Challenges of the Anthropocene epoch – supporting multi-focus research

André Santanchè[1] **, Claudia Bauzer Medeiros**[1] **, Geneviève Jomier**[2] **, Michel Zam**[2]

[1]Institute of Computing, UNICAMP, Brazil , LAMSADE - Universite Paris-IX Dauphine, France

Abstract. *Work on multiscale issues presents countless challenges that have been long attacked by GIScience researchers. Most results either concentrate on modeling or on data structures/database aspects. Solutions go either towards generalization (and/or virtualization of distinct scales) or towards linking entities of interest across scales. However, researchers seldom take into account the fact that multiscale scenarios are increasingly constructed cooperatively, and require distinct perspectives of the world. The combination of multiscale and multiple perspectives per scale constitutes what we call* multi-focus *research. This paper presents our solution to these issues. It builds upon a specific database version model – the multiversion MVBD – which has already been successfully implemented in several geospatial scenarios, being extended here to support multi-focus research.*

1. Introduction

Geological societies, all over the world, are adopting the term "Anthropocene" to designate a new geological epoch whose start coincides with the impact of human activities on the Earth's ecosystems and their dynamics.

The discussion on the Anthropocene shows a trend in multidisciplinary research directly concerned with the issues raised in this paper – scientists increasingly need to integrate results of research conducted under multiple foci and scales. Anthropocenic research requires considering multiscale interactions – e.g., in climate change studies, this may vary from the small granularity (e.g., a human) to the macro one (e.g., the Earth). To exploit the evolution and interaction of such complex systems, research groups (and disciplines) must consider distinct entities of study, submitted to particular time and space dynamics. Multiscale research is not restricted to geographic phenomena; this paper, however, will consider only two kinds of scales – temporal and geographic.

For such scenarios, one can no longer consider data heterogeneity alone, but also the heterogeneity of processes that occur within and across scales.This is complicated by the following: (a) there are distinct fields of knowledge involved (hence different data collection methodologies, models and practices); and (b) the study of complex systems requires complementary ways of analyzing a problem, looking at evidence at distinct aggregation/generalization levels – a *multi-focus* approach. Since it is impossible to work at all scales and representations at once, each group of scientists will focus on a given (sub)problem and try to understand its complex processes. The set of analyses performed under a given focus has implications on others. From now on, this paper will use the term

*Work partially financed by CAPES-COFECUB (AMIB project), FAPESP-Microsoft Research Virtual Institute (NavScales project), and CNPq

”multi-focus” to refer to these problems, where a ”focus” is a perspective of a problem, including data (and data representations), but also modeling, analysis and dynamics of the spatio-temporal entities of interest, within and across scales.

This scenario opens a wide range of new problems to be investigated [Longo et al. 2012]. This paper has chosen to concentrate on the following challenges:

- How can GIScience researchers provide support to research that is characterized by the need to analyze data, models, processes and events at distinct space and time scales, and represented at varying levels of detail?
- How to keep track of events as they percolate bottom-up, top-down and across space, time and foci of interest?
- How to provide adequate management of these multi-focus multi-expertise scenarios and their evolution?

A good example of multi-focus Anthropocene research in a geographic context is multimodal transportation. At a given granularity, engineers are interested in individual vehicles, for which data are collected (e.g., itineraries). Other experts may store and query trajectories, and associate semantics to stops. At a higher level, traffic planners study trends - the individual vehicles disappear and the entities of study become clusters of vehicles and/or traffic flow – e.g., [Medeiros et al. 2010]. A complementary focus comes from climate research (e.g., floods cause major traffic disturbances) or political upheavals. This can be generalized to several interacting granularity levels. In spite of advances in transportation research, e.g., in moving objects, there are very few results in representation and interaction of multiple foci.

Environmental changes present a different set of challenges to multi-focus work. Studies consider a hierarchy of ecological levels, from community to ecosystem, to landscape, to a whole biome. Though ecosystems are often considered closed systems for study purposes, the same does not apply to landscapes, e.g., they can include rivers that run into (or out of) boundaries[1]. A landscape contains multiple habitats, vegetation types, land uses, which are inter-related by many spatio-temporal relationships. And a study may focus on vegetation patches, or in insect-plant interactions.

In agriculture – the case study in this paper – the focus varies from sensors to satellites, analyzed under land use practices or crop strains and lifecycles. Each of the disciplines involved has its own work practices, which require analyzing data at several granularity levels; when all disciplines and data sets are put together, one is faced with a highly heterogeneous set of data and processes that vary on space and time, and for which there are no consensual storage, indexation, analysis or visualization procedures.

Previous work of ours in traffic management, agriculture and biodiversity brought to light the limitations of present research on spatio-temporal information management, when it comes to supporting multi-focus studies. As will be seen, our work combines the main solution trends found in the literature, handling both data and processes in a homogeneous way, expanding the paradigm of *multiversion databases*, under the model of [Cellary and Jomier 1990]. We have recently extended it to support multiple spatial scales [Longo et al. 2012], and here explore multiple foci and interactions across scales.

---

[1] Similar to studies in traffic in and out of a region...

2. Related work

Research on multiscale data management involves state-of-the-art work in countless fields. As pointed out in, for instance, [Spaccapietra et al. 2002], multiple cartographic representations are just one example of the need for managing multiple scales. In climate change studies, or agriculture, for instance, a considerable amount of the data are geospatial – e.g., human factors.

Present research on multiscale issues has several limitations in this broader scenario. To start with, it is most frequently limited to vectorial data, whereas many domains, including agriculture, require other kinds of representation and modeling (including raster data) [Leibovicia and Jackson 2011]. Also, it is essentially concerned with the representation of geographic entities (in special at the cartographic level), while other kinds of requirements must also be considered.

The example reported in [Benda et al. 2002], concerning riverine ecosystems, is representative of challenges to be faced and which are not solved by research on spatio-temporal data management. It shows that such ecosystems involve, among others, analysis of spatio-temporal data and processes on human activities (e.g., urbanization, agricultural practices), on hydrologic properties (e.g., precipitation, flow routing), and on the environment (e.g., vegetation and aquatic fauna). This, in turn, requires cooperation of (at least) hydrologists, geomorphologists, social scientists and ecologists.

Literature on the management of spatio-temporal data and processes at multiple scales concentrates on two directions: (a) generalization algorithms, which are mostly geared towards handling multiple spatial scales via algorithmic processes; and (b) multi-representation databases (MRDBs), which are geared towards data management at multiple spatial scales. These two approaches respectively correspond to Zhou and Jones' [Zhou and Jones 2003] multi-representation spatial databases and linked multi-version databases[2]. Most solutions, nevertheless, concentrate on spatial "snapshots" at the same time, and frequently do not consider evolution with time or focus variation.

Generalization-based solutions rely on the construction of virtual spatial scales from a basic initial geographic scale - for instance, [Oosterom and Stoter 2010] in their model mention that managing scales require "zooming in and out", operations usually associated with visualization (but not data management). Here, as pointed out by [Zhou and Jones 2003], scale and spatial resolution are usually treated as one single concept. Generalization itself is far from being a solved subject. As stressed by [Buttenfield et al. 2010], for instance, effective multiscale representation requires that the algorithm to be applied be tuned to a given region, e.g., due to landscape differences. Generalization solutions are more flexible than MRDBs, but require more computing time.

While generalization approaches compute multiple virtual scales, approaches based on data structures rely on managing stored data. Options may vary from maintaining separate databases (one for each scale) to using MRDBs.The latter concern data structures to store and link different objects of several representation of the same entity or phenomenon [Sarjakoski 2007]. They have been successfully reported in, for instance, urban planning, or in the aggregation of large amounts of geospatial data and in cases that applications require data in different levels of detail [Oosterom 2009, Gao et al. 2010,

[2] We point out that our definition of *version* is not the same as that of Zhou and Jones

Parent et al. 2009]. The multiple representation work of [Oosterom and Stoter 2010] comments on the possibility of storing the most detailed data and computing other scales via generalization. This presents the advantage of preserving consistency across scales (since all except for a basis are computed), but multiple foci cannot be considered.

The previous paragraphs discussed work that concentrates on spatial, and sometimes spatio-temporal issues[3]. Several authors have considered multiscale issues from a conceptual formalization point of view, thus being able to come closer to our focus concept. An example is [Spaccapietra et al. 2002], which considers classification and inheritance as useful conceptual constructs to conceive and manage multiple scales, including multiple foci. The work of [Duce and Janowicz 2010] is concerned with multiple (hierarchical) conceptualizations of the world, restricted to spatial administrative boundaries (e.g., the concept of rivers in Spain or in Germany). While this is related to our problem (as multi-focus studies also require multiple ontologies), it is restricted to ontology construction. We, on the other hand, though also concerned with multiple conceptualizations of geographic space, need to support many views at several scales – e.g., a given entity, for the same administrative boundary, may play distinct roles, and be present or not.

We point out that the work of [Parent et al. 2006] concerning the MADS model, though centered on conceptual issues concerning space, time and perspective (which has similar points with our focus concept), also covers implementation issues in a spatio-temporal database. Several implementation initiatives are reported. However, a perspective (focus) does not encompass several scales, and the authors do not concern themselves with performance issues. Our extension to the MVBD approach, discussed next, covers all these points, and allows managing both materialized and virtual data objects within a single framework, encompassing both vector and raster data, and letting a focus cover multiple spatial or temporal scales.

3. Case study

Let us briefly introduce our case study - agricultural monitoring. In this domain, phenomena within a given region must be accompanied through time. Data to be monitored include, for instance, temperature, rainfall, but also soil management practices, and even crop responses to such practices. More complex scenarios combine these factors with economic, transportation, or cultural factors.

Data need to be gathered at several spatial and temporal scales – e.g., from chemical analysis on a farm's crop every year, to sensor data every 10 minutes. Analyses are conducted by distinct groups of experts, with multiple foci – agro-environmentalists will look for impact on the environment, others will think of optimizing yield, and so on.

We restrict ourselves to two data sources, satellite images (typically, one image every 10 days) and ground sensors, abstracting details on the actual data being produced. From a high level perspective, both kinds of sources give origin to *time series*, since they periodically produce data that are stored together with timestamps. We point out that these series are very heterogeneous. Sensor (stream) series data are being studied under distinct research perspectives, in particular data fusion and summarization e.g.,

[3]The notion of scale, more often than not, is associated with spatial resolution, and time plays a secondary role.

[McGuire et al. 2011]. Some of these methods are specific for comparing entire time series, while others can work with subsequences. Satellite images are seldom considered under a time series perspective: data are collected less frequently, values are not atomic, and processing algorithms are totally different – research on satellite image analysis is conducted within remote sensing literature – e.g., [Xavier et al. 2006]. Our multi-focus approach, however, can treat both kinds of data source homogenously.

Satellite time series are usually adopted to provide long-term monitoring, and to predict yield; sensor time series are reserved for real time monitoring. However, data from both sources must be combined to provide adequate monitoring. Such combinations present many open problems. The standard, practical, solution is to aggregate sensor data temporally (usually producing averages over a period of time), and then aggregate them spatially. In the spatial aggregation, a local sensor network becomes a point, whose value is the average of the temporal averages of each sensor in the network. Next, Voronoi polygons are constructed, in which the ”content” of a polygon is this global average value. Finally, these polygons can be combined with the contents of the images. Joint time series evolution is not considered. Our solution, as will be seen, allows solving these issues within the database itself.

4. Solving anthropocenic issues using MVDBs

Our solution is based on the Multiversion Database (MVDB) model, which will be only introduced in an informal way. For more details the reader is referred to [Cellary and Jomier 1990]. The solution is illustrated by considering the monitoring of a farm within a given region, for which time-evolving data are: (a) satellite images (database object S); (b) the farm's boundaries (database object P), and (c) weather stations at several places in the region, with several sensors each (database object G).

4.1. Introducing MVBD

Intuitively, a given real world entity can correspond to many distinct digital items expressing, for example, its alternative representations, or capturing its different states along time. Each of these ”expressions” will be treated in this work as a *version* of the object. Consider the example illustrated in Figure 1. On the left, there are two identified database objects: a satellite image (`Obj S`) and a polygon to be superimposed on the image (`Obj P`). delimiting the boundaries of the farm to be monitored.

As illustrated by the table on the right of the figure, both objects can change along time, reflecting changes in the world, e.g., a new satellite image will be periodically provided, or the boundaries of the farm can change. For each real world entity, instead of considering that these are new database objects, such changes can be interpreted as many versions of the same object[4]. This object has a single, unique, identifier – called an Object Identifier `Oid`[5].

A challenge when many interrelated objects have multiple versions is how to group them coherently. For example, since the satellite image and the farm polygon change along time, a given version of the satellite image from 12/05/2010 must be related with a temporally compatible version of the farm polygon. This is the central focus of

[4]Here, both raster and vector representations are supported. An MVDB object is a database entity

[5]Oids are artificial constructs. The actual disambiguation of an object in the world is not an issue here

the Multiversion Database (MVDB) model. It can handle multiple versions of an arbitrary number of objects, which are organized in *database versions - DBVs*. A DBV is a logical construct. It represents an entire, consistent database constructed from a MVDB which gathers together consistent versions of interrelated objects. Intuitively, it can be interpreted as a *complex view* on a MVDB. However, as shall be seen, unlike standard database views, DBVs are not constructed from queries.

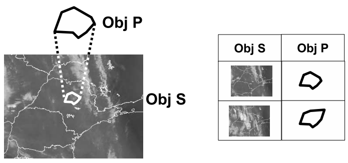

Figure 1. Practical scenario of a polygon over a satellite image.

To handle the relation between an object and its versions, the MDBV distinguishes their identifications by using object and physical identifiers respectively. Each object has a single object identifier (`Oid`), which will be the same independently of its multiple versions. Each version of this object, materialized in the database by a digital item – e.g., an image, a polygon etc. – will receive a distinct physical version identifier *PVid*. In the example of Figure 1, there is a single `Oid` for each object – satellite image (`Obj S`) and the farm boundaries (`Obj P`). Every time a new image or a new polygon is stored, it will receive its own `PVid`.

DBVs are the means to manage the relationship between an `Oid` (say, `S`) and a given `PVid` (of `S`). Figure 2 introduces a graphical illustration of the relationship among these three elements: `DBV`, `Oid` and `PVid`. In the middle there are two DBVs identified by `DBVids` – `DBV 1` and `DBV 1.1` – and represented as planes containing logical slices (the "views") of the MVDB. The figure shows that each DBV has versions of `P` and `S`, but each DBV is monoversion (i.e., it cannot contain two different versions of an object). The right part of the figure shows the physical storage, in which there are two physical versions of `S` (identified by `Ph1` and `Ph9`), and just one version of `P`.

`DBV 1` relates `S` with a specific satellite image and `P` with a specific polygon, which form together a consistent version of the world. Notice that here nothing is being said about temporal or spatial scales. For instance, the two satellite images can correspond to images obtained by different sensors aboard the same satellite (e.g., heat sensor, water sensor), and thus have the same timestamp. Alternatively, they can be images taken in different days. The role of the DBV is to gather together compatible versions of its objects, under whichever perspective applies.

Since DBVs are logical constructs, each object in a DBV has its own logical identifier. Figure 2 shows on the left an alternative tabular representation, in which `DBVids` identify rows and `Oids` identify columns. Each pair (`DBVid`, `Oid`) identifies the logical version of an object and is related to a single `PVid`, e.g., $(DBV1, ObjS) \rightarrow Ph1$. The asterisk in cell (`DBV 1.1, Obj P`) means that the state of the object did not change from `DBV 1` to `DBV 1.1`, and therefore it will address the same physical identifier `Ph 5`.

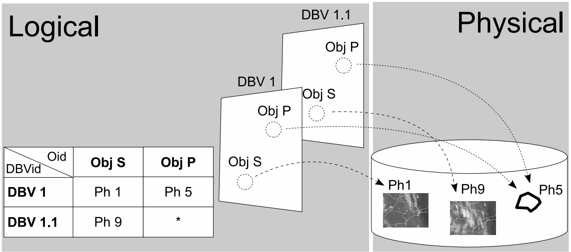

Figure 2. The relationship between DBVs, logical and physical identifiers.

4.2. DBV Evolution and Traceability

DBVs can be constructed from scratch or from other DBVs[6]. The identifier of a DBV (DBVid) indicates its derivation history. This is aligned to the idea that versions are not necessarily related to time changes, affording alternative variations of the same source, as well as multiple foci – see section 5.

The distinction between logical and physical identifications is explored by an MVDB to provide storage efficiency. In most of the derivations, only a partial set of objects will change in a new derived DBV. In this case, the MVDB has a strategy in which it stores only the differences from the previous version. Returning to the example presented in Figure 2 on the left table, `DBV 1.1` is derived from `DBV 1`, by changing the state of `Obj S`. Thus, a new `PVid` is stored for it, but the state of `Obj P` has not changed – no new polygon is stored, and thus there is no new `PVid`.

The evolution of a DBV is recorded in a derivation tree of DBVids. To retrieve the proper `PVid` for each (virtual) object in a DBV, the MVDB adopts two strategies: provided and inferred references[7], through navigation in the tree. This allows keeping track of real world evolution. We take advantage of these concepts in our extension of the MVDB model, implemented to support multiple spatial scales [Longo et al. 2012]. First, we create one tree per spatial scales, and all trees grow and shrink together. Second, the notion of object id is extended to associate the id with the scale in which that object exists - (Oid, Scaleid). This paper extends this proposal in two directions: (1) we generalize the notion of spatial scale to that of focus, where a given spatial or temporal scale can accomodate multiple foci, and the evolution of these foci within a single derivation tree; (2) we provide a detailed case study to illustrate the internals of our solution.

5. From Multiversion to Multi-focus

This paper extends the MVDB model to support the several flavors of multi-focus. This implies in synthesizing the multiple foci which can be applied to objects – scales, representations etc. – as specializations of versions. Figure 3 illustrates an example of this extension. There are three perspectives within the logical view - see the Figure.

In the Physical perspective, there are three objects – two versions of satellite image S (with identifiers `Ph1` and `Ph2`), and one version of a set of sensor data streams, corresponding to a set of weather stations G – global identifier `Ph7`). Satellite image and

[6]DBV derivation trees, part of the model, will not be presented here.

[7]For the logical version (`DBV 1.1, Obj P`), the reference will be inferred by traversing the chain of derivations.

sensor data are to be combined in Applications, which can only access DBVs (and not the database). So, several DBVs are built, each of which corresponding to a distinct focus. The arrows between DBV objects and stored objects appear whenever an object is copied into a DBV, without any additional computation. In the figure, the DBV corresponding to Focus 1 makes available the satellite image version `Ph1` and all data from all weather stations G. The DBV corresponding to Focus 2 makes available the satellite image version `Ph2`, and *computes* a set of Voronoi polygons from the weather station data streams – the resulting polygon is displayed in the figure with a dotted line to show that it is not directly copied from the database, but is computed from it. Finally, DBV-Focus3 contains only one image, which has been computed from DBV-Focus2.

Applications access these three DBVs in the following way. Application Scale A is built from DBV-Focus2; it corresponds to a particular spatio-temporal focus of the database, in which the image is directly extracted from the DBV, and a set of Voronoi polygons is computed from the DBV. Application Scale B is built from DBV-Focus1; it corresponds to another spatio-temporal focus of the database, in which the image and the polygons are directly copied from the DBV. The third DBV is not being used by any application.

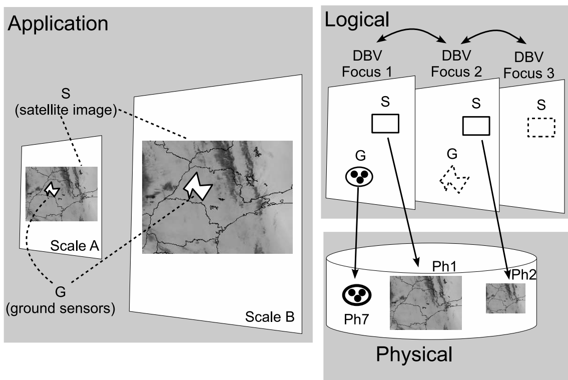

Figure 3. Handling multiple foci.

Figure 3 reflects the following facts. First, DBVs can contain just objects that are in the database, or computed objects, or a mix of both. Second, applications constructed on top of the DBVs can use exactly the same objects (the one on Scale A directly uses the same contents of DBV-Focus2), but also compute other objects (the polygon on Scale B, computed from DBV-Focus1). Third, DBVs now can be interrelated by many kinds of derivation operations.

In our case study, each application corresponds to one spatial scale (scale B smaller than scale A), and sensor data are preprocessed either at the application, or by the DBMS, to allow combination of these distinct data sources. DBV-Focus 3 is an example of at least three possible scenarios: in one, S corresponds to an even smaller spatial scale, for which sensor data do no longer make sense; in another, S is the result of combination of satellite image and sensor data; in the third, the focus is in some characteristics of the satellite image, and sensor data can be ignored for the purposes of that DBV.

In order to support these kinds of DBV, the classical MVDB model was extended: (i) we added more types of relationships between DBVs; (ii) we introduced the notion of scale to be part of an OID. In the classical MVDB the only relationship between two DBVs is the derivation relationship, explained in the previous section. Our multi-focus approach requires a wider set of relationships. Therefore, now the relationship between two DBVs becomes typed: generalization, aggregation etc. This typing system is extensible, affording new types. This requires that new information be stored concerning each DBV, and that the semantics of each object be stored alongside the object, e.g., using ontologies.

Returning to our example in Figure 3 consider an application that will access the contents of `S` in `DBV-Focus3`. Since there is no explicit reference to it in the `DBV-Focus2`, the only information is that the state of S in the third focus has been derived in some kind of relationship with the state of S in the second DBV. Let us consider that this is a generalization relationship, i.e., the state of S in the third DBV is a cartographic generalization of the state of S in the DBV-Focus2. In order to use this logical version of S in an application, the construction of DBV-Focus3 will require an algorithm that will: (1) verify that the type of the relationship is generalization; therefore, S must be transformed to the proper scale; (2) check the semantics of S, verifying that it is a satellite image, and therefore generalization concerns image processing, and scaling.

6. Conclusions and ongoing work

This paper presents our approach to handling multi-focus problems, for geospatial data, based on adapting the MDBV (multiversion database) approach to handle not only multiple scales, but multiple foci at each scale. Most approaches in the geospatial field concentrate on the management of multiple spatial or temporal scales (either by computing additional scales via generalization, or keeping track of all scales within a database via link mechanisms). Our solution encompasses both kinds of approach in a single environment, where an *ad hoc* working scenario (the focus) can be built either by getting together consistent spatio-temporal versions of geospatial entities, or by computing the appropriate states, or a combination of both. Since a DBV can be seen as a consistent view of the multiversion database, our approach also supports construction of any kind of arbitrary work scenarios, thereby allowing cooperative work. Moreover, derivation trees allow keeping track of the evolution of objects as they are updated, appear or disappear across scales.

Our ongoing work follows several directions. One of them includes domain ontologies, to support communication among experts and interactions across levels and foci. We are also concerned with formalizing constraints across DBVs (and thus across scales and foci).

References

Benda, L. E. et al. (2002). How to Avoid Train Wrecks When Using Science in Environmental Problem Solving. *Bioscience*, 52(12):1127–1136.

Buttenfield, B., Stanislawski, L., and Brewer, C. (2010). Multiscale Repreentations of Water: Tailoring Generalization Sequences to Specific Physiographic Regimes. In *Proc. GIScience 2010*.

Cellary, W. and Jomier, G. (1990). Consistency of Versions in Object-Oriented Databases. In *Proc. 16th VLDB*, pages 432–441.

Duce, S. and Janowicz, K. (2010). Microtheories for Spatial Data Infrastructures – Accounting for Diversity of Local Conceptualizations at a Global Level. In *Proc. GIScience 2010*.

Gao, H., Zhang, H., Hu, D., Tian, R., and Guo, D. (2010). Multi-scale features of urban planning spatial data. In *Proc 18th Int. Conf. on Geoinformatics*, pages 1 –7.

Leibovicia, D. G. and Jackson, M. (2011). Multi-scale integration for spatio-temporal ecoregioning delineation. *Int. Journal of Image and Data Fusion*, 2(2):105–119.

Longo, J. S. C., Camargo, L. O., Medeiros, C. B., and Santanche, A. (2012). Using the DBV model to maintain versions of multi-scale geospatial data. In *Proc. 6th International Workshop on Semantic and Conceptual Issues in GIS (SeCoGIS 2012)*. Springer-Verlag.

McGuire, M. P., Janeja, V. P., and Gangopadhyay, A. (2011). Characterizing Sensor Datasets with Multi-Granular Spatio-Temporal Intervals. *19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*.

Medeiros, C. B., Joliveau, M., Jomier, G., and Vuyst, F. (2010). Managing sensor traffic data and forecasting unusual behaviour propagation. *Geoinformatica*, 14:279–305.

Oosterom, P. (2009). Research and development in geo-information generalisation and multiple representation. *Computers, Environment and Urban Systems*, 33(5):303–310.

Oosterom, P. and Stoter, J. (2010). 5D Data Modelling: Full Integration of 2D/3D Space, Time and Scale Dimensions. In *Proc. GIScience 2010*, pages 310–324.

Parent, C., Spaccapietra, S., Vangenot, C., and Zimanyi, E. (2009). Multiple Representation Modeling. In LIU, L. and OZSU, M. T., editors, *Encyclopedia of Database Systems*, pages 1844–1849. Springer US.

Parent, C., Spaccapietra, S., and Zimanyi, E. (2006). *Conceptual Modeling for Traditional and Spatio-Temporal Applications - the MADS Approach*. Springer.

Sarjakoski, L. T. (2007). Conceptual Models of Generalisation and Multiple Representation. In *Generalisation of Geographic Information*, pages 11–35. Elsevier.

Spaccapietra, S., Parent, C., and Vangenot, C. (2002). GIS Databases: From Multiscale to MultiRepresentation. In *Proc. of the 4th Int. Symposium on Abstraction, Reformulation, and Approximation*, SARA '02, pages 57–70.

Xavier, A., Rodorff, B., Shimabukuro, Y., Berka, S., and Moreira, M. (2006). Multitemporal analysisof MODIS data to classify sugarcane crop. *International Journal of Remote Sensing*, 27(4):755–768.

Zhou, S. and Jones, C. B. (2003). A multirepresentation spatial data model. In *Proc 8th Int. Symposium in Advances in Spatial and Temporal Databases – SSTD*, pages 394–411. LNCS 2750.