Land Use Policy 49 (2015) 565–576

Contents lists available at ScienceDirect

Land Use Policy

journal homepage: www.elsevier.com/locate/landusepol

A model for the creation and progressive improvement of a digital cadastral data base

Rodney James Thompson[a,b,*]

[a] Delft University of Technology, Jaffalaan 9, 2628 BX Delft, The Netherlands
[b] Department of Natural Resources and Mines, Cadastral and Geodetic Services, GPO Box 2454, Brisbane, QLD 4001, Australia

ARTICLE INFO

Keywords:
Cadastral schema
Progressive development
Topology
Land administration domain model
Data cleansing
Open source

ABSTRACT

A digital cadastral data base (DCDB) is a big investment for a jurisdiction tasked with the administration of land boundaries. In the past, the development of such a database produced no real pay-back on investment before many years of time, and millions of dollars in cash had been committed.

The Land Administration Domain Model (LADM) (ISO-TC211, 2012) provides a schema in which the progressive creation and improvement of a DCDB is possible; to allowing benefits to be obtained even in the early stages of effort. It also incorporates the necessary structure to ensure that a useful historical record of the cadastre can be kept. This paper explores issues to be faced in the development of software based on the LADM, which retains the history of the cadastre, and allows for progressive improvement of the data. From experience gained in the development of cadastral databases of the Queensland (Australia) Department of Natural Resources and Mines, and the Netherlands Kadaster, a suggested logical schema is presented and discussed with respect to the requirements of a progressively developed and refined cadastral database.

Rather than each cadastral jurisdiction developing its own database structure from basic geometric primitives, this paper proposes the establishment of a cadastral schema, based on the LADM, which can support all levels of encoding, variable accuracy and topological purity, while maintaining a comprehensive history. This would allow data quality to vary by geographic and temporal location and would be configurable to allow for country profiles under ISO 19152; thus permitting local terminology and language to be retained. Many jurisdictions are having extreme difficulty in successfully creating a cadastral database, so an open source type of software development may be indicated and desirable.

This paper presents findings based on theoretical consideration and the construction of a proof of concept database, which indicate that such a schema is a practical proposition for the development of a digital cadastral data base.

Introduction

Typically, a DCDB is repository which is developed as an adjunct to the administration of interests in land “It usually includes a geometric description of land parcels linked to other records describing the nature of the interests, the ownership or control of those interests, and often the value of the parcel and its improvements. It may be established for fiscal purposes (e.g. valuation and equitable taxation), legal purposes (conveyancing), to assist in the management of land and land use (e.g. for planning and other administrative purposes), and enables sustainable development and environmental protection” (Österberg et al., 1995). Such a rich set of data is also frequently used for many other purposes, providing background mapping for assets management, network administration, and other high value activities.

A problem has been that database structures chosen to support a DCDB have been such that data cannot be stored until it has passed stringent tests of validity, therefore much manual cleansing and correcting is necessary. This is exacerbated by the fact that a DCDB is of limited usefulness until it is complete. The classical approach to data capture in a spatial database has been for the incoming data to be validated against a set of rules, usually set by the database vendor, and often not well defined. Any failure of these rules results in the data being rejected.

Unfortunately, this puts a giant hurdle in front of any organisation. If the data cannot be entered without being correct, it

* Correspondence to: Department of Natural Resources and Mines, Cadastral and Geodetic Services GPO Box, 2454 Brisbane, QLD 4001, Australia.
Tel.: +61 07 38963286; fax: +61 07 38963697.
E-mail address: Rod.Thompson@qld.gov.au

http://dx.doi.org/10.1016/j.landusepol.2014.12.016
0264-8377/

cannot be made visible to a wider audience, and no pay-back can be obtained. Correcting data on input is a difficult (and therefore expensive) process, and the only eyes on the data are those of the data capture operators. As an indication, the Queensland DCDB took about 12 years to capture to an acceptable standard of quality, at the estimated cost of AU$50 million in the currency of the 1980s (Diggles, 2008, p. 209 in Part 3). (This equates to approximately $25 per parcel.)

By contrast, many of the uses to which cadastral data is put do not necessarily do not need highly validated data, and can accept small imperfections such as “knots”, “overshoots”, “undershoots”, etc. It is important to remember that spatial data invariably has an intrinsic limit to its accuracy. For example, various mapping functions, including Web Map Services (WMS), Web Feature Services (WFS) and cadastral maps, searches etc., may be adequately supported by data with small imperfections. It is also important to be aware that validity and correctness are distinct concepts.

Traditionally, where information is publicly owned and maintained, and particularly when that data provides a legal framework for decision making, it has been the aim to prevent the release of data that might not be completely correct. An alternative viewpoint is that the possibility of errors in the database could be a reason to allow public viewing, so users might detect and report these errors (especially the ones that cannot be detected automatically).

This line of thinking supports the OpenCadastre concept (Keenja et al., 2012). As occurs in the OpenStreetMap, volunteers can enter data. Similarly, as in OpenStreetMap, users may correct each other’s entries. In cases of ‘conflicts’, cadastral experts could be consulted to resolve these issues. This may be counter-intuitive as cadastre is about authoritative registration and the guarantee of land ownership and title, but provided a distinction can be made in the metadata between volunteered and authoritative information, and this distinction can be held in the public view, it may be an effective way to achieve clean and complete data. At the very least, making data visible to the public and providing an error reporting mechanism will lead eventually to higher quality data.

There are several possible ways to encode the geographic information in a cadastral database. The LADM (ISO-TC211, 2012) defines 5 levels of encoding:

1. “Text-Based” Spatial Unit
2. “Point-Based” Spatial Unit
3. “Line-Based” Spatial Unit
4. “Polygon-Based” Spatial Unit
5. “Topology-Based” Spatial Unit

(with “Sketch-Based” as a sub category of Text-Based).

These are discussed and described in Thompson (2013).

As a DCDB matures, it can be expected that its quality will be improved both in terms of its accuracy (Tarbit and Thompson, 2006), and in terms of its topological correctness (Thompson, 2013). This may also involve changes in the level of encoding. For example, a jurisdiction with polygon-based encoding might convert to a topology based form. In the past, an improvement in encoding would require a reworking of the database, with re-programming, data conversion and very probably a loss of history.

It is an important part of any cadastral database (though sometimes overlooked) to maintain the historical record of land use in digital format. In providing this functionality the Queensland Government, like the Netherlands Kadaster, adopted forms of what is now known as the “Versioned Object” pattern (van Oosterom, 1997). The LADM itself uses this pattern, permitting a permanent and efficient storage of cadastral history within the database. In a progressively developed database with history, it must be recognised that older historic data will usually be of a lower state of accuracy and topological purity than has been achieved later and may contain errors that have subsequently been detected and corrected. It is however commonly accepted that history of “the cadastre as we knew it” is a valuable resource. One important issue with history is that it must not be necessary to jettison many years of that history if the level of encoding is changed, or to partition the database into incompatible layers to allow progressive improvement.

With regard to the LADM, Lemmen raises a critical question “Is the design implementable and applicable in a real life situation?” (Lemmen, 2012, p. 14). The “FLOSS Cadastre Project” (Steudler et al., 2010) and “OSCAR” (Hay and Hall, 2009) argue that this is the case, and that a practical database can be built on the principles of the Social Tenure Domain Model (STDM) which is a profile of LADM. This paper explores the question further in terms of the LADM support of a fully mature cadastral database.

Original research presented in this paper includes: (1) The building of a database closely based on the LADM structure, and the loading of that database with realistic data quantities. (2) The use of that database to explore complexity issues. (3) The finding that levels of encoding can co-exist within the same cadastral database and that 2D and 3D parcels can be mixed.

The paper is structured as follows: the next section discusses the issue of data quality within a cadastral database; third section considers the requirement to record history; fourth section proposes a data model, and explores its capabilities; fifth section presents findings of an experimental database used to investigate the model; sixth section summarises the conclusions and the last section suggests further work.

Quality of a developing DCDB

Positioning accuracy

Measurement accuracy has improved over the years, but there is and will always remain a limit to the accuracy that can be achieved by any measurement. Typically, as a DCDB is being developed, the accuracy of the earliest capture will be lower than that of data added later (Effenberg and Williamson, 1996). The other major issue in this regard is that a DCDB may be the most useful and complete (or only) base mapping layer available, so it is often used as a background for assets management and for the positioning of street furniture by local authorities, electricity supply organisations, telecom, etc. (Priebbenow, 1993). It is usually true that the local relative accuracy of a DCDB is significantly higher than its absolute positional accuracy. This is certainly true in Queensland, where individual surveys are carried out to high accuracy, but the positioning of the property in absolute terms may have been done using a significantly lower accuracy technique (Diggles, 2008).

As an example, in Fig. 1A, an underground cable junction may be positioned 1 m from a property boundary. If a later survey is done which improves the positioning of the land parcels, it is not acceptable to lose this relativity between the cable junction and the parcels (Fig. 1B). The approach used in Queensland was that each time a vertex in the database was moved, a “point movement” record was generated, giving the old and new location of the vertex. These could be processed by the infrastructure authority to keep its asset locations up to date. The approach was not totally satisfactory, as it relied on the update operators maintaining point integrity, and not simply deleting linework and entering new points and lines. The result of this is that the major (paying) customers for the DCDB data are loath to see a large number of small adjustments to the positions of cadastral boundaries, and prefer that point positions be held. Thus the update process in use in Queensland is that incoming survey information is adjusted to fit the existing (probably lower accuracy) points. Only when a certain number of new surveys are available is a general adjustment of a region carried out.

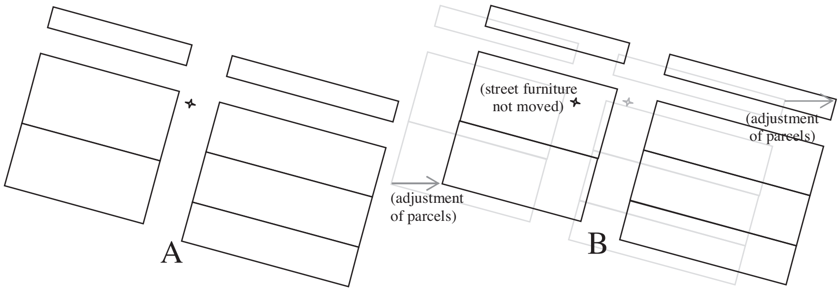

Fig. 1. (A) Positioning street furniture in relation to cadastre. (B) Where the positioning of the parcels is adjusted due to an improved survey, the position of the street furniture must also be changed to maintain relativity.

Hard boundaries in a cadastre

Almost uniquely in spatial data, cadastre is characterised by a preponderance of crisp boundaries. By contrast, most spatial features (at a large enough scale) have soft boundary definitions, although they may be represented by hard-edged database objects (Frank, 1995). That is to say, that while for example, real world land use features have soft edges (where is the boundary of scrubland?); it is possible to define the boundary of the land owned by a person to a hard edge. This is not to say that the position of a boundary is known exactly to the millimetre, but the aim of the cadastre is to divide the land surface (and the space above and below it) so that there is no ambiguity about the physical extent of the ownership or other rights that exist.

Unlike most spatial features, there is no "real world object" that corresponds to the boundary of a spatial unit. There may be a fence line between two properties, but that is not necessarily the true definition of the boundary between them. Nevertheless, a line is defined, for which all points on one side to belong to party A, and points on the other to belong to party B. This line has zero width. That is to say, most boundary representations in a DCDB are not representations of real world objects, but are representations of a legal definition. The "art" of surveying is to use all the available evidence to determine where on the ground this legally defined boundary falls.

Soft boundaries in a cadastre

Although most cadastral boundaries are crisply defined by straight line segments, or sometimes circular arcs or other mathematically calculable shapes, there are some that are soft. An example from the Queensland cadastre is the ambulatory boundary based on a watercourse bank (Brown, 1980, p. 142).

In the example of Fig. 2, Lots 1 and 2 are defined with sharp boundaries on three sides (each), but on the remaining side they are defined by the current position of the bank of Mary Smokes Creek. If we were to visit the property today, the definition of the boundary would be today's position of the creek, and not the position as marked on the plan of survey. This is subject to the proviso that any movement of the creek bank has been "gradual and imperceptible" (Queensland-Government, 2003, p. 44). In this context, it is important to remember that the definition of the boundary is the physical location of the river bank, and the linework stored in the DCDB is an approximation to the boundary at a point of time. In the presence of ambulatory boundaries, it may not be possible to determine an operation such as "find all parcels within region" rigorously, even if the region is sharply defined.

Other forms of soft boundaries in a DCDB occur in secondary interests, particularly if the precepts of Cadastre 2014 are followed – in particular statement 1 "Cadastre 2014 will show the complete legal situation of land, including public rights and restrictions!" (Kaufman and Steudler, 1998, p. 15).

Many cases occur here:

- *No boundaries defined*: In this case, the area or volume of a spatial unit is defined, but its position is only defined within a base spatial unit. An example in Queensland is the "road reservation area", where a certain area of a (usually large) property is reserved for future road building, but at the time of registration, the actual location of the road has not been decided.
- *Boundaries defined texturally*: e.g. within a nominated catchment area, certain land use is not permitted.
- *Boundaries defined by a physical object with a non-sharp boundary*: Recall from Frank (1995) that most natural objects have a soft boundary at some scale of representation. For example, within a region of old-growth forest, certain land use is not permitted.
- *Boundaries defined by reference to an object or location based on a different reference system*: For example, building height restrictions imposed by proximity to an airport. (Note that this could be a 3D restriction).
- *Boundaries defined but not recorded*: For example, it may be known that a property is affected by a network infrastructure, but the DCDB does not record the location of that infrastructure.
- *Extent of region defined by raster image*: For example, certain land cover may impose restrictions on development.

Mixing hard and soft boundaries

The LADM allows for the geographical extents of all interests to be defined as spatial units, so that for example, a soft-edged restriction as described above could be considered to be a non-base spatial unit overlaying the base cadastre. Since multiple levels of encoding are permissible, there is no problem mixing parcels based on hard and soft boundaries in the same database. This is not the whole answer of course; the more interesting question is how to process the mixture of boundary types.

As mentioned above, a request for polygon overlay operations, in the presence of soft edged regions cannot be expected to give a rigorous answer. The methods of fuzzy logic and fuzzy geometry (Dilo, 2006) will be needed. It will thus be necessary for each soft edge to be represented by a nominal location which is attributed with a measure of its accuracy. The same principle applies when cadastral data is overlaid with land use data. Unless provision is made for the mixing of boundary types, misleading results are obtained.

Other land use data

Many other land use and administrative regions need to be correlated with property interests, which are not under the direct control of the cadastral jurisdiction. For example, old-growth forest, noise limitation zones, buffer zones around dangerous industries.

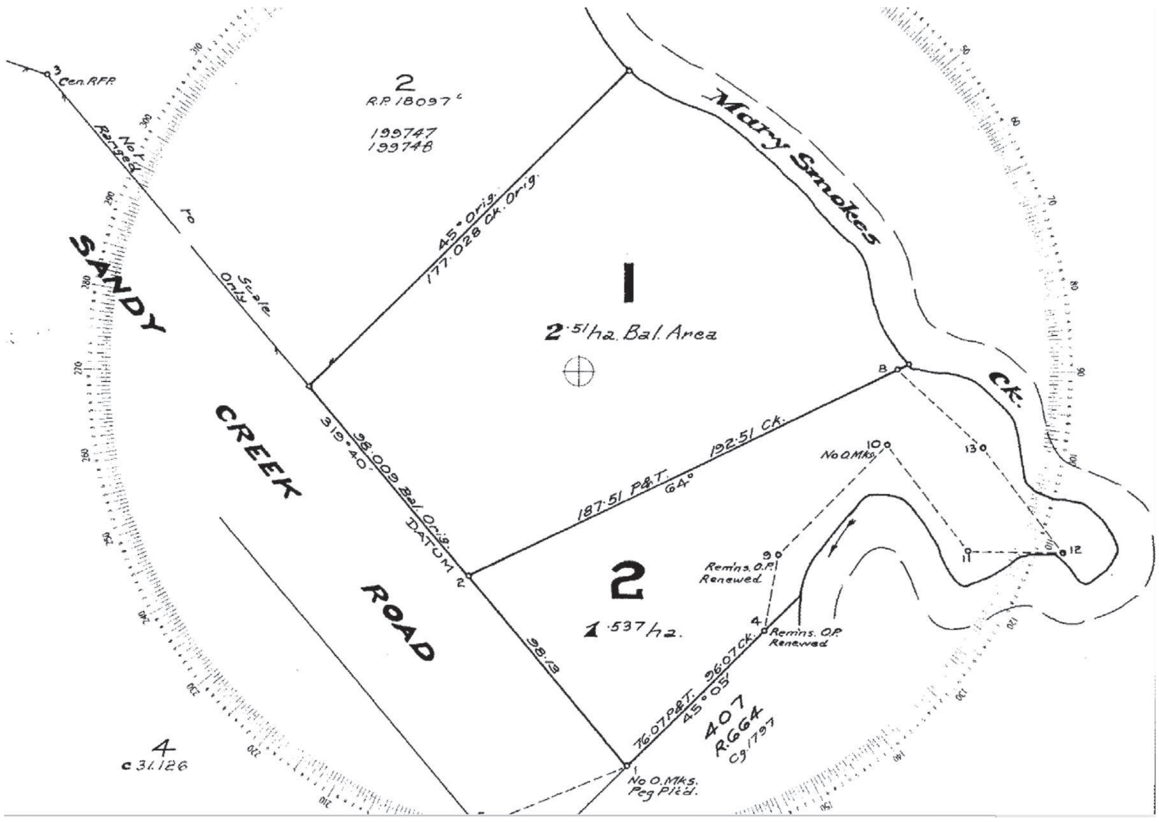

Fig. 2. Surveyed spatial units with ambulatory boundaries.

Typically, these regions will have been sourced independently from the cadastre, and will not be in close positional agreement. These regions have traditionally been viewed as 2D objects, but many 3D regions apply – such as extents of mining.

An important secondary use of the cadastral database is in the generation of land-use mapping, whereby a number of known attributes of cadastral parcels, such as zoning and land ownership are used to determine the usage of individual parcels. These parcels are then aggregated to give a picture of land-use across the jurisdiction (Morse-McNabb, 2011).

History in a developing DCDB

History is an important part of any cadastral database. The apparently simple process of archiving a copy of the database regularly (say once a year) has proved to be extremely difficult in practice. The archived data is in the format of the day, and usually in a structure that requires special (soon obsolete) software to read it (Sweetkind-Singer et al., 2006; Janée et al., 2008). McGarva et al. (2009) recommend to consider "keeping archival data in live access systems". It is vital that if an archiving strategy is adopted, the archived copies must be recognised as "live data", and any redevelopment of the DCDB must address the conversion of the history into a format compatible with the current data.

The ideal form of history in a DCDB would provide functionality to show a user today what the cadastre was like at any point in the past. That is to say changes in the cadastre should be recorded in "valid time" (Snodgrass, 1996; Snodgrass et al., 1998), but further, as errors are detected in current data, the corrections and adjustments should also be applied to the historic record. For example, Fig. 3 shows an excerpt from the Queensland Cadastre in 2001 and as it is now. Observing the result of superimposing them (Fig. 4) it can be seen that some parcel boundaries have been corrected (highlighted "D"). The actual parcels have not changed, so that the historic record as at 2001 should ideally have been corrected. In fact, of course, any other records between 2001 and now should also have been corrected.

It must be recognised that any such corrections to history would be time consuming and hard to cost-justify, however it is commonly accepted that history of "The cadastre as we knew it" is still a valuable resource. Thus the Queensland Cadastre and the Netherlands Kadaster have both used variants of what is now known as the "Versioned Object" pattern of ISO 19152 (van Oosterom, 1997). This is a largely automatic process whereby the details of the cadastre are kept in the live database both before and after any change. The historic records are not updated. In Queensland, an event driven approach is used which automatically records the metadata of the change (Thompson, 2003).

Advantages of the approach:

- It is largely automatic.
- It shows the status of the DCDB as at any point in time (since initial data capture).
- It does not appreciably increase the complexity of the database schema or access SQL.
- It automatically records metadata of the update (who, when and why the update was done) against the affected objects.

Disadvantages:

- It must be recognised that it records history of the database, not of the real world.
- Older historic items may be of lower standard of accuracy or consistency than the current data.

Fig. 3. (A) Queensland Cadastre in 2001. (B) Queensland Cadastre now.

For example, in Queensland, the DCDB from the 1990s contains adjacent spatial units with mismatched boundaries. This was a known problem with the data at that time which has since been corrected, but persists in the historic record. Thus the levels of encoding and accuracy may be non-homogeneous in time as well as in space. Figs. 3 and 4 illustrate the differences between corrections and new data.

As can be seen in Fig. 4, there are additional parcels (A), a change in the definition of the river boundary (B and C), and an adjustment of parcel boundaries (D). The parcels in darker shading on Fig. 3B and 4 are strata parcels and volumetric parcels which had not been captured in 2001. It is not easy to determine if the recorded change in the river's edge (E) has been the result of improving the measurement, or whether it is due to the erosion and accretion of the natural boundary.

Lemmen (2012, p. 145) notes that "all kind of quality improvements can be seen as transactions: 'improve' geometric quality e.g. from point based to polygon based". The corollary to this is that there can be a difference in geometry encoding between the historic record and the current data. To illustrate this issue, consider a spatial unit in a hypothetical database:

> At time t_0 it is created by subdivision of a large land parcel. At this time the database is encoded at the line-based level (level 3). Although the encoding looks correct, there are some digitising errors in the data – small knots and gaps.

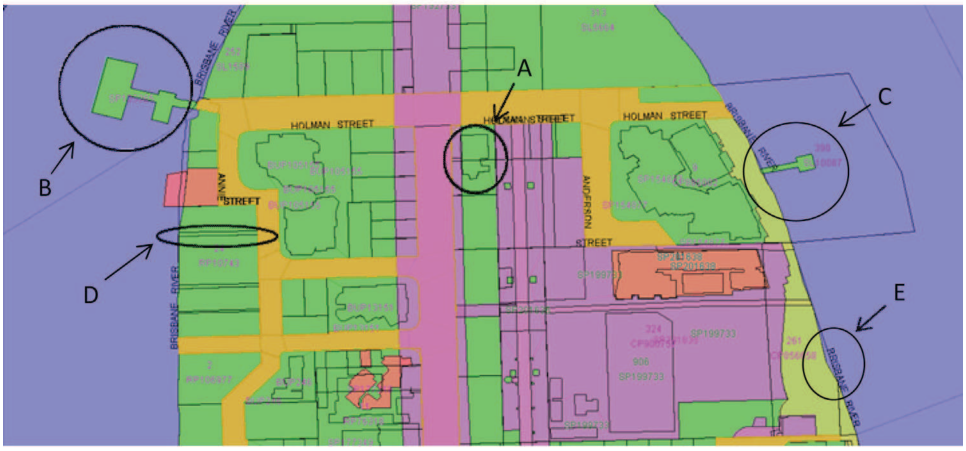

Fig. 4. Detail of 2001 superimposed over 2013. Note the change in definition of the river boundary.

At time t_1, the spatial unit is truncated slightly to widen a road. In the process, some small errors are removed.

At time t_2, there is a general improvement of the accuracy of the database, which causes a small movement in all corners of the spatial unit.

At time t_3, this spatial unit, in conjunction with nearby spatial units, is topologically cleaned, and upgraded to polygon-based encoding (level 4).

At time t_4, this spatial unit is developed into a volumetric parcel, and becomes the "base parcel", with a number of 3D spatial units over it.

The critical issue here is that in the progressive improvement of the database, each of the actions at times t_0 to t_4 is seen as a transaction, which preserves a historical record of the spatial unit as it was prior to each event.

A possible database schema

The requirements to mix data of different accuracy and encoding level argue for a schema which can simultaneously accommodate all levels of encoding. This schema should be designed with consideration of the LADM, be capable of delivering data to the specifications of the LADM, and be capable of accepting such data. The internal structure of the database may differ in some respects from the LADM model, but this should be limited. The nomenclature used in the LADM should be used as far as possible. Some possible areas of difference could be:

1. Additional classes and attributes as per the local profile.
2. Additional linkages and redundancies for performance reasons.

Lazy cleansing of data

The traditional approach to database construction is that the data are validated before being allowed into the database and where possible automatic correction is used to remove small errors and inconsistencies. This is in conflict with the argument that data in a database should be, as far as possible, original, with the minimum of derived or manipulated data being stored. If data is "cleansed", with the removal of small errors as described above, some information is lost. Original point positions are moved, angles are changed, the number of points is changed, etc. Also, in some cases, it is possible to introduce errors – even accidentally removing correct small parcels.

The suggestion here is that the "glitches" be detected with a warning given, but nevertheless loaded into the database. Provided the software is robust, the presence of such issues need not be a problem, and they can be removed as convenient. It is no more difficult to clean the data within the analysis software than it is at load time. It also quite often happens that there is no need for the data to be cleaned. In mapping applications, the presence of small overlaps, knots, etc. will not be noticed.

As an example, in Fig. 5, if an attempt is made to clean the data automatically on entry to the database, it will be rejected because the arrowed parcel E will be lost or corrupted. If, however, a request is made to locate all parcels that overlap the dashed rectangular window, a robust algorithm should be capable of locating the parcels 172, 172, 181, 182, and E. Likewise, calculation of the area of parcel 181 should be possible at an accuracy comparable to the accuracy of the data whether the wedge parcel is included or not. (Parcel E in Fig. 5 is based on a real easement parcel in the Queensland Cadastre, which is 15 mm at the widest – see Fig. 6).

Metadata

The database must also have an easily accessible metadata repository which automatically records and displays the level of encoding and the accuracy (and currency) of the cadastre in any area and any era of interest. It must be able to query and display the cadastre at any level of encoding up to the limit set by the data in any area and era of interest. (e.g. if a user wants line-based encoding where the data is polygon based, the conversion should occur automatically "on the fly").

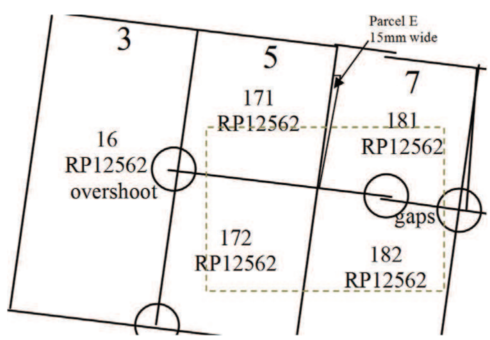

Fig. 5. The arrowed small parcel is similar in size to the gaps and overlaps in the data. Any automated cleansing will cause it to be lost or corrupted.

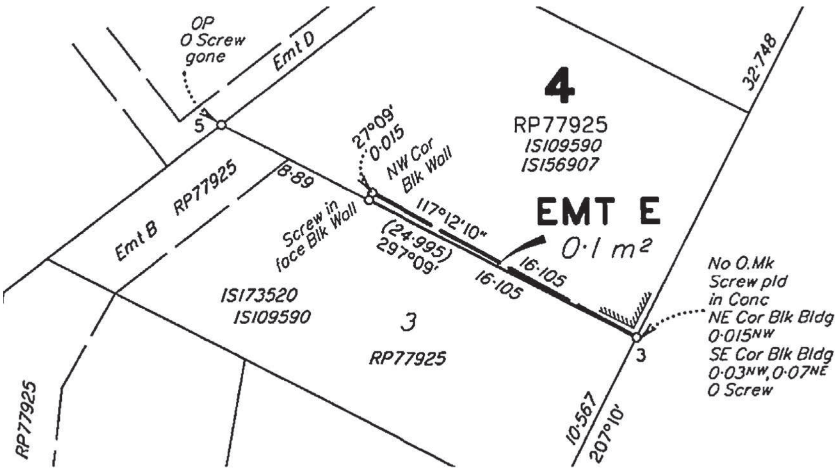

Fig. 6. Cadastral parcel which is 15 mm wide at widest point (From the Queensland Cadastre).

Generic schema

Fig. 7 shows a possible database schema, based on a pilot project attempted in Queensland.

Notes on Fig. 7:

1. All relationships are temporal – that is to say, the create/destroy time stamps must be respected in following any relationship links.
2. All identifiers are persistent – that is to say, when a new representation of an object is created in the process of an update, the identifier remains the same.
3. The classes “Boundary”, “Boundary3D” and “Corner” are associations, to resolve the many-to-many links in the LADM definition.
4. Where a LA_BoundaryFaceString is shared by one or more LA_SpatialUnit(s), there will be a Boundary object for each spatial unit.
5. Where a LA_BoundaryFace is shared by one or more LA_SpatialUnit(s), there will be a Boundary3D object for each spatial unit.
6. A LA_SpatialUnit may be bounded by a combination of LA_BoundaryFaceString(s) and LA_BoundaryFace(s).
7. Following the Queensland conventions, a LA_Point is considered to be a 2D point. A Corner is a 3D point along the vertical line defined by the LA_Point. It is identified by an alphabetic suffix, and an elevation. (e.g. Corners 2a and 2b are vertically one above the other at the location given by LA_Point 2).
8. RRR details and LA_BAUnit information has not been included in the model at present.
9. LA_Point has only been shown in simplified form.
10. The metadata requirements have not yet been included.

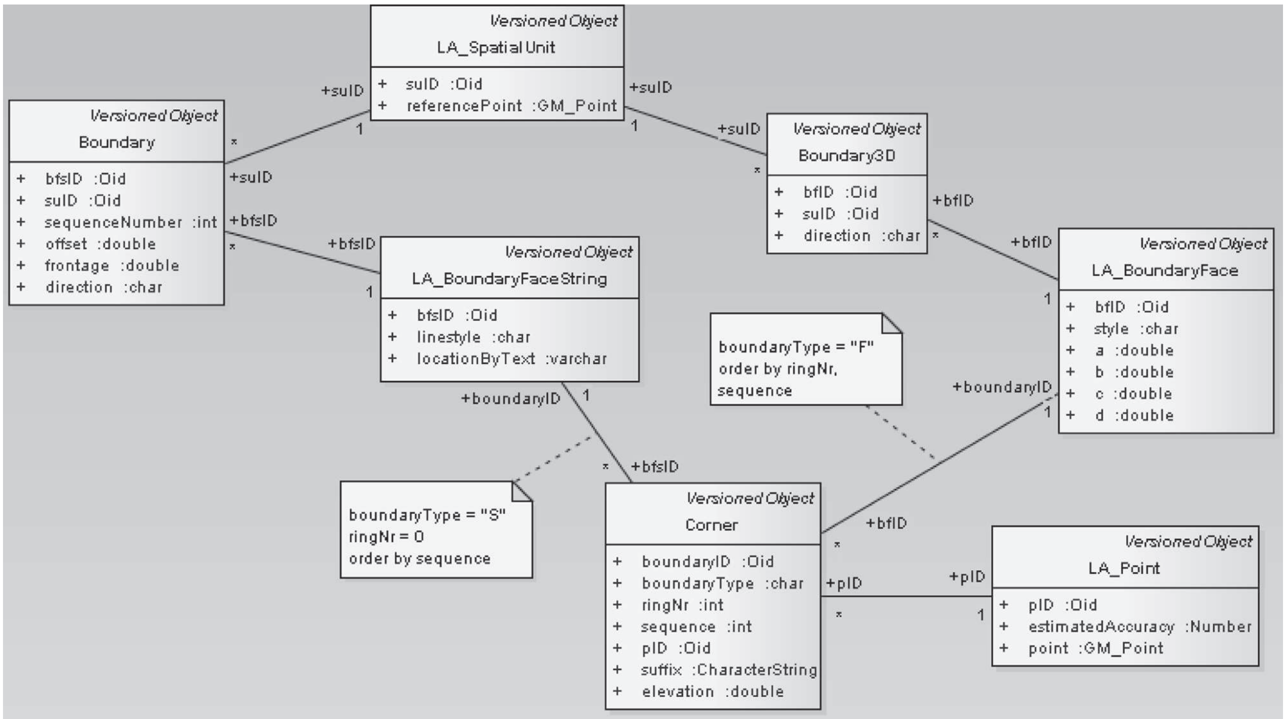

Fig. 7. Suggested data model, based on proof of concept database piloted using the Queensland DCDB data.

Fig. 8. Point-based encoding against a geo-referenced image (mock-up).

In the following text, abbreviated forms of the classes will be used where there is no likelihood of confusion – spatial-unit, face-string, face and point.

Text-based encoding: Each spatial-unit can be connected via boundary records to one or more face-strings which carry the text description. Ideally these would be in anti-clockwise order around the spatial-unit.

Point-based encoding: A database using only the point-based encoding (Profile E1 in ISO 19152) (ISO-TC211, 2012) can be used to place identifying text against a geo-referenced photo (see Fig. 8). This can be used as a very basic property map, as the basis of a property location system, and allowing searching by identifying attributes (e.g. property identifier, address, etc.). In fact, at any level of encoding point-based or better (level ≥ 2), such a product is possible.

The referencePoint in the spatial-unit records the location of the spatial unit's labelling point.

Line-based encoding: The cadastre is represented by a collection of face-strings defining the linework and spatial-units defining the text labels. Boundary objects are not present. The face-strings are connected via corner records to points. As in the point-based encoding, the labelling points of each spatial unit are indicated by referencePoint.

In Fig. 9, different face strings have been given different line fonts. There is no connection between the linework and the text. There is also no guarantee that edges do not lie over one another and there may be cases of multiple points at the same physical location. There are well-known parcelling routines that can detect and correct gaps, overlaps and mismatches. While it is easy for a person to recognise an easement "E" as a non-base parcel, it may be difficult for a parcelling routine to do so, especially in the case of multiple overlapping easements.

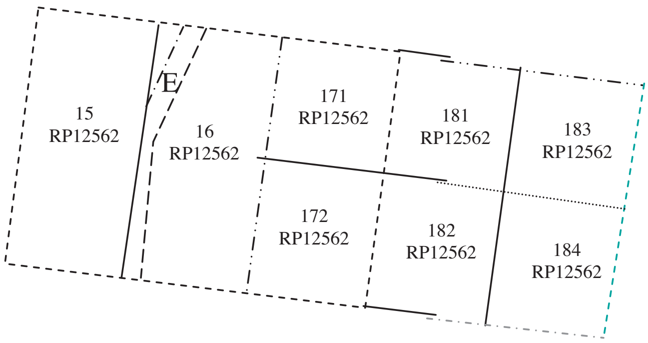

Fig. 9. Line based encoding.

Polygon-based encoding: Here, each spatial unit is represented as a polygon (so that the dividing line between two parcels will be recorded twice). In the simplest case, each polygon is stored as a closed cycle LA_BoundaryFaceString. e.g. in Fig. 10 lot 171 would be represented by face-string,[1] (*abce*); Lot 16 by (*fkji*). Note that line segment *f* in boundary of lot 16 does not need to be broken to match segments *b* and *h*. Note also the sliver between *e* and g. This is not to say that constraints cannot be built into the database software to detect and even correct these issues, but the database does not preclude them. In this case an easement is simply a polygon "on top" of the base parcel polygon e.g. easement E represented by (*nmrpoq*). If there is need to record attributes of the lines (e.g. that a line is an

[1] Here, straight line segments are indicated by a lower case italic letter – *a*, while a face string by a series of lower case letters in parentheses thus (*ab*) indicates the face string made by concatenating line segments *a* and *b*.

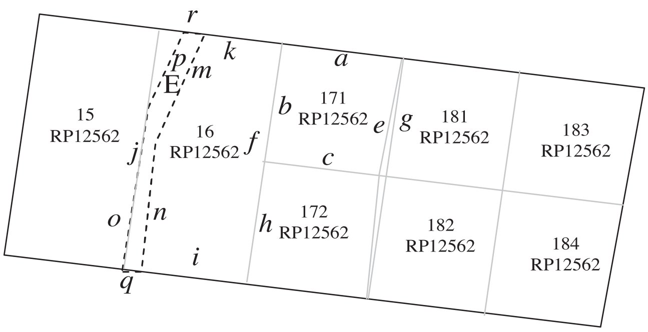

Fig. 10. Polygon based encoding.

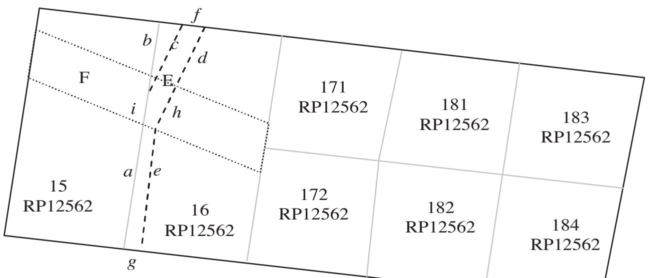

Fig. 11. Overlaying easements.

ambulatory natural boundary), the polygon may be represented by multiple line-string objects. For example, lot 171 could be stored as line-strings (*bce*) and (*a*), with (*a*) carrying the attribute "road boundary".

Topology-based encoding: Here the face-strings are stored once only, and linked to the spatial-unit(s) on the left, and those on the right. In Fig. 11, a face-string (*ehd*) (in that direction) will be linked to easement E via a Boundary record with direction "left". (*a*) will have lot 15 on the left, and lot 16 and easement E on the right. Where easements or other secondary interests overlay, various encoding strategies are possible (see Fig. 11): the most parsimonious of these being:

1. Face string (*i*) has lot 15 on left; easement E and lot 16 on right (no linkage to easement F).
2. Face string (*h*) has easement E on left; nothing on right (no linkage to lot 16 or easement F).

Where easements or other secondary interests overlay, two alternative strategies are possible:

Maximal encoding:

1. (*i*) has easement F and lot 15 on left; easement E, F and lot 16 on right.
2. (*h*) has easement E and F and lot 16 on left; easement F and lot 16 on right

(i.e. if a line goes through a parcel, that parcel is linked to both sides).

Or always forcing base parcels into the links, but not forcing non-base parcels:

1. (*i*) has lot 15 on left; easement E and lot 16 on right.
2. (*h*) has easement E and lot 16 on left; and lot 16 on right

(i.e. lot 16 is forced into both links of face string (*h*), but easement F is not)

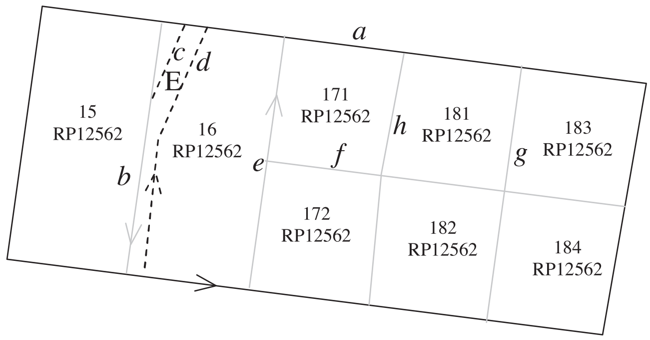

Fig. 12. Generic encoding – note that in contrast with earlier figures, lower case letters are used to label face-strings rather than individual line segments. For example, '*a*' refers to the outer (road) boundary starting and ending at the top left corner.

Topological breakdown

Encoding levels 3–5 can exhibit breakdown of the topology. At level 3, this can be the omission of line(s) causing adjacent spatial units to be combined, leading to more than one centroid in a parcel; or an extra line creating a spurious parcel. At level 4, the boundaries of adjacent spatial units may mismatch, leading to gaps or overlaps of neighbouring parcels. At level 5, the encoding of the left or right parcel(s) on a line may be omitted – leading to an unclosed boundary on a spatial unit – particularly a non-base parcel.

A parcel encoded at level 1 may exhibit an invalid topological relation with another (for example – "parcel A is bounded to the north by the Mary River" and "parcel B is within parcel A and is north of the Mary River", however any detection of this kind of error is beyond the scope of this work). Level 2 encoding cannot exhibit detectable topological breakdown apart from the possibility of two labelling points coinciding.

Generic encoding

The schema shown in Fig. 7 allows, in effect, a line-based encoding with additional topological linkages. In this approach, lot 16, as shown in Fig. 12, would be linked (via boundary records) to face-strings '*b*', '*a*', '*e*', and '*a*' (a second time). Easement E would be linked to face strings '*b*', '*a*', '*d*', '*a*', '*c*', and lot 171 by '$\bar{e}$', '*f*', '*h*' and '*a*'. (where '$\bar{e}$' is face string '*e*' with a direction of "right"). The boundary attributes "offset" and "frontage" indicate the part of the face-string that makes up the boundary of the spatial unit.

On entry to the database, or when updates are applied to improve the data, validation routines are applied to the incoming "spaghetti" data. If it is sufficiently topologically clean, the linkages between the face strings and the spatial units are generated and stored. If not, the region is marked as "line based spatial units only". Where a topological clean operation is possible at a given acceptable tolerance level, data are not adjusted, but the acceptable tolerance value is recorded.

3D spatial units

The LADM permits the mixing of 2D and 3D spatial units. As discussed by Thompson (2013), when the LADM approach is taken to mixing 2D and 3D parcels, the parcels fall into three broad categories:

1. *The base parcels*: These form a complete non-overlapping coverage in 3D (although some parcels may be encoded in 2D).

2. *Secondary interest parcels*: these overlap base parcels (in 2D and 3D), but do not subtract from the area or volume of the base parcel(s). For example, easements.
3. *Excision parcels*: Commonly network or other 3D parcels are defined within what would otherwise be considered to be 2D parcels. These might be stored without their volumes being subtracted from the base parcel(s), with the actual remnant 3D balance parcel only being calculated when needed.

In a progressively developed 3D database, mixtures of encoding will co-exist in the database. This data model can accommodate many combinations. For example:

1. The ground plan of an apartment defined by polygon encoding, and the *z*-component as a text encoding (e.g. "On Level 5").
2. As in Queensland, building unit lots defined by text encoding (within a base parcel which is represented as a polygon).
3. Horizontal subdivision of a building defined by a topological encoding of face strings to define the *x*/*y* extents; horizontal faces used to define the *z* extent, and defined by elevation only.

In Table 1, various combinations of encoding are shown. Clearly where a fully general 3D object is being represented, the *x*/*y* encoding and the *z* encoding must be the same. However, there are many cases in the cadastral domain where the form of the objects is restricted. A combination of encodings is only acceptable if an automatic conversion to a homogeneous form is possible.

"Y" indicates that the encodings are compatible, "N" that they are not, "Y^1" takes the form of a definition of the floor plan of the spatial unit, with a point to indicate the approximate elevation, without specifying the vertical extent.
"Y^2" line encoding in 3D means that space is divided by a combination of face strings and faces, with no attempt to connect them with the spatial unit labelling points.
"Y^3" This is means that 2D space is broken up by face strings, but that certain spatial units have an upper or lower limit imposed on them by horizontal or nearly horizontal faces.
"Y^4" Here, the ground plan is defined as polygons, but the horizontal dividing faces are shared.

Mixtures of encoding, accuracy, datum and dimensionality

Encoding

Any two spatial features, whether spatial units from the DCDB, or more generic features from other sources, can always be compared if they can be converted to a compatible form. The nature of the possible comparison will depend on the most suitable common form that is available. For example, a polygon encoded feature can be overlayed with a line based feature (by converting them both to polygon or both to line based form). Thus the full set of spatial predicates and functions can be supported. By contrast, if a point-based feature is compared with a polygon based feature, only a limited set of functions is possible (such as "certainly overlaps", "possibly overlaps", "approximate distance" etc.).

If data are initially captured using line-based encoding, and later upgraded to polygon or topology, using the "lazy cleansing" approach mentioned above, approximate polygons may be generated, allowing the full range of functionality – albeit with an "approximation" rider.

Accuracy

Where features come from different sources (or from different historic eras in the same database), there will be differences in accuracy. There has been considerable work done on mixing of accuracies, and soft/fuzzy region logic (Dilo, 2006).

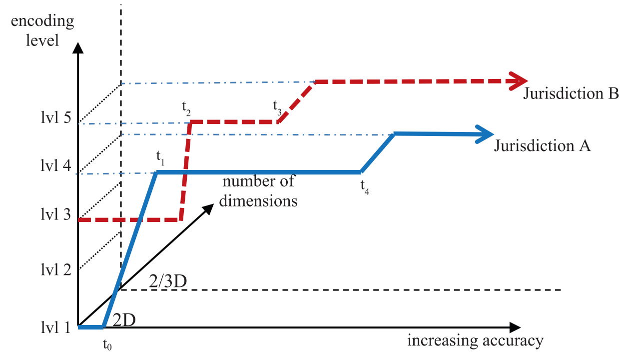

Fig. 13. Possible progressions of a Digital Cadastral Data Base.

Datum

In all cases of mixture of data from different sources, every attempt must be made to ensure that all are based on the same datum. This is common for all spatial data manipulations, and is sufficiently well covered provided a common datum for the functions can be chosen, and all incoming features can be converted to that datum.

Dimensionality

The structure of the encoding of 2D parcels within the LADM as collections of face strings means that it is easy to cast a 2D parcel as a 3D object. Thus there is a rigorous way to define spatial operations between mixed 2D and 3D spatial units. All are converted to 3D, and the operation is evaluated.

It is also possible to do an approximate operation (say as a first pass), by converting all to 2D and applying the operation, which may improve responsiveness of the system. For example, to find all parcels (2D or 3D) within a 3D region, it may be faster to "flatten" the region to 2D, do a search for all 2D or flattened 3D objects within the 2D region, and only apply the rigorous "within" test to those that pass the rough test.

Progressive development

The actual path that the development of a cadastre and the cadastral database clearly depends on the requirements of the individual jurisdiction. Fig. 13 shows the progress of two hypothetical cadastral databases. In Jurisdiction A, at time t_0, progressive conversion of a text-based database into polygon-based form begins (level 4, completed at t_1). During the period between t_0 and t_1, some spatial units are in text-based form, some in polygon form. At time t_4, 3D spatial units are added to the database. Jurisdiction B, begins at level 3, at time t_2 a project is undertaken to topologically clean the structure. This allows it to move to topology-based encoding. At time t_3, it then begins accepting 3D spatial units. In both cases, there is a continuing improvement in accuracy and completeness of both databases.

The critical issue is that, using the approach suggested here, there is no need to convert or discard the history. For example at time t_5, the history of spatial units from the era before t_2 is still available to Jurisdiction B, albeit in level 3 encoding.

Methodology

Based on the schema shown in Fig. 7, a proof of concept database was created, and loaded with the full set of Queensland Cadastral parcels (Table 2).

Table 1
Mixed x/y encoding vs. z encoding from (Thompson, 2013).

3D	2D				
	Text based x/y	Point basedx/y	Line based x/y	Polygon based x/y	Topology based x/y
Text z	Y	Y	Y	Y	Y
Point z	N	Y	Y[1]	Y[1]	Y[1]
Line z	N	N	Y[2]	N	N
Polygon z	N	N	Y[3]	Y	Y
Topology z	N	N	Y[3]	Y[4]	Y

Table 2
Proof of concept database sizes.

Parcels	2,990,794
Parcel corners	19,594,607
LA_points	5,906,777
Face strings	3,046,748
Faces	299
Boundaries	12,221,150

This was not used for formal timing tests, because no significant tuning of access techniques had been done, but indicative speeds were recorded. The major reason for loading the data into this form was to determine the relative cardinalities of the various tables. For example, it can be seen that an average of 3.3 parcels meet at any point (19,594,607 corners meeting at 5,906,777 points). A critical finding was that the number of face strings is approximately equal to the number of parcels. This was not known beforehand, and is a positive finding because it implies a balance in the retrieval speed of parcels and linework. Note that the “LA_BoundaryFaces” table was only loaded with a small number of hand-encoded 3D parcels, so 3D timings were not attempted.

Variations on data model

All of the testing was done assuming an encoding level of 3. Similar or better results could be expected for higher levels of encoding. The major variation to the data model proposed in Fig. 7 was the introduction of a redundant geometry column (GM_Curve) into the LA_BoundaryFaceString. This was not done for speed of access, but rather to avoid the possibility of missing a face string in a “fill window” operation. The retrieval strategies used were:

1. *Web Map Server*: simply retrieve all the parcels and face strings within the window, using the usual PostGIS spatial indexing. The results were as expected – timings were very fast, and retrieval speed was in linear proportion to the area of the window. This would be acceptable as a commercial service.
2. *One or more Parcel(s) and their edges*: the parcel(s) is retrieved by identifier, and the face strings linked via boundary records are retrieved. At this level of encoding, the linework still needs to be parcelled to determine which parts of the face strings make up the parcel boundary.
3. *Parcels in a window, with their edges*: all parcels and face strings in a window are retrieved (as in 1. above), and parcelled to produce complete parcels. Note that this is not sufficient as a Web Feature Service as it stands, because there may be empty spaces within the window where a parcel’s labelling point is off-window.

Table 4
Speed in milliseconds of retrieval of parcel(s) and edges.

	No. of spatial units	No. of corners	Extraction time first run	Extraction time second run
1	1	150	94	0
2	1	48	109	16
3	1	6	32	0
4	1	5	16	15
5	4	231	62	15
6	1	42	62	0
7	10	207	94	47
8	3	158	79	15
9	1	9	109	16
10	1	8	78	16

Speed issues

As mentioned above, timing tests were not rigorous, due to the uncontrolled nature of the systems in use and the lack of optimisation, but some indicative speeds were determined. Each test was run twice to detect any effects of caching in the PostGres server. The test of method 1 consisted of extracting a window of parcels and face strings, and writing a “kml” file for checking using Google Earth (the time to write this file is included in the timing). Table 3 shows some indicative results. The geographic size of the window increases in the later tests, but test 5 is in a rural region with fewer parcels per km^2. As a rule of thumb, the method retrieves approximately 2000 parcels per second (with linework). Indications are that this would support a practical Web Map Server.

The timings of retrieval of individual spatial units and their edges were consistent and good. Indications are that the schema would also support a Web Feature Server (Table 4).

The parcelling routine proved very effective. For example, where a single parcel is being processed as in method 2 (above), it was frequently less than 1 millisecond. This algorithm would be expected to be $O(s\sqrt{s})$, where s is the number of line segments. Applying this to our small number of test results, we obtained $T = s\sqrt{s}/949$ where t is the time in milliseconds. This seems to be valid up to about 40,000 line segments.

Table 3
Timings of window extraction.

	No. of parcels	No. of face-strings	No. of line segments	Window width (km)	Window height (km)	Time (μs) first run	Time (μs) second run
1	80	91	539	0.33	0.34	121	33
2	358	649	3168	0.59	0.77	186	156
3	162	259	2461	9.9	1.1	96	88
4	2782	3210	24,187	9.9	11	1044	1023
5	996	1252	13,482	226	130	541	482

Conclusion

This paper has indicated that a database built on the model of generic encoding is a practical proposition. Such a database would support the LADM, being able to receive data encoded to any level that that standard defines. It would also be able to supply fit-for-purpose data in various levels of encoding at an earlier opportunity than is usual with geospatial database projects. As data quality is improved, so the database matures, increasing the functionality it delivers.

A support for history of the cadastre is included, maintaining an automatic record of the knowledge of the cadastre as it was at the time of recording, based on transaction time and the versioned object pattern. This model ensures that as the database is improved over time the history is not lost.

Further research

The jurisdictions considered in this paper (Queensland and the Netherlands) reflect the background of the author, and a review of this approach with respect to a wider range of jurisdictions would be appropriate; However it is difficult to imagine any cadastral jurisdiction that could not be accommodated by such a model, at least in terms of the geometric layer.

The storing of metadata needs to be fully defined. It must be possible to determine the level of topological purity, accuracy, resolution, etc. available on the basis of geographic region, at a specified epoch, or on specified spatial units, linework or points. The metadata must be active–meaning that it should be computer-readable, and used in the control of access to the data. Also, techniques of fuzzy logic are needed to accommodate soft boundaries.

More complete evaluation is needed of the lazy cleansing options. For example, where the incoming validation has determined that the data is not valid but can be automatically cleansed at a specified tolerance parameter setting, what needs to be recorded to make the subsequent processing of these data more efficient?

The model has been validated by loading data from the Queensland Cadastre. It would be instructive to extend this to another jurisdictional Cadastre.

Acknowledgments

I would like, particularly, to thank Peter van Oosterom for his comments and assistance, and Jaap Zevenbergen and Hendrik Ploeger for detail of the approaches taken by the Netherlands Kadaster.

References

Brown, A.G., 1980. Law Relating to Land Boundaries and Surveying. Association of Consulting Surveyors Queensland.

Diggles, D., 2008. Mapping Queensland: History of Cadastral and Topographic Mapping in Queensland since 1859. Mapping Sciences Institute Australia, Queensland Division, Brisbane.

Dilo, A., 2006. (PhD thesis). In: Representation of and reasoning with vagueness in spatial information. Wageningen University.

Effenberg, W.W., Williamson, I.P.,1996. Data flows, standards and incremental cadastral update. In: Spatial Information Research Centre's 8th Colloquium. University of Otago, New Zealand.

Frank, A.U., 1995. The prevalence of objects with sharp boundaries in GIS. In: Burrough, P.A., Frank, A.U. (Eds.), Geographic Objects with Indeterminate Boundaries. Taylor & Francis.

Hay, G., Hall, G.B.,2009. Architecture for an Open Source Land Administration Application. In: FIG Working Week 2009. Eilat, Israel.

ISO-TC211, 2012. Geographic Information – Land Administration Domain Model (LADM) (ISO 19152).

Janée, G., Mathena, J., Frew, J.,2008. A data model and architecture for long-term preservation. In: 8th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL). ACM, Pittsburgh.

Kaufman, J., Steudler, D., 1998. Cadastre 2014 – A Vision for a Future Cadastral System, Retrieved from https://www.fig.net/cadastre2014/translation/c2014-english.pdf (June 2014).

Keenja, E., de Vries, W., Bennett, R., Laarakker, P., 2012. Crowd Sourcing for Land Administration: Perceptions within Netherlands Kadaster. In: FIG Working Week 2012: Knowing to Manage the Territory, Protect the Environment, Evaluate the Cultural Heritage, Rome, Italy.

Lemmen, C., 2012. A Domain Model for Land Administration. NGC, Nederlandse Commissie voor Geodesie, Netherlands Geodetic Commission.

McGarva, G., Morris, S., Janée, G., 2009. Technology watch report: preserving geospatial data. In: Digital Preservation Coalition May 2009, Document 09-01.

Morse-McNabb, E., 2011. The Victorian Land Use Information System (VLUIS): A new method for creating land use data for Victoria, Australia. In: Surveying & Spatial Sciences Biennial Conference 2011, Wellington, New Zealand, pp. 155–168.

Österberg, T., Hawerk, W., Henssen, J., Kaufmann, J., Li, Z.S., Nichols, S., Williamson, I., 1995. FIG Statement on the Cadastre, Retrieved from http://www.fig.net/commission7/reports/cadastre/statement_on_cadastre.html (August 2014).

Priebbenow, R., 1993. Survey Integration – The Future, Seminar and Workshop. ISA Qld and Surveyors Board of Qld.

Queensland-Government, 2003. Surveying and Mapping Infrastructure Act. Queensland-Government.

Snodgrass, R.T., A Road Map of Additions to SQL/Temporal, Report: ANSI-96-013 9th Feb 1996.

Snodgrass, R.T., Böhlen, M.H., Jensen, C.S., Steiner, A., 1998. Transitioning temporal support in TSQL2 to SQL3, vol. 1399., pp. 150–194.

Steudler, D., Törhönen, M.-P., Pieper, G., 2010. FLOSS in Cadastre and Land Registration: Opportunities and Risks.

Sweetkind-Singer, J., Larsgaard, M.L., Erwin, T., 2006. Digital Preservation of Geospatial Data. Library Trends 55 (2), 304–314.

Tarbit, S., Thompson, R.J., 2006. Future Trends for Modern DCDB's, a new Vision for an Existing Infrastructure. In: Combined 5th Trans Tasman Survey Conference and 2nd Queensland Spatial Industry Conference, Cairns, Queensland, Australia.

Thompson, R., 2013. Progressive development of a digital cadastral database. In: Van Oosterom, P., Lemmen, C., Fendel, E. (Eds.), 5th Land Administration Domain Model Workshop. International Federation of Surveyors (FIG), Kuala Lumpur, Malaysia, pp. 361–380.

Thompson, R.J., 2003. Metadata and Timestamping in RIME. In: Spatial Sciences 2003, Canberra.

van Oosterom, P., 1997. Maintaining consistent topology including historical data in a large spatial database. Seattle, WA, Auto Carto 13.