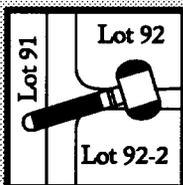
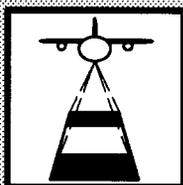


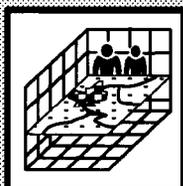
GÉODÉSIE



LÉGISLATION FONCIÈRE



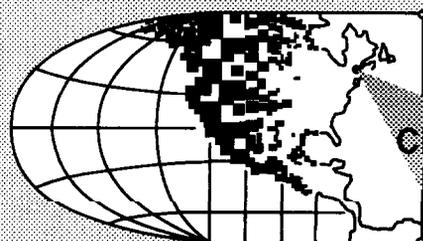
PHOTOGRAMMÉTRIE



SIRS et CARTOGRAPHIE



TELÉDÉTECTION



CENTRE
DE RECHERCHE
EN GÉOMATIQUE

ADAPTING DATA MODELS FOR
THE DESIGN OF SPATIO-TEMPORAL DATABASES

Bédard, Y., C. Caron, Z. Maamar,
B. Moulin & D. Vallière

Computer, Environment and Urban Systems
an International Journal

Vol. 20, No. 1
p. 19-41



SO198-9715(96)00008-7

ADAPTING DATA MODELS FOR THE DESIGN OF SPATIO-TEMPORAL DATABASES

Yvan Bédard*, Claude Caron*, Zakaria Maamar
Bernard Moulin and Denis Vallière*

*Department of Geomatics Sciences and Information Technology
Department, Center for Research in Geomatics, Lava/ University,
Quebec, Canada

MODELING FUNDAMENTALS 1

A data model is above all a *model*. Therefore, it is an approximation of the reality with regard to a particular purpose (Bedard & Chevallier, 1989) which is useful to understand, memorize, communicate and simulate. In everyday life, various types of models have different applications: texts, diagrams, maps, etc. Generally, the models created have to be as representative as possible of the reality they reflect ("homomorphism") since they constitute knowledge communication channels between different interlocutors (Bidard, 1986). Therefore, a model is built to be ultimately communicated, read, interpreted and its content assimilated.

Every model is necessarily developed using a *language of formalism*. A data model is no exception. A formalism or language is made up of *semantic components* (the content, e.g., the set of notions represented by the words of the French language dictionary), *structural rules* (the directions for use, e.g., French *grammar* rules) and a *formal notation* (the form, e.g., *letters* and *punctuation* to present the content) (Caron, 1991).

A formalism must make it possible to create *efficient* models. This efficiency is rendered by a richness of expression (meaning the ability to express as many concepts of the reality as possible required to achieve the objective), along with the capacity to be easily read and comprehended (the language must be as simple and intuitive as possible), qualities that often find themselves in opposition (Caron, 1991). In the field of geomatics, we more often find data models in a textual form described in a data dictionary or a procedures manual. There are then two questions that can be asked: (1) Should the formalism to be used for data modeling be preferably *graphical* or totally *literal* (textual)? and (2) Can the use of data dictionaries be enhanced? Before answering these questions, let us first look at what the human brain consists of and how it functions.

*This article is addressed to readers who already have certain notions regarding data modeling. As a result, the various formalisms and models presented in this article are described concisely for the purpose of comparison and analysis. We invite readers interested in knowing more about the formalisms and models presented to refer to the works of the authors mentioned throughout the article.

Composition and Functioning of the Brain

The brain is divided into two hemispheres, the left and the right, which function in a complementary manner: one side deals with object and event associativity, while the other deals with components or parts, as well as sequences (Rico & Tarcher, 1983). It has been observed that from adolescence the human brain tends towards a total lateralization and, in most cases, the left hemisphere becomes completely dominant. In fact, until recently, it was believed that the right part of the adult brain was practically not participating at all in the language process since this hemisphere functions essentially with images rather than with words (concepts). Contrary to what was believed, the right side of the brain, although not very efficient in aligning words, also participates in language, more specifically by creating global concepts (patterns) and by creating metaphors essential to language enrichment. Let us not lose sight of the fact that language as we use it makes it possible to generate models used to communicate knowledge stored in our minds, essentially kept in holographic form (Buzan, 1983; Rico & Tarcher, 1983). Therefore, we can already see the emergence of arguments in favor of what is “graphic”. These concepts are used in data processing for the creation of graphical user interfaces which entail extensive use of the “mouse” and symbols or pictograms on the screen, to benefit from the use of the right side of the brain. We might then hastily conclude that the old saying “a *picture is worth a thousand words*” seems to be true. However it is not quite so simple. When the two brain hemispheres work together, a synergy is created (Buzan, 1983; Rico & Tarcher, 1983). As an analogy, we could say that the right hemisphere is responsible for the overall melody and all existing feelings, while the left hemisphere focuses on the notes and the musical mechanics. Another confirmation that the brain works more efficiently when engaged in its totality comes from empirical studies on the efficiency of the memory with respect to the five senses and how the brain is employed:

When we pay attention, we retain approximately 10% of what we read, 20% of what we hear, 30% of what we see, 50% of what we see and hear at the same time. On the other hand, we retain 80% of what we say, 90% of what we say while doing something related to what we are thinking about and which involves us (Mucchielli, 1979).

In light of what we now know on the way the brain functions, it is understood that the greatest *efficiency* with respect to modeling is obtained during the simultaneous and integrated use of both brain hemispheres, which can help us identify the most appropriate type of formalism for data modeling.

Formalism Typology

It is possible to classify the various formalisms in a continuum between “totally literal” and “totally graphic” (see Figure 1). Figure 2 presents a model in a “totally literal” form whereas Figure 3 shows the same model in a “totally graphic” form. From looking at these two models, it appears that some graphic elements can be *easily* and *quickly* understood, while certain concepts are very difficult to illustrate without adding text (“totally literal” formalism). It can therefore be said that graphic and totally literal formalisms each have their advantages and disadvantages. Finally, a “hybrid” type of formalism, ideally combining qualities from both types of formalisms, would perhaps be more appropriate (see Figure 4). The hybrid formalism used in Figure 4 is closer to a language in a literal

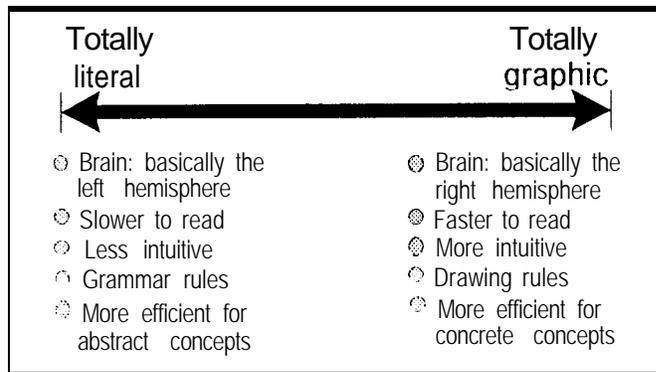


FIGURE 1. Continuum between “totally literal” and “totally graphic”.

The sun's shining, and there is a fish which is jumping in the lake near the poplar where I get up now because I had an idea.

FIGURE 2. Model using a “totally literal” formalism.

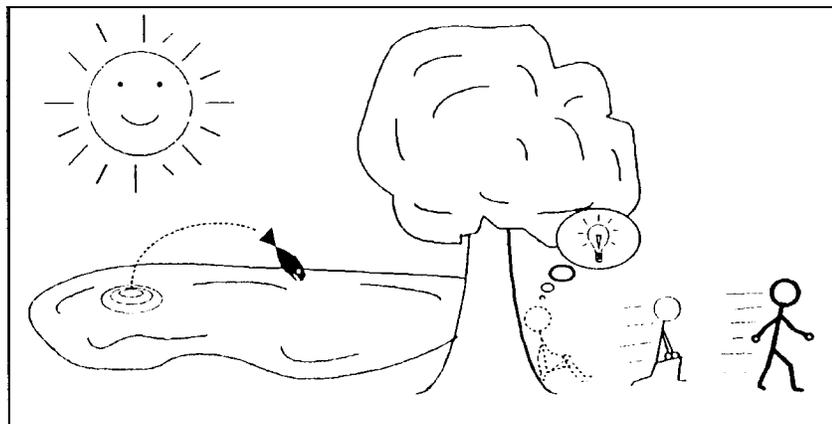


FIGURE 3. Model using a “totally graphic” formalism.

form than to one in a graphic form even though it uses the 2-D characteristics of the latter, as well as symbols such as rectangles and arrows.

It should be pointed out that some tangible concepts, such as *fish*, *lake*, *poplar* could have been represented by drawings rather than modeled in the form of text. On the other

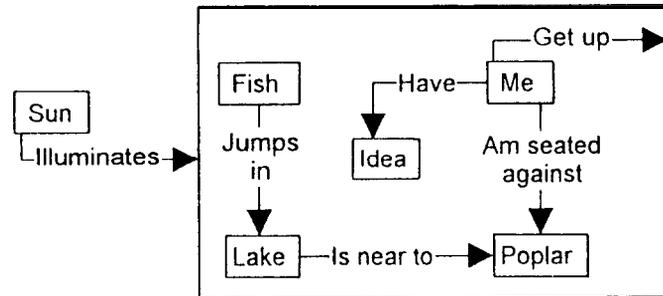


FIGURE 4. Model using a "hybrid" formalism.

hand, this does not apply to intangible concepts such as me illuminates and idea.

Hybrid languages are similar to recommendations and developments carried out by certain authors (Proulx, 1995; Calvinelli & Mainguenaud, 1994; Lee & Chin, 1991). Some caution should be used (Proulx, 1995; Chan, 1987) with respect to the poor legibility of formalisms that are too graphic and whose symbols or pictograms are not very meaningful for the reader of the model. A "hybrid" formalism has the advantage of making the two parts of the brain work-- the left hemisphere, which partitions the reality into multiple concepts, and the right hemisphere, which proceeds by association to create a synthetic image: "If the brain is to relate to information most efficiently the information must be structured in such a way as to 'slot in' as easily as possible. [...] our notes and our word relations should in many instances be structured in this way rather than in traditional 'lines'" (Buzan, 1983). The use of a "hybrid" formalism should enable the creation of **efficient** models, i.e., those which have both a **richness of expression** and a **capacity to be easily read and comprehended**. The **richness of expression** is favored since the capacities and forces of expression of each of the two types of formalisms are combined. The **capacity to be easily read and comprehended** is favored due to the fact that the brain is engaged in its totality, making models possible (among others, data models) which are easier for the brain to assimilate and memorize, and easier to communicate.

Conceptual Data Models and Data Dictionaries

When we speak of **conceptual** data modeling, we are referring to the various "levels" of modeling as defined originally in ANSI/X3/SPARC (1975). The **conceptual** level generally constitutes the first level during which data are modeled, placing the emphasis on the user's reality rather than on the physical structure of the database. These models are sometimes called information models, organization data models or semantic models and are commonly used in information technology. They are also common in object-oriented design despite the use of different terminology. Conceptual data models (CDMs) are therefore developed independently of the technology to be used, however they are translatable into structures that can be implemented in these different technologies. But what is the relative efficiency of a CDM (e.g., Entity-Relationship models) as compared to a data dictionary in expressing and communicating data structures?

The data dictionary is presented in a textual and linear form, i.e., essentially according to one dimension. This form of representation is efficient in presenting data in the form of

lists. Furthermore, a literal form is especially efficient in expressing details difficult to grasp or abstract concepts.

On the other hand, a CDM built in accordance with a “hybrid” form makes it possible to graphically represent **relationships** between elements to be modeled, relationships fundamental to the efficient comprehension of a database’s content and its implementation. In practice, a CDM constitutes a mirroring tool that is more efficient than a data dictionary because it accurately highlights the structure of the elements. It is easier to create and communicate since it is synoptic, a quality indispensable to the **validation** of the model (by people who have indicated their needs and by database developers) and to its **comprehension** by people other than developers (e.g., administrators, engineers or other specialists).

It can therefore be said that the hybrid and textual models complement each other. While the CDM provides a synoptic view of the database, the dictionary contains a large amount of supplementary information indispensable to those responsible for implementing the database. This way of proceeding makes it possible to engage the totality of the brain in a synergetic manner, which promotes the comprehension and communication of the models.

NEEDS CONCERNING TEMPORAL AND SPATIAL REFERENCES

It would be advisable to specify the **specific needs** related to the **spatio-temporal** modeling of data, in order to determine which existing formalisms are the most appropriate for the design of spatio-temporal databases.

Needs Concerning the Spatial Reference

Before studying the problems concerning the modeling of the spatial reference of objects or entities, it is first important to specify what the **spatial reference** is. It can be presented as a gradual process, going from a very concise (qualitative) to a very detailed (quantitative) spatial reference. A minimal spatial reference gives a somewhat vague **position**, while a complete spatial reference includes the **dimension, orientation** and **detailed form** of objects.

Depending on whether or not it is necessary to map an entity, two entity types can be distinguished: **spatial entities** (e.g., building, bus route, fire hydrant) and **non-spatial or traditional entities** (e.g., people, permit).

Spatial entities are cartographically represented by **geometric entities** (e.g., point, line, surface). Therefore the need arises here to indicate in a CDM both whether or not the client wishes to map an entity and the choice of its geometric form, if applicable.

Since the late 1980s, methods and formalisms already used to develop traditional information systems have been studied in order to evaluate their adaptation to a Geographic Information System (GIS) context. The weakness of expression of existing formalisms in representing the spatial reference and the associated characteristics was stressed [at that time, the Entity-Relationship formalism was studied more thoroughly in Boutin (1988), and Bédard and Paquette (1989)]. Emphasis was also placed on the fact that no formalism proposed additional rules for modeling the spatial reference (Boutin, 1988). Still today, geometric entities and the relationships between them are represented by a typical Entity-Relationship formalism (E/R) (Laurini & Thompson, 1992) or OO.

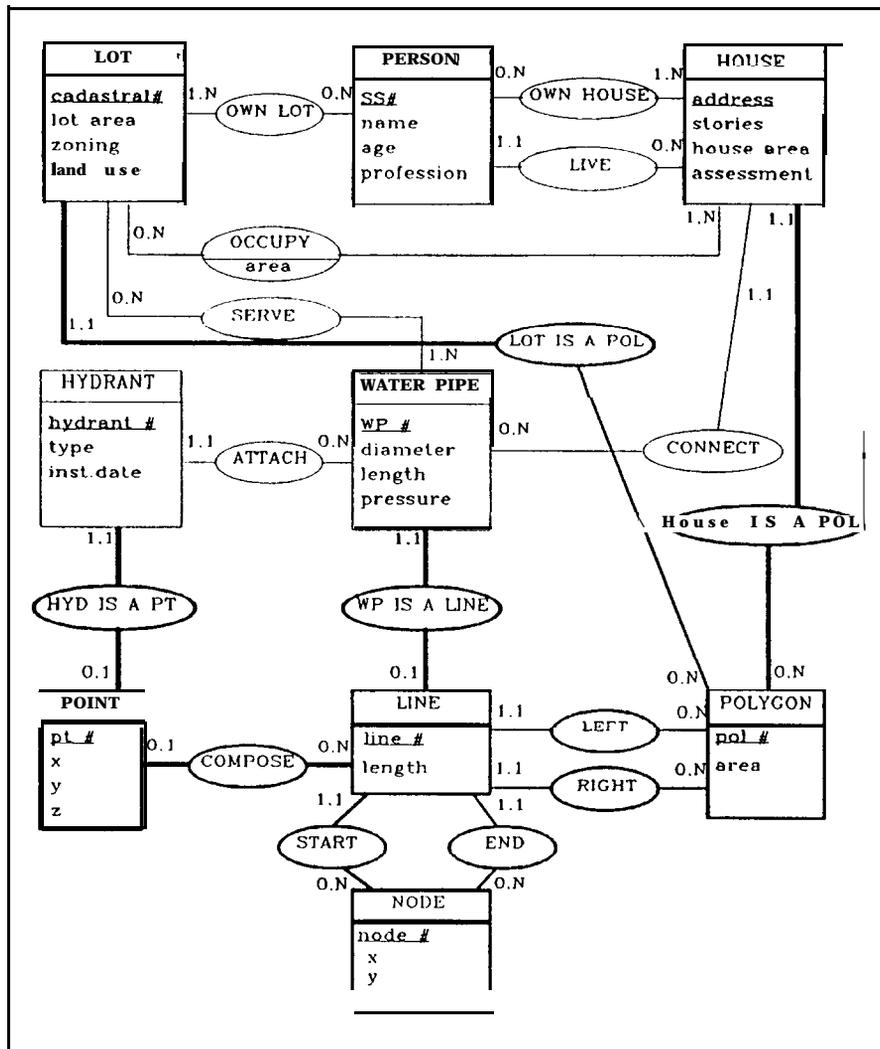


FIGURE 5. Data model including geometric entities.

Figure 5 shows a CDM built with the E/R formalism (following the MERISE method—see next section for more details on this formalism) where geometric entities and their relationships to traditional entities are explicitly included. This way of modeling data is still common today and entails some disadvantages that reduce the efficiency of the resulting model (Caron, 1991):

(1) **Large size of the resulting model.** This *sub-model* of geometric entities (lower part of the model in Figure 5) significantly increases the overall size of the model. In the context of spatio-temporal databases (STDBs), the size is further increased with the presence of primitive temporal entities and their relationships to semantic and geometric entities (Gagnon, 1993a).

(2) **Relevance of specifying the temporal and geometric structure at the conceptual level.** It appears difficult for the STDB developer to establish a precise structure of geometric and

temporal data at the conceptual level, since this structure depends on the technology that will be subsequently used although it is too early to make the technological choices at this stage. Furthermore, since the GISs (Geographic Information Systems) have their own internal structures of geometric data, which are often poorly documented, the STDB developer can only guess, knowing the type of system ("CAD" or topological), which geometric entities are manageable. In any case, it seems inconsistent to explicitly include in the same CDM entities that come from the client's reality (evaluation unit, street...) and entities that do not exist in the same reality (point, line...) (Caron, 1991; Buogo, 1995). This argument is in keeping with the fact that most users of GIS technologies do not consider geometric entities as real objects (Paquet, 1990). During conceptual data modeling, we must however determine the geometric forms required to map spatial entities since a data model is used to identify the data required to complete the databases. This can be done without specifying the complete structure of geometric entities at the conceptual level. It is possible to specify this structure during the translation of the conceptual model for a particular technology (ArcInfo, MGE, Map-Info, etc.).

(3) Absence of contextual information. Another difficulty remains during data modeling: specifying the contextual information. The contextual information is generally made up of some elements on the basic map that we wish to see in order to get our bearings (streets, trees...) but whose attributes we do not want to manage.

(4) Difficulty in specifying the dimension of the spatial data model. In the majority of cases, it is only possible to specify whether one desires to model data in a 2-D or 3-D spatial universe by adding a "z" value to the geometric entity. It is also difficult to distinguish whether the geometric representation of the entities to be modeled is two or three dimensional (e.g., managing and representing the volume of wood piles independently of the dimension of the working environment).

(5) Difficulties in specification of alternative geometric forms. With traditional formalisms, it is difficult to specify the fact that a spatial entity can be represented by several **alternative** geometric forms. For example, the spatial entity BUILDING could have some of its occurrences represented by a SURFACE, while other occurrences with smaller areas could be represented by a symbolized POINT.

(6) Difficulty in the specification of complex geometric forms. It is equally difficult to specify **complex** geometric forms: e.g., the spatial entity WATER SYSTEM whose occurrences are simultaneously represented by an aggregation of POINT and LINE geometric forms.

(7) Difficulty in the specification of multiple geometric forms. It is becoming increasingly complex to represent in a model the fact that an entity occurrence must be digitized two ways, as one is not deductible from the other. For example, each spatial entity MUNICIPALITY can be geometrically represented by a polygon and a point on its demographic center, paving the way to more processing possibilities.

Needs Concerning the Temporal Reference

Since entities evolve in **space and time**, the form and the spatial and temporal behavior of entities must be managed. More over, temporal and spatial aspects influence each other; for example, the geometric representation of a localized phenomenon can evolve in time (Boutin, 1988). The trend until now in spatial data modeling has been either to not take into consideration the temporal aspect of data, or to use partial solutions regarding the

structure of data (e.g., by the addition of a DATE attribute or a DATE entity to the data model). Techniques for representing temporal data in traditional databases have long been as inefficient as techniques for representing geometric data (Paquette, 1990). The hypothesis that the temporal reference is similar to the spatial reference is often used as a basis for further study (Dean & McDermott, 1987; Gagnon, 1993b) as is the idea that as (Buogo, 1995) it should behave and be modeled more or less the same way and using the same techniques.

Needs for temporal reference modeling are therefore exactly the same as those for spatial reference modeling except for the dimensions of the working environment, which are replaced by the need to define as many temporal referentials as there are object existences, attribute evolutions and geometry changes to manage. Finally, it must also be possible to express the management requests for the presence/absence periods of the objects in the territory and their activation/deactivation periods, which all have different impacts on the STDB.

In the following sections, we will present three different types of formalisms used for data modeling: Entity- Relationship, Object-oriented, and a type developed for the STDB. More specifically, we will take a look at how they can be used to model *spatio-temporal data*.

USE OF FORMALISMS FOR SPATIO-TEMPORAL DATA MODELING

In the following section, we will present various data modeling formalisms: the Entity-Relationship formalism based on the MERISE method (Tardieu, Rochfeld, & Colletti, 1986); object-oriented modeling based on the OMT method (Rumbaugh, Blaha, Premerlani, Eddy, & Lorenzen, 1991) and spatio-temporal data modeling based on the MODUL-R formalism (Bédard, Gagnon, & Vallière, 1994). The field of application chosen to support this article represents a simplified part of the road network database of the province of Quebec. In this context, a road is made up of road sections and intersections. They can include different types of structures such as bridges and viaducts and can be affected by obstacles. Each road section, the smallest managed element of the road network, can be delimited by an administrative zone or an intersection. The geometry and temporality of the road network objects and some of their attributes are identified on these road sections (e.g., *geometry* of the road section, *evolution* of the pavement type, etc.).

E/R Formalism (MERISE Method)

Presentation of the Formalism 2

The Entity-Relationship (E/R) formalism is commonly used for conceptual data modeling. It is based on three main concepts: entity, relationship and attribute, as presented in Figure 6. An entity represents an object endowed with its own existence in accordance with the management choices of the organization whose data we want to manage. A relationship is a representation of associations between entities, also in

²We have used the notation conventions of the MERISE method instead of those proposed by Chen (Chen, 1976) for two main reasons: (1) the lack of complexity of the rules used, and (2) the clarity and simplicity of the model produced.

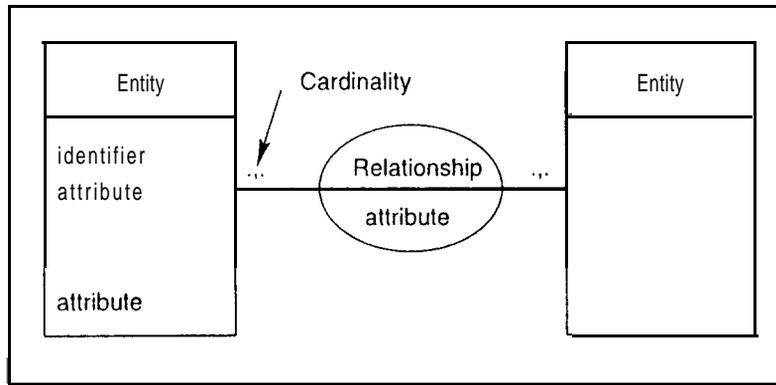


FIGURE 6. E/R formalism concepts (MERISE method).

accordance with the management choices of the organization. An attribute contains the basic data that makes it possible to describe entity and relationship characteristics. All of the concepts defined above can be materialized as occurrences and consequently receive values. There are three types of occurrences: (1) occurrence of an attribute—one of the values the attribute can have in its field, (2) occurrence of an entity—all of the occurrences of its attributes (one occurrence per attribute), and (3) occurrence of a relationship, which associates one and only one occurrence of each of its entities and of each of the relationship's attributes.

The E/R formalism has other characteristics, including (1) the concept of identifier, which can be found at the entity level to characterize each of its occurrences in a unique manner and at the relationship level from the concatenation of the identifiers of the entities they associate, (2) the dimension of a relationship, which defines the number of entities participating in the relationship, and finally (3) the cardinalities of an entity, which measure the minimum and maximum number of the entity's participations in each relationship.

Model representation

The conceptual data model built with the Entity- Relationship formalism is presented in Figure 7. For the purposes of this article, we have distinguished 2 types of entities: the first represents the semantic aspect of the model with the entities ROAD-SECTION, STRUCTURE, ADMINISTRATIVE -ZONE, etc., while the second type reflects the spatio-temporal aspect of the model with the entities ARC, ISOLATED-NODE, TEMPORAL-VECTOR, etc. Furthermore, it has been observed that some entities possess attributes which have the same semantic designation, for example INTERSECTION and HIGHWAY-EXIT with the attributes snow-removal-type and pavement-type.

This is caused by the specialization relationship that exists between them; a HIGHWAY-EXIT is a special kind of INTERSECTION. Composition relationships between entities can also be observed, e.g., ROAD is composed of ROAD-SECTION, INTERSECTION and HIGHWAY-EXIT.

Object-Oriented Formalism (OMT Method)

Our choice concerns the OMT method (Rumbaugh et al., 1991) because the formalism of its data model is very representative of data formalisms used by most of the Object-oriented methods (Monarchi & Puhr, 1992). Furthermore, it is the most widely used OO formalism both in Europe and America. Generally, two different families of concepts can be found in the Object-oriented method (OO) (Moulin & Nguyen, 1993). They consist of those that are specially used for the analysis phase with object classes, objects, relationships and inheritance relationships, and those that are used in the implementation phase with polymorphism, dynamic allocation and encapsulation. In this article, we only deal with concepts associated with the analysis phase (therefore the conceptual level). Furthermore, in OO formalisms, some basic concepts used in the conceptual data model are similar to those of the E/R formalism. Thus, the entities and relationships become classes and associations respectively.

of the Formalism

The OMT (Object Modeling Technique) method was developed at the General Electric Research and Development Center. It is based on 3 different models used to describe the problem to be solved (Carmichael, 1994): the **object model**, the **dynamic model**, and the **functional model**. We are more specifically interested in the **object model**. This model evolves throughout the three development phases: analysis, design and implementation. During the analysis phase, the abstraction level of the model is higher than in the subsequent phases. It is detailed and expanded as the passage is made from one phase to another.

The object model describes the static structure in terms of objects and relationships that exist between them (Harvey & Moulin, 1993). It is graphically represented by object diagrams, using the concepts presented in Figure 8. A class represents a set of objects with common attributes. Various procedures, methods, or functions that define operations to be carried out on a particular object are associated with this object. A class is defined by a data structure (static aspect) and operations (dynamic aspect). All objects that belong to the same class have similar characteristics with respect to their form (attributes) and their behavior (operations), and communicate between themselves using messages. In OO, the associations enable the connection of classes that make up a system. These associations are characterized by cardinalities often called multiplicity. It is impossible to speak of inheritance without introducing the notion of **hierarchy**. The two most important hierarchies of a system are the class structure (hierarchy "kind of" or "is a"), also called generalization or specialization e.g., STRUCTURE with BRIDGE, VIADUCT and OTHER STRUCTURES, and the object structure (hierarchy "part of" or "has") also called aggregation relationship, e.g., ROAD PART and ROAD. Thus, the inheritance makes it possible to define new classes from the attributes and operations of another existent class.

Model Representation 3

The representation of the model in OO that corresponds to Figure 9 is simple and legible in comparison with that of the E/R model. Among the changes brought is the introduction

¹The constrained-alternative-geometry method for obstacle and structure classes makes it possible to support alternative constraints, characteristics of spatio-temporal representation.

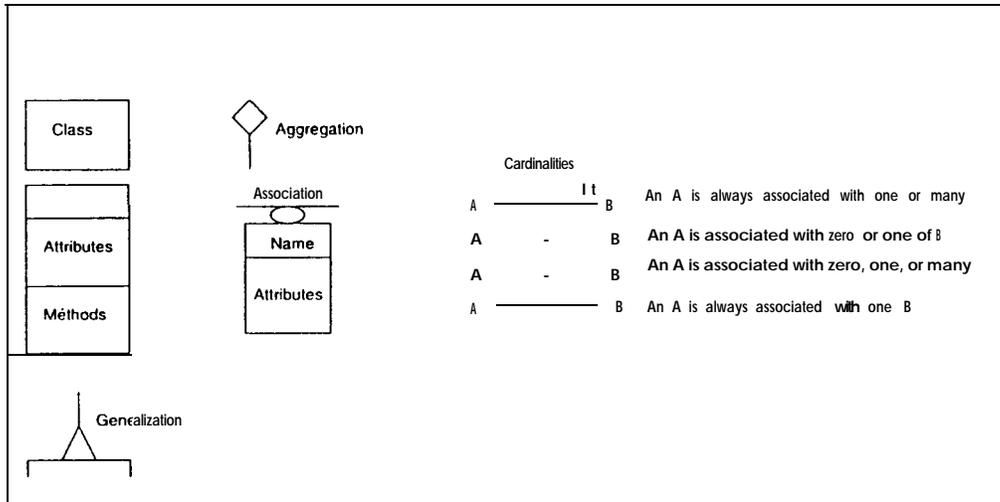


FIGURE 8. Object-oriented formalism concepts (OMT method).

of the ROAD-PART class, making it possible to highlight (1) the aggregation relationship with the ROAD class, and (2) the generalization/specialization relationship with the ROAD-SECTION and INTERSECTION classes. Furthermore, the number of relationships and attributes recorded in the model has decreased as a result of the notion of inheritance, which enables a class to use the attributes and operations of a parent class, e.g., INTERSECTION (parent class) and HIGHWAY-EXIT (son class). The spatio-temporal aspect is always brought to the fore by the same classes (entities) as those of the E/R model.

MODUL-R Formalism

Presentation of the Formalism

MODUL-R is a conceptual level formalism adapted to spatio-temporal databases. MODUL-R has enabled the new needs of spatio-temporal modeling to be taken into account by proposing techniques for representing the temporal and geometric characteristics of entities. The main characteristic of the MODUL-R formalism is its capacity to propose in a unified form complex, *temporal* and *spatial* reference modeling, using *modules* intended for each design context. All of these modules are represented in Figure 10. Other research has also focused on the creation of sub-models, in order to, among other things, model and manage the geometric structure of data (Webster & Omare, 1994). The atomic concepts of the MODUL-R formalism are those of the E/R model, namely entities, relationships, attributes. Cardinalities and functional dependencies are added to these concepts. The hierarchical principle highlights the contribution of the OO approaches to the MODUL-R formalism by proposing the integration of new concepts such as aggregation, specialization and generalization.

The explicit spatial reference enables the determination of the geometric forms required

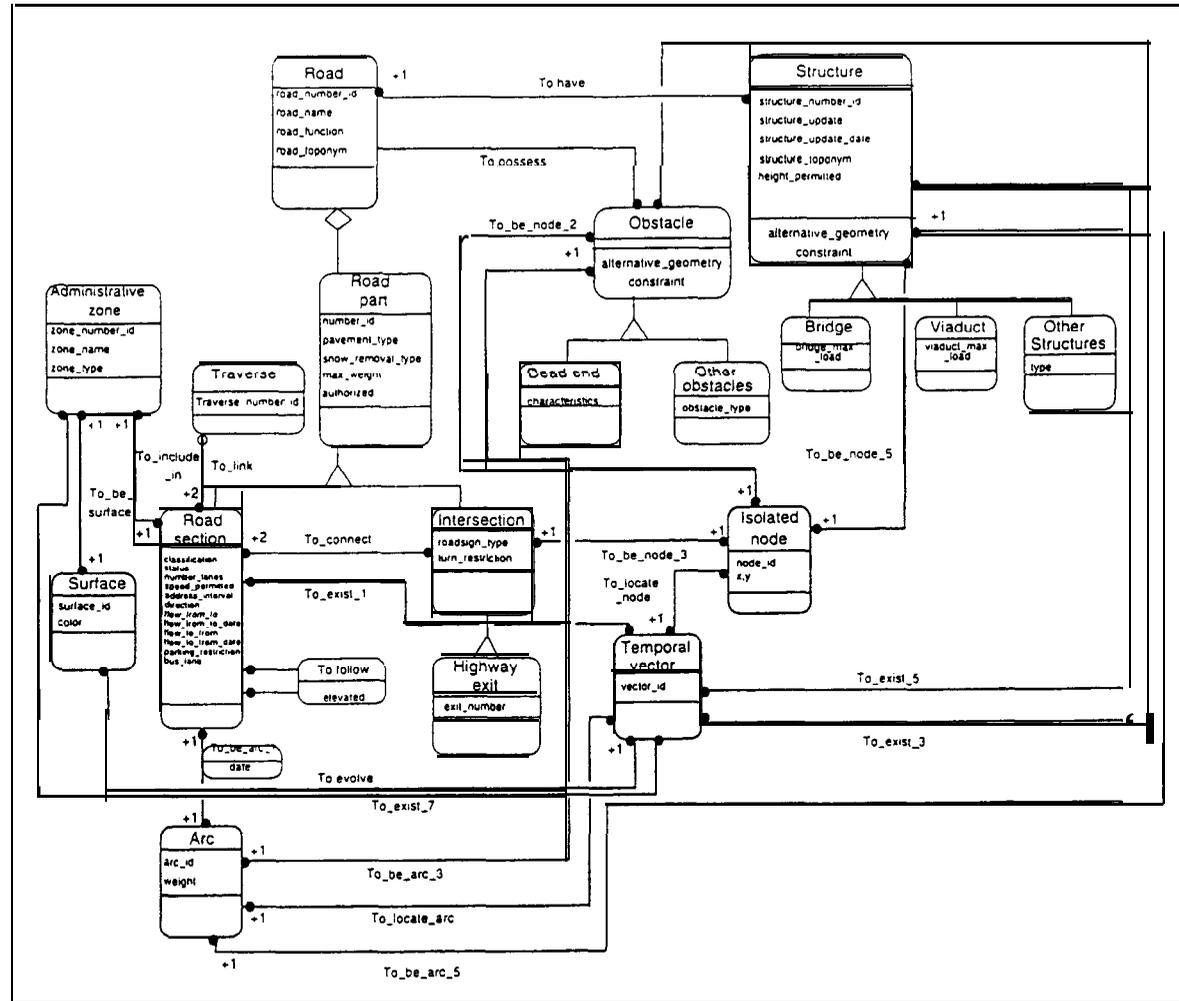


FIGURE 9. Data model with Object-oriented formalism (OMT method).

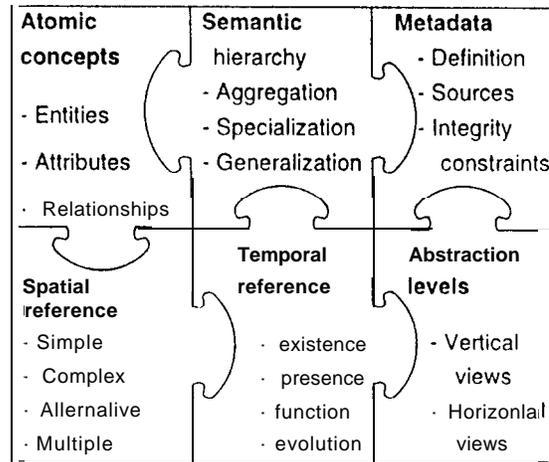


FIGURE 10. MODLR formalism modules.

to represent spatial entities, without specifying the complete structure of the geometric entities at the conceptual level, as only the dimension of the spatial entity is indicated (0-D, 1-D, 2-D, 3-D). To do this, the geometric entities are considered as a set of sub-models which are substituted by pictograms (Bédard & Paquette, 1989). These spatial reference pictograms, which appear in the spatial entity to the left of the name of the entity are substituted, at the conceptual level, for one or more geometric entities and geometric/spatial entity relationships. The spatial pictograms are represented in Figure 11. Thus, the spatial reference makes it possible to define the type of geometry of the entities to be modeled, to (1) state the needs regarding the cartography of objects, (2) specify the type of cartographic information to be digitized and (3) facilitate the passage towards implementation on a GIS. A semantic entity can be characterized by a combination of pictograms (Caron & Bédard, 1993). We then speak of spatial reference (1) **simple**, an entity with only one geometry per occurrence, and whose geometry is the same type for all occurrences, (2) **complex**, an entity which possesses several geometries simultaneously for each instance (spatial aggregation), (3) **alternative**, an entity whose occurrences can be represented by different types of geometry but by only one at a time, or (4) **multiple**, an entity with two or several geometric non-deductible forms for the same occurrence. The three latter possible combinations are represented in Figure 12.

The temporal reference functions almost exactly the same way as the spatial reference. It defines the type of temporal management to be undertaken as well as the temporal forms

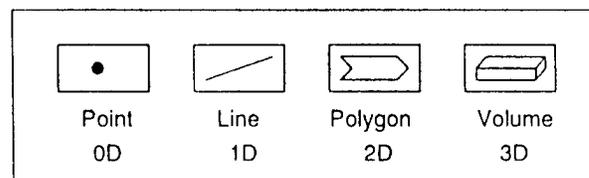


FIGURE 11. Spatial pictograms.

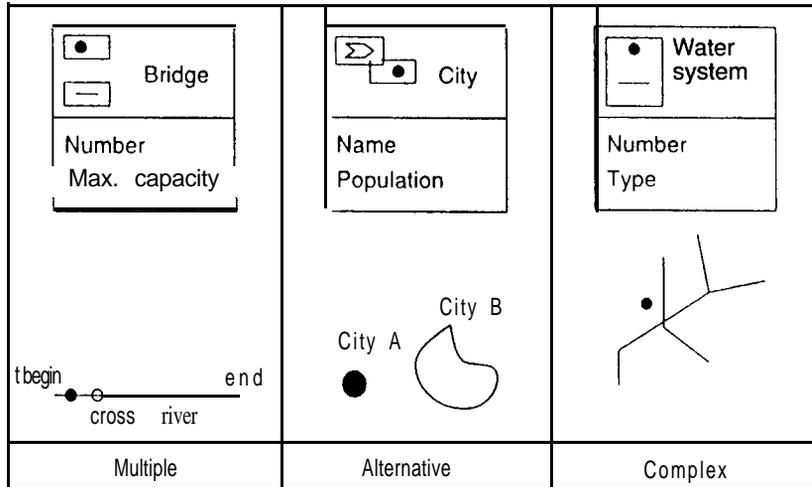


FIGURE 12. Entity types according to the pictograms.

(O-D or instant, 1-D or duration). Thus, there are temporal pictograms of existence, presence and function for the entities, and temporal pictograms of evolution for the attributes. A semantic entity can also be characterized by a combination of pictograms, thus resulting in a complex, alternative or multiple temporal reference. Each pictogram is characterized by its punctuality or its durability, as it is indicated in Figure 13. Each pictogram interpretation is a function of its graphic layout within the entity or the relationship (cf. Figure 14). The existence pictograms make it possible to indicate whether an entity has a durable or punctual existence. The stability pictograms enable characterization of the behavior of the attribute values and the geometry in time, to determine whether these values and this geometry are valid only for an instant or whether they are durable. Besides these concepts of stability and existence, the temporal aspect defines spatial entities in terms of presence and function. These two elements are quite rare. The presence informs us when the entity is physically present or absent in the territory managed (e.g., emergency vehicles), while the function tells us when an entity is active or inactive (e.g., street sections reserved for buses during rush hours). The MODUL-R formalism also proposes simplification techniques in order to facilitate the comprehension of the model. These techniques correspond to:

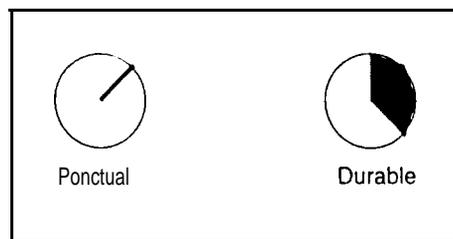


FIGURE 13. Temporal pictograms.

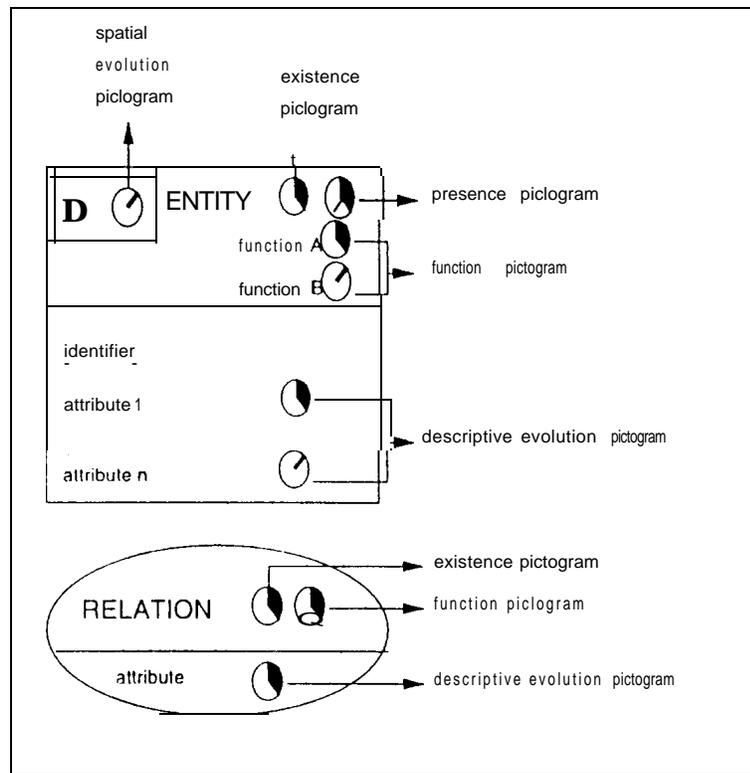


FIGURE 14. Pictogram interpretation according to their position.

1. Groups of entities, enabling the "n" occurrences of component entities that form a complex entity to be grouped together, while respecting certain well-defined grouping rules (Gagnon, 1993a).
2. Abstraction levels or views, whose semantic components are themes, linking relationships, entities and super-entities.
3. Generalizations or specializations; generalization groups entities with common characteristics, and the resulting super-entity is virtual, while specialization illustrates specialized entities linked to a more general entity. The resulting super-entity is real.

In a MODUL-R formalism, the data dictionary associated with the conceptual data model is explicitly developed as supplementary information thus encountering concepts defined in the **Conceptual data models and data dictionaries** section. Its purpose is to provide enough information for the people who collect data, to inform users of the future system and its content, and to describe certain work-related rules of the organization. This explicit aspect of metadata makes the MODUL-R formalism more comprehensible and explicit than the other methods by informing the user about each attribute used during system development.

Model Representation

Due to the concepts proposed by the MODUL-R formalism, the model represented by Figure 15 is less dense and consequently more legible. This is essentially the result of two important factors: (1) the exploitation of certain elements taken from Object-oriented formalisms with concepts such as aggregation, generalization/specialization and inheritance by the creation of super-classes, and (2) the replacement of the sub-model intended for spatio-temporal management by pictograms.

In this section, we have presented various formalisms of data modeling. Based on analyses done for each of them, we can observe that the MODUL-R formalism has many more concepts available for modeling and proposes a model that is more compact and intuitive than the other formalisms. MODUL-R therefore seems to be the most efficient formalism for modeling spatio-temporal databases. This first observation is analyzed in greater detail in the next section.

COMPARATIVE ANALYSIS

The choice of a type of formalism depends on several criteria, including the degree of comprehension for the user, the proof of aptitude for the description of the structure, the basis of discussion and communication between those involved in the project and the simplicity of its design tools (Gagnon, 1993b).

The objective of this section is to present the results of a comparative analysis between the formalisms presented by emphasizing differences existing in terms of concepts supported by each formalism and the fact that they can be created easily.

The suggested comparison criteria essentially concern data modeling concepts and take into account the presence of:

- Elements of conceptual modeling with such concepts as object, object attribute, relationship attribute, cardinality, semantic relationship name, relationship recursive-ness, and finally, dictionary as semantic complement.
- . Elements of object-oriented modeling with such concepts as aggregation and sub-type.
- . Elements of spatio-temporal representation with dictionary as information complement.
- . Elements of dynamic modeling or process representation.
- . Elements of hierarchical modeling or by abstraction level.

In the following table, the terms “yes” and “easily” are used to indicate that the comparative element to be considered is supported by the formalism or is part of the concepts, whereas the opposite is indicated respectively by the terms “no” and “with difficulty”; the term “no” indicates that the element is absent, whereas the term “with difficulty” indicates that the formalism can be adapted to support this comparative element (this implies inevitable additions or modifications to the concepts of the formalism). The following table outlines the differences that exist between the formalisms with respect to modeling concepts.

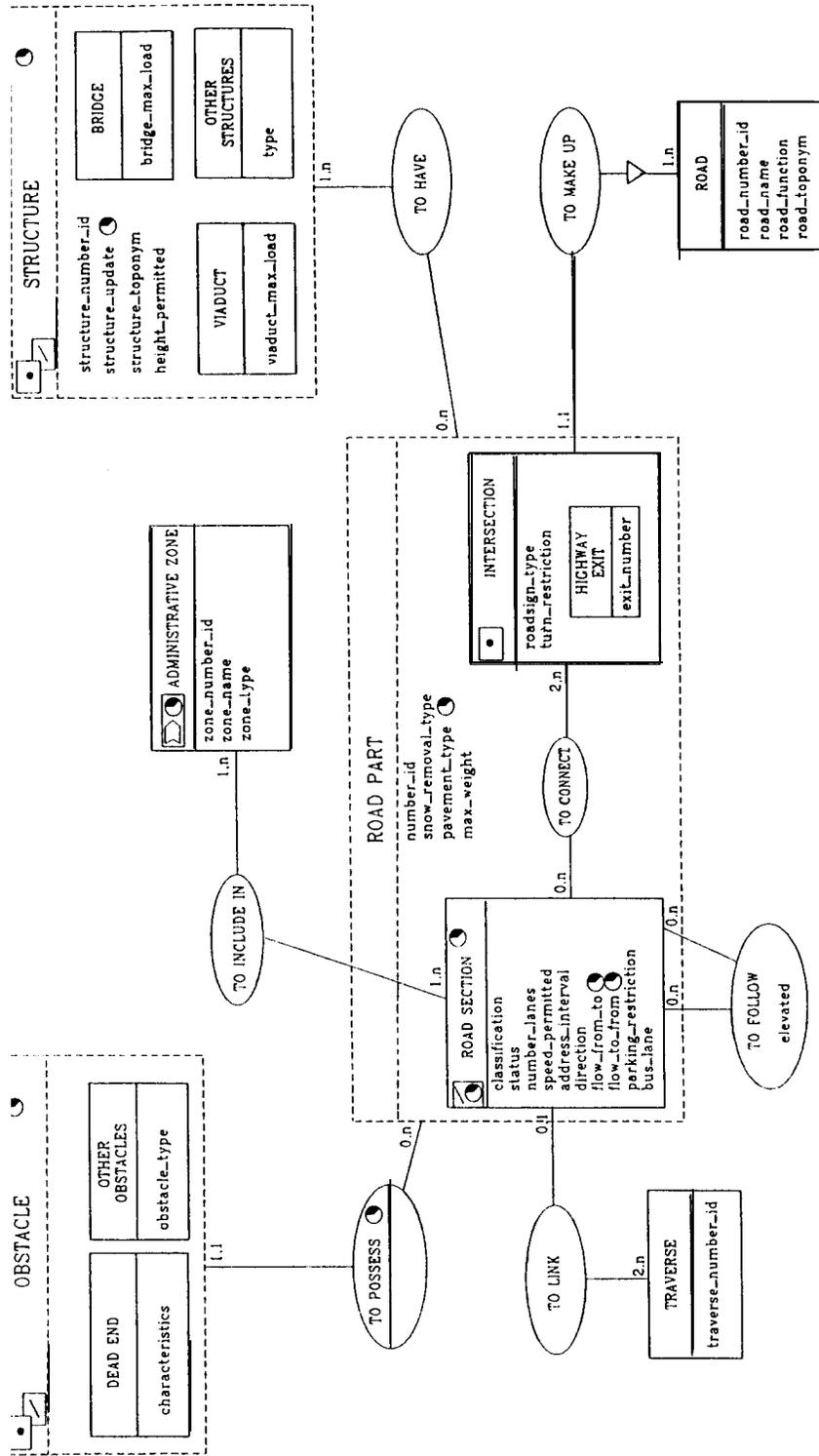


FIGURE 15. Data model with the Modut-R formalism.

Elements of comparison	E/R (MERISE)	Object-oriented (OMT)	MODUL-R
Object	entity	class	entity
Object attribute	attribute	attribute	attribute
Relationship attribute	attribute	no (use class)	attribute
Cardinality	yes	yes (and/or symb.)	yes
Semantic relationship	yes	yes	yes
Recursive relationship	yes	yes	yes
Aggregation	no	yes (aggregation)	yes (aggregation)
Sub-type	no	yes (generalization)	yes (generaliz./specializ.)
Constraint on relationship: and/or	no	no	no
Spatio-temporal representation	with difficulty	with difficulty	easily
Semantic complement in the dictionary	no (method) yes (software)	no (method) yes (software)	yes
Spatio-temporal complement in the dictionary	no	no	yes
Abstraction levels (vertical views)	no	yes (not standardized), in modules	yes (standardized: summary, themes and detailed)
Dynamic aspect (method in the CDM)	yes (conceptual process model)	yes (object diag. / state graph.)	no

Our main observations concerning richness of expression are as follows:

- . Basic notions such as object, object attribute, relationship attribute, cardinality, etc. are found in all of the formalisms.
- . Object-oriented characteristics (inheritance, aggregation, generalization/specialization) are found in both the OO and MODUL-R formalisms.
- . Spatio-temporal modeling concepts are only associated with MODUL-R. With regard to the other approaches, similar extensions could, however, be added to their formalisms (the authors of this article have recently begun a related project).
- . Unlike Object-oriented methods, MODUL-R does not propose modeling techniques for the dynamic components of a system (processing); this deprives this particular formalism of an important stage in system development.
- . It is possible to model the geometric structure of the objects/entities in the OMT and MERISE formalisms, by adding supplementary objects/entities corresponding to the geometric primitives to the model (which makes the model more dense and results in inconsistencies concerning its objective). As for MODUL-R, it does not allow for the expression of the geometric **structure** (a structure that is not always known at the time the conceptual model is created), but rather for the expression of the geometric **forms** of entities to be referenced in space; in this case, the use of **pictograms** makes the resulting model much less dense and preserves consistency in terms of its objectives.
- . The data dictionary used with the MODUL-R formalism contains many **extensions**, particularly some enabling specification of **contextual information** related to the model, as well as the **spatial dimension** of cartographic data (O-D, I-D, 2-D or 3-D). Finally, concerning the expression of **alternative, complex and multiple geometric**

forms, it is also possible, in the OMT and MERISE formalisms, to add supplementary objects/entities (which makes the model cumbersome). In comparison, the MODUL-R formalism makes it possible to combine different spatial pictograms, making the task easier while more or less preserving the same size for the resulting model.

To enable the support of certain characteristics essential to spatio-temporal modeling (particularly alternative constraints on entities, cf. section on MODUL-R formalism), some modeling tricks are introduced in the OO methods. In this way the alternative constraint is represented by a method associated with the class/object (corresponding to the entity); e.g., Class: Obstacle; Method: Alternative-Geometry-Obstacle Constraint.

From a more global perspective, if we compare each of the formalisms studied to needs concerning spatial and temporal references described in the section on Needs concerning temporary and spatial references, (cf. Figures 7, 9 and 15) it is possible to emphasize the formalisms that are more able to adequately meet these needs. The following table synthesizes the results of this comparison.

	E/R (MERISE)	Object-Oriented (OMT)	MODUL-R
1) Size of the resulting model	Big (345 elements)	Medium (250 elements)	Small (15.5 elements)
2) Specification of the geometric structure	Possible, but resulting in a cumbersome model	Possible, but resulting in a cumbersome model	Possible, with pictograms
3) Inconsistencies in the model aim	Possible inconsistencies	Possible inconsistencies	Inconsistencies eliminated by the formalism
4) Contextual information	Not explicit	Not explicit	Explicit
5) Specification of the model dimension (O-D, I-D, 2-D, 3-D)	Not explicit	Not explicit	Explicit
6) Expression of alternative geometric forms	With difficulty	With difficulty	Easily
7) Expression of complex geometric forms	With difficulty	With difficulty	Easily

With respect to efficiency in creating and editing the data models, the quantitative comparison of the components of the three models is very revealing. Furthermore, the explicit spatio-temporal structure contextual information, dimensions of the working environment, different temporal and spatial forms, and the elimination of spatio-temporal inconsistency enhance the efficiency of MODUL-R.

CONCLUSION AND REMARKS

In the light of the human brain structure and functioning described above, fundamental arguments can be brought forward to determine the characteristics of an optimal formalism for spatio-temporal conceptual data modeling.

Thus, the comparison of two extremes, that is to say, a “totally literal” formalism and a “totally graphic” formalism, have enabled us to compare the strengths and weaknesses of both possibilities. These reflections have then allowed us to conclude that an “hybrid formalism” could most effectively use all of the brain’s capacities.

On the basis of this acknowledgment, three existing formalisms of the “hybrid” type were the subject of a detailed comparative study regarding their effectiveness to model spatio-temporal data: Entity- Relationship, OMT and MODUL-R.

From that comparison, it appears that MODUL-R has certain specific characteristics that greatly facilitate the modeling of the spatio-temporal aspect of data, by allowing the creation of models that are more compact than with the two other formalisms. Furthermore, by comparing with traditional data dictionary, the one used with MODUL-R contains several extensions allowing to detail the spatial and temporal characteristics of data.

As an Object-oriented formalism, OMT is, on the other hand, the only studied formalism to integrate modeling techniques for the dynamic components of a system (processing). We believe that this dynamic aspect could eventually be integrated to a future version of MODUL-R. Incidentally, the same could also be applied for the spatial and temporal extensions exclusive to MODUL-R that could be adapted and integrated to other formalisms, among which the ones that have been studied. In fact, researches in these two directions are already underway by the authors.

Finally, whether we add the spatio-temporal extensions to formalisms such as OMT and E/R, or we add process modeling to MODUL-R, the trend is to head towards formalisms with a greater richness of expression.

This *richness of expression* as mentioned at the beginning of this article, is frequently in opposition with the *capacity to be easily read and comprehended*; these two qualities are nonetheless essential to an efficient formalism, which engages the brain in its totality. It is a matter of trade-off that is probably very difficult to achieve in the creation process of any language. This trade-off determines what is found in the data schema and what is found in the data dictionary.

To those two qualities a third one is necessary for a database developer: the *easiness to create* models. This quality often comes in opposition to the *richness of expression of a formalism*.

Fortunately, the use of computerized modeling tools,⁴ among them the *Computer-assisted software engineering (CASE)* tools, facilitates the creation of data models while preserving the richness of expression.

To be truly efficient for spatio-temporal databases, from a Geographic Information System point of view, a modeling tool has to support the necessary extensions identified in the *Comparative analysis* section, as much for the formalism as for the computerized data

⁴There are various types of software on the market that can facilitate the creation and editing of data models. These tools can be classified into four categories: (1) Vectorial or matrix drawing software (e.g., Autocad, CorelDraw, PhotoPaint); (2) Flowchart software (e.g., CorelFlow, ABCFlowCharter, Visio); (3) Computer assisted software engineering (CASE) tools (e.g., Silverrun, Designer, System Architect); and (4) Meta-CASE (e.g., ObjectMaker, Developer, Paradigm plus). Each of these categories is superior to that which precedes it, e.g., by the CASE tools to a computerized data dictionary, and validation and automatic code generation rules for DBMSs. Another such example would be the addition to the meta-CASE tools of a feature that modify the formalisms and dictionaries provided with the software or to create a new one by including all of “the intelligence” that we find in the traditional CASE tools.

dictionary. It must also take into account these features with respect to the validation rules for the model and offer automatic code generation for GISs.

There is currently no commercial CASE tool which offers such possibilities. Spatio-temporal database developers must therefore turn towards meta-CASE software and adapt it to their needs (this is what these tools are for). This is the solution adopted by a team of researchers of the Center for Research in Geomatics at Laval University, who have developed such a tool (**ORION**), using a meta-CASE tool (ObjectMaker) This functional prototype supports the MODUL-R formalism, includes a data dictionary for temporal and spatial references, and automatically generates the code for GISs. Other technical developments following the evolution of the theoretical concepts described in the present article are presently in progress.

REFERENCES

- ANSI/X3/SPARC (1975). Interim report, study group on data base management systems. FDT Bulletin of ACM, 7(2).
- Bédard, Y. (1986). A study of the nature of data using a communication-based conceptual framework of land information systems, Ph.D. Thesis, Department of Civil Engineering, University of Maine, Orono.
- Bédard, Y., & Paquette, F. (1989). Extending entity/relationship formalism for spatial information systems (pp. 818-828). *Auto Carto 9*, April 2-7, Baltimore, Maryland.
- Bédard Y., & Chevallier, J.-J. (1989). Processus de modélisation dans les systèmes d'information à référence spatiale. Internal document, GIS Laboratory, Department of Geomatic Sciences, Faculty of Forestry and Geomatics, Laval University, Quebec.
- Boutin, G. (1988). Étude de l'applicabilité d'une méthode traditionnelle de conception de systèmes d'information dans le contexte d'un système d'information à référence spatiale. M.Sc. Thesis, Department of Geomatic Sciences, Faculty of Forestry and Geomatics, Laval University, Quebec, April.
- Bédard, Y., Gagnon, P. D., & Vallière, D. (1994). Le formalisme MODUL-R 2.01 et le dictionnaire de données pour la conception des bases de données spatio-temporelles, Research document, Center for Research in Geomatics, Laval University.
- Buogo, A. (1995). L'intégration de l'information et les systèmes d'information à référence spatiale. M.Sc. Thesis, Department of Geomatic Sciences, Faculty of Forestry and Geomatics, Laval University, Quebec.
- Buzan, T. (1983). *Use both sides of your brain*. Revised and updated edition. New York: E. P. Dutton Inc.
- Carmichael, A. (1994). Object development methods. *Advances in Object Technology*. New York: SIGS Books Inc.
- Calvinelli, D., & Maiguenaud, M. (1994). CIGALES a visual query language for a geographical information system: The user interface. *Journal of Visual Languages and Computing*, 5 (June), 113 - 132.
- Caron, C. (1991) Nouveau formalisme de modélisation conceptuelle adapté aux SIRS. M.Sc. Thesis, Department of Geomatic Sciences, Faculty of Forestry and Geomatics, Laval University, Quebec.
- Caron, C., & Bédard, Y. (1993). Extending the individual formalism for a more complete modeling of urban spatially referenced data. *Computers, Environment and Urban Systems*, 17, 337-346.
- Chan, S. K. (1987). Visual languages: A tutorial and survey. *IEEE Transactions on Software*, January, pp. 29- 39.
- Chen, P. P.-S. (1976). The entity-relationship model: Toward a unified view of data. *ACM TODS*.
- Dean, T. L., & McDermott, D. V. (1987). Temporal data base management. *Artificial Intelligence*, 32(1), 1 - 55.
- Gagnon, P. D. (1993a) Concepts fondamentaux de la gestion du temps dans les SIG. M.Sc. Thesis, Department of Geomatic Sciences, Faculty of Forestry and Geomatics, Laval University, Quebec.
- Gagnon, P. D. (1993b) MODUL-R version 2.0. Internal document-SDIAM Project for Intergraph, center for Research in Geomatics, Laval University, September.
- Harvey, G., & Moulin, B. (1993). Revue et comparaison de quelques méthodes de conception de systèmes orientés-objet. *ICO*, 5(2), 38-51.
- Laurini R., & Thompson, D. (1992). *Fundamentals of spatial information systems*. The APIC Series, No. 37. London: Academic Press.
- Lee, Y. C., & Chin, F. L. (1991). A query language based on a spatially intuitive grammar. *Proceedings: Canadian Conference on GIS*. March 18-22 Canadian Institute of Surveying and Mapping, Ottawa, pp. 868-875.
- Monarchi, D. E., & Puhr, G. I. (1992). A research typology for object-oriented analysis and design. *Communications of the ACM*, 35(9), 35 -47.
- Moulin, B., & Nguyen, C. L. (1993). La technologie orientée-objet: Évolution ou révolution? *ICO*, 5(2), 4-10.

- Mucchielli, R. (1979). *Les méthodes actives dans la pédagogie des adultes* Paris: Editions ESF.
- Paquette, F. (1990). *Conception d'un système d'information à référence spatiale pour la gestion et l'aménagement de la forêt Montmorency* M.Sc. Thesis, Department of Geomatic Sciences, Faculty of Forestry and Geomatics. Laval University, Quebec.
- Proulx, M.-J. (1995). *Développement d'un nouveau langage d'interrogation de bases de données spatio-temporelles* M.Sc. Thesis, Department of Geomatic Sciences, Faculty of Forestry and Geomatics. Laval University, Quebec, May.
- Rico, G. L., & Tarcher, J. P. (1983). *Writing the natural way Using right-brain techniques IO release your expressive power* U.S.A.
- Rumhaugh, J., Blaha, M., Premerlani, W., Eddy, F., & Lorensen, W. (1991). *Object-oriented modeling and design*. Englewood Cliffs. NJ: Prentice-Hall.
- Tardieu, H., Rochfeld, A., & Colletti, R. (1986). *La méthode Merise tome 1, principes et outils* Paris: Les éditions d'organisation.
- Webster, C. J., & Omare, C. N. (1994). *Structured methods for GIS design --Part 2: An object-oriented system for physical plan monitoring*. *Computer Environment and Urban Systems*, 18(1) 19-41.