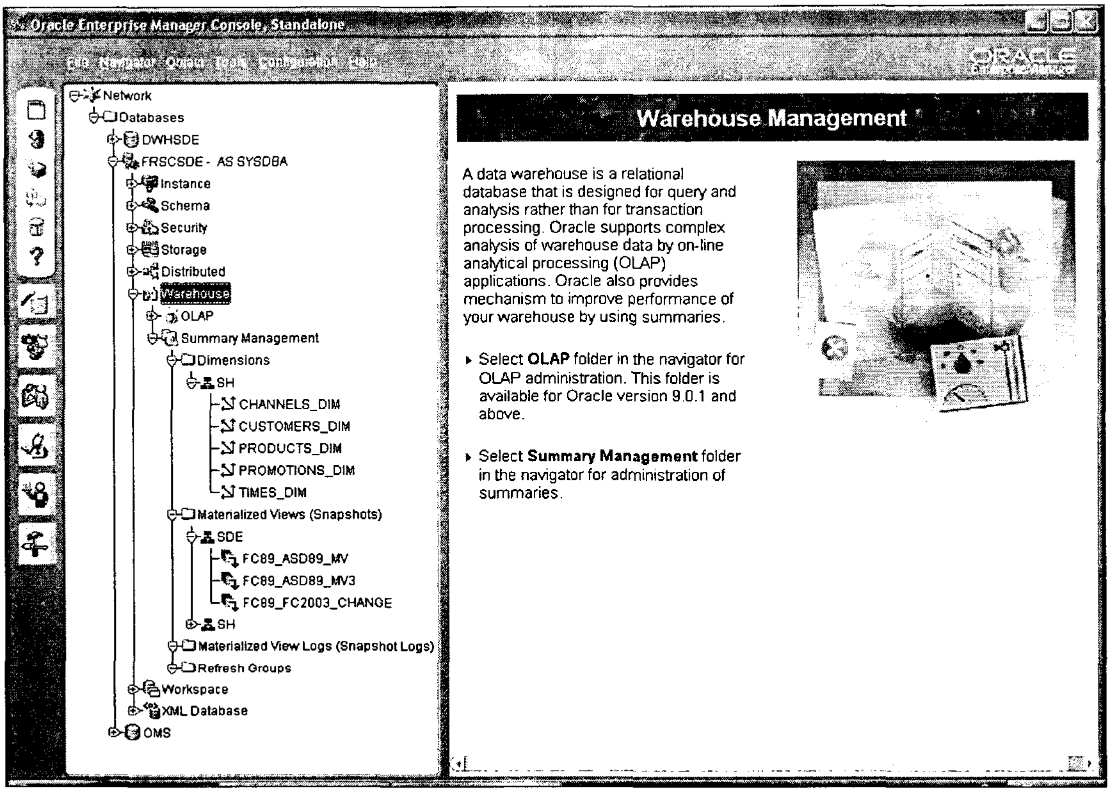
objects are included in a member of homogeneous domain maps, e.g., in this case, a hydrographic domain that has streams, lakes, or reservoirs. The map is the highest level of the spatial set of domains and the map consists of several objects. The spatial object consists of a set of maps as layers.

3.2.2 Temporal Dimension

A temporal aspect in the generic DW data model is recorded by design, in terms of years. For example, days, weeks, and months will not be factored into the design, simply because the effects of most forestry operations are observed after a number of years have elapsed. Therefore, warehouse updates will be occurring annually. The temporal notations by Snodgrass (1993) and Slivinskas et al., (2001) were considered. Two kinds of temporal dimensions in this paper are considered, as a time point and as a period. The schema of temporal dimension is composed of start-time and end-time, so the time point can be represented by either one of the two, and the period of time is represented by both temporal dimensions.

3.2.3 Relational Dimension

Three relational dimensions are considered in our generic model : Organization dimension, Tree Species dimension, and Management Activity dimension, representing who, what, and how of the model, respectively. These three relational dimensions are mandatory factors in the generic model. The most important factor is the Management Activity dimension. We apply these dimensions consistently all though the management activity options (e.g., preparing sites, planting, harvesting etc.), so it can



〈Figure 3-2〉 Modeling for Data Warehouse Management

be called a generic model. With the same generic data model, we can generate forestry management points of view based on activity options such as harvesting a cutblock, and prescribing management activities to rehabilitate the harvest unit, such as a planting a desired tree species, and conducting plant growth surveys.

4. Formal Representation

4.1 Terms and Definitions

In this section, we present a conceptual geospatial DW model independent of any database representation. DW data are defined as a set of database states consisting of current database states as well as the history of the states. So we assume that DW schema includes a database schema, which means that the database is focused on the current state of a base relation where as a DW is focused on collecting the state trajectory (history) of the base relation.

We define a DW schema as a triplet component (i.e., spatial, relational, and temporal) as follows. It is assumed that data warehouse objects include data, operator, and queries in the relational terminology for clarity and simplicity. Only spatial objects and spatial operations are redefined in the following paragraphs : spatial selection, spatial projection, and spatial join.

Assumed are a finite set of domain names and an infinite set of attributes [Abiteboul and Kanellakis, 1998 ; Voisard and David, 2002]. Types are generated from domain names, attributes, and the tuple constructors. If $T_1, T_2, \cdots, T_n$ are types, and $A_1, A_2, \cdots, A_n$ are attributes, then $<A_1, A_2, \cdots, A_n>$ is a tuple type ; if T is a type and A an attribute then {A : T} is a set type, where A : T is a named type with name A.

Spatial data are of three types : s_i : T are represented as vector format having 0 dimension (e.g., location points on the map), 1 dimension (e.g., boundary line, stream, or road), and 2 dimensions (e.g., FCP, harvest area, lake, or reservoir). In this paper reference will be made to the FCP, though other spatial objects may apply equal well.

(Definition : Geospatial DW Schema) A geospatial DW schema $\Omega = <\Sigma, \mathcal{R}, \Psi>$ follows a triplet where $\Sigma = \{s_1, s_2, \cdots, s_n\}$ is a spatial schema consisting of FCP s_i, DB schema $\mathcal{R} = \{r_1, r_2, \cdots, r_m\}$ be a set of base relation instances, and $\Psi = \{t_1, t_2, \cdots, t_l\}$ be a set of temporal instances t_i for all i representing a finite number of an index set.

(Definition : Geospatial DW instance and operation) A triplet instance $\omega = \{<s_i, r_i, t_i> |$ condition C} of Ω follows operations on the condition C. Operations are represented by the operators in terms of the corresponding instances : e.g., selection (σ), projection (Π), and join ($\bowtie$)) w.r.t. the components of triplet ω of spatial, relational, as well as temporal respectively.

(Definition : Spatial entity) We define a thematic map as a collection of spatial objects as follows. A Spatial object is derived from the real world forest. A spatial object can consist of other spatial objects, such as a stream is composed of tributaries. The objects are included in a member of homogeneous domain maps, e.g., in this case, hydrographic domains that have stream, lake or reservoir. The map is the highest level of spatial set domain and the map consists of several objects. The spatial object consists of a set of maps as layers. Below is a definition of a map schema.

Spatial objects (s_i) = {MapID : alphanumeric type, Map : Spatial type}
Map = {OID : alphanumeric type, FCP object : Spatial type
[OID : alphanumeric type, Harvest Unit : Spatial type],
Position : Spatial type, description : alphanumeric type}

Where the Harvest Units are subcomponents of a map object, called the Forest Cover Polygon (FCP), position is represented by (longitude, latitude), and description details can be included by height, direction, etc. Note that the signature is a set of function symbols, { } represents a set, < > represents a database tuple, [ ] represents optional condition, represents Cartesian product, and → represents implication.

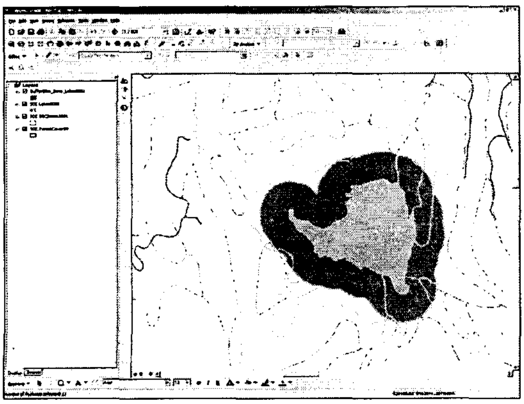
4.2 Spatial Selection

The spatial selection (σ^s) is defined as selecting a portion of a map that matches a condition of a predicate of alphanumeric attribute. The signature of spatial selection can be represented as Map × condition → Map, where condition is a predicate of one or many alphanumeric attributes : condition = predicate (Ai). The spatial selection is defined as follows:

$$\text{Spatial selection} = \{<s_i, r_i, t_i>| \; x = \sigma^s (s_i.\text{attribute } \theta \text{ Constant}) ; \; x \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$$

Note that the notation on spatial objects such as spatial selection, spatial projection, spatial join (including temporal notations) is followed by standard database representation with superscripts (s for spatial, t for temporal, no superscript for relational) for the sake of consistency.

Example 4.1 : A user query is issued as follows : '*Show Forest Cover Polygons within 50m from a lake*'. It can be represented formally : $\{<s_i, r_i, t_i>| \; x = \sigma^s (|s_i - \text{a lake}| < 50 \text{ m}) ; x \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$. The query result is represented in <Figure 4-1>.



⟨Figure 4-1⟩ Spatial Join Example

4.3 Spatial Projection

The spatial projection (Π^s) is defined as selecting a Map layer from the set of Maps whose signature is Map × MapID → Map, where MapID (an identifier of Map) is an alphanumeric attribute of the map. The result of the operation gives back a Map whose description is made of the MapID and whose spatial part is unchanged and retrievable. The spatial projection is defined as follows :

$$\text{Spatial projection} = \{<s_i, r_i, t_i>| \; x = \Pi^s(s_i.\text{attribute}), x \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$$

Example 4.2 : A query is issued : '*Project BECZone and lake layers.*' It can be represented formally : $\{<s_i, r_i, t_i>| \; x = \Pi^s(s_i.\text{MapName} =$ '*BECZone*' and $s_i.\text{MapName} =$ '*lake*'$), x \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$.

4.4 Spatial Join

The spatial join ($\bowtie^s$) is defined as generating a new map by overlaying two or more maps. The geometry of the maps should be computed by applying the intersection operation to the geometry of the participating maps, whose signature is map×map → map. The result of the operation gives back a map whose spatial location is unchanged and retrievable. The spatial join (assuming that the participating maps are (s_i, s_j), [note that the two Maps can be identical, i.e., spatial self-join]) is defined as follows :

$$\text{Spatial join} = \{<s_i, r_i, t_i> | \bowtie^s(x, y \mid x = \text{condition}, y = \text{condition}), (x, y) \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$$

Example 4.3 : A spatial selection projection and join example can be issued as follows: '*Show the joined Map of lake with Forest Cover Polygons within 50m from a lake*'. It can be represented formally: $\{<s_i, r_i, t_i>| \bowtie^s(x, y \mid x = \Pi^s(s_i.\text{MapName} = \text{'Forest Cover Polygon'}, y = s_j.\text{MapName} = \text{'lake'}) \sigma^s(|s_i - lake| < 50m)), (x, y) \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$.

Proposition (Set theoretical Spatial Operators) The spatial operators like spatial selection (σ^s), spatial projection (Π^s), and spatial join ($\bowtie^s$) are commutative, associative, and distributive.

Note that by the proposition, topological subsumptions on the operators and multiple query optimizations in terms of database literatures onto more than 2 spatial objects can be applied. For example, it would be trivial that joining by three Maps can be done without considering the topological location of Map, or it would be guaranteed that the spatial projection followed by a spatial selection and the spatial selection followed by the spatial projection are the same regardless of the order of operators. We focused on the basic operators, and other operators such as spatial merge, windowing, clipping, map overlay, and spatial aggregation operators, are not considered here [Voisard and David, 2002].

4.5 Interfaces with Spatial Objects to Temporal and/or Relational Objects

Now we can consider the integration of spatial objects with temporal and/or relational objects. We assume that those elements s_i, r_i, and t_i are orthogonal and are connected with the three subspaces Σ, $\mathcal{R}$, and Ψ. Note that this paper uses the relational algebra by Codd [1990], temporal notations by Snodgrass [1993], Slivinskas et al., [2001], spatial and spatio-temporal representations by Tryfona and Hadzilacos [1998], Hadzilacos and Tryfona [1997] and Tryfona et al. [1999].

Spatial Relational Join

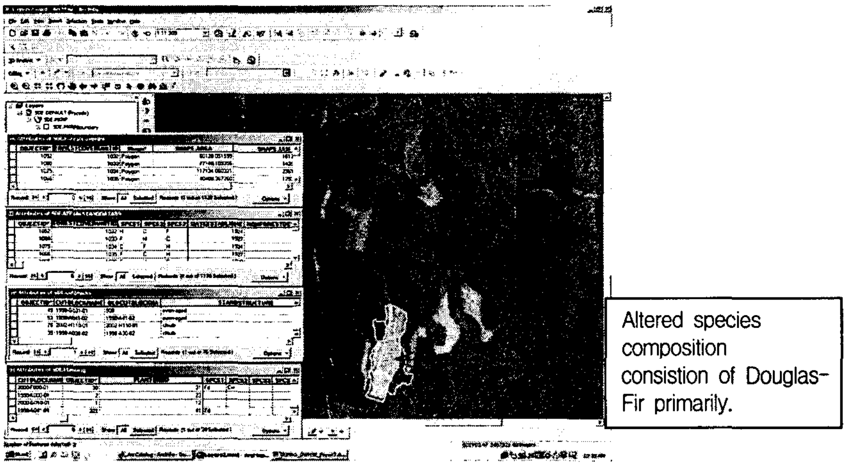
It is assumed that joining a spatial object with a relational object can be done through a fact table that has a spatial dimension and relational dimension(s). As long as the join is not done between spatial objects, the join path should connect the DW fact table. Therefore, unless the join is accomplished within the DW fact table, it is a DW relational join that is a normal theta (θ) join, by definition. For $\theta \in \{=, >, <, >=, <=\}$. The spatial relational join is defined as follows :

$$\text{Spatial relational join} = \{<s_i, r_i, t_i>| \bowtie(x, y \mid x = s_i.\text{attribute Fact.attribute}, y = r_i.\text{attribute Fact.attribute}) ; s_i \in \Sigma, r_i \in \mathcal{R}, t_i \in \Psi\}$$

Spatial Temporal Join

It is assumed that joining a spatial object with a temporal object can be done to which a fact table has a spatial dimension and temporal dimension. The temporal operators are assumed to be relational ones. Therefore, the spatial temporal join is accomplished within the DW fact table; that is a normal theta (θ) join by definition. For $\theta \in \{=, >, <, >=, <=\}$.

The spatial temporal join is defined as follows :



〈Figure 4-2〉 Spatial Temporal Relational Join result

Spatial relational join = $\{<s_i, r_i, t_i>|$
$\bowtie(s_i$.attribute Fact.attribute,
t_i.attribute
Fact.attribute) ; $s_i \in \Sigma,\ r_i \in \mathcal{R},\ t_i \in \Psi\}$

4.6 Spatial Temporal Relational Join

According to the above definitions, an integrated form of Spatial Temporal Relational Join can be suggested.

Example 4.6 : A query is issued : *'Show species composition changes on a harvest unit before (year 1992) and after cutblock (year 2003).'* It is an integrated spatial temporal join that can be represented formally: $\{<s_i, r_i, t_i>$ | Π(Tree Species.TreeName | σ(Time.Time = '2003'), TreeSpecies.TreeName | σ(Time.Time = '1992')) $\bowtie^s$(x, y | x = *'LandSat Image'*, y = 'Cutblock') $\bowtie$(x, y | x = Time.TimeID, y = Map.MapID), Map $\in \Sigma$, Tree Species $\in \mathcal{R}$, Time $\in \Psi\}$

The result of the query is that the dominant species composition of Fir, Hemlock and Cedar are altered to Douglas-Fir primarily.

5. Software Considerations

ESRI's suite of GIS application software provides solutions for integrating large GIS data objects into the relational model. With this software, GIS data are integrated with standard attribute data in a relational format.

ArcGIS is a scalable system that operates on a high-end object oriented database management systems such as Oracle and Microsoft SQL family of servers. ArcSDE is a core component in the ESRI family of application software that enables object-based spatial data access through client/server architecture. ArcSDE provides a broad range of analytical tools, which are lacking in similar GIS applications. It provides both a data model to the data warehouse, a geographic analysis engine for OLAP, data transformation, a spatial storage engine and ease of access. All things considered, ArcSDE is a more superior tool.

A few corporations who produce high-end

software dominate the market for GIS application software, and a large number of people rely on this software. However, there is also on the market open source GIS software that is largely free. Among these is GIS software called GRASS (Geographic Resources Analysis Support System), which was originally developed for land planning and site management. GRASS, like its cohorts has a more limited focus and is topic oriented [Neteler, 2000]. Hence, cannot be effectively used to maintain a STDW.

6. Conclusions and Future Work

The objectives of this paper were to develop an architectural framework for integrating geospatial and temporal data with relational information from which a geospatial data warehouse (STDW) was successfully built and implemented. The DW is considered an architectural framework for the following reasons: (1) it maintains both current and historical data ; (2) it is scalable to handle a sheer volume of data that may be collected from various data sources and users ; and (3) it is a real-world stable technology guaranteed for high performance. Maintaining both current and historic data in the same database structure requires a non-conventional spatially enabled object-relational data warehouse that integrates spatial, temporal, and relational objects. In this paper, we have developed a generic model for forestry management that can be applied to different management operations. The generic model has a fact table with five dimensions. Using the generic model, an interface has been developed that creates links to spatial, temporal and relational operators. This interface is simple and yet powerful enough to organize and maintain the integrity of the data in the STDW. The integration of spatial and temporal and/or relational objects is still a challenging issue and there remains a lot of work to be done particularly in the area of higher level data integration using more sophisticated operators than our basic operators. We also plan to extend our model to the web environment using XML technology, as our next research topic.

References

[1] Abiteboul, S., and Kanellakis, P., "Object Identity as a Query Language Primitive", *Journal of the ACM*, 45(5), 1998, pp. 798-842.

[2] Belussi, A., Negri, M., Pelagatti, G., *An integrity constraints driven system for updating spatial databases*, In Proc. ACM-GIS, 2000, pp. 121-128.

[3] Booch, G., Rumbaugh, J., Jacobson, I., *The Unified Modeling Language User Guide*, Addison-Wesley, Reading, MA, 1999.

[4] Cassidy, S., Bird, S., *Querying Databases of Annotated Speech*, In Proc, Australasian Database Conference, 2000, pp. 12-20.

[5] Chen, Y., Davidson, S. and Zheng, Y., *3XNF : Redundancy eliminating XML storage in relations*, In Proc. VLDB, 2003.

[6] Coad, P., Yourdon, E., *Object Oriented Analysis*, Yourdon Press/Prentice-Hall, Englewood Cliffs, NJ, 1991.

[7] Codd, E., *The Relational Model for Database Management*, Version 2. Addison-Wesley, 1990.

[8] Dawson, L., *The Use of Formal and Informal Models in Object-Oriented Requirements Engineering*, ICEIS (2), 2001, pp. 753-760.

[9] Erwig, M., Schneider, M., *Vague Regions*. In Proc. Symp. on Advances in Spatial Databases, LNCS 1262, 1997, pp. 298-320.

[10] Fernández, R., Rusinkiewicz, M., "A Conceptual Design of a Soil Database for a Geographical Information System", *International Journal of Geographical Information Systems*, 7(6), 1993, pp. 525- 539.

[11] Graham, I., *Object Oriented Methods*, Addison-Wesley, Reading, MA, 1994.

[12] Gray, J., "Evolution of Data Management", *IEEE Computer*, 29(10), 1996, pp. 38-46.

[13] Hadzilacos, T., Tryfona, T., "An Extended Entity-Relationship Model for Geographic Applications", *SIGMOD Record*, 26(3), 1997, pp. 24-29.

[14] Hjelsvold, R., Midtstraum, R., *Modeling and Querying Video Data*, In Proc. VLDB, 1994, pp. 686-694.

[15] Jacobsen, I., Christerson, M., Jonssen, D., Overgaard, G., *Object Oriented Software Engineering*, Addison-Wesley, Reading, MA, 1992.

[16] Jones G., and Edens R., *Automated Alignment and Annotation of Audio-Visual Presentations*, In Proc. ECDL 2002, pp. 276-291.

[17] Lim, T., Kang, S., "Effectively Managing Data Warehouse System for Geographic Woods Information," *Database Res*, 20(3), 2004, pp. 1-8.

[18] Marshall, C., *Toward an Ecology of Hypertext Annotation*, Hypertext, 1998, pp. 40-49.

[19] Neteler, M., *Advances in open GIS software*, In Proc. First National Geoinformatics Conference, 2000.

[20] Ng, Y., Melton, A. and Unger, E., *A method for constructing generalized non-normal-form models*, In Proc. 19th annual conference on Computer Science, 1999.

[21] Papadias, D., and Arkoumanis, D., *Approximate Processing of Multiway Spatial Joins in Very Large Databases*, In Proc. EDBT, 2002, pp. 179-196.

[22] Pratt, P., Adamski, J., *Concepts of Database Management*, Third Edition Course Technology, 2000.

[23] Raza, A., Kainsz, W., *Cell Tuple Based Spatio-Temporal Data Model : An Object Oriented Approach*, In Proc. ACM GIS, 1999, pp. 20-25.

[24] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W., *Object-Oriented Modeling and Design*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[25] Shekhar, S., Coyle, M., Goyal, B., Liu, D., Sarkar, S., "Data Models in Geographic Information Systems", *CACM*, 40(4), 1997. pp. 103-111.

[26] Slivinskas, G., Jensen, C., Snodgrass, R., "A Foundation for Conventional and Temporal Query Optimization Addressing Duplicates and Ordering", *TKDE*, 13(1), 2001, pp. 21-49.

[27] Snodgrass, R., *An Overview of TQuel*, Temporal Databases, 1993, pp. 141-182.

[28] Stonebraker, M.. *Object-Relational DBMSs : Tracking the Next Great Wave*, Morgan Kaufman Publishers, 1998.

[29] Sudarsky, S., Hjelsvold, R., *Visualizing Electronic Mail*, In Proc. IV 2002, pp. 3-9.

[30] Takayama, H., and Couclelis, M., "Map Dynamics Integrating Cellular Automata and GIS Through Geo-Algebra", *International Journal of Geographical Information Science*, 11(1), 1997, pp. 73-91.

[31] Tryfona, N., Busborg, F., Christiansen, J., *starER : A Conceptual Model for Data Warehouse Design*, In Proc. DOLAP, 1999, pp. 3-8.

[32] Tryfona, N., Hadzilacos, T., *Logical Data Modeling of Spatio Temporal Applications Definitions and a Model*, In Proc. IDEAS, 1998, pp. 14-23.

[33] Vasudevan V., and Palmer M., *On Web Annotations : Promises and Pitfalls of Current Web Infrastructure,* In Proc. HICSS, 1999.

[34] Voisard, A. and David, B., "A Database Perspective on Geospatial Data Modeling", *IEEE TKDE,* 14(2), 2002, pp. 226-243.

저자소개

이 우 기

서울대학교 산업공학과 학사, 석사 및 박사를 취득하였고, 현재 성결대학교 컴퓨터학부 부교수로 재직하고 있다. 카네기멜런대학교에서 MSE 과정 수료, University of British Columbia 교환교수, ISI Canada에서 TEFL deploma를 취득하였으며, 한국경영과학회에서 최우수논문상 수상(2004년)했다. 관심분야는 데이터웨어하우스, 정보검색, Web Structure Mining 등이다.

◈ 이 논문은 2004년 8월 14일 접수하여 1차 수정을 거쳐 2004년 11월 6일 게재확정되었습니다.