Flexible Database Structures for Land Records

Richard MOLERO, Venezuela, Michael BARRY and Andrew HUNTER, Canada, Thaer SHUNNAR, Palestinian Authority

Key words: Land tenure, LIS, flexible databases, computerised and computer assisted LIS, post-conflict, land registration.

SUMMARY

Considerable effort has been put into creating standardised conceptual models for land records, such as the Land Administration Domain Model (LADM). At the same time, there are a number of initiatives to develop land records systems for uncertain situations where standardised models are unlikely to represent the situation on the ground adequately. These initiatives include the Talking Titler model, the Social Tenure Domain Model, and the Open Source Cadastral and Registry (OSCAR) toolset and its accompanying OSCAR Conceptual Data Model (OCDM).

Uncertain situations include complex customary systems, urban informal settlements where tenure practices draw on both western and traditional custom, post-conflict situations where power vacuums allow powerful elites to grab land, and changing administrative situations where institutional structures change as a situation stabilises. If justice and fairness are the principal values which drive the improvement of a land administration system, then there is an argument that the technical design of a land record system should be flexible. It should facilitate the merging of records if the initial design is found to be unsuitable or records may be held by different private institutions and government. It should also allow for the continual specialisation of data classes as a situation stabilises and for methods of reflecting unofficial or informal transactions. However, there is a tension between the two antecedents of what will make users likely to use such a system, ease of use and usefulness. Simple systems are most likely to be easy to use and flexible systems which are most likely useful in a complex situation, but which may not be easy to use. In this paper we examine the three different initiatives and consider methods for addressing specific scenarios in a changing situation.

Flexible Database Structures for Land Records

Richard MOLERO, Venezuela, Michael BARRY and Andrew HUNTER, Canada, Thaer SHUNNAR, Palestinian Authority

1. INTRODUCTION

Flexible database structures are likely to form a significant area of interest in land administration systems in the next decade or two. Over the past decade, there has been much interest in standardized conceptual models for land records, such as the Land Administration Domain Model (LADM). However, uncertain land administration scenarios are unlikely to be well served by standardized models. Typically, user needs are ill defined and if justice and fairness are driving principles in a land administration programme, then we should be careful not to inadvertently extinguish the interests of vulnerable groups. We argue that the technical design of a land record system in uncertain situations should be sufficiently flexible to handle frequent change in system requirements, unconventional data forms and structures, and unforeseen user requirements. Flexible land record systems are best suited to complex, perhaps uncertain, land tenure situations. These include customary systems, urban informal settlements where tenure practices draw on both western and traditional custom, post-conflict situations where power vacuums allow powerful elites to grab land, changing administrative situations where institutional structures change as a situation stabilises, and land restitution evidence gathering exercises.

Land record systems may be paper based, computer assisted or fully computerised. We argue that any of these systems need to be flexible in order to respond meaningfully to what is required of the system as a situation changes. In this discussion we focus on flexibility in particular technical aspects of the system. In the software development literature, flexibility has been applied somewhat loosely. For our purposes, we draw on Johnson *et al.* (2005) and IEEE (1990) to define it as a design which can accommodate changes demanded of the software with minimal modification of its components.

We examine land tenure record models that are currently being developed and then look at a number of technical options to create flexible systems. We first describe four existing conceptual models: the Land Administration Domain Model (LADM), the Social Tenure Domain Model (STDM), OSCAR Conceptual Data Model and the Talking Titler Model. Our research group is developing the last mentioned system. Following this is a discussion of technical options, the scope of which we limit to schema evolution and schema versioning.

2. EXISTING MODELS

2.1 Land Administration Domain Model (LADM)

Drawing on Cadastre 2014, a team in the Netherlands developed the Core Cadastral Domain Model (CCDM) (van Oosterom and Lemmen 2003). The initiative is linked to initiatives by the Open GIS Consortium (OGC) and Infrastructure for Spatial Information in the European Community (INSPIRE), amongst others. The CCDM's primary goals were: (1) to allow efficient and effective cadastral system development without the need to re-implement the same functionality in different systems, and (2) to provide a shared ontology to facilitate better communication and sharing of cadastral data. The latter objective is particularly relevant in Europe (van Oosterom and Lemmen 2003). The user community raised some concerns about the broad definition of cadastre (a definition with which we concur incidentally), and so Version 1.1 was renamed the Land Administration Domain Model (LADM) to allow a broader semantic interpretation (Hespanha *et al.* 2008).

The LADM core is based on three classes Person, RegisterObject, and RRR (Right, Restriction, and Responsibility, which can be expanded into a number of specialised child or sub-classes. The relationship between Person and RegisterObject is represented indirectly via RRR as per the UML class diagram below (figure 1). This diagram is the simplest representation of part of 5 UML packages that represent the whole LADM model. Each package represents an independent aspect of the cadastral domain in the model and includes packages that represent: people (Person); things that can be registered by the system (RegisterObject); the legal/administrative environment that recognizes or formalizes rights in things (RRR); the survey system that support rights in things; and a geometric/topological package as defined by OGC/TC 211 (Lemmen and van Oosterom 2006; Hespanha *et al.* 2008).

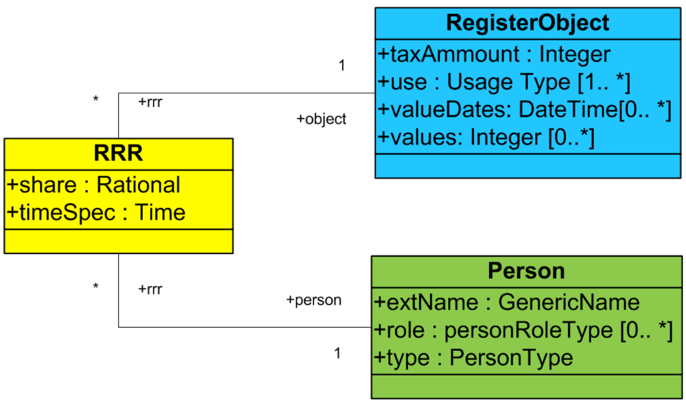

Figure 1: Core of LADM after Lemmen and van Oosterom (2006).

The representation of rights as a separate class in LADM stresses their importance in land administration and makes it easier to find information about rights in any implementation of this model. It also makes it easier to represent relations between rights and other classes.

Our understanding is that the LADM has been designed with a conventional government based land administration information system in a stable situation in mind. However, in situations where there is uncertainty and systems are vulnerable to rapid change, a rigid model based on the LADM is unlikely to be suitable. We may not know which of the classes, sub-classes or attributes are relevant at a particular time as, for example, institutions and their mandates change, power relations change, and land policy, laws and regulations develop. However, the different object classes, sub-classes and their interrelationships specified in the LADM are, of course, relevant as a framework for designing a flexible land records system.

2.2 Social Tenure Domain Model (STDM)

To address situations where the LADM might be ill suited, the Social Tenure Domain Model (STDM), a specialization of LADM, was developed in partnership between a number of academics and land professionals in the Netherlands in conjunction with the International Federation of Surveyors (FIG) and UN-Habitat. The STDM aims to model the person-land relationship regardless of its formal/legal status (Lemmen *et al.* 2007). The STDM conceptual model is based on the same three core classes as LADM but class names have been changed, and attributes have been added. In STDM the RegisterObject class has been changed to SpatialUnit and the RRR class changed to SocialTenureRelation. Lemmen *et al.* (2007) argue that it is possible to merge formal and informal tenure systems in STDM. This is achieved by introducing lookup tables and keywords to represent different kinds of SocialTenureRelations (formal and customary relations, instead of rights, restrictions and responsibilities) such as ownership, apartment rights, possessory rights, Waqf (Islamic law), occupation interest, and other similar rights and interests (see figure 2).

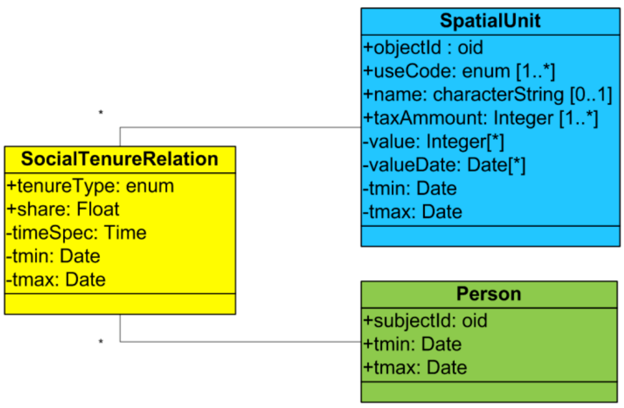

Figure 2: Core of the Social Tenure Domain Model (STDM) after Lemmen *et al.*(2007).

2.3 Open Source Cadastre and Registry (OSCAR)

Open Source Cadastre and Registry (OSCAR) is a framework that uses the LADM as a point of departure. It offers a set of tools for software implementation in an effort to overcome the limitations (e.g. global consensus for the model and stable user requirements) of the LADM in building software solutions for land administration (Hay and Hall 2009). OSCAR makes use of semantic Web technologies to free system development from a particular domain ontology (Hay and Hall 2009).

OSCAR has two main components, the conceptual data model (see figure 3) and the architecture guidelines for the software implementation of a model. The conceptual data model is built on the LADM classes, but replaces the state-based temporal representation with an event-based representation in order to include better support for temporal processes. Pelekis *et al.*'s (2005) review of space-time databases focus' on these temporal concepts in some detail.

The architecture guidelines suggest thinking of the system as a centralized core that can be adapted by inserting new software components (plug-ins) or extending existent components. For example, an initial implementation might comprise only the classes represented in figure 1 (e.g. Person Class for a census survey prior to adjudication). As local conditions stabilize, and become more clearly defined, additional software components (packages/plug-ins) might be added that legitimize rights captured by the initial system (i.e. a legal/administrative package is added to the system), or it becomes evident that other attributes are required in order to substantiate a claim. OSCAR, as the name suggests has adopted a free open source software (FOSS) philosophy, which refers to four freedoms of use (Steiniger and Bocher, 2009): the freedom to run the program for any purpose; the freedom to study how the program works and adapt it to your needs; the freedom to redistribute copies; and, the freedom to improve the program and to release your improvements to the public so that the whole community benefits. OSCAR has been developed using the Eclipse Rich Client Platform, and as such, its software implementation guidelines mirror the guidelines defined by Eclipse (http://www.eclipse.org/) for open source project development (Rubel 2006).

The OSCAR conceptual data model (OCDM) is based on a legal document (instrument) against which transactions and their modifications (events and amendments) are stored (see figure 3). The legal instrument is the only relationship between people (agents) and land units (registry object). Although OCDM was initially based on LADM, its creators envisage that OCDM will evolve independently of the LADM (Hay and Hall 2009).

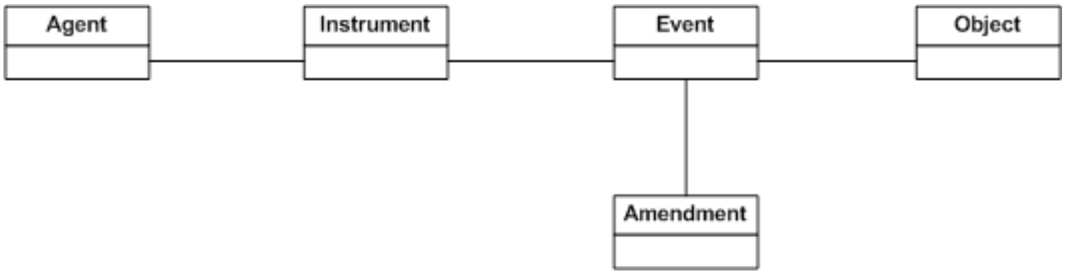

Figure 3: Core of OCDM after Hay and Hall (2009).

The OSCAR implementation guidelines cater for flexibility in the application by incorporating the following: semantic web technologies (XML and RDF), workflow process modelling, document repositories, the Eclipse Rich Client Platform (Eclipse RCP), and programming skills in order to extend or adjust the existing functionality (via plug-ins) to meet specific requirements of a registration system for a local jurisdiction.

Hay and Hall (2009) argue that the OSCAR initiative is suited to situations where both IT skills and funding are scarce. However, the implementation guidelines suggest that considerable IT skills may be required to adapt a system to a user's unique needs and hardware infrastructure. Additionally, as with all of the systems described here, the framework provides for a great deal of flexibility in building a system, and as system functionality is increased, it is probable that the complexity of the system will also increase.

2.4 Talking Titler Model

The Talking Titler (TT) system was conceived based on observations in urban informal settlements and rural land restitution projects in South Africa. Part of the original thinking – influenced in part by the Cybertracker system (see Cybertracker 2009) - was to develop systems where members of local communities could be trained to collect, and perhaps manage, land administration information data themselves using field computers, video cameras and digital audio recorders. Early tests involved using a palm computer to conduct a census survey and then developing a community based information system for a land reform project (e.g. Barry *et al.* 2002, Barry and Mayson 2000). In uncertain situations, if fairness and equity are a major objective in addressing the situation, there is often a need to collect a range of data types that would not normally be included in a conventional land information system. In concept, the initial thinking was for a means of relating multi-media data, such as video and audio clips, in a land record system. Later, the thinking was extended to how to synthesise data from a variety of sources and institutions in post-conflict situations (e.g. Augustinus and Barry 2006) and customary systems (Barry and Khan 2005). In essence the challenge is to incorporate data from disparate sources, in a variety of formats in an information system and relate these data in a meaningful way. The data may be structured (as you would expect in the database supporting an established land registration system), semi-structured, and unstructured (e.g. a video interview).

The Object Manager is the simplest form of the Talking Titler system. As can be seen in figure 4, the conceptual data model is similar to both the STDM and the OCDM. Each object class can have a relationship with itself or a similar object in the same class. For example, a land parcel may enjoy an easement over another parcel. As with the OCDM, an instrument, such as a title or a property administration file, is generally expected to guide the process of recording interests in land. Accompanying this, media files such as the physical title itself, the accompanying covenants and a range of video files might provide the actual evidence and so these would be linked to the title, perhaps to each other and perhaps, in the case of videos to the persons appearing in the videos. The design also allows for events which conventional land registration system are generally not designed to handle. For example, in a case when no reference instrument exists for a particular relationship, or when buyers and sellers choose to ignore a formal legal system such as land registration and engage in a private or informal land transaction. In the latter case a direct relationship between persons (e.g. vendor and purchaser) and a land object may be most suited to reflecting this relationship. There may or may not be some form of media item to reflect this latter arrangement (e.g. a private contract, an audio recording of an oral agreement, a photograph of the symbolic delivery process), and so perhaps links between the land object and the media items and links between the media items and the persons who are party to the transaction may be required too.

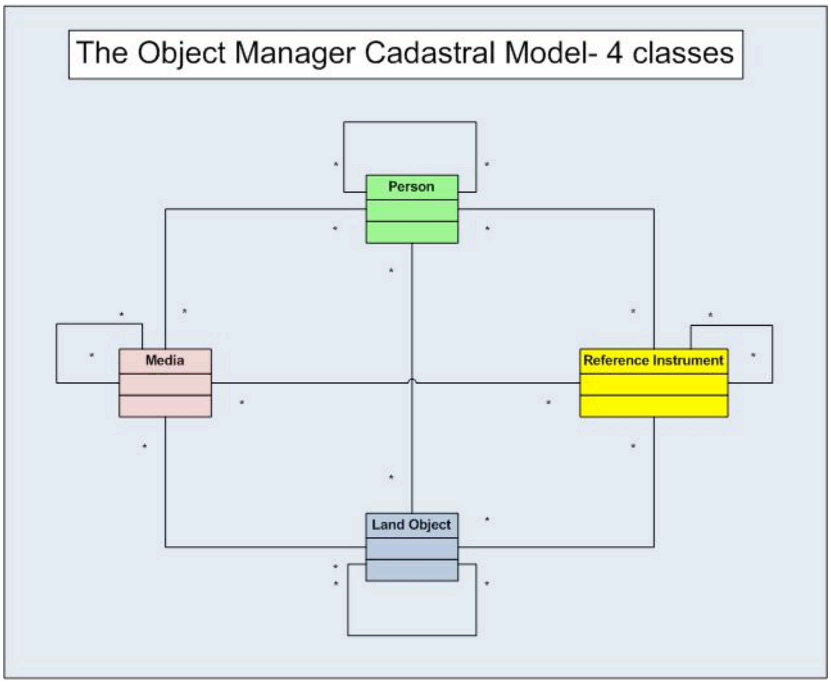

Figure 4: The Talking Titler Object Manager Model (Muhsen and Barry 2008).

The system is designed to allow for the greatest level of flexibility. Each object class can be related to itself, and it can also be related to every other object class to deal with the range of relationships, formal and informal, legal and illegal, private or public, that may occur in an uncertain situation.

At present a software version is still in prototype form. A major challenge is how to tailor the system to address the specific requirements of a particular situation and to make the system easy to use. In the physical implementation of the model, each parent class has a free form description field in which data can be described and detailed. However, while this allows a great deal of flexibility in how the system may be used, it increases the level of complexity and so the software is not easy to use and requires a skilled person to design rigorous procedures. In addition it is a major challenge to design user interfaces which are user friendly and which can handle complexity in the system.

2.5 Comparison and Critique

Direct comparison between these different models is difficult without subjecting them to tests on the same data sets. There are major advantages to the LADM's standardised ontology and semantics for government information systems in stable environments. However the LADM was not designed for the uncertain situations that we have discussed, but rather for situations such as the European Union where major changes in the socio-political environment would be most unusual. In addition, its developers continually change and develop the design in response to user needs (e.g. the concept of a parcel has been reinstated as a class) (Lemmen *et al.* 2009). OSCAR is a development methodology which should be applicable to any of LADM, STDM and the Talking Titler (TT) model.

In superficial analysis, the TT and STDM are similar in concept and design and intended to serve the same or at least similar purposes. At a glance the TT system allows for more relationships and therefore more flexibility and complexity in the situation on ground, but there is nothing to prevent a developer working on the STDM from providing this level of flexibility too. Likewise, the TT system can use the keyword method advocated in the STDM.

There are a number of possible problems associated with the TT approach. One of these is that by allowing every single type of relationship possible to be created, a user might create a database which has a host of meaningful and perhaps not so meaningful relationships modelled in it. Extracting meaningful information from it may be difficult and operators may spend a great deal of time creating unnecessary relationships during data entry. Thus the users' processes and office procedures need to be carefully designed, documented and managed for a particular application.

The STDM differentiates between tenure types (social tenure class specialisation) by assigning specific keywords to these types. As we see it, the major advantage of this approach is that this is a simple and arguably user friendly way to approach recording land tenure relations in many of the situations alluded to earlier. However, rigorous process design and strict adherence to defined procedures may be very important – as they are with the Talking

Titler system in its current form. We have observed a similar system in a land taxation records system (i.e. keyword identifiers), where users generated a vast number of keywords over a number of years. Many of these keywords were semantically similar; users could apply similar meanings to two or more words in the set and thus the datasets ended up being inconsistent.

Thus the tension between flexibility and simplicity is clearly apparent. Flexibility may contribute to achieving justice and fairness in a complex socio-political situation, but at the cost of creating a complex information system which turns out to be difficult to use. Simplicity tends to make the information system easier to use and therefore more likely to be effective as an information system on its own, but it may not be effective in serving some of the main purposes for which the system was created. These tensions are not new. Taylor (2008) describes how similar debates raged in legal and administrative circles around proposed introductions of the Torrens system of land registration during the mid nineteenth century.

3. FLEXIBLE DATABASE STRUCTURES

In our observation, the way the land records are stored and organized differ from one LIS to another, and the selection of a specific architecture will have an impact on the way the land record can be accessed, the hardware resources required to accommodate it, and the overall performance of the system. In addition, land records differ from typical data records because they are both time and location sensitive, and they often incorporate large amounts of semi-structured data.

From a technical point of view, in general data can be stored and organized in three ways: structured data, semi-structured data, and unstructured data. Typical relational databases manage structured data, as the structure is known in advance and constraints are defined to store data only if the constraints are satisfied. Semi-structured data are less constrained than data in well designed relational databases. Semi-structured data are often organised using tags that arrange the data according to semantic meaning or hierarchies within the data (Papakonstantinou *et al*. 1995). XML (W3C 2008), RDF (W3C 2004), GML (Cox *et al*. 2005), and LandXML (LandXML 2008) are examples of widely used schemas to organise, store and exchange semi-structured data, and enable a data structure to be adapted as requirements change. Finally, unstructured data is raw data that does not follow any structure at all. Multimedia data such as video and audio data, and free format written documents, are examples of unstructured data. We note that video and audio data file formats are rigorously structured according to international standards, but their content can be considered unstructured.

XML has emerged as the data format of the World Wide Web. It has been used primarily to exchange data between applications even when these applications are heterogeneous (applications using different data structures from different sources). Its success has been due to two main factors: simplicity and flexibility. Simplicity is due to its easy to understand tree structure that keeps the data organized, instead of a non-hierarchical or flat structure typical of relational databases. Its flexibility allows frequent changes to the XML schema, thereby allowing a designer to adjust the structure of the data according to specific needs. XML and its derivatives' popularity have created a need for efficient storage and query capabilities (search and retrieval). Progress in this sense has been achieved with ArcGIS including its own proprietary XML format to store cadastral information, CadastralXML (ESRI 2007). A more recent development is Apache CouchDB (http://couchdb.apache.org/), an Apache open source project that offers a JSON (JavaScript Object Notation) document-oriented database with query and indexing capabilities. JSON contains its own native language compliant data structure that allows data to be processed more efficiently than the equivalent DOM (Document Object Model) calls required to process an XML document, and there are numerous tools available that can translate XML to JSON (Anderson *et al.* 2009).

As mentioned earlier, land records might comprise structured, semi-structured and unstructured data. Additionally, due to the complexity of diverse land administration situations, the major challenge for developing land administration systems is to keep reliable land records for situations undergoing constant change (van der Molen 2002) and uncertainty (Barry and Fourie 2002). Therefore, new land administration systems should be flexible enough to cope with unavoidable and unexpected changes occurring in the land administration scenario while maintaining persistent land records – records that remain consistent with one another and are not lost as the information system changes.

3.1 Dealing With Changes

Developing specific information systems typically involve gathering requirements, modelling the situation, and then implementation. When requirements are changed, or new requirements arise, reengineering is required and a new version of the software is delivered. Changes in requirements often involve changes in data structure and changes in software components that use the modified data structures. Reengineering of this nature often implies high implementation costs, long turnaround times and a need for highly skilled developers and IT support.

In the case of land records, the land administration scenario is represented by a conceptual model. This model is later implemented in a land information system. The architecture behind a land information system can be seen, to keep it simple, as the combination of a user interface, software logic components, and the database. Now, changes in the user environment (requirement changes) imply changes in the conceptual model (conceptual changes), which implies changes in the land information system implementation (application changes). Finally, changes in the application imply changes to the user interface, software logic components and database. A new requirement might include adding new attributes to the existing data so the database must be updated to store these new attributes, then the software components must be updated to include support for the new attributes (including user interface and routines that process them).The process is illustrated in figure 5.

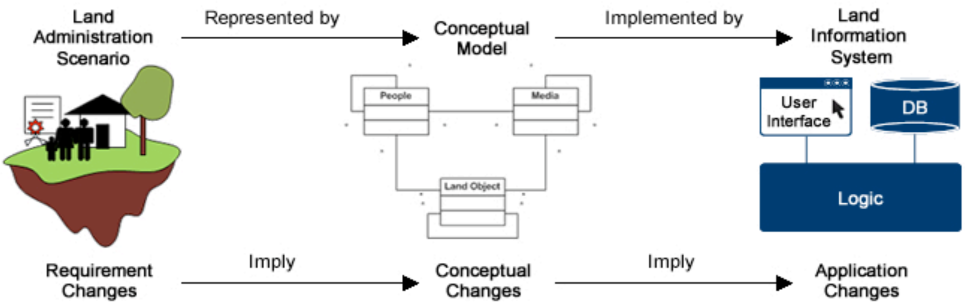

Figure 5: Simplified Land Information System Development Process.

It is our observation that regardless of the type of changes encountered in a land administration scenario they require translation into conceptual changes such as the addition of new classes, modification of existing classes, addition of attributes, modification of existing attributes, creation of new associations between classes (relations), and specialization of classes, to mention just a few. This characterization of changes allows, to a degree, anticipation of the kind of changes expected at the application level each time a change is made.

3.2 Self Adapting Software

Application changes at the user interface or to software logic components are beyond the scope of this paper. Some work has been done in this sense with the OSCAR framework where software components can be added and modified by skilled IT professionals (Hay and Hall 2009). However, as indicated earlier, a lack of skilled IT staff is often a characteristic of uncertain land administration scenarios. Therefore, it is desirable, as a long term objective, to reduce the need for skilled personnel in order to alleviate this problem.

Our research extends beyond the database level in an effort to incorporate as many IT roles within the application as possible. One approach is to develop software that self-adapts in response to the changes in the conceptual model mentioned previously. This is also outside the scope of this discussion, but it is one approach to the challenges we have mentioned earlier that we are exploring. More information on a self-adaptive software architecture can be found in Bradbury *et al.* (2004).

3.3 Database Schema Evolution and Schema Versioning

Regardless of whether land records are stored in a structured way (tables in a relational database), or in semi-structured way (e.g. derivates of XML), both require a schema that dictates the organization of the data. A schema, the conceptual model translated into the physical model, is the collection of items that define the structure of the data in a database. In the case of relational databases the schema consists of the definition of the tables, attributes, indexes, constraints and the relationships between them, while in XML the schema comprises tag definitions, their hierarchy and constraints.

Designing a land information system which is expected to evolve through frequent changes in user requirements should be able to cope with both changes in the schema at the database level (see figure 6), and changes in the software components that comprise it. At the database level there are two methods to address changes in the schema: schema evolution and schema versioning. The wide spread accepted definitions for both approaches have been provided by Roddick (1995) as follows:

- Schema Evolution. *A database system supports schema evolution if it permits modification of the database schema without the loss of extant data. No support for previous schema is required*.

- Schema Versioning. *A database system accommodates schema versioning if it allows the querying of all data, both retrospectively and prospectively, through user-definable version interfaces*.

Two major challenges arise when implementing these methods: the level of granularity of the versions in the first place (versions of parts of the schema vs versions of the whole schema), and then how to propagate the changes on the data (Roddick 1995). Roddick *et al.* (2001) later extended schema versioning to consider spatial and spatio-temporal data. More recently, schema evolution and schema versioning have been applied to semi-structured data by Symanosvky (2008). The major benefits obtained when applying these methods include allowing changes in the schema without affecting the data already stored in the database, and still providing mechanisms for access to the data (Symanosvky 2008).

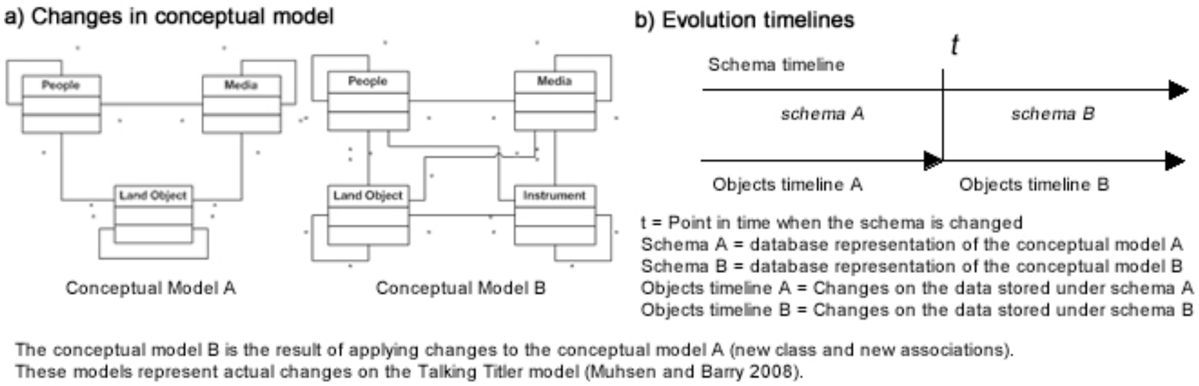

Figure 6: Evolution timelines of changes in the database.

The approach adopted in our research is under experimentation and consists of keeping track of changes according to two timelines: schema timeline and objects timeline. This means that both changes on the schema and changes on the objects under each schema are recorded in order to be able to reconstruct the state of the database (not only the structure but also the state of the objects) at any point in time. Figure 5 illustrates these timelines. The figure shows a point at time *t* where a change in the schema occurred (e.g. addition of the new class and relations with the existent classes). It also shows that changes on the objects are recorded under each schema.

It is important to mention that change propagation (i.e. migrating data between different versions of the schema) is avoided and views (i.e. users' view of the data) are adapted to handle this instead. This way a new schema will be translated into changes on the views that reflect the current state of the database under the current schema, initially under the assumption that the data comes from a single source (i.e. one database that incorporates and manages all the different versions of the schemas). The next step would be to consider more than one source, with each source having its own schema, so as to enable integration of data from different sources.

4. DISCUSSION AND CONCLUDING REMARKS

In essence, in uncertain situations land records system users require a system that remains useful during periods of significant change; that is an information system that is sufficiently flexible to handle complexity and continual change. The other aspect of the challenge is that the system should be easy to use; complexity should remain hidden, and user interaction should be sufficiently intuitive such that the user is able to perform their functional land administration duties via the software system without hindrance, hesitation, or question (Rubin and Chisnell, 2008). Herein lies the major tension and the major challenge. Arguably the problem cannot be solved to the extent that we remove the tension completely, but we do need to explore ways of reducing this tension.

Strategies to reduce this tension should attempt to lower the level of IT skills required to cope with changing system requirements. This involves using built-in software evolution mechanisms which reduce the need for highly skilled IT staff. Self adapting software and schema versioning are complementary strategic options which we are exploring.

We envisage a database that supports schema evolution and schema versioning, allowing the coexistence of different schemas or variations of the same schema. This is useful where merging of records is needed if the initial design is found to be unsuitable, or if records are held by different private institutions and/or government and each organization maintains different schemas. It should also allow for the continual specialisation of data classes as a situation stabilises, and for methods that enable the documentation of unofficial or informal transactions. Interoperability is also an important consideration. Data exchange via OGC/ISO compliant standards such as GML, LandXML and/or CadastralXML should be supported.

Once the land records database is established, it is possible to extend the software evolution mechanisms to the software architecture as a whole. Self-adaptive software principles may prove to be useful in this regard and so they need to be considered. Furthermore, data mining techniques can be applied to look for non-trivial patterns in the data, to extract information beyond simple queries on the land records, and to detect irregularities such as fraud.

In conclusion, the incorporation of all the mechanisms mentioned here should result in a sufficiently flexible land information system that can handle constant changes in user requirements to make the information system both useful and easy to use. However, we have no illusions about the magnitude of the challenge.

ACKNOWLEDGMENT

This study has been conducted with the support of the John Holmlund Chair in Land Tenure and Cadastral Systems and the Canadian National Science and Engineering Research Council Talking Titler project

5. REFERENCES

Anderson, J.C., Lehnardt, J., and Slater, N., 2009, *CouchDB: The Definitive Guide*, O'Rielly Media, Retrieved 2010, January 18, from http://books.couchdb.org/relax/.

Augustinus, C., and Barry, M., 2006, Land Management Strategy Formulation in Post Conflict Societies, *Survey Review,* 38(302), 668-681.

Barry, M., and Mayson, D., 2000, Informal Settlement Characteristics in a Rural Land Restitution Case: Elandskloof, South Africa, *Sociological Research On-Line*, 5(2), August 2000:70:2.

Barry, M., and Fourie, C., 2002, Analyzing cadastral systems in uncertain situations: a conceptual framework based on soft systems theory, In *International Journal of Geographic Information Science*, 16(1), 23-40.

Barry, M., Roux, L., Barodien, G., and Bishop, I., 2002, Video-evidencing and Palmtop Computer Technology to support formalising Land Rights, *Development Southern Africa* 19(2), 261-272.

Barry, M., and Khan, K., 2005, *Law and Policy Implications of Multimedia Land Records: The Talking Titler Project*, FIG Working Week and 8th International Conference of the Global Spatial Data Infrastructure, Cairo, Egypt, 16-21 April 2005.

Bradbury, J.S., *et al.*, 2004, A survey of self-management in dynamic software architecture specifications, In *Proceedings of the 1st ACM SIGSOFT workshop on Self-managed systems*. 33-38.

Cox, S., Daisey, P., Lake, R., Portele, C., and Whiteside, A, 2005, *OpenGIS® geography markup language (GML) implementation specification, version 3.1.1.* Wayland, MA, USA: Open Geospatial Consortium, Inc.

Cybertracker, 2009, Retrieved 2010, January 18, from http://www.cybertracker.co.za/.

ESRI, 2007, ArcGIS® Survey Analyst Cadastral Editor: Cadastral XML—A Technical Description, Retrieved 2010, January 18, from http://downloads2.esri.com/support/whitepapers/ao_/ArcGIS_Survey_Analyst_Cadastral_XML.pdf.

Hay, G., and Hall, B., 2009, Architecture for an open source land administration application, In *FIG Working Week 2009*, Eilat, Israel, 3-8 May, 2009.

Hespanha, J., van Bennekom-Minnema, J., van Oosterom, P., and Lemmen, C., 2008, *The Model Driven Architecture approach applied to the Land Administration Domain Model version 1.1 - with focus on constraints specified in the Object Constraint Language*, Paper presented at the Integrating Generations, FIG Working Week, Stockholm, Sweden.

IEEE, 1990. IEEE standard glossary of software engineering terminology. *IEEE Std 610.12-1990*.

Johnson, B., Woolfolk, W.W., Miller, R., and Johnson, C., 2005, *Flexible software design - systems development for changing requirements*. Boca Raton: Taylor & Francis Group.

LandXML, 2008, LandXML-1.2, Retrieved 2009, September 22, from http://www.landxml.org/spec.htm.

Lemmen, C., Augustinus, C., van Oosterom, P., and van der Molen, P., 2007, *The Social Tenure Domain Model - Specifications of a First Draft Model*. Paper presented at the Strategic Integration of Surveying Services, FIG Working Week, Hong Kong SAR, China.

Lemmen, C., and van Oosterom, P., 2006, Version 1.0 of the Core Cadastral Domain Model. In *Proceedings XXIII FIG Congress*, 8-13, Munich, Germany.

Lemmen, C., van Oosterom, P., Uitermark, H., Thompson, R. and Hespanha, J., 2009, Transforming the Land Administration Domain Model (LADM) into an ISO Standard (ISO19152) (3282).

Muhsen, A-R, and Barry, M., 2008, Technical Challenges in Developing Flexible Land Records Software, *Surveys and Land Information Science*, 68(3), 171-181.

Papakonstantinou, Y., Garcia-Molina, H., and Widom, J., 1995, *Object Exchange Across Heterogeneous Information Sources*, Paper presented at the Eleventh IEEE International Conference on Data Engineering (ICDE 1995), Taipei, Taiwan.

Pelekis, N., Theodoulidis, B., Kopanakis, I., and Theodoridis, Y., 2005, Literature review of spatio-temporal database models, *The Knowledge Engineering Review*, 19(03), 235–274.

Roddick, J.F., 1995. A survey of schema versioning issues for database systems. *Information and Software Technology*, 37(7), 383-393.

Roddick, J.F., Grandi, F., Mandreoli, F., and Scalas, M.R., 2001, Beyond schema versioning: A flexible model for spatio-temporal schema selection, *Geoinformatica*, 5(1), 33–50.

Rubel, D., 2006, The Heart of Eclipse, *Queue*, 4(8), 36-44.

Rubin, J., & Chisnell, D. (2008). Handbook of usability testing: How to plan, design, and conduct effective tests. 2. Retrieved April 14, 2009, from http://common.books24x7.com.ezproxy.lib.ucalgary.ca/book/id_25203/book.asp.

Simanovsky, A.A., 2008. Data schema evolution support in XML-relational database systems. *Programming and Computer Software*, 34(1), 16–26.

Steiniger, S., & Bocher, E. (2009). An overview of free and open source GIS developments. *International Journal of Geographical Information Science*, *23*(10), 1345 - 1370.

Taylor, G., 2008, *The Law of the Land, The Advent of the Torrens System in Canada,* University of Toronto Press, Toronto.

van der Molen, P., 2002, The dynamic aspect of land administration: An often-forgotten component in system design, *Computers, Environment and Urban Systems*, 26(5), 361-381.

van Oosterom, P., and Lemmen, C., 2003, Towards a standard for the cadastral domain, *Journal of Geospatial Engineering*, *5*(1), 11-27.

W3C, 2004, RDF/XML Syntax Specification (Revised). Retrieved 2010, January 14, from http://www.w3.org/TR/REC-rdf-syntax/.

W3C, 2008, Extensible markup language (XML) 1.0 (Fifth edition), Retrieved 2009, September 22, from http://www.w3.org/TR/REC-xml/.

BIOGRAPHICAL DATA

Richard Molero is a MSc graduate student in the Department of Geomatics Engineering at the University of Calgary. He obtained a BSc degree in Computer Science from La Universidad del Zulia - Venezuela, in 2003 and has work experience since in several IT roles including DBA, Software Architect, Web Application Development and Systems Auditor. His research interests include evolutionary land information systems, self-adapting software and current trends in database management systems.

Mike Barry holds the John Holmland Chair in Land Tenure and Cadastral in the Geomatics Engineering Department at the University of Calgary, where has been since the end of 2002. Prior to this, he was at the Department of Geomatics at the University of Cape Town. He has PhD, MBA and BSc(Survey) degrees. His research interests are in land tenure, cadastral systems, and spatial data analysis.

Andrew Hunter is an Assistant Professor in the Department of Geomatics Engineering at the University of Calgary. He is a Professional Registered Surveyor with the New Zealand Institute of Surveyors. His research is focused on the acquisition, use, and analysis of dynamic spatial data.

Thaer Shunnar is a MSc student in the Department of Geomatics Engineering at the University of Calgary. He has a B.Sc. in Computer Engineering from An-Najah National University (Palestine). Prior to commencing his MSc he worked as a software developer and software engineer. His research interests are in data mining and knowledge discovery with a focus on land registration systems.

CONTACTS

Mr. Richard Molero
Department of Geomatics Engineering
University of Calgary
2500 University Dr. N.W.
Calgary, Alberta T2N 1N4
CANADA
Email: rmmolero@ucalgary.ca
tel +1 (403) 220-4981

Dr Michael Barry
John Holmlund Chair in Land Tenure and Cadastral Systems
Department of Geomatics Engineering
University of Calgary
Email: barry@geomatics.ucalgary.ca
tel +1 (403) 220-5834

fax +1 (403) 284-1980
www.geomatics.ucalgary.ca/~barry

Dr Andrew Hunter
Department of Geomatics Engineering
University of Calgary
Email: ahunter@ucalgary.ca
tel +1 (403) 220-7377
fax +1 (403) 284-1980

Mr. Thaer Shunnar
Department of Geomatics Engineering
University of Calgary
Email: taishunn@ucalgary.ca
tel +1 (403) 890-9133