SPATIO-TEMPORAL DATA MODEL AND QUERY LANGUAGE FOR BIODIVERSITY DATA

Mohd Taib Wahid[1], A.Z.M Kamruzzaman[2], Abdul Rashid Mohamed Shariff[3], Suhaimi Napis[4]
[1]taib@fsksm.utm.my, [2]akzaman@siswa.utm.my, [3] rashid@eng.upm.edu.my,
[4]suhaimi@putra.upm.edu.my

[1&2]Faculty of Computer Science & Information Systems
University Technology of Malaysia, K.B. 791, 81310 Skudai, Johor, Malaysia
Tel: (607)-5532422, Fax: (607) 5565044

[3]Faculty Engineering, [4]Institute of Multimedia and Software
Universiti Putra Malaysia 43400 UPM Serdang, Selangor, Malaysia
Tel: (603)-89467801, Fax: (603)-89483514

Abstract: *A Spatio-temporal data model and query language for biodiversity is growing importance to the biodiversity data management, forest and environment control. Spatio-temporal data model and query language have received much attention in the database research community because of their practical importance and interesting challenges they pose. This paper discusses about spatio-temporal information, query language and problems with existing models. In this paper our main objective to minimize the extension required in SQL language. This paper also focuses on the unified models of space and time using object-relation approach. In particular, we propose a conceptual object-relational spatial temporal data model based on Donna J. Peuquet's pyramid framework. Spatio-temporal queries are expressed by user defined aggregates, such as duration and during temporal relationships. Therefore, query language (such as SQL^{ST}) or other relational languages are required to support spatio-temporal queries for the biodiversity data model.*

Keywords: Spatio-temporal, Data model, Query language, Object-relational, SQL^{ST}.

1. Introduction

Time and space are important aspects of all real world phenomena. Database applications must capture the time and space varying nature of the phenomena they model. For example, biodiversity data collector needs to maintain a record of the data he collects from the area and when he has collected. This information certainly varies over times: data collects during particular time; data maybe collected same time by several collectors from different places; and also data collects periodically such as daily, monthly and yearly. Many applications demand for spatial temporal support [1], such as biodiversity applications, GIS applications, traffic management systems computing traffic network and traffic volume at different time, forecast-prediction systems recording weather change process, etc. But existing database techniques couldn't support spatial temporal information management well [2].

After almost two decades research, representation of space and time in databases and functional applications are still problematic [3]. This paper presents a universal object-relational framework for spatial temporal data modeling. Spatial temporal data modeling aims to extend the existing data models to include space and time in order to better describe our dynamic real world.

Problem with previous data models is that spatial and temporal aspects of databases are modeled separately [4], [5]. Spatial database focuses on supporting geometries [6], while temporal databases focus on the past state [7]. But in many circumstances, such as environmental monitoring, resource management, transportation scheduling, etc, spatial and temporal attributes should be connected together. Many current systems can handle only one aspect of space and time. Spatial systems always fail to cater for many temporal aspects in a dynamic environment [8]. Though many researchers have found the necessity of integration of space and time in one environment, by far, little such work has been done.

Another problem is representation of data should be natural to human. The structures of space and time are identified as essential for the realization of cognitive systems [12]. According to Donna J. Peuquet and her group [13], models of spatial temporal data in geographical database representations must incorporate human cognitive principles. Human knowledge of the dynamic geographical world comprises of three different (and interrelated) subsystems that handle *what*, *where* and *when* aspects of object properties [10]. Theme-based model, location-based model and time-based model separately describe one subsystem. From this view, these three kinds of model are all single semantic models. According to human cognitive principle, the model we build should cover all of these 3 subsystems. Donna J. Peuquet and her group's pyramid framework show how to incorporate three subsystems of human cognition into data modeling. From above analysis, we should build a unified infrastructure that integrate space and time, mediate discrete and continuous representation, describe when/where/what systematically, and be extensible to various applications. A multi-semantic model cannot ultimately be generated from extension of current single semantic models such as ER model and location/time/theme-based model. Object relational approach with its characteristic of inheritance and aggregation is capable of capturing the various notions of space and time and reflecting them into a single framework extensible to different applications.

Many database systems concentrate on the definition of a particular spatial temporal model that is related to certain application. The result is that more and more different models appeared. Each model focuses on a specific set of spatial temporal features [9]. When encountering other features and applications, the model doesn't work. So what we should do is to build an overall framework that can be extended to various applications, not driven by various applications generating different models [10]. Furthermore, application specific modeling will be more efficient if it is based on a generic model [11].

This paper focuses on the unified models of space and time using object-relation approach. In particular, we propose a conceptual object-relational spatial temporal data model based on Donna J. Peuquet's pyramid framework. However, even in the latest generation of OR (Object Relational) systems, extensibility comes with many limitations and requires significant expertise and programming effort. Indeed, the main extensibility, mechanism of OR (Object Relational) systems is allow SQL queries to call external functions coded in a procedural language (such as C or Java). Spatio-temporal queries are expressed by user defined aggregates, such as duration and during temporal relationships. Therefore, query language (such as SQL^{ST}) or other relational languages are required to support spatio-temporal queries for biodiversity data. In addition, spatio-temporal primitives and SQL^{ST} allows user to introduce additional extensions to the data model and query language.

This paper is organized as follows: Section 2 discusses about spatio-temporal data model and related works and its requirement. In Section 3, discusses about data modeling approach, and data modeling techniques. In section 4, discuss about conceptual structure of biodiversity data model. In section 5, discuss about query languages, temporal selection and projection of spatio-temporal queries, and aggregations of temporal queries. Also this section tried to examine the spatio-temporal queries for biodiversity data. We conclude, in Section 6, with the summary of this work and future research plans.

2. Spatio-temporal Model related works

Spatio-temporal data models and query languages are a topic of growing interest. A spatio-temporal database is a database that embodies spatial, temporal and spatio-temporal database concepts and captures simultaneously spatial and temporal aspects of data. It deals with the data those changes with time such biodiversity data and geographical data.

Spatio-temporal databases have been the focus of considerable research attention [8]. For example, proposals have been made that support the conceptual modeling of spatio-temporal plant biodiversity data model [18], the classification of spatio-temporal applications [14], the indexing of spatio-temporal data [15], and the modeling of moving objects [16]. However, despite progress in certain aspects of spatio-temporal modeling and implementation, there are still very few complete prototypes of spatio-temporal databases, far less products that provide effective support for applications tracking changes to spatial and aspatial data over time.

However, modeling the spatio-temporal phenomenon remains a complex task due to the large diversity of the entities and phenomena and their possible evolution in the real world. We need a data model and corresponding technologies to depict the dynamic entities and phenomena. Basically, models of this kind should treat an entity as a whole and can provide us as much more rich information as possible, not only data about its current state, but also other data about its past evolution.

A great number of data models have been suggested for a spatio-temporal system. Some of them are based on the raster approach, others vector approach and yet other concepts can be applied to both vector and raster data. Some models are based on the object-oriented and object-relational approach. As was pointed out in Table 1, researchers have been trying to incorporate the time component in the data models while maintaining their original design.

Table 1: Summary of the data models

Models	Problems
Space-time cube	Individual Geographic Feature
Space-time cube, Triad model	No stored topology
Space-time cube, Triad, Snapshot (vector, Raster), Space-time composite.	Large amount of data
Space-time cube, Update Snapshot (vector, raster).	Complex Algorithm to recover topology
Space-time cube, Snapshot (vector, Raster), Update Snapshot (vector, Raster), 3D/4D.	Complex Temporal Queries
Snapshot (raster, vector)	Data inconsistency
Snapshot (vector, Raster), Space-time composite.	Difficult to detect change
Space-time cube, Triad, Snapshot (vector, Raster), Space-time composite.	Data Redundancy
Space-time composite, Snapshot (vector, Raster), Update Snapshot (vector, Raster)	Poor Temporal Resolution
Space Time Composite, 3D/4D, Triad	Complex Model
Relational	Difficult to represent Geographical Data
Snapshot (vector, Raster), Update Snapshot (vector, Raster)`	Time is not included in the Model
Space-time cube, Triad, Space-time composite, 3D/4D,	Not Implemented

Models	Problems
Object-Oriented.	
Spatio-temporal Object-relational Model(STORM)	Not Implemented

There are many models that have been proposed for incorporating temporal information into spatial databases as in table 1. Some of these models treat time as an attribute of spatial objects while others treat time as a separate dimension. Some of the main models are only briefly introduced here.

This data model is similar to the boundary representation mode in a GIS. Few years back space-time cube model developed [26] to describe how the hydrology data will be store and retrieve in relational database. There are three main coordinate: Feature Identity (FID), Attribute Identity (DID) and Date and Time (TID). These coordinates are use to determine the data that stored in the database. FID refers to data value that had measured or value key for spatial data. DID refers to key that describes about FID that is key for spatial data that had taken. TID is a key for the time the data is taken. TID is not referring to spatial data but only refer non spatial data (temporal data) such as rainfall and water level. These features will be referred when accessed data from the database.

This data model can make data analysis and manipulation efficiently, still lacking to perform the best data retrieval time, Data Redundancy Problems, representation of real world phenomena and the model itself cannot handle the spatial data due to constraints in relational approach. The main problem with cube model is not suitable for Geographical Phenomenon, required 3D space and temporal queries are difficult

The *snapshot approach* stores (as in Figure 1) all versions of the map as series snapshots or time-slices. This approach represents a state but does not represent the events that cause changes. Instead of storing the change, its stores the states of the whole map or layer. This quickly leads to large data redundancy, and has serious implications for data management and temporal analysis. The *update model (*or *base state with amendment)* stores only the changed objects but not a completely new snapshot of the whole data. Changed objects are represented as versions. This reduces data redundancy and allows object history to be traceable.

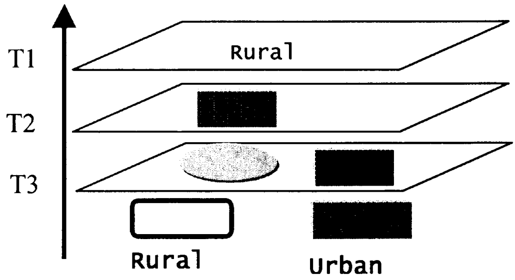


Figure 1: A snapshot model of timestamped layers

The weakness is that two snapshots contain much the same data and changes do not exist as explicit entities. In addition, the root problem is that snapshots represent states but not the events changing one state into the next. Judging from time-stamp layers, the snapshot models are based on representations of raster or vector styles

The Space-time composite model stores all past and present features by overlapping all timestamped layers to produce a space-time composite layer [19]. This model is not very different from the update model except that past and current features are stored in the same layer. Event-based spatiotemporal data model; are the examples of the models used the object-oriented technology. Scientist proposes event-based spatio-temporal data model by using the idea of Triad framework data model.

The space-time composite approach requires re-construction of thematic and temporal attribute tables whenever operations involve any changes in spatial objects (shape, size, or configuration). Consequently, geographic entities tend to be decomposed into fragments of spatial objects. For example, a wildfire event can be represented by a set of polygons with descriptions of burn severity and burn time.

The four basic models described above have been adapted to facilitate various kinds of analysis and case studies through the years. Location-based and feature-based representations are not well suited for the analysis of overall temporal relationships of events and patterns of changes through time such as the analysis of raster data. The ESTDM (Event-based Spatio-Temporal Data Model) is an example of a model developed to facilitate the analysis of raster-based data. The ESTDM model groups time-stamped layers to show temporal observations of a single event in a temporal sequence. The most significant feature of the ESTDM model is its capabilities to support both spatial and temporal manipulations on data. However, since the ESTDM is based on raster data, modeling a vector-based system using this model will be difficult since changes stored by the model are based on grid cells, not spatial topology. Our application requires the storage of both raster and vector temporal data. Concepts presented in the ESTDM model will be useful in the design and implementation of our spatio-temporal database.

After the analysis of all the above given models and problem with those models stated above. After the review presented here by no means represent the complete research work that has been done in the field of biodiversity data model and spatio-temporal data, but it representing subset of such data models. However, this review shows that although several data models have been proposed and some of them have been implemented, there is no one which is universally accepted. The main reason of this, no data models were not originally designed to support time. As we point out, researchers have been trying to incorporate the time component in these data models while maintaining their original design [20].

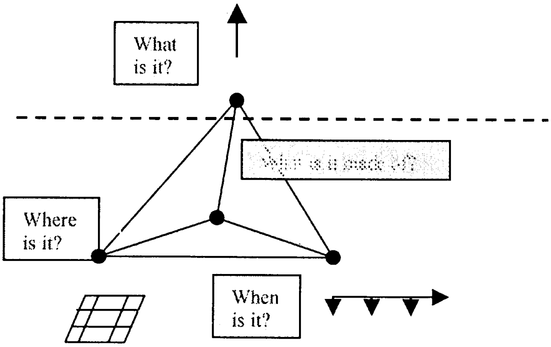
The main reason for the lack of universal data model is that, models were not originally designed to support time. Researchers have been trying to incorporate the time component in data models while maintaining their original design. To design a biodiversity data model to support spatio-temporal data, we have chosen pyramid frame work to support a biodiversity data model design. The challenge consists in the design of a spatio-temporal biodiversity data model, in parallel with the development of appropriate databases, the development of efficient spatio-temporal biodiversity query languages

3. Data Modeling Approaches

3.1 Pyramid framework and event based approach

In the design of object-relational biodiversity data model to support spatio-temporal data, pyramid frame can make enhancement to support a biodiversity data model design. A conceptual framework (also known as pyramid framework) was designed which guide the implementation of the semantic geographical information system (GIS) data model [23]. Conceptual frame work (pyramid) is interrelated with two separate parts one is data component and another is knowledge component (Figure 2). Data component can be divided into three categories: location (position in the spatial three dimensions), time (position alone a time line) and theme (a set of observations, measurements, or attribute values associated with a particular location and time). Data components stores spatio-temporally referenced observational data such as *spectral, climate*, *vegetation* and *other environmental attributes* that maybe queried and visualized to reveal embedded spatio-temporal pattern and relationships. Data component can be computerized as a multi-dimensional, spatio-temporal referenced 'hyper-cube' of observational data that is similar to the 'feature space' concept commonly cited in analysis of remote sensing imagery.

Knowledge component stores information about higher-level semantic 'objects' the geographic entities or process that are describe by the data. Information concerning on object's location, time and composition. All the objects are also placed within two hierarchical relationship structures central to cognitive knowledge representational and object-oriented modeling, plant taxonomy (generalization) and partonomy (aggregation). The taxonomy structures groups similar objects within a category and stores a rule-base that describes how those objects may be identified within the data space. These rules may be derived from expert knowledge or from inductive analysis of the observation data. In an object-relational data model consists of a set of *object classes* (of different *types* or *schemas*). Each object class has an associated set of *objects*; each object has a number of *attributes* with values drawn from certain *domains* or *atomic data types*. Of course, there are additional features, such as object valued attributes (Oid), methods, object class hierarchies, etc. Besides objects, attributes describing *geometries* including *time* are of particular interest. Hence we would like to define collections of *abstract data types*, or in fact *many-sorted algebras* containing several related types and their operations, for spatial values changing over time. This section presents a simple and expressive system of abstract data types, comprising data types and encapsulating operations, which may be integrated into a query language, to yield a powerful language for querying spatio-temporal data.



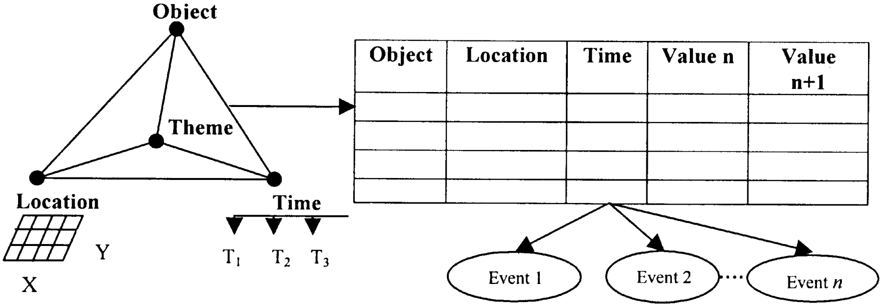


Figure 3: From the framework data transform to the table and time modified as events

To support temporal data, system needs another model such as event-based model. Pyramid system is a model to support multi-dimensional, spatio-temporal and geo-graphic objects (such as location and space). All the objects are also placed within two hierarchical relationship structures central to cognitive knowledge representational and object-oriented modeling, plant taxonomy (generalization). The plant taxonomy structures groups of similar objects within a category and stores a rule-base that describes how those objects maybe identified within the data space. Pyramid framework consists of main three components; objects, location and time. These features will be referred to accessed data from the database. This framework can be converted to table in stated figure 3.

3.2 Modeling Temporal Information

Time is an integral piece of information in the description of an object and / or process in a constantly involving real world. The database requirements (indicated as gray arrows) directly lead to the specification of data types, operations, and integrity constraints for real world object (in this case biodiversity data). As an example in a spatial data model is implemented within a relational DBMS (i.e., Database Management System), which raises database requirements for the data model to consists of three data model components –spatial data types (e.g., points, lines, polygons), spatial operations (e.g., collection of biodiversity data from different places), and relational integrity constraints (including accessing methods). Specific user needs as functional requirements are proposed for different applications. For example, biodiversity data and forest area maps are specified as spatial data types (i.e., point, lines polygons) according to the need from exploring spatial and temporal data relevant to biodiversity data interaction. Operations and constraints corresponding to the interaction can also be developed for handling and facilitating biodiversity data management and manipulations.

Alternatively, a data model **M** can be described, modified, and accessed in a uniform way, is expressed as **M= (DT, OP, and C)**. The data model **M** includes three components data types **DT**, query enhancing operation **OP** and integrity constraints **C** [21]. A corresponding spatial data model enhancing the three components of the data model can be expressed as $\mathbf{M^S= (DT^S, OP^S, C^S)}$. Meanwhile, as a spatial-temporal data model extended from $\mathbf{M^S}$ is denoted as $\mathbf{M^{ST} = (DT^{ST}, OP^{ST}, C^{ST})}$ where time is stamped o every component of the spatial data model. Its data types accommodate time-varying spatial data. Its query operations are redefined with spatial, temporal and spatio-temporal functions [22]. Additionally, each integrity constraint in the spatial data model $\mathbf{C^S}$ has a temporal version $\mathbf{C^{ST}}$.

The data types **DT** are the first, compared with the operations OP and the constraints C, to reflect changes enforced for applications. The literature reveals that spatio-temporal data models developed form spatial data models usually start with the design and implementation of new data types and data structures. Query languages and access methods, although they are components equally important to the data types in the data model, are the next for modification.

Usually, extending the data structures with time attributes does not cause any severe problems. In temporal databases, it is customary to include a number of different time dimensions. The most common kinds of time are: valid time, transaction time, record time, event time, future time, and end user-defined time. We adopt the taxonomy of time terminology given in [23] for valid time and record time in [24] for event time. Valid time consists of the start time and end time used to delimit the records of an object history. Start time is the time when new object become valid in the database. The end time is the time when information of an object is recorded in the database. The event time is when the event about an object actually occurs.

In our approach, each object is stamped with either a valid time interval, <start time, end time>, or an event time. The reasoning behind making a difference between them can be explained by the following examples. A data collector may collect biodiversity data for few days from 10th January 2003 to 20th January 2003 which can be conveniently represented by a time of the interval. However a flora or fauna image / data represents of plant and animal species in particular growth stage. It is valid when image / data have been collected and there fore, it is more appropriate to describe it by an event time.

3.3 Object- relation Data Model Techniques

The object relational model (ORM) is a way of describing or representing objects, classes of objects, relationships between objects and classes, and real world memberships. The ORM can be considered the static part of object-oriented systems modeling. That is, the ORM describes the "database" of a model: what objects may exists, what objects classes they belong to and what relationships exists between objects. In the object-relational model also consists of object, object classes, relationships relationship sets and constraints.

3.3.1 Objects Types:

An object includes a static part, a set of data, and a dynamic part, a set of procedures manipulating this data. an object is defined by its behavior, and represented by the set of its procedures, rather than by its structure. An object has a unique and immutable identified that differentiates it from other objects. An object identity (OID) comes into being when the object is created and can never be confused with that on an other object even if the original object has been deleted. The OID never changes even if all the properties of the object change. It is depend on the object's state. An object identifier further has the following characteristics: 1) it identifies an object not only in an extension of a class, but also in database, 2) it is constant throughout the lifetime of the object, 3) it is not visible to the user, and 4) it is created and maintained by the system.

An object has a set of attributes (Instance variables), and relationship with other objects. An attribute models the state of an object. Spatial and/or temporal information may be associated to objects, independently from the characteristics of their attributes. Consequently, an object type can be plain (neither spatial nor temporal), spatial, temporal, or spatio-temporal. The term object, however, is fundamentally different from the term entity. Entity is concerned merely with data. We typically store a record from each entity. Object concerned with data and the methods with which it is manipulated. For example, in biodiversity data model different object represents the different real world entities as shown in Figure 4.

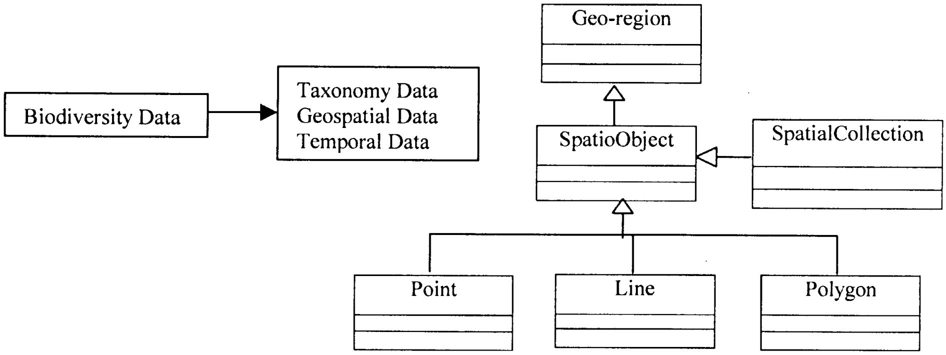


Figure 4: Object type and spatial data as object

There are two types of object diagrams: class diagrams and instance diagrams. A class diagram is a schema, pattern, or template for describing many possible instances of data. A class diagram describes object classes. An instance diagram describes how a particular set of objects relates to each other.

3.3.2 Abstraction mechanism

The object-relational data model is built on few basic concept of abstraction: relationship, generalization, association and aggregation. These abstraction concepts may be described by the means of relationship sets. A relationship set is a set of relationships, all of which have the same role to between objects. In an object-relational diagram, a relationship set is represented by lines connecting associated objects. As for objects, relationship may be located in space and time, via two attributes. In this case they are referred to as spatial and/or temporal relationships.

3.3.3 Generalization and specialization

Generalization and specialization represents the "Is-a" relationship set, an essential element of object-relational paradigm. The main idea in generalization and specialization is that one object class (specialization) is a subset of another (the generalization). The process of generalization consists of putting classes together in what is called a superclass. Generalization enables us to perceive that all instances of more specific concept are also instance of general concept. Specialization, on the other hand, breaks apart a class by differentiation of properties or forms a set of lower class called subclass. Specialization allows us to describe each subclass of a more general class by specifying only the additional details necessary for us define. Subclass describes a specialization of the superclass.

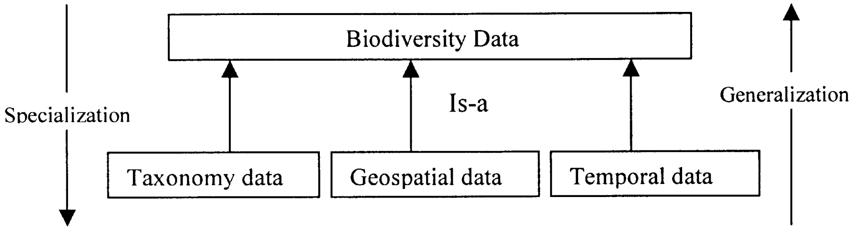


Figure 5: Generalization and Specialization

The terms superclass and subclass characterize generalization and specialization. It is important to note that superclass a subclass are abstraction of the same object, and do not describe two different objects. The direction of the "Is-a" relationship goes from the specialization to the generalization (Figure 5), that is, it may be stated as, "Specialization Class is-a Generalization Class".

3.3.4 Association

An association describes a group of links with common structure and common semantics. A link is a physical or conceptual connection between object classes. All the links in an association connect objects from the same classes. Association and links often appear as verbs in a problem. An association describes a set of potential links in the same way that a class depicts a set of potential objects. In fact, the notion of an association is certainly not a new concept. Associations are widely used throughout the database modeling community for years.

In an object-relational model diagram, association is also known as the "member of" relationship set (Figure 6). For example, clubs and teams are usually modeled using association. Accordingly, an object in the Taxonomy data object class is a set of objects from the Species, IdentCharacteristic, IdentLevel. The relationship set is read "Species is a member of Taxonomy Data".

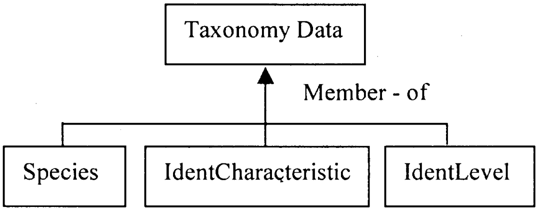


Figure 6: Association

3.3.5 Aggregation

An aggregation represent the "a -part-of" or "is subpart of" relationship set, in which objects representing the components of something are associated with an object representing the entire assembly. Aggregation is a strong from of association in which an aggregate object is made of components. Components are part of the aggregate. The aggregate is semantically an extended object that is treated as a unit in many operations, although physically it is made of several lesser objects. A single aggregate object may have several parts: each part-whole relationship is treated as a separate aggregation in order to emphasize its similarity to association.

However, there are may be some discussions of the differences between association and aggregation. Aggregation is a spatial form of association, not an independent concept. Aggregation adds semantic connotations in certain cases. If two objects are tightly bound by a part-whole relationship, it is an aggregation. If the two objects are usually considered as independent, even though they are often linked, it is an association. But, the decision to use aggregation is often arbitrary a being matter of judgment. Often it is not obvious if an association should be modeled as an aggregation [25]. Most of the objects dealt with in biodiversity databases are composite, i.e., they exhibit an aggregation hierarchy. They follows are some of the aggregations in the biodiversity data models:

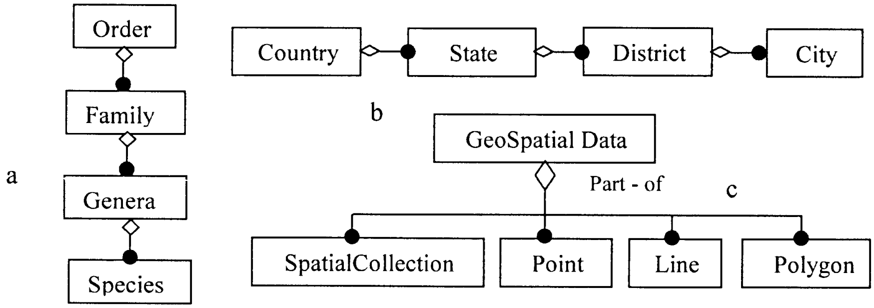


Figure 7: Aggregations in biodiversity data model

An order contains several families. A family in turn contains several genuses and genus contains closely related species. Since a family cannot belong to two orders, the order-family relationship is modeled as an aggregation rather than an association. Similar arguments hold for genera and species also. Hence, taxonomy hierarchy is modeled as a four level containment hierarchy in biodiversity data model (figure 7a)

A country contains number of states, each states is made up of districts and districts made up of cities. The relationship is also modeled as a four level containment hierarchy. Similarly, the association between a city and the forest department in the city and the association between a country and a forest are modeled as aggregations (figure 7b). Figure 7c is an example of an aggregation. Geospatial object in as aggregation of the SpatialCollection, point, line and polygon that are the part of it.

3.3.6 Constraint

Most of the system can be model with objects, objects classes, and relationship sets. Constraints are functional relationships between entities of an object relationship model. Constraints provide one certain for measuring the quality of an object relational model. In object relational model, a constraint restricts the membership of one or more object classes and relationship sets. The object relational model allows analysis to express several different types of constraints. However, in the object behavior model and interaction model, it may be useful to use this constraint in order to describe some restricted condition.

4. Conceptual Biodiversity Data Model Structure

After explanation of several components of object relational model, this section shows some relationships among the components of object relational model for biodiversity. The proposed spatio-temporal data model is described by means of collection of one or more class diagrams that form the object model and connected with its related relational tables to form an object-relational data model.

The class diagram which describes the structural characteristics of the proposed spatio-temporal data model is presented in Figure 8. It shows that five major classes were identified and incorporated into the data model: BiodiversityFeature (BF), BiodiversityData (BD), Spatial Data (SD), Taxonomy Data, and Temporal Data (TD).

Object-relation is the most powerful techniques to design object-relational data model. The design of an object-Relational Data Model generally involves two basic steps. First, structural modeling, which describes the structure of similar objects in terms of classes, their similarities and differences (generalization), the associations or connections among these classes, and the structural constrains. The second step corresponds to behavioral modeling, which describes the behavior of different classes in terms of operations and relationships.

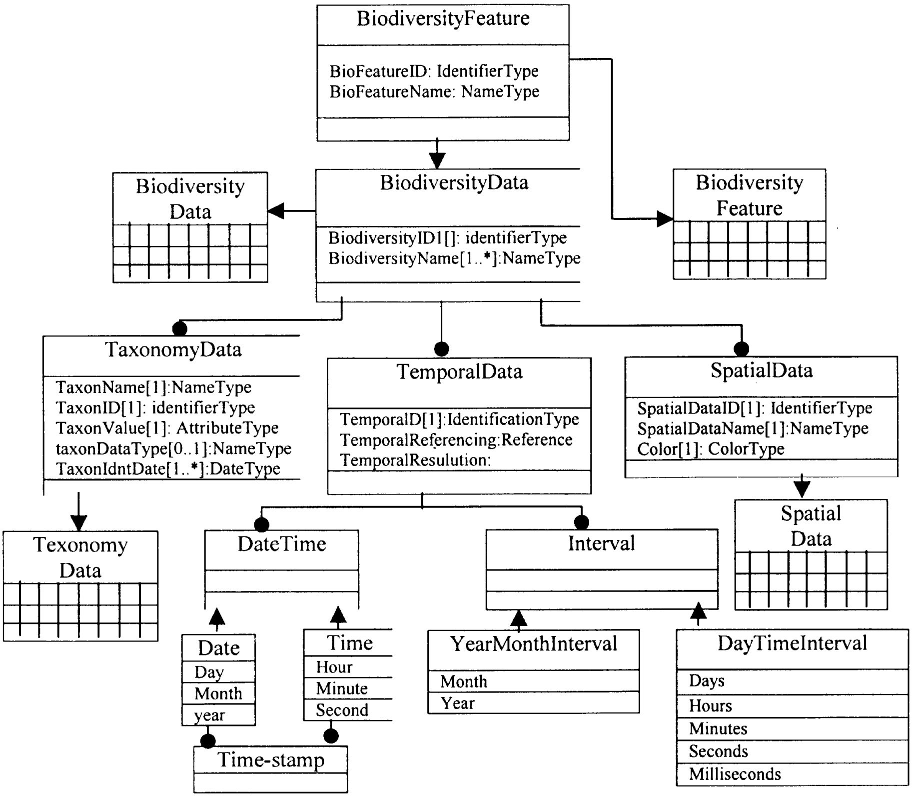


Figure 8: Detailed Structure of Spatio-temporal Object-Relational Model for Biodiversity

5. Querying with Spatio-temporal Operations

Our intention is not to devise a new spatio-temporal query language from scratch but to appropriately extend the widespread database query language SQL. The main focus is on integrating developments. We profit from the fact that the underlying data model rests on the Abstract Data Type (ADT) approach which necessitates only conservative extensions to SQL which are: (i) a set of basic spatiotemporal predicates, and (ii) an extension mechanism for new, more complex spatio-temporal predicates. We call this extended query language *STQL* (*Spatio-Temporal QueryLanguage*). All added functionality is captured by ADT objects and operations. The benefit of this approach is the preservation of well known SQL concepts, the high-level treatment of spatio-temporal objects, and the easy incorporation of spatio-temporal predicates.

The query facility of SQL is provided by the well known SELCT FROM WHERE clause. The integration of predicates like "<" or "<>"" for standard data types such as integers or strings is well understood. In particular, there are only a few of them which allow one to include them as built-in predicates. When considering more complex and more structured data such as points, lines, or regions, one can try to systematically derive all reasonable predicates. Temporally enhanced object-relational system can express temporal queries as powerful as those of TSQL2 with only minimal extensions to standard SQL by using construct such as user-defined functions and table expressions supported in Object-Relational systems.

We will consider queries from two (simplified) application scenarios. The first scenario is related to biodiversity data (such as flora, fauna) management which pursues the important goal of learning from past biodiversity data and their evolution. We assume a database containing relations with schemas

Flora (FloraName: VARCHAR2, Territory: Species name)
Flora TypeType (ForestArea: VARCHAR2, Extent: species)
Area (ForestArea: VARCHAR2, Location: Point)
Collectors (CollectorName: VARCHAR2, Actor: Person)

The relation flora records the location and the development of different species (attribute Territory) growing and shrinking over time through clearing, cultivation, and destruction processes, for example. The relation of flora collects from different places and it growth of different data by different people to their extinction (attribute Extent). The relation collectors describes about each flora being on duty from their start at the collection data up to their return (attribute Location).

The second scenario, finally, relates to a database about the migration of birds in order to explore their behavior patterns over the years.

Birds (Swarm: VARCHAR2, Movement:Point)

5.1 Temporal Selections

The first queries refer to the simple temporal range query. A *temporal selection* extracts the value of an object at a certain instant or the temporal development over a certain period. We can then ask queries like "what happened to the data between days January 10, 1995 to January 20, 1995?"

```
SELECT Flora_ID FROM flora
WHERE (((Flora_collection_year) Between # 01/10/1990# AND #01/20/1995#));
```

This query shows the functional character of a spatio-temporal object by determining the value of the object at a certain time through a simple function application. A more general version of this query asks what happened to the data between January 10, 1990 and January 20, 1995.

```
WHERE (((Flora_collection_year) Between # 01/10/1990# AND #01/20/1995#));
```

The "Between" notation specifies a range of time values, that is, a time interval. If a spatio-temporal object is applied to a time interval (or a collection of disjoint time intervals separated by AND), this expression yields a spatio-temporal object restricted to that time interval (function restriction).

5.2 Projections to Space and Time

Projection operations on moving objects map either to their spatial or to their temporal aspect. Assume that we are interested in the geometric locations where the data was changed at the year January 20, 1990. These can be obtained by:

```
SELECT Flora_ID, AreaName
FROM Flora, Area
WHERE (((ForestArea =#Johor# AND #01/20/1990#));
```

This operation computes the *spatial projection* of a spatio-temporal object for the Johor forest area. For an evolving region the trajectory operation returns an object of the spatial type *region* which results from projecting the union of the region values for the Date 20th January, 1990.

The next query asks for the lifespan of a spatio-temporal object: "How many times data have been collected from Johor before 1995?"

```
SELECT COUNT (Flora_ID) AS Ex1, COLLECTION_DATE AS Ex2
FROM Flora
GROUP BY COLLECTION_DATE
HAVING (COLLECTION_DATE <CONVERT (DATETIME, '1990-30-12'))
WHERE Area = "Johor"
```

The count operator collects the times when the area Johor is defined (*temporalprojection*). In this way inverse temporal functions can be computed. The duration operation computes the length of an interval or of several intervals.

5.3 Aggregations

The following query inquires about the largest collection of flora areas.

```
SELECT Area (max (Extent)) FROM ForestArea
WHERE Type = "Flora"
```

The query demonstrates an example of a *spatio-temporal aggregation* operation max which is an extension of the well known aggregation operator in SQL of the same name. It is here applied to a collection of evolving regions contained in a relation column and computes a new evolving region. Internally, this operator is based on a binary function $\boldsymbol{MAX_{ST}}$ applied to two evolving regions $\boldsymbol{R_1}$ and $\boldsymbol{R_2}$ 'and yielding a new evolving region in the following way:

$$MAX_{ST}(R_1, R_2) := \{(t, r) \mid t \in \text{time} \wedge r\ MAX_{geo}\ (R_1\ (t), (R_2\ (t))\}$$

This definition uses a function $\boldsymbol{MAX_{geo}}$ which is applied to two regions R_1 and R_2 and which returns larger of both regions.

$$MAX_{geo}(R_1, R_2) = \begin{cases} \underline{r_1} & \text{If area } (r_1) > \text{area } (r_2) \\ r_2 & \text{otherwise} \end{cases}$$

Altogether this means that for 2 evolving regions $R_1, \ldots, R_n$ we first compute the evolving region R = MAX_{st} (R_1,....MAX_{st}(R_{n-1}, R_n)...). Afterwards, we apply the raise area of R, which computes the area of R at all times as a temporal real number.

Peuquet and Wentz (1994) propose three classes of queries on a spatio-temporal database. The first is change in an object or feature. The second is change in the spatial distribution of an object or set of objects. The third is the temporal relationships among multiple geographic phenomena. But most current languages which develop temporal query languages are a result of extending an existing query language based on business type applications. There are, however, additional temporal query language requirements in biodiversity systems, because biodiversity data changes over temporal factor rather than spatial issues. The problems of biodiversity data are closely associated with spatial matters. To examine the relationship between time and a player (who observe and collects data) or location of biodiversity area, we consider the following questions:

- General temporal query: when and who collects data? *Time + Participant*
- Specific temporal query: how much data has collected at time T?
- Temporal range query: how much data change before T1, between T1 and T2, and after T2?
- Spatio-temporal query: who, when, and where collection take place at time T1, between T1 and T2 and after T2? *Time + Location + Participant*

There are few conceptual differences among temporal queries. But, spatio-temporal queries are very diverse. The second query considers a time instant and other three associated with time intervals. An important question is whether we need to represent more details of the timeline in a biodiversity system. Even if it is always possible to do that, it is seldom needed. Because the importance lies with time instant and the relationship among time intervals are most often used in GIS. Temporal events and intervals can be represented by a discrete and totally ordered time model. Judging from the requirements of temporal models, they should adequately represent and manage real world entities as they evolve over time. An entity assumes different values over time. The set of these values from the temporal history of the entity.

6. Conclusion

Data model has been developed either spatial and temporally. In this paper we have discussed issue of spatio-temporal and data modeling and query language. A number of data models have been suggested for a spatio-temporal system. Some of them are based on the raster approach, others on the vector approach and yet other concepts can be applied to both vector and raster data. Some models are based on the object-oriented paradigm, an approach that is capable of integrating both vector and raster data into one data model. This paper takes into consideration the development of spatio-temporal data model for biodiversity data and minimizes those problems. For the first experimental investigation purpose, we have tried to applied johor forest data for the suitability of the model for handling spatio-temporal data. For this purposes, we inserted and manipulated the biodiversity data through proposed model and perform a specific set of pre-defined temporal and spatio-temporal queries on it as discussed above. The proposed model supports well the spatio-temporal data to perform spatio-temporal queries.

Experimental works still under process to obtain better result than previous experiments. Few more steps to be done within few months such as comparison between proposed data model and existing model, user interface design and system development.

References

[1] Spaccapietra, S. (2001). "Editorial: Spatio-Temporal Data Models and Languages." Geoinformatica 5(1): 5-9.

[2] Tryfona, N. and C. S. Jensen (1999). "Conceptual Data Modeling for Spatiotemporal Applications." Geoinformatica 3(3): 245-268.

[3] Langran, G. (1992), "Time in Geographic Information Syetems:, Taylor & Francis, London

[4] Sellis, T. (1999). Research issues in Spatiotemporal Database Systems, Springer-Verlag Berlin Heidelberg.

[5] Martin Erwig, R. H. G., Markus Schneider, Michalis Vazirgiannis (1999). " Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases." Geoinformatica 3(3): 269-296.

[6] Guting, R. H. (1994). "An Introduction to spatial Database Systems." VLDB Journal 4: 357-399.

[7] Tansel, A. U. and J. Clifford (1993). Temporal Databases: Theory, Design, and Implementation, Benjamin/Cummings Publishing Company.

[8] Abraham, T. and J. F. Roddick (1999). "Survey of Spatio-Temporal Databases." Geoinformatica 3(1): 61-69.

[9] Goralwalla, I. A., M. T. Ozsu, et al. (1998). An Object-Oriented Framework for Temporal Data Models, Springer-Verlag ABerlin Heidelberg.

[10] Mennis, J. L., D. J. Peuquet, et al. (2000). "A Conceptual Framework for Incorporating Cognitive principles into geographical database representation." International Journal of Geographical Information System 14(6): 501-520.

[11] Raza, A. and W. Kainz (1999). Cell tuple based spatio-temporal data model. Proceedings of the 7th international symposium on Advances in geographic information systems, ACM Press.

[12] Freksa, C. (1998). Spatial and Temporal Structures in Cognitive Processes, Springer Verlag Heidelberg.

[13] Peuquet, D. J. (2001). "Making Space for Time: Issues in Space-Time Data Representation." GeoInformatica 5(1): 11- 32.

[14] Dieter Pfoser, Nectaria Tryfona, (1998), "Requirements, definitions, notations for spatio-temporal application environment", Geographic Information Systems, Proceedings of the 6th ACM international symposium on Advances in geographic information systems, Washington, D.C., United States, Pages: 124 - 130

[15] Bonan Li and Guoray Cai, (2002), "A general object-oriented spatio-temporal data model" Symposium on Geospatial Theory, Processing and Applications, Symposium sur la théorie, les traitements et les applications des données Géospatiales, Ottawa 2002

[16] Dieter Pfoser, Yannis Theodoridis, "Generating Semantics-Based Trajectories of Moving Objects", In *Proceedings of the Int'l Workshop on Emerging Technologies for Geo-Based Applications*, pp. 59-76, Ascona, Switzerland, May 2000.

[17] Aritsugi and Tagashira, (1997), “An Approach to Spatio-Temporal Queries - Interval-Based Contents Representation of Images”, Proceedings of the 8th International Conference on Database and Expert Systems Applications, Pages: 202 - 213

[18] Mohd Taib Wahid, A.Z.M Kamruzzaman, Abdul Rashid Mohamed Shariff, Harihodin Selamat (2005), "Conceptual spatio-temporal data model design for plant biodiversity" Malaysian Science and Technology Congress(Costam),18th-20th April, CitiTel MID Valley, Kuala Lumpur, Malaysia

[19] Xiaoyu Wang, Xiaofang Zhou and Sanglu Lu. “Spatio-temporal data modeling and management: A survey”. IEEE TOOLS-Asia’00, 2000.

[20] Peuquet, Donna J. and Niu Duan., (1995),"An Event-Based Spatio-Temporal Data Model for Geographic Information Systems," *International Journal of Geographical Information Systems*, 9(1):7-24

[21] Steiner, A. (1998). *A generalization approach to temporal data models and their implementations,* Swiss Federal Institute of Technology, Zurich, Zurich

[22] Faria, G., Mederios, C. B. and Nascimento, M. A. (1998). "An extensible framework for spatio-temporal database applications", TimeCenter

[23] M. Bouzeghoub, “Morse: a functional query language and its semantic data model”, proc. of int. Conf. on Trends an applications of databases, IEEE-NBS, Gaithersburg, USA, 1994

[24] E.Rose and A. Segev, “TOODM- A temporal object-oriented data model with temporal constrains”, University of California, technical report LBL-30678, April 1991

[25] James E. Rumbaugh, Michael R. Blaha, William J. Premerlani, Frederick Eddy, William E. Lorensen: Object-Oriented Modeling and Design Prentice-Hall 1991

[26] Shafry Mohd Rahim (2002). *Spatial and Non Spatial Enhancement Database for Hydrological Information System (HIS)*, MSc Thesis.

[27] Guting, R. H., B¨ohlen, M. H., Erwig, M., Jensen, C. S., Lorentzos, N. A., Schneider, M., and Vazirgiannis, M. (2000). A Foundation for Representing and Querying Moving Objects. *ACM Transactions on Database Systems*, 25(1):1–42.