

## Article

# LandXML Encoding of Mixed 2D and 3D Survey Plans with Multi-Level Topology <sup>†</sup>

Rodney James Thompson <sup>1,\*</sup>, Peter van Oosterom <sup>1</sup> and Kean Huat Soon <sup>2</sup>

<sup>1</sup> Department OTB, GIS Technology Section, Delft University of Technology, P.O. Box 5030, 2600 GA Delft, The Netherlands; P.J.M.vanOosterom@tudelft.nl

<sup>2</sup> Singapore Land Authority, 55 Newton Road, #12-01, Revenue House, Singapore 307987, Singapore; soon\_kean\_huat@sla.gov.sg

\* Correspondence: R.J.Thompson@tudelft.nl; Tel.: +31-15-2786950

<sup>†</sup> This paper is extended from the version presented at the 5th International FIG 3D Cadastre Workshop, Athens, Greece, 18–20 October 2016.

Academic Editors: Efi Dimopoulou and Wolfgang Kainz

Received: 31 March 2017; Accepted: 5 June 2017; Published: 12 June 2017

**Abstract:** Cadastral spatial units around the world range from simple 2D parcels to complex 3D collections of spaces, defined at levels of sophistication from textural descriptions to complete, rigorous mathematical descriptions based on measurements and coordinates. The most common spatial unit in a cadastral database is the 2D land parcel—the basic unit subject to cadastral Rights, Restrictions and Responsibilities (RRR). Built on this is a varying complexity of 3D subdivisions and secondary interests. Spatial units may also be subdivided into smaller units, with the remainder being kept as common property for the owners/tenants of the individual units. This has led to the adoption of hierarchical multi-level schemes. In this paper, we explore the encoding of spatial units in a way that highlights their 2D extent and topology, while fully defining their extent in the third dimension. Obviously, topological encoding itself is not new. However, having mixed a 2D and 3D topological structure is rather challenging. Therefore, despite the potential benefits of mixed 2D and 3D topology, it is currently not used in LandXML, one of the main and best documented formats when representing survey data. This paper presents a multi-level topological encoding for the purposes of survey plan representation in LandXML that is simple and efficient in space requirements, including the question of curved surfaces, (partly) unbounded spatial units, and grouping and division of 2D and 3D spatial units. No “off the shelf” software is available for validating newly lodged surveys and we present our prototype for this. It is further suggested that the conceptual model behind this encoding approach can be extend to the database schema itself.

**Keywords:** 3D cadastres; survey plans; LandXML; InfraGML; multi-level topology; geographic information systems; rights; restrictions; responsibilities; spatial data infrastructure; real property

## 1. Introduction

In many jurisdictions, the cadastral survey plan is a critical instrument in the administration of property rights, being the starting point that defines the extent and location of the property. The secondary purpose of such a plan is as a data source for a database (and map) of cadastral information. With the growing trend towards digital submission of cadastral plans, there is a need to maintain the authoritative nature of the plan in the absence of a paper document. It is critical that the definitions of properties are correct and topologically sound, with adjoining properties identified in 2D and 3D.

2D land parcels (3D columns of space) or 3D spatial units may be subdivided into smaller spatial units, with the remainder being kept as common property for the owners/occupiers of the individual

units—for example, townhouses within a 2D land parcel (spatial unit), with the driveways and gardens being held for common use, or a 3D building with both private spaces and common use spaces for the elevators, foyer etc. Multi-level schemes have been used to alleviate this complexity, with base 2D land parcels (3D columns of space) being subdivided into volumetric spatial units, which are in turn further subdivided into individual units.

In another example, a building may be placed on a base parcel, leaving property in common. It can be subdivided into volumetric spatial units for different classes of units (commercial, residential, etc.), leaving common property for entrance, elevators, etc. The volumetric spatial units can then be subdivided into individual units, with common property for the use of these unit owners/occupiers (but not for owners/occupiers of units in other volumes). It should be noted that when a volumetric unit is excised from a 2D parcel, the common property left will be partially unbounded: above and/or below.

The Land Administration Domain Model (LADM) provides for all of these levels of complexity [1,2], and it has been shown by Thompson et al. [3] that a mixed representation allows a relatively simple encoding of the full range of cadastral spatial units. The latter paper, however, does not address the issue of topological encoding of such a mixture of spatial units. The issues involved in encoding a survey plan (as distinct from a cadastral database) include some extra complexity.

This paper explores the practicality of topologically encoding spatial units, initially in LandXML [4], but with a view to also supporting InfraGML [5,6] when it is more mature; by demonstrating a topologically structured conceptual model for the purposes of survey plan representations, addressing the questions of curved surfaces, (partly) unbounded spatial units, and hierarchical grouping/division of 2D and 3D spatial units. The suggested method uses a form of mixed-dimensional topological structuring—sharing boundary definitions between spatial units that are simple and efficient in space requirements. It prevents problems of accidental overlap between spatial units in 3D, while providing a data source for a mixed 2D/3D digital cadastral database that minimizes redundancy and inconsistency. It is suggested that the conceptual model behind this approach can be extended to the cadastral database itself, including the requirement to maintain a historical record of the spatial unit structure (lineage).

### *Methodology*

Several cases of plans of survey have been chosen, and “proof of concept” software written to (1) accept an encoding of the plan in a simplified form, (2) write the spatial units of the plan to a Postgres database, and (3) translate those spatial units into LandXML.

The path of the research has been: (1) the definition of axioms to ensure the validity of 3D spatial units [7], (2) the categorization of spatial units in terms of complexity and relative frequency of occurrence [8], (3) a representation of cadastral data enabling the mixing of 2D and 3D spatial units [3], and (4) the extension of the approach to a topological encoding scheme [9].

In what follows, Section 2 discusses the concept of survey plans and the research on topology for cadastral data. Section 3 describes a selection of 2D and 3D theoretical and real cases. Section 4 describes the conceptual encoding in these cases and demonstrates the actual encoding in LandXML and in future, InfraGML. In Section 5, the findings with respect to a proof-of-concept implementation are analyzed. Section 6 concludes the paper by summarizing the main results and indicating future work.

## **2. Background**

This section first elaborates the purpose and role of the survey plans in the context of land administration (Section 2.1). Next, the importance of topology in a cadastral database is reviewed (Section 2.2).

## 2.1. Survey Plans

Typically, cadastral jurisdictions separate land administration into the act of defining the extent of a piece of land on a “Survey Plan” and the parties involved on a Title document (as in the case of the Torrens Titling System) or Deed document. The “Rights Responsibilities and Restrictions” (RRR) recorded on a title or deed associates it with the land parcel as defined on the plan. Even though the term “Survey Plan” is used, in practice it may not always involve a survey in the conventional sense. Many types of technology are used, particularly where 3D spatial units are involved (e.g., laser scanner), to produce a combination of sketches, tabular data, and measurements that serve the purposes to which a survey plan is put. That is, the survey plan is the source document for the spatial unit.

Traditionally, a survey plan has been a paper document, which was reassuring in that it carried seals and signatures, and was suitable for long-term archive and storage. However, in this form, it was clumsy as a data source, especially when the preparation of these paper documents started to be a computer process. Recently, there has been an effort to switch to digital plans, containing, in structured and semantically enriched manner, the spatial and measurement data [10–13]. This has not changed the fundamental requirements of the survey plan as noted above. In addition, the move to 3D spatial units has led to a much greater complexity of the plans—needing to carry elevation diagrams and/or isometric views to make the geometry comprehensible. In Singapore, New Zealand, and several Australian States, the LandXML format [4] has been chosen to transport digital plans. The essential information carried on a survey plan is typically collected into a database to provide a multiple use cadastral database.

Currently, the Open Geospatial Consortium (OGC) is developing a standard named InfraGML [6], which is intended to replace LandXML [14]. This development is still in progress, but the standard has provision for survey data and land division. It is confidently expected that the techniques used to express this conceptual model in LandXML will carry across in large part to InfraGML, but not sufficient detail is present in the current draft to permit any detailed design at present.

## 2.2. Topology in Cadastral Data

The current way of using LandXML only defines geometry and not topology. For topology, it depends upon the software that reads it. For instance to encode a doughnut parcel, one not only has to define the two polygons but also to specify explicitly the relationship between these two polygons. When software reads in, it will only need to interpret the relationship that has syntactically been defined in the LandXML, even though the two polygons may follow the same parcel orientation.

The explicit encoding of the relationship will limit its use to a small group of software that can understand the relationship. By contrast, if a topology is encoded, say the outer ring of a parcel as counter-clockwise, and the inner ring as clockwise, a wider range of software can support the LandXML as this is the general rule of topology at least for ISO and OGC.

This becomes significantly more important in 3D modeling when perspective becomes significant. Let us imagine a void space (think it as a 3D empty box) contained within a 3D parcel. If one has to define explicitly every topological relationship between one surface to another, this is certainly not elegant and will be computing-power intensive. With topological encoding by specifying the parcel orientation into a certain direction for inner rings and outer rings, the proposed topological approach is a great improvement.

In 3D modeling for strata for instance, there exists typical floors or units within a building (typical floors refer to the exact same shape across different levels, while typical units refer to units of the same shape in different levels). The topological encoding approach will reduce tremendously the need to encode the common surfaces twice (or more), by simply adding an indicator of orientation to the encoding of a face or line.

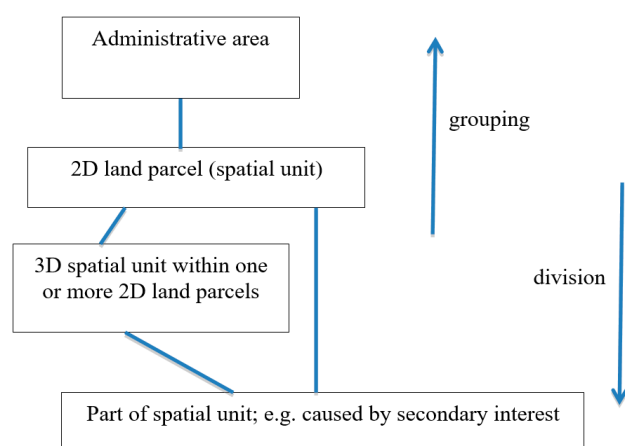
The 2D spatial unit has a special place in a Cadastre. Often there is an identifiable “base layer” of 2D spatial units—interpreted as 3D prisms [15], which comprises a complete, non-overlapping coverage of the area administered by the jurisdiction. With the scarcity and value of land in modern cities, there is a strong trend to subdivision into explicit 3D spatial units. Typically, a 3D spatial unit which is to be associated via RRR with a party (person etc.) will be a closed volume, with a complete

and well defined boundary (shell), but each time a closed volume is defined within a 2D spatial unit, it leaves a 3D “object” (a prism with a cavity). There is no volume to be determined for such a remainder spatial unit as it has an undefined top and bottom.

Alternatively, 3D spatial units may be defined, not by measurements but by the references to walls and floors/ceilings in a building that encloses them (or are planned to enclose them when the building is completed). There may be sketches on the plan of their location within the building, but the sketch is not the definition of the extents of the spatial unit (and may not have any measurements marked). In Queensland and Singapore, these do not have volumes defined, but are defined to have a certain floor area. These are known as Building Format Plans, and the spatial units defined by them as Building Format Units. This form of 3D spatial unit is the most common internationally, and it appears that all jurisdictions that recognize 3D subdivision use this form [11,16–18].

There has been considerable discussion on the subject of topological encoding of cadastral data in 2D over the years [19–21]. One major advantage of the topological approach is the reduction in redundant storage of linework. There are different types of topology—from the simple single layer complete non-overlapping coverage, to the multi-layer with topological connections maintained between levels. In practice, a cadastral database needs to accommodate multiple levels of data—ranging from the simple property spatial unit, aggregated into administrative regions, and subdivided into 3D spatial units and into secondary interests (such as easements). This concept is addressed by the ISO19152 LA\_Level class.

Figure 1 gives a rough schematic of the sub and super-sets of a basic spatial unit. Administrative areas may not consist of an integral number of whole spatial units, and secondary interests may span more than one base spatial unit.



**Figure 1.** A rough “hierarchy” of spatial units.

It has been shown in [22] that such partial hierarchies can be accommodated in 2D, with the decision to use a number of single valued vector map (SVVM) layers, or alternatively more tightly as a multi-valued vector map (MVVM). Similarly, such SVVM or MVVM’s could be defined in 3D. This is a question of balancing consistency against complexity. Integrating multiple levels in an MVVM enables the reuse of boundaries needed at two or more levels (good for consistency), but also causes some geometry fragmentation in other cases.

The question of extending the principle of topological encoding to 3D has been covered in detail, and in several ways, but always with the aim of a true 3D coverage—where all objects are volumetric [23,24]. By contrast, it has been shown that the vast majority of 3D cadastral parcels are relatively simple [8], and savings can be made by using that fact [3].

It is quite common that a complex 3D spatial unit can be comprised of a set of simpler units, either by the process of adding (forming the union of) smaller units, or subtracting inner excised areas. Regardless of what the process is, it raises a question of whether it is preferable to share

common sub-unit definitions as far as the LandXML encoding is concerned. The following sections will address this question specifically.

### 3. Topological Representation of 2D and 3D Spatial Units

In this section, examples of three typical core spatial units is described: 2D Cadastral Spatial Units (Section 3.1), Simple 3D Spatial Units (Section 3.3), and Complex 3D Spatial Units (Section 3.4). Real-world cases of multi-level and hierarchical properties are introduced in Sections 3.5 and 3.6. The scheme of categorization used here is from Thompson et al. [8], and is briefly:

- The 2D Land Parcel;
- The Building Format Unit (where the definition of the parcel is the actual building walls);
- The Polygonal Slice (where the units are described by vertical planes, with height limits);
- The Single-Valued Stepped Slice (where all surfaces are vertical or horizontal, and the unit does not have parts overlying other parts);
- The Multi-Valued Stepped Slice (where all surfaces are vertical or horizontal, but parts may overlie others); and
- General 3D Parcels (with few restrictions beyond validity).

#### 3.1. 2D Cadastral Spatial Unit

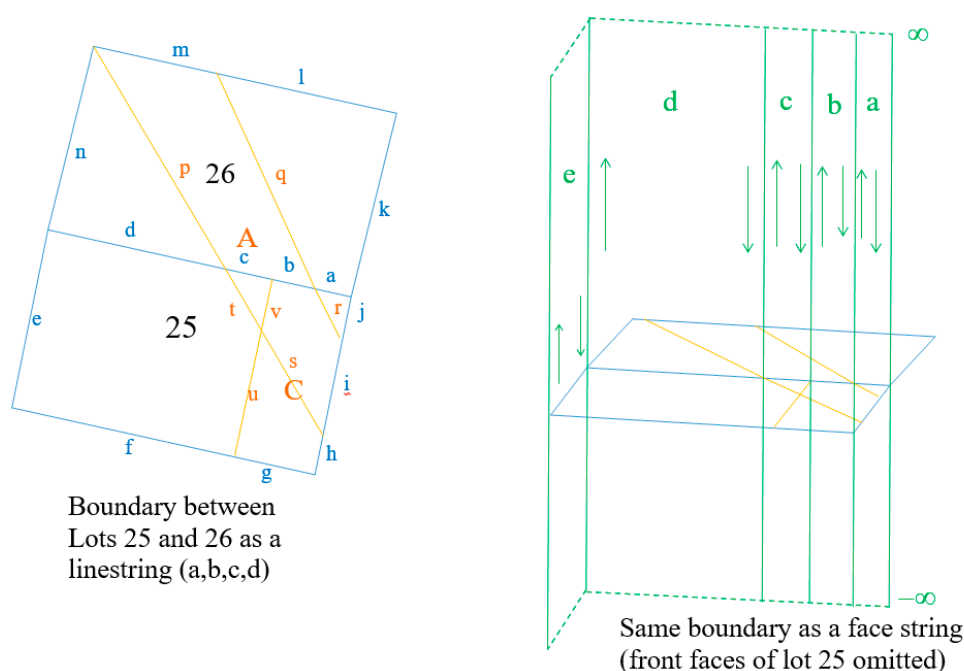
2D topology can be modeled by representing cadastral boundaries as line strings, with encoding for the “left” and “right” base cadastral units [22,25,26]. For example, a “winged edge” structure can be extended to include non-base spatial units by including additional left/right encodings for non-base 2D spatial units (administrative areas, secondary interests, easements, etc.) [22]. This is the approach taken to develop a 2D topologically structured multi-layer Cadastral database such as the Netherlands Kadaster, with left and right references at multiple-levels: parcel, cadastral section, and municipalities [27].

For a mixed 2D/3D cadastral database, the 1D linestrings in 2D space that are used to define the cadastral boundaries are re-interpreted as 2D face strings in 3D space (as defined in the LADM) [1]. This does not change the database representation at all, adding nothing to the storage requirements because the storage of a LA\_BoundaryFaceString is simply as a 1D linestring in 2D space. To illustrate this, let us consider Figure 2 as an example of a multi-valued-vector-map (MVVM) style encoding. In Figure 2,

- (1) Line segment *a* has Lot 25 and Easement C on **left**; Lot 26 on **right**;
  - (2) Line segment *b* has Lot 25, Easement A and C on **left**; Lot 26 and Easement A on **right**;
  - (3) Line segment *c* has Lot 25 and Easement A on **left**; Lot 26 and Easement A on **right**;
  - (4) Line segment *d* has Lot 25 on **left**, Lot 26 on **right**;
  - (5) Line segment *e* has Lot 25 on **left**;
- Therefore, Lot 25 is defined as being on the left of line segments *a,b,c,d,e,f,g,h,i,j* and both sides of *t,s,r,v,u*;
  - Lot 26 is left of *k,l,m,n*, right of *a,b,c,d* and both sides of *q,p*;
  - Easement A is left *i* and *m*, right of *t,p,q,r,s* and both sides of *b,c,v*.

Note that in some implementations, links where the same parcel is on both sides of a line are omitted. This will be the approach taken here.

The 2D spatial units are “converted” into 3D spatial units by replacing each line segment in their boundary by a face running from  $-\infty$  to  $+\infty$ , and passing through the endpoints. The outside of any face is that side from which it appears to be anticlockwise (i.e., from the positive direction of the normal vector), so that the same definitions apply, with the words “on left” replaced by “positive side of” or “+”, and “on right” with “negative side of” or “−”. In this way, Parcel 25 is now the positive side of face string *a,b,c,d,e*, and Parcel 26 is on the negative side of *a,b,c,d*.



**Figure 2.** Interpreting a line string topological boundary as a LA\_BoundaryFaceString. Base Parcels 25 and 26 are burdened by two easements, A and C. A runs diagonally across both base parcels, while C overlays a rectangular area of Parcel 25 only.

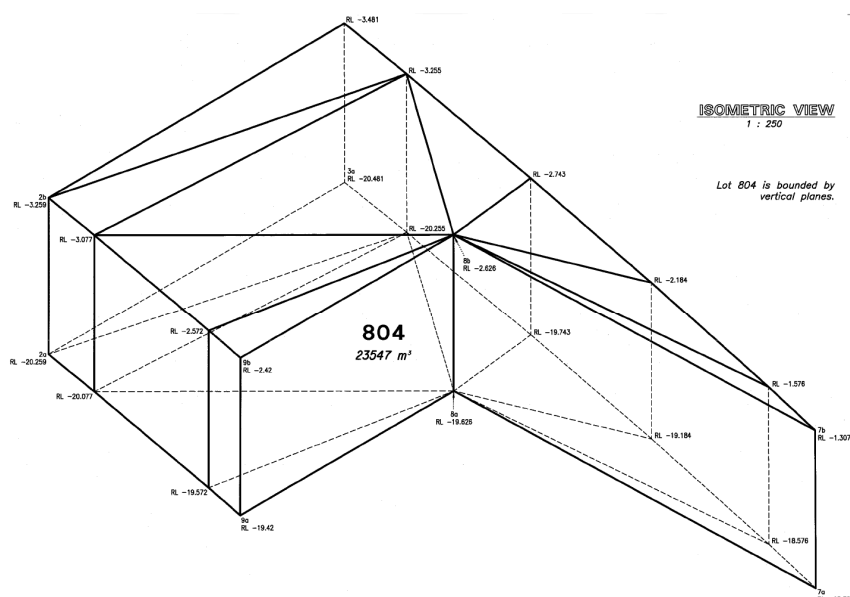
### 3.2. Extension to 3D

The approach as suggested by [3] makes use of some of the specific features of cadastral data, and is suggested by the conventional form of survey plans. When defining a 3D subdivision, the survey is commonly first introduced by a “plan view,” which defines the 2D shape of the subdivision. This is followed by various elevation and isometric views that define the faces that bound the volumes in question.

In brief, 2D spatial units (which correspond to the plan view) are viewed as a column of space, unbounded above and below. The “polygon” that conventionally defines a 2D parcel then is interpreted as the “footprint” of this column. Extending this, a 3D spatial unit is represented by a footprint, which is then restricted by faces above and below the actual parcel. That paper [3] did not address the topology either within individual parcels or between parcels. This paper and [9] use the mixed approach, extended to define a topological structure. Using the language of the LADM, the footprint is defined as a set of “face strings” (LA\_BoundaryFaceString) [2], while the faces use LA\_BoundaryFace definitions.

### 3.3. Simple 3D Spatial Unit

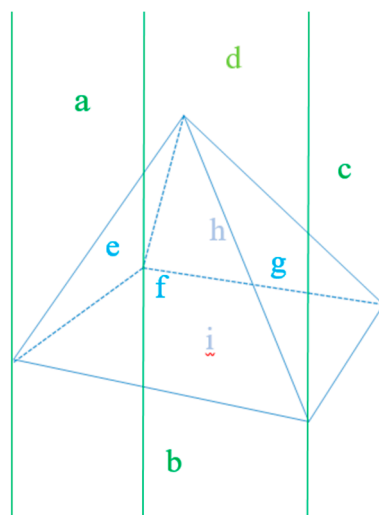
The simplest of the 3D spatial units are the “Polygonal Slice” and the “Above/Below Elevation” spatial units [3]. An example is shown in Figure 3. These are a prism of space with vertical faces, delimited above and/or below by surfaces—usually horizontal. Only slightly more complex, those with a well-defined top and bottom surface, which are not per se exactly planar (hence the triangulation of these surface in the example of Figure 3).



**Figure 3.** A simple 3D (underground) spatial unit with vertical walls corresponding to surface parcel boundaries, but with non-horizontal top and bottom reflecting the slope of the tunnel.

### 3.4. Complex 3D Spatial Units

This is the most generic situation, which does not necessarily define the 3D Spatial Unit as face strings, as the boundaries have arbitrary orientation and might even be curved. Figure 4 illustrates an artificial example of this kind of spatial unit. These Spatial Units are relatively rare, and in some cases, our proposed storage scheme may require more storage space for consistency than would the conventional 3D polyhedron stored as a set of faces. However, having the 1D line string in 2D space is still useful as it can be applied to depict the footprint of the 3D spatial unit on the traditional 2D cadastral map, so it is always included in this representation.



**Figure 4.** Complex spatial unit in the shape of a pyramid.

### 3.5. Multi-Level Case of a Tunnel Below a Building

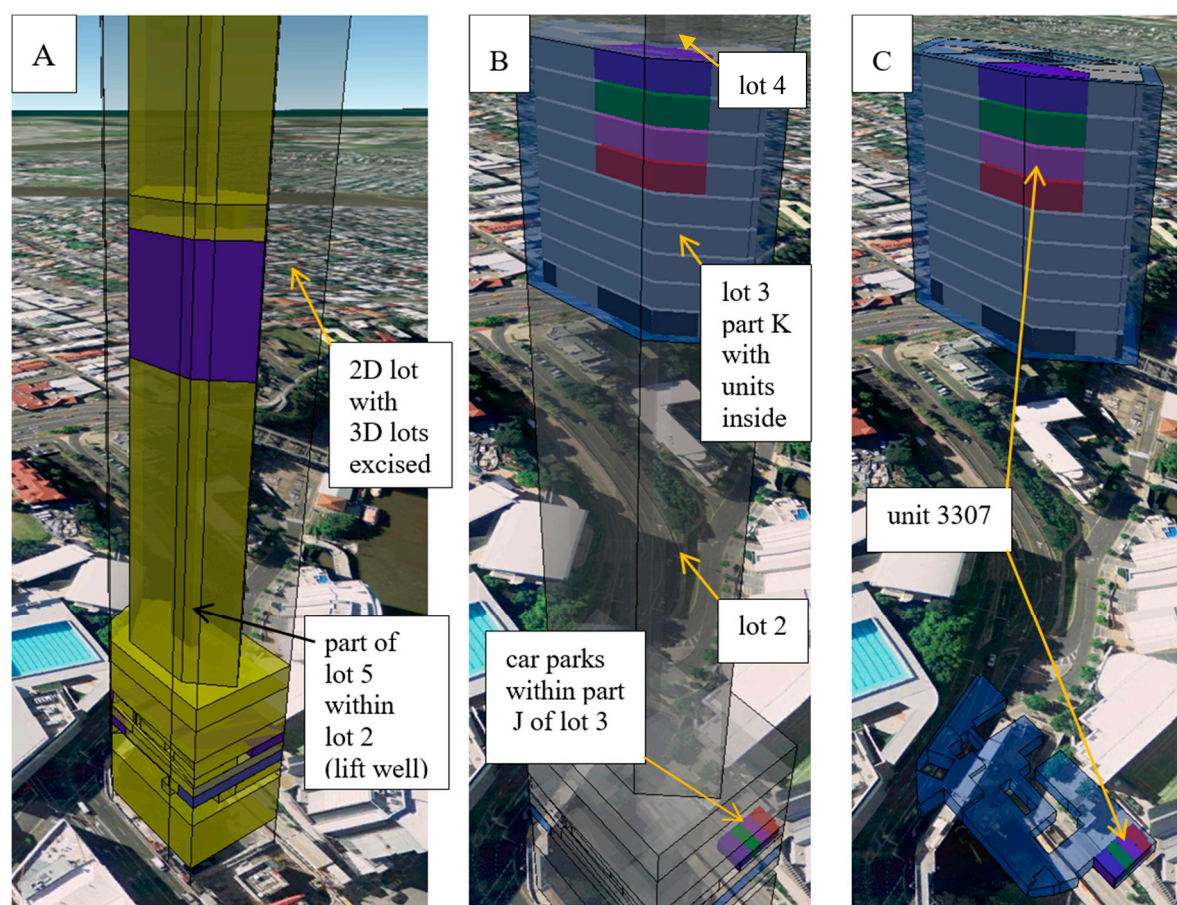
This case consists of a partially unbounded spatial unit with development where a tunnel (3D spatial unit) has been created underneath. At and above ground level is a multi-unit building, while below ground is a road tunnel unit. Thus, it combines the issues of a building format plan, a volumetric excision from a 2D parcel, and a complex spatial unit unbounded above and below. This was described in some detail by [9], with the encoding addressed in Section 4.4.



The subdivision of surface parcels where they are affected by subterranean infrastructure, and the converse breaking of infrastructure at surface parcels is a complex question, which is described in Section 4.1.1 of Stoter and van Oosterom [15]. The case illustrated here (Figure 3) is defined as required by Queensland legislation, where new infrastructure must be legally “broken” at existing surface parcel boundaries, but newer surface boundary changes do not necessarily affect the subsurface parcels (Queensland Government Cadastral Survey Requirements, Section 10.2.4) [28].

### 3.6. Multi-Level Hierarchical Case—A High-Rise Building

This case is a highrise building (Figure 5). It is subdivided into sections, with different uses (residential, commercial etc.) e.g., Lot 3, and these sections are further subdivided into units—e.g., Unit 3301. As a result of this hierarchical subdivision, there are volumes of common property, which are available to all tenants of the building (Lot 5, defined as the remains of original 2D lot after the 3D excisions), but also more restricted volumes such as the remainder of Lot 3, which is only available to owner/occupiers of the units of Lot 3, but not those of Lots 2 or 4. This applies in particular to restricted lift wells. Note that the subdivision of lots into parts (Parts A to M) is for convenience of definition.



**Figure 5.** (A) The 2D lot shown in grey with the 3D lots (Lots 2, 3, and 4) shown in solid fill. The lot of interest here is Lot 3, shown in blue; (B) the 3D lots (2, 3, and 4) made transparent to allow the building format lots to be seen; (C) Lot 3 only is shown with some of its building format lots highlighted (Red Lot 3603, Purple Lot 3703, Green Lot 3803, Blue Lot 3903). It should be noted that this figure reflects quite a complex situation (together with the text in Section 3.6 and additional explanation of the structure in Section 4.5, the complex situation should become clear).

The subdivision began life as a simple 2D spatial unit, which was divided into four volumetric spatial units (defined by measurements before the building was constructed). Then three of these volumetric units were further subdivided into units, based on the walls of the building under construction. The remainder of the original 2D parcel is the grey prism in Figure 5A and is identified



as Lot 5 on SP217742. On plan SP217742, it is subdivided in 3D into Lots 2, 3, and 4, and some road, with Lot 5 being the remainder of the prism of space after these are excised. In Figure 5A, the outer boundary of Lot 5 is made transparent, with Lots 2 and 4 being shown in yellow. Lot 3 is highlighted in blue.

Lot 3/SP217742 was then further subdivided on plan SP217743. Note that Lot 3 consists of two disconnected parts. These are joined by common property (a lift well, part of Lot 5 which runs up the interior and can be glimpsed in Figure 5A). The conceptual modeling and expression of the plan in LandXML is discussed in Section 4.5.

### Lot 3 Part K

As an example of a relatively simple part of this complex building, consider Part K of Lot 3 (Figures 5B and 6). This is a simple slice, with a polygonal footprint, and three lift shafts running through it. These shafts are not the same cases—two of them (Lot 2 Part K, and Lot 4 Part K—with the lot number in dashed font) are sections of other volumetric spatial units, excised from Lot 3, but Lot 5 (with the lot number in solid font—indicating a base parcel) is a remnant of the original 2D spatial unit (and is connected to  $+\infty$  and to  $-\infty$ ). Encoding of this is discussed in Section 4.5.

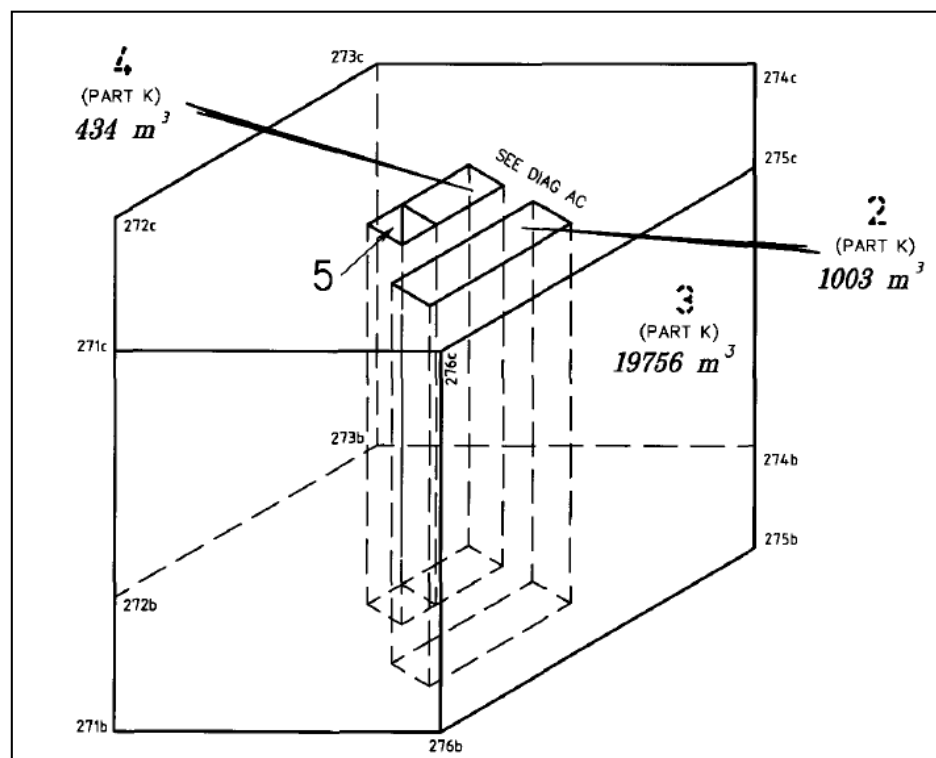


Figure 6. Excerpt from plan SP217742 showing part of Lot 3 (part K).

## 4. Representation in LandXML

The LandXML specification is quite complex, but for the purposes of this discussion, only three elements are required: **Parcel**, **Parcels**, and **CoordGeom**:

1. **Parcel**: in LandXML, the term “parcel” is overloaded to include volumes, faces, and face strings. The class attribute of the parcel element makes this explicit (Face, Lot, FaceString, etc.)
2. **Parcels**: Parcels are a collections of parcels that are used to define a higher-level parcel. For example, a set of Faces and FaceStrings that define a volume.
3. **CoordGeom**: Strictly speaking, this defines a chain of straight line segments, but traditionally in LandXML encoding, a closed CoordGeom is assumed to be a planar surface (polygon).

In this section, the topological modeling of the cases identified in Section 3 are addressed in terms of the conceptual structure and possible implementation in LandXML at varying levels of

detail. The simpler cases are presented as a simple element model, while the more complex cases have been in part encoded in LandXML. Section 4.4 revisits the case introduced in Section 3.5, while Section 4.5 expands on the case from Section 3.6. The spatial logic of the encoding is addressed in Section 4.6, the encoding issues particular to curved surfaces in LandXML in Section 4.7, and alternate encodings in Section 4.8.

#### 4.1. 2D Cadastral Spatial Units

Returning to the case of Section 3.1, Figure 2, a possible element structure, in part, would be as shown in Table 1. The sharing of face string elements should be noted, and compared with a non-topological approach, in which the elements for FaceString d and m would need to be duplicated.

**Table 1.** Excerpt from Face Strings Table.

Face String	+ Spatial Unit(s)	– Spatial Unit(s)
a	Lot 25	Lot 26
b	Lot 25	Lot 26
d	Lot 25	Lot 26
i	Easement A, Lot 25	
p		Easement A
q		Easement A

The encoding in LandXML could be as follows:

```
<Parcel class="FaceString" name="a">
  <CoordGeom><Line><Start pntRef="1"/><End pntRef="2"/></Line>
</CoordGeom></Parcel>
<Parcel class="FaceString" name="b">
  <CoordGeom><Line><Start pntRef="2"/><End pntRef="3"/></Line>
</CoordGeom></Parcel> (etc.)
<Parcel class="Lot" name="25" parcelFormat="Standard">
  <Parcels>
    <Parcel pclRef="a" />    <Parcel pclRef="b" /> (etc.)
    <Parcel pclRef="i" /> (etc.)
  </Parcels>
</Parcel>
<Parcel class="Lot" name="26" parcelFormat="Standard">
  <Parcels>
    <Parcel pclRef=" a" />    <Parcel pclRef=" b"/> (etc.)
    <Parcel pclRef="k" /> (etc.)
  </Parcels>
</Parcel>
<Parcel class="Easement" name="A" parcelFormat="Standard">
  <Parcels>
    <Parcel pclRef="i" />    <Parcel pclRef="r" /> (etc.)
  </Parcels>
</Parcel> ... etc.
```

Note that a negative reference to a face string is indicated by the character “¬” rather than a minus sign. This will be discussed later in the paper (Section 4.6).

#### 4.2. Simple 3D Spatial Unit

The example in Figure 3, Section 3.3 (Reproduced in Figure 7), is clearly well catered for in this approach, where the top and bottom are not horizontal (as indicated by the reduced level values of the vertices—RL in the plan) and therefore have been triangulated (by the surveyor) to ensure the planarity of all faces.

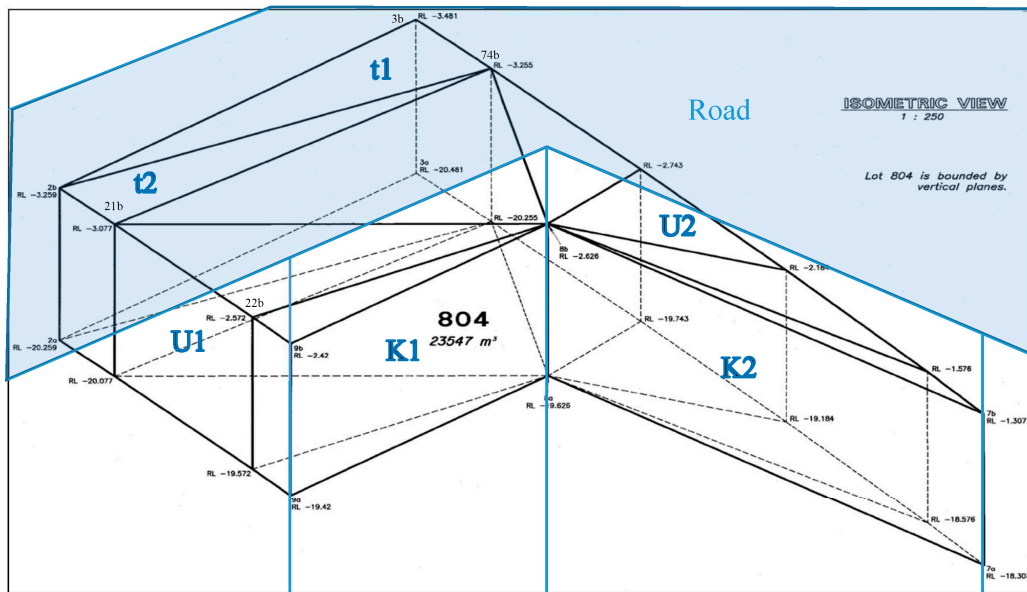


Figure 7. Figure 3, with the addition of the surface road parcel.

The encoding in LandXML could be as follows:

```
<Parcel class="FaceString" name="U1">
  <CoordGeom><Line><Start pntRef="2"/><End pntRef="21"/></Line>
  <Line><Start pntRef="21"/><End pntRef="22"/></Line>
  <Line><Start pntRef="22"/><End pntRef="9"/></Line>
</CoordGeom></Parcel>
<Parcel class="FaceString" name="K1">
  <CoordGeom><Line><Start pntRef="9"/><End pntRef="8"/></Line>
</CoordGeom></Parcel>
<Parcel class="FaceString" name="K2">
  <CoordGeom><Line><Start pntRef="8"/><End pntRef="7"/></Line>
</CoordGeom></Parcel>
<Parcel class="Face" name="t1">
  <CoordGeom><Line><Start pntRef="2b"/><End pntRef="74b"/></Line>
  <Line><Start pntRef="74b"/><End pntRef="3b"/></Line>
  <Line><Start pntRef="3b"/><End pntRef="2b"/></Line>
</CoordGeom></Parcel>
(etc.)
<Parcel class="Lot" name="904" parcelFormat="Volumetric">
  <Parcels>
    <Parcel pclRef="U1"/>    <Parcel pclRef="K1"/>
    <Parcel pclRef="K2"/> (etc.)
    <Parcel pclRef="t1"/>    <Parcel pclRef="t2"/>
    (etc.)
  </Parcels>
</Parcel>
<Parcel class="Road" parcelFormat="Standard">
  <Parcels>
    <Parcel pclRef="K1" />    <Parcel pclRef="K2"/> (etc.)
    <Parcel pclRef="-t1"/>    <Parcel pclRef="-t2"/>
  <Parcel pclRef="u1"/> (etc.)
  </Parcels>
</Parcel>
etc.
```

Here, compared to the non-topological approach, face strings K1 and K2 are shared by the underground spatial unit (Lot 804) and the 2D road parcel. By contrast, the face string U1 is not a boundary of the road parcel, and would not be shared, but a face u1 would be created, as the part of

U1 that separates Lot 804 from the remainder of the road. The faces t1, t2, etc. are shared between Lot 804 and the remainder of the road parcel, but in this case, the sense is reversed.

#### 4.3. Complex 3D Spatial Units

In the case shown in Figure 4, the face strings *a*, *b*, *c*, *d* of the outer face string do not add anything to the definition of the pyramid—which is fully defined by the tilted faces *e*, *f*, *g*, *h*, and horizontal face *i*, but do provide the consistency that permits the database to be viewed as a 2D repository, by simply accessing the boundary face strings as linestrings [3]. This is a small cost for a significant advantage.

The encoding in LandXML could be as follows:

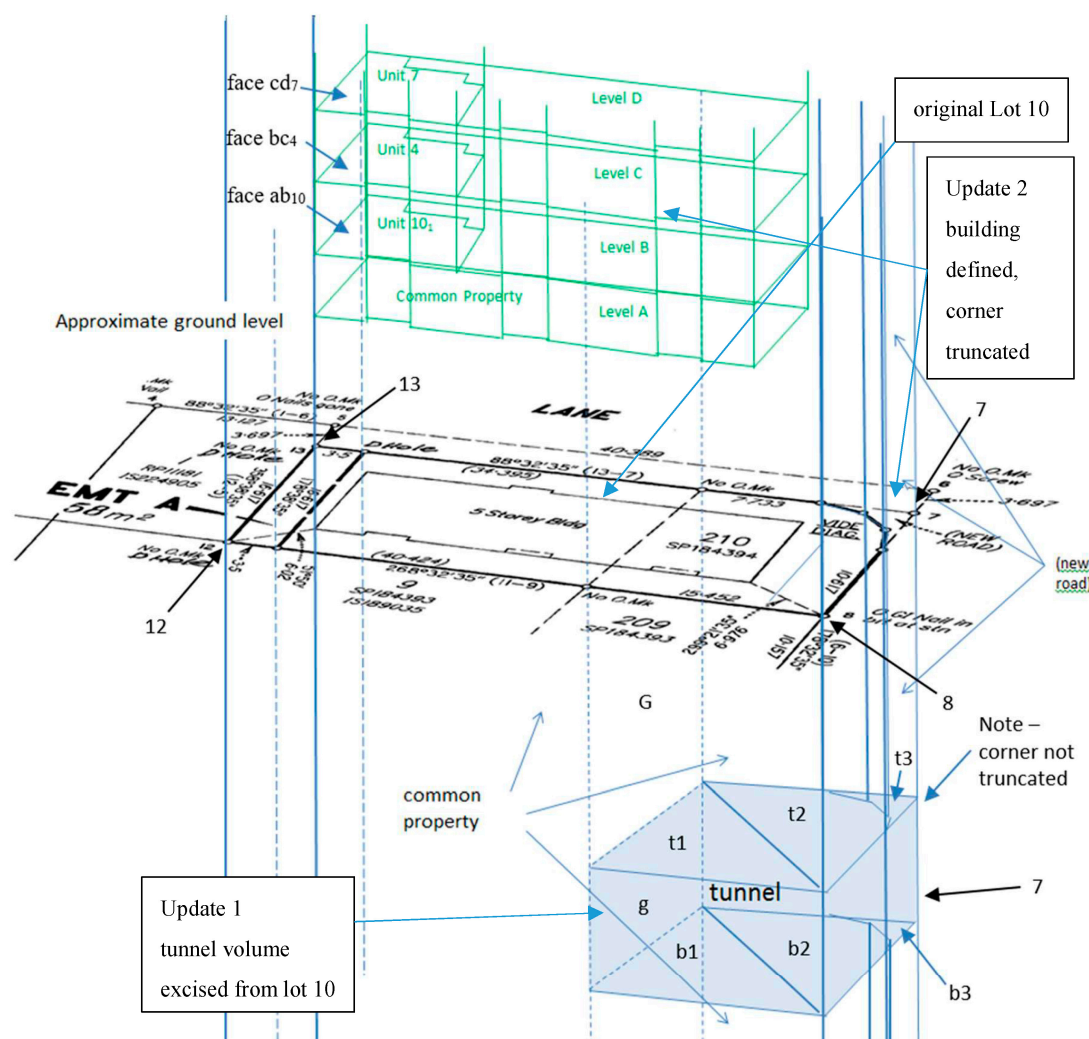
```
<Parcel class="FaceString" name="a">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="FaceString" name="b">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="FaceString" name="c">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="FaceString" name="d">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="Face" name="e">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="Face" name="f">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="Face" name="g">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="Face" name="h">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="Face" name="i">
  <CoordGeom> detail </CoordGeom></Parcel>
<Parcel class="Lot" name="L1" parcelFormat="Volumetric">
  <Parcels>
    <Parcel pclRef="a"/>      <Parcel pclRef="b"/>
    <Parcel pclRef="c"/>      <Parcel pclRef="d"/>
    <Parcel pclRef="e"/>      <Parcel pclRef="f"/>
    <Parcel pclRef="g"/>      <Parcel pclRef="h"/>
    <Parcel pclRef="i"/>
  </Parcels>
</Parcel>
<Parcel class="Lot" name="Common Property" parcelFormat="Standard">
  <Parcels>
    <Parcel pclRef="a"/>      <Parcel pclRef="b"/>
    <Parcel pclRef="c"/>      <Parcel pclRef="d"/>
    <Parcel pclRef="e"/>      <Parcel pclRef="f"/>
    <Parcel pclRef="g"/>      <Parcel pclRef="h"/>
    <Parcel pclRef="i"/>
  </Parcels>
</Parcel>
etc.
```

This storage schema does not depend on the concept of a top surface and a bottom surface—it is only described in these words for clarity. Where there is no clear separation, the set of faces is sufficient as long as the correct orientation of the faces is maintained. Further, if in the survey plan there are multiple 3D spatial units defined, then faces of adjoining spatial units are also shared—one unit on the positive side and one on the negative. Again, this is realized by using signed references to the shared elements, i.e., the shared faces.

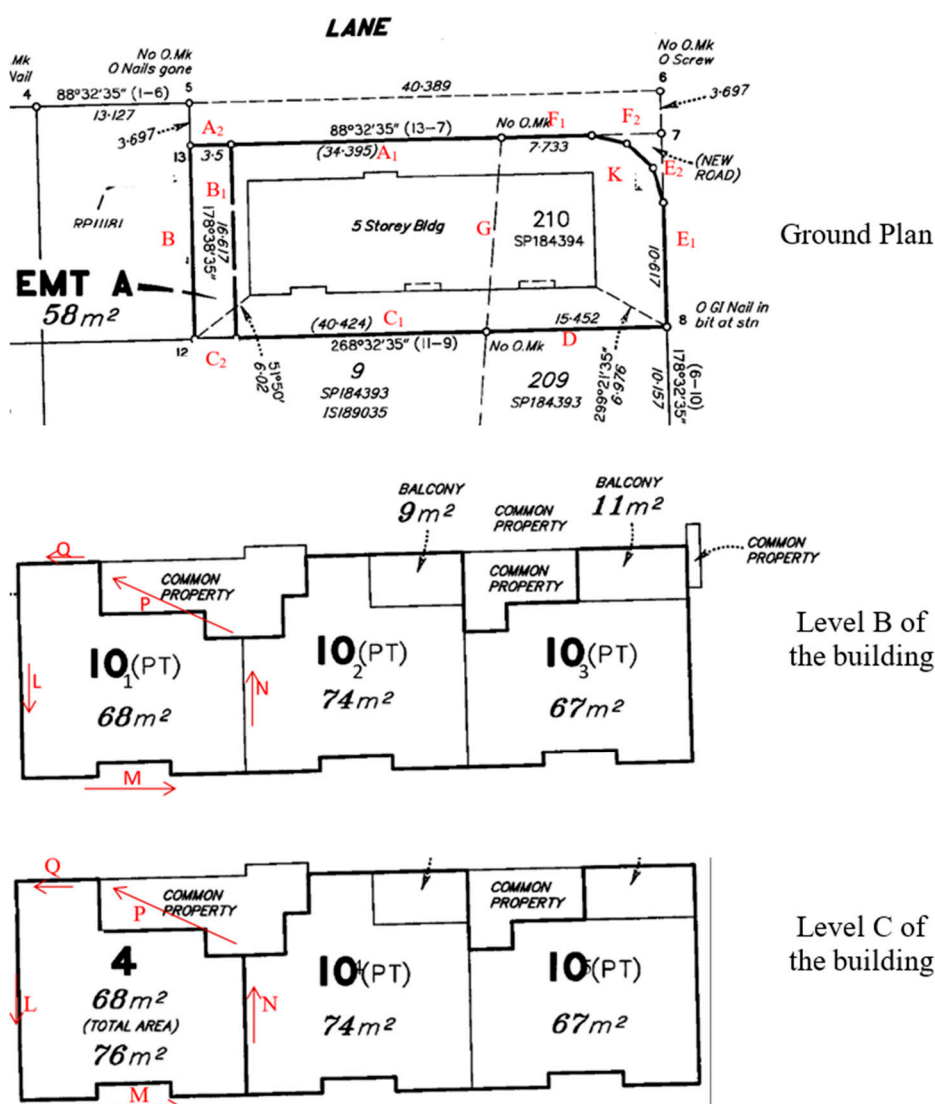
#### 4.4. Case of a Tunnel Below a Building (from Section 3.5)

In the case shown in Figures 8 and 9, initially the spatial unit was a simple 2D lot with 4 sides (Lot 10), adjoining other lots on two sides, and road on the others. A tunnel was constructed below,

the volume being excised from it below ground, leaving a remainder lot unbounded above and below. At a later date, a five-story building was constructed, an easement (labeled as “EMT A” in Figure 8) was put through the lot, and the road corner was truncated. This resulted in the creation of the building format plan but had no effect on the tunnel lot (Lot 210). Although Lot 210 is noted on the new plan, its definition is still provided by the previous plan, and the truncation of the corner does not apply to it (This is in line with the Queensland regulations as mentioned in Section 3.4). Lot 10 then becomes the base lot from which the building format lots are excised. Lot 10 is known from this time on as Lot CP (for Common Property) on the new plan. The tunnel parcel (Lot 210) is unchanged except for the splitting of the face strings and top and bottom faces (Figure 9). The resultant face strings and faces tables are illustrated in Tables 1 and 2.



**Figure 8.** Side view, showing original 2D lot (Lot 10 bounded by corners 12, 13, 7, and 8), the volume of the part of the tunnel below this lot, the corner truncation (removing corner 7 from the surface lot), and four floors of the building near and above ground level. It should be noted that this figure reflects quite a complex situation with 3D parcels above and below the surface at the same location (together with the earlier text in Section 3.5 and the explanation in the current Section 4.4, the complex situation should become clear).



**Figure 9.** Detail of the building subdivision showing labels used in Table 2. Note that the face string definitions L, M, N, P, and Q are used in the definition of units on multiple floors.

Using the labels from Figures 8 and 9, where face strings are shown as capital letters, and faces as lowercase, the following encoding is possible (Tables 2 and 3).

**Table 2.** Face strings table.

Line	+ Spatial Unit(s)	– Spatial Unit(s)
A <sub>1</sub>	Lot CP	Mark Lane (road)
A <sub>2</sub>	Lot CP, Easement A	Mark Lane (road)
B	Lot CP, Easement A	Lot 1/RP11181
B <sub>1</sub>	Easement A	
C <sub>2</sub>	Lot CP, Easement A	Lot 9/SP184393
C <sub>1</sub>	Lot CP	Lot 9/SP184393
D	Lot CP, Lot 210	Lot 9 and Lot 209/SP184393
E <sub>1</sub>	Lot CP, Lot 210	Main Street (road)
E <sub>2</sub>	Lot 210, New Road	Main Street (road)
F <sub>2</sub>	Lot 210, New Road	Mark Lane (road)
F <sub>1</sub>	Lot CP, Lot 210	Mark Lane (road)
G	Lot 210	
K	Lot CP	New Road



L	Unit 10 part 1, Unit 4	
M	Unit 10 part 1, Unit 4	
N	Unit 10 part 1, Unit 4	Unit 10 part 2, Unit 10 part 4
P	Unit 10 part 1, Unit 4	
Q	Unit 10 part 1, Unit 4	

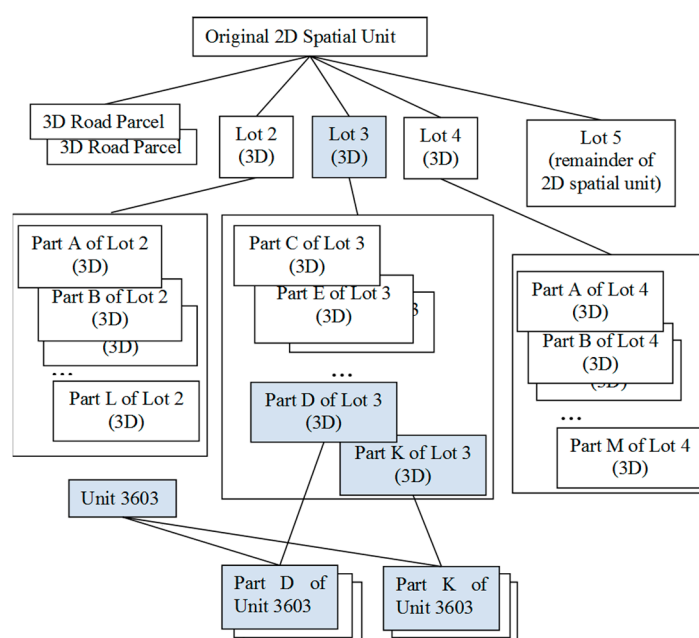
**Table 3.** Faces table.

Face	+ Spatial Unit(s)	– Spatial Unit(s)
$t_1$	Lot 210	Lot CP
$t_2$	Lot 210	Lot CP
$t_3$	Lot 210	New Road
$b_1$	Lot CP	Lot 210
$b_2$	Lot CP	Lot 210
$b_3$	New Road	Lot 210
$g$	Lot 210	Lot CP
$ab_{10}$	Unit 10 part 1	Common Property
$bc_4$	Unit 4	Unit 10 part 1
$cd_7$	Unit 7 (not in diagram)	Unit 4

Thompson et al. (2016b page 148) presents excerpts from an encoding of this case, which is not repeated here.

#### 4.5. Multi-Level Hierarchical Case—High-Rise Building from Section 3.6

This was pictured in Figure 5 as a 2D parcel subdivided into volumetric parcels, some of which are further subdivided by building format plans. The hierarchy of this decomposition is shown in Figure 10.



**Figure 10.** The hierarchy of spatial units and part spatial units, and their use in the definition of basic allocation units.

#### 4.5.1. Encoding of Lot 3 Part K

Please see Figures 5, 6, and 10. The encoding proceeds by defining the outer and inner polygonal boundary of the slice as follows:

```
<Parcel class="FaceString" name="TowerOuter">
  <CoordGeom><Line><Start pntRef="271"/><End pntRef="276"/></Line>
  <Line><Start pntRef="276"/><End pntRef="275"/></Line>
  <Line><Start pntRef="275"/><End pntRef="274"/></Line>
  <Line><Start pntRef="274"/><End pntRef="273"/></Line>
  <Line><Start pntRef="273"/><End pntRef="272"/></Line>
  <Line><Start pntRef="272"/><End pntRef="271"/></Line>
</CoordGeom></Parcel>
<Parcel class="FaceString" name="LiftWell5Inner">
  <CoordGeom><Line><Start pntRef="23"/><End pntRef="20"/></Line>
  <Line><Start pntRef="20"/><End pntRef="21"/></Line>
  <Line><Start pntRef="21"/><End pntRef="22"/></Line>
  <Line><Start pntRef="22"/><End pntRef="23"/></Line>
</CoordGeom></Parcel>
```

Noting that both face strings are encoded in strict order—outer is anticlockwise, inner clockwise. Note also that the repetitive form of the line definition, repeating the node identifiers, is the only form permitted by LandXML at this time.

Then the top and bottom planes are defined. In this case, as described in earlier papers [3,9], the horizontal faces that define the tops and bottoms of the polygonal slice parcels can be expressed as an infinite plane and do not need to repeat the polygon definitions—so that:

```
<Parcel class="Face" name="KTop">
  <CoordGeom><Line><Start pntRef="271b"/><End pntRef="276b"/></Line>
  <Line><Start pntRef="276b"/><End pntRef="275b"/></Line>
  <Line><Start pntRef="275b"/><End pntRef="274b"/></Line>
  <Line><Start pntRef="274b"/><End pntRef="273b"/></Line>
  <Line><Start pntRef="273b"/><End pntRef="272b"/></Line>
  <Line><Start pntRef="272b"/><End pntRef="271b"/></Line>
</CoordGeom></Parcel>
```

and, with a minor extension of the LandXML schema, this could be replaced by the following:

```
<Parcel class="Face" name="KTop" a="0" b="0" c="-1" d="137.625"/>
```

(which is equivalent to defining the boundary face as " $Z < 137.625$ ", or equivalently  $aX + bY + cZ + d \geq 0$  with  $a = b = 0$ ,  $c = -1$ ,  $d = 137.625$ ).

Note that this simplification does require a change to the LandXML specification. If this is not possible, the original encoding can be used, but at the cost of more redundancy in encoding.

Next, the two other lift well face strings are defined as follows:

```
<Parcel class="FaceString" name="LiftWell2Outer">
  <CoordGeom><Line><Start pntRef="25"/><End pntRef="24"/></Line>
  <Line><Start pntRef="24"/><End pntRef="28"/></Line>
  <Line><Start pntRef="28"/><End pntRef="29"/></Line>
  <Line><Start pntRef="29"/><End pntRef="25"/></Line>
</CoordGeom></Parcel>
<Parcel class="FaceString" name="LiftWell4Outer">
  <CoordGeom><Line><Start pntRef="23"/><End pntRef="132"/></Line>
  <Line><Start pntRef="132"/><End pntRef="131"/></Line>
  <Line><Start pntRef="131"/><End pntRef="22"/></Line>
  <Line><Start pntRef="22"/><End pntRef="23"/></Line>
</CoordGeom></Parcel>
```

Naturally, the face strings LiftWell2Outer and LiftWell4Outer will be used in the definitions of Lot 2 and Lot 4. Finally, Part K of Lot 3 can be defined as follows:

```
<Parcel class="LOT" name="3K" parcelFormat="Volumetric">
  <Parcels>
    <Parcel pclRef="TowerOuter"/> <Parcel pclRef="LiftWell15Inner"/>
    <Parcel pclRef="KTop"/> <Parcel pclRef="~JTop" />
    <Parcel pclRef="~LiftWell2Outer"/>
    <Parcel pclRef="~LiftWell4Outer"/>
  </Parcels>
```

where the "~" (not sign) preceding the pclRef of a 2D "parcel" indicates that the "minus" side of the face string or face is used in the definition.

LiftWell5 defines part of Lot 5, which is a 3D spatial unit, but is open above and below. It can be quite simply defined by taking the original 2D lot, excising the volumetric Lots 2, 3, and 4 and the 3D road parcels. So Lot 5, (the remainder) would be defined as follows:

```
<Parcel class="FaceString" name="Lot5Outer">
  <CoordGeom> (the definition of the original 2D lot as a footprint
  face string)
</CoordGeom></Parcel>
<Parcel class="LOT" name="5" parcelFormat="Volumetric">
  <Parcels>
    <Parcel pclRef="Lot5Outer" /> <Parcel pclRef="~4" />
    <Parcel pclRef="~3"/> <Parcel pclRef="~2" />
    <Parcel pclRef="~Road" /> ...
  </Parcels>
</Parcel>
```

where the "~" preceding a pclRef for a 3D parcel indicates that it is subtracted from the main parcel. Since Lots 2, 3, and 4 all individually will have LiftWell5 excised from them, the space is included in Lot 5.

#### 4.5.2. Encoding the Units within Lot 3

The next plan further subdivides Lot 3 into building unit lots, most of which are in two parts—the actual apartment (shown in Figure 11) and the associated car park. The apartments are defined purely by the walls and floor/ceiling of the structure—hence no measurements have been shown on the plan. The car parks are defined by bearing and distance in plan (there being no walls separating car parks in this case), but by the building floor and ceiling.

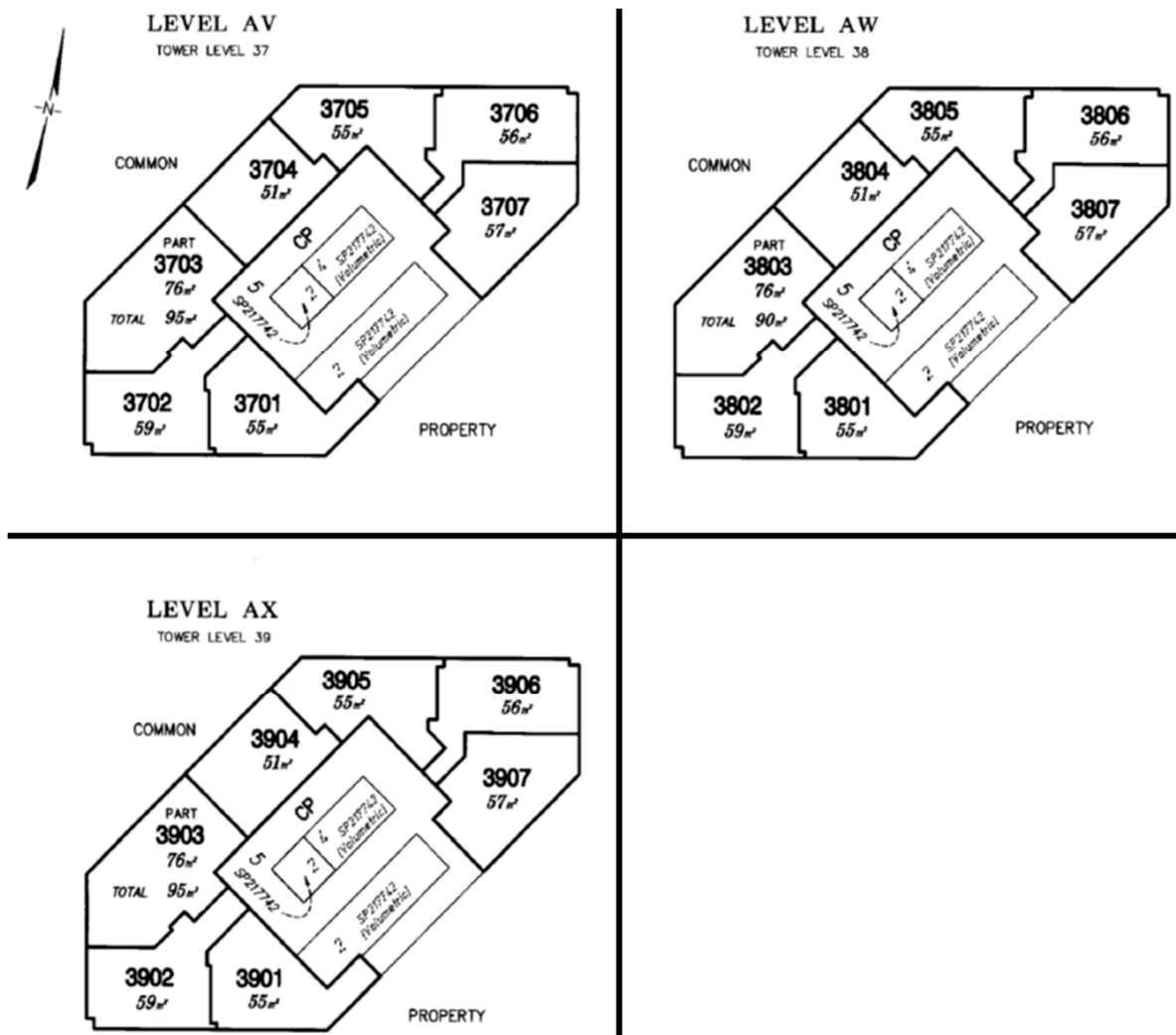


Figure 11. Subdivision of Lot 3 part K into 63 units and common property.

Floors 31 to 39 are subdivided in the same basic format, so the face string definition of the unit footprints can be shared between floors, thus allowing the residential parts of the units to be encoded quite simply as follows:

```
<Parcel class="LOT" name="3701U" parcelFormat="Building">
  <Parcels>
    <Parcel pclRef="Unit3101-3102" /> <Parcel
      pclRef="Unit3101Outer"/>
    etc.
    <Parcel pclRef="~36Top"/> <Parcel pclRef="37Top"/>
  </Parcels>
</Parcel>
<Parcel class="LOT" name="3801U" parcelFormat="Building">
  <Parcels>
    <Parcel pclRef="Unit3101-3102" /> <Parcel
      pclRef="Unit3101Outer"/>
    etc.
    <Parcel pclRef="~37Top"/> <Parcel pclRef="38Top"/>
  </Parcels>
</Parcel>
```

The car parks can be defined in a similar fashion as 3201P etc..., allowing the units to be defined as follows:

```

<Parcel class="LOT" name="3201" parcelFormat="Building">
  <Parcels>
    <Parcel pclRef="+3201U" />    <Parcel pclRef="+3201P"/>
  </Parcels>
</Parcel>

```

Note that the face 38Top could have been defined as 39Bottom, but with  $c = "1"$  and  $d = "-116.5"$ . In that case, the reference in the definition of lot 3801U would have read  $\text{pclRef} = "-39\text{Bottom}"$ .

#### 4.6. Spatial Logic Issues

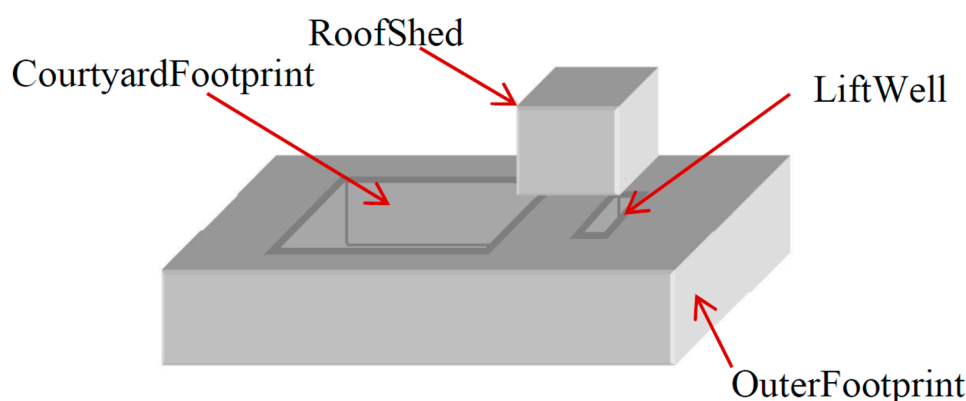
The interpretation of the LandXML is critical, especially since XML requires that the physical order of elements within the file should never be assumed to be significant. Thus in this encoding:

```

<Parcel class="LOT" name="L" parcelFormat="Volumetric">
  <Parcels>
    <Parcel pclRef="OuterFootprint"/>
    <Parcel pclRef="CourtyardFootprint"/>
    <Parcel pclRef="Top1"/>    <Parcel pclRef="¬Top2" />
    <Parcel pclRef="-LiftWell"/> <Parcel pclRef="+RoofShed"/>
  </Parcels>
</Parcel>

```

The references should be interpreted as follows: a  $\text{pclRef}$  to a 3D parcel (such as RoofShed and LiftWell) must have a "+" or "-", with "+" interpreted as the union of the referenced parcel with the current parcel, "-" indicates a subtraction (current parcel "and not" the referenced parcel). Two-dimensional referenced parcels (faces such as Top1 or face strings such as OuterFootprint) either have a not symbol "¬" to indicate that the face is to be reversed, or no special character. The priority should be that the 2D face strings and fully defined faces should be processed first, followed by the faces defined using the "a,b,c,d" notation, followed by the 3D referenced parcels, subtractions before unions. In this example, this would lead to the same order as the XML implies (if LiftWell and RoofShed are pre-defined 3D spatial units) (see Figure 12).



**Figure 12.** Note that the roof shed overlaps and partially blocks the CourtyardFootprint and LiftWell. This is because the "+" operation is done after the boundaries are assembled, and after the lift well is subtracted.

#### 4.7. Curved Surfaces

Several jurisdictions, as in Queensland or the Netherlands, permit curved boundary lines, and curved boundary surfaces in the definition of cadastral spatial units. As mentioned before, where a spatial unit boundary is defined as curved, the plan must record the details of that curve. While approximating this with short lines and facets is acceptable and often necessary for presentation, the definition in a survey plan is the legal situation. As an example, LandXML is used in a number of

jurisdictions [29] and does have some options to define curved lines and surfaces. In practice, the commonly used curves can be accommodated in LandXML (circular arcs, cylindrical faces, conic faces), but not spherical/ellipsoidal surfaces or elliptical lines, which would need to be notated as text on a paper survey plan, and could not therefore be processed automatically.

LandXML, having been defined originally for different purposes has an unusual combination of primitives: It has 0D primitives (Centre), 1D primitives (Line, Irregular line, etc.), and 3D primitives (VolumeGeom), but no explicit 2D primitive that can be used in the definition of a “Parcel” (spatial unit). The only geometric elements available within the Parcel element are Center (a point), CoordGeom (collection of line, curve, and/or spiral elements) (spiral curves are used in road and rail design), and VolumeGeom. An assumption is made that a CoordGeom that closes defines a surface and that, if any of the CoordGeom elements are curved, then the simplest curved surface that passes through those lines is intended. For example, in Figure 13, the definition of the two volumes would be identical in LandXML, but the simplest, A, would be inferred. This cannot be said to provide a truly unambiguous definition of the spatial unit. There is no agreed LandXML encoding for a negatively curved surface such as Figure 13B at present.

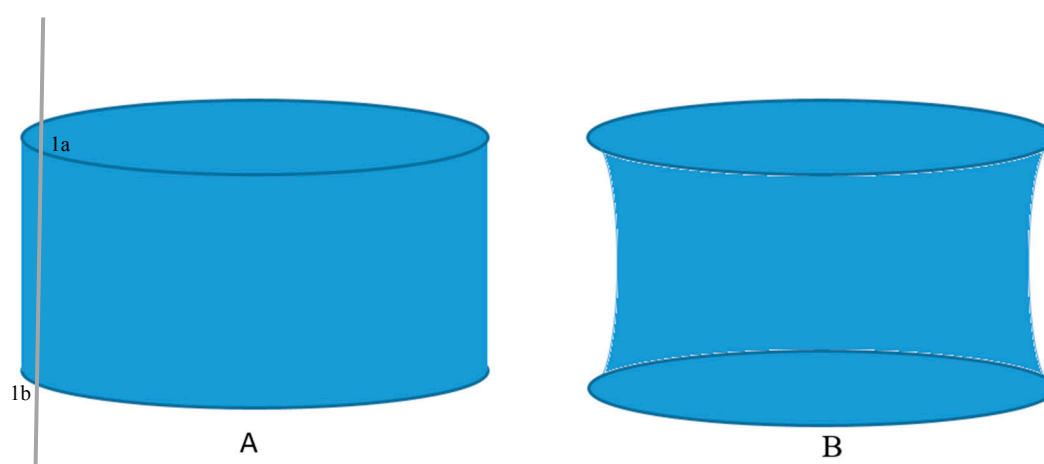


Figure 13. Encoding curved surfaces in LandXML.

While there is no unambiguous and agreed encoding for parcels with general curved surfaces in LandXML (as used for Cadastral plans), it should be noted that the volume of Figure 13A can be encoded in the mixed format as follows:

```
<Parcel class="FaceString" name="Outer">
  <CoordGeom><Curve radius="25" rot="acw">
    <Start pntRef="1"/><End pntRef="1"/></Curve>
  </CoordGeom></Parcel>
<Parcel class="Face" name="Top" a="0" b="0" c="-1" d="130"/> <Parcel
class="Face" name="Bottom" a="0" b="0" c="1" d="-120"/>
```

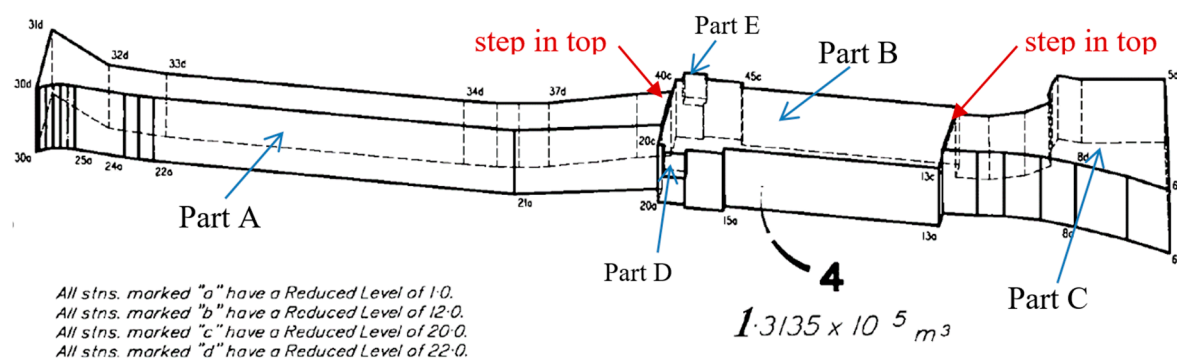
The storage of spatial units with curved boundaries is identical to that for conventional planar-faced or linear faced spatial units, but the processing is more complex. It must be remembered that the curve of the intersection of two simple curved surfaces can be complex to determine [30]. Note that the footprint boundary face strings may or may not need to be curved to define a volume with a curved surface.

#### 4.8. Alternate Encodings

It is clearly possible, for any moderately complex spatial unit, that different encodings could be used. For example, the space in Figure 14, can be readily decomposed into a single “footprint” of vertical walls, with a well-defined set of faces defining the top and bottom. Alternatively, it could be encoded in parts (A to E)—each of which would be a simple slice with horizontal tops and bottoms.



Any combination of these encodings is also possible. Similarly, as discussed with respect to the lift wells in Figure 6, an inner face string can be used; a separate 3D parcel could define the well, to be subtracted from the target spatial unit. It is important that the end-result is functionally equivalent, irrespective of the encoding. To ensure this requires a computable test for equality between alternate representations of the same spatial unit—see Section 5.5.



**Lot 4 is bounded by vertical & horizontal planes.**

**Figure 14.** A Spatial unit that can be encoded in many ways.

## 5. Implementation Issues

A proof of concept implementation has been written to allow encoding of 2D and 3D into LandXML, and to read a limited subset of the LandXML as described here. The remainder of this section discusses various findings from or suggested by this work. Conversion to and from polyhedral form has been implemented, as a way of demonstrating that the necessary functionality is accommodated. Section 5.1 discusses the completeness of the proof of concept implementation, Sections 5.2–5.4 indicate the methods needed to convert to and from more conventional schemas. Section 5.5 addresses the critical issue of computability.

### 5.1. Completeness of Implementation

As part of the proof of concept, the encoding of face-string and face defined spatial units using this conceptual model, the conversion of them to or from conventional polyhedra (see Section 5.2), and the validation of these according to the axioms proposed by Thompson and van Oosterom [7,31] have been demonstrated. However, not all the required combinations of addition and subtraction of sub-units have yet been implemented. Enough has been implemented to allow the examples of Sections 4.1–4.5 to be encoded.

### 5.2. Conversion to Polyhedron Form

The proof of concept implementation converts plan data encoded using this schema to a set of polyhedra. Because some of the spatial units are not fully bounded (e.g., Lot 5 in the example of Section 4.5), special values of  $z$  to represent  $\pm\infty$  have been chosen. As mentioned above, this implementation is not complete, but it is expected that no major obstacles will be encountered. The output polyhedra have been validated, using the proof of concept coding described by Thompson and van Oosterom [7].

### 5.3. Conversion from Polyhedra Form

In practice, the creation of plans using this schema would be the result of the calculations and data entry carried out in the survey process (in a “user-friendly” equivalent to the process described in Section 1.1, but conversion from polyhedra may be necessary where the 3D plan information is sourced from a CAD/CAM package, where the individual spatial units are already described as

polyhedra. The secondary reason for this implementation is that it demonstrates that the storage method is complete.

Assuming planar faces, the procedure is as follows:

- (1) The parameters ( $a, b, c, d$ ) of each face are calculated. This is not trivial, involving a least-squares fitting of the function  $aX + bY + cZ + d$  to the set of vertices of the face, but this is a necessary action in any case to validate the face. Where  $c < 0$ , this is a top face, and it is replaced by a 2D polygon, with  $Z = 0$  for each vertex. Where  $c \geq 0$ , the face is discarded.
- (2) These polygons are then dissolved to produce a (multi) polygon (with possible holes). This becomes the footprint boundary face string.
- (3) The faces are then re-scanned, and any with  $c = 0$  are tested against the boundary face string. If they fall within it, they are discarded. Otherwise, they are retained, along with all faces with  $c \neq 0$ , as boundary faces.
- (4) This produces a non-topological encoding of the spatial units of the plan. Normal 2D algorithms can be used to convert the face strings to a topological network that encodes the individual spatial units in 2D. In a similar way, shared faces can be detected and encoded.

A major issue in this, that must be handled very carefully is the test for  $c = 0$ . Rounding errors in this can lead to mis-calculation of the boundary face string, and overlaps and gaps in the final result. This will be discussed further in Section 5.5.

#### 5.4. Viewing and Manipulating Data Using 2D Software

It is very simple to view a set of spatial units encoded in this form using 2D viewing software. If the face strings that define the spatial units are interpreted as line strings, and the faces are simply ignored, what is left is a 2D topologically encoded plan view of the data. There is little software development needed to achieve this, and the resulting visualization can be useful for a first enquiry on a set of plans—including the production of thumbnail views.

It is even possible to allow updates to the data using 2D software. If the positions of the points are adjusted (in the  $X, Y$  position), the topological structure of the data will not be affected. In many cases, for example, where the spatial units are of the “stepped slice” form (or simpler), they will remain valid. For more complex spatial units, the adjustment can be checked by revalidating the data (in particular, checking planarity of the faces).

#### 5.5. Calculations and Rounding Errors

As a result of the proof of concept implementation, the following observations are made on various issues—which are important to consider if a reliable and robust data transfer is to be achieved.

In calculations, it is important that the coordinates are comparable. Having  $X$  and  $Y$  coordinates in degrees of longitude and latitude while  $Z$  is in meters makes nonsense of the parameters ( $a, b, c, d$ ) of a plane. It is, however, not necessary to convert to a rigorous rectangular coordinate system in general. The assumption is made that the  $Z$  direction is such that two points with the same  $X, Y$  coordinates are on a vertical line, and so a simple approximation is acceptable—such as choosing a local origin and multiplying the latitude and longitude displacements by a constant (about 111,000 for latitude and  $111,000 \times \cos(lat)$  for the longitude)—so that all dimensions are approximately in meters.

This then allows the ( $a, b, c, d$ ) parameters of a plane to be calculated such that  $a^2 + b^2 + c^2 = 1$ , and  $d$  is in meters (the distance from the plane to the local origin). The value of  $c$  indicates how nearly vertical the plane is, and  $1/c$  is approximately the distance in the  $Z$  direction that a point needs to move on the plane to make a difference of 1 m in the  $X, Y$  position. Alternatively, if  $h$  is the maximum range of  $z$  values on the face (its “height”), then  $hc$  is the distance that the face deviates from the vertical. If this value for any face is smaller than the planarity tolerance ( $\epsilon'$  in [7]), then it is worthwhile setting  $c = 0$ , adjusting  $a, b, d$  so that  $a^2 + b^2 = 1$ , and re-applying the tolerance test for the vertices of the face.

Ideally, an effort should be made to flatten a set of faces that were intended to be planar. This can be significant where a planar surface is common to a number of spatial units, but should be

determined by calculating a set of plane parameters for all faces together, and checking that all vertices are within tolerance based on these parameters.

Because the same plan can be encoded in several different ways in general, the logic operations must be rigorous and repeatable. For example, there must be a method of testing whether two representations are equal. The point set definition ( $A = B =_{\text{def}} p \in A \Leftrightarrow p \in B$ ) is not programmable, especially where a tolerance is permitted. As argued in [32], a more realistic approach is to provide a process that can detect an empty parcel ( $\text{isEmpty}(A)$  meaning that there can be no points within  $A$ ), then use the subtraction operation to define  $A \subseteq B =_{\text{def}} \text{isEmpty}(B - A)$ , and then define  $A = B =_{\text{def}} A \subseteq B$  and  $B \subseteq A$ .

There are broadly three approaches to fully rigorous logic within spatial database structures that are immune to rounding errors (where all faces are planar): 1. The infinite precision rational number approach [33], 2. The Dual Grid [34,35] (which also uses rational numbers, but restricts the sizes to which they can grow, and 3. The Regular Polytope [32,36], which uses a different form of storage of spatial objects.

An approach to rigor in the definition of spatial units in 3D is being adopted informally by some surveyors, and perhaps should be taken up as a standard or guideline. This is to ensure that all faces are either horizontal or vertical, or are fully triangulated. (Face strings are by definition vertical and thus fit this scheme very well. Spatial units defined in this way in the Queensland Cadastre can be found in Figure 3 and the underground portion of Figure 8. By contrast, it would be wrong to “correct” a non-planar surface by triangulating it post-registration of the plan, as this introduces an interpretation that could lead to disagreements.

The proof of concept uses an approach where two tolerances are imposed to ensure robustness [37],  $\epsilon$  a larger value, which sets a limit to the closeness of points, flatness of faces, etc., and a much smaller value  $\iota$ , which is assumed to be the accuracy to which the computations can be made. The difficulty here is that  $\epsilon$  is a fixed distance—for example, 1 mm—while  $\iota$  varies depending on the size of the arguments being applied to the calculation. It is not certain that the requirement  $\iota \ll \epsilon$  can necessarily be achieved with 8 byte floating point arithmetic and real-world sized spatial units.

## 6. Analysis of Results

### *Sharing of Data*

The sharing of face and face string definitions in this approach had proved to operate in many cases at a very high level. The following variants occur in the plans investigated here:

1. Some definitions are not shared: Face strings at the edge of the plan, and faces that surround a secondary interest (not excised from the enclosing spatial unit) are examples.
2. Many definitions are shared by exactly two spatial units—where they form the boundary between them.
3. A significant number of face strings are shared a large number of times: For example, the face string “Unit3101–3102” defined in Section 4.5.2 would be used in the units 3101 to 3910 (9 units), and reversed in 3102 to 3902. A total of 18 usages.

Further, using the infinite plane form of the horizontal boundaries between floors (Section 4.5.2), which admittedly does require a change to the LandXML specification, the definition of “38Top” can be shared between 7 units and the common property above and the same below (16 uses in total).

## 7. Conclusions and Future Research

### *7.1. Major Results*

This paper has presented a conceptual model applied to topology encoding of range of spatial units (2D, simple 3D, complex 3D). It supports elegant fusing of the 2D and the 3D representations with a key role for the 1D line string in 2D space, which becomes a 2D face string in 3D space. Additionally, this line string is used to depict footprints of the 3D spatial units in traditional 2D

cadastral maps. The further benefits of the proposed conceptual model and the topology encoding are as follows:

1. It is a close fit to current cadastral survey practice (Section 3.2).
2. It permits efficient encoding, explicit semantics on neighbors, reduction of redundancy, and avoids inconsistencies. (Section 6.1)
3. It is capable of being viewed and even manipulated by software that is strictly 2D (not 3D-aware) (see Section 5.4).

Further, it achieves this with any combination of simple 2D and/or a mixture of 2D and 3D spatial units in an individual survey, combined with a hierarchical/multi-layered decomposition of spatial units.

The conceptual model for survey plans has been expressed in the language of the LADM. Two multi-step real world examples are given and encoded according to this conceptual model in LandXML for exchange purposes, including the initial registration.

## 7.2. Future Work

Future work includes investigating more types of 3D Cadastral parcel real world cases. It should be noted that, in the future not all survey plans may originate from surveys, but quite a number of them might result from design; e.g., BIM/IFC (Building Information Modeling/Industry Foundation Classes) [18]. Additionally, here too the role of topology is of equal importance, and topology encoding from this source needs further investigations. Currently, in much legislation, the survey plan (paper hard copy) is an official, legal document. The issue of the digital document being the official document needs further investigation (on aspects such as long-term preservation, digital signatures, approvals, etc.). This is quite similar to the treatment of administrative legal documents (Deeds) in the Netherlands. The digital deeds have become legal documents and are submitted via the ELAN system [38]. These digital deeds now form the large majority of submitted, approved, and recorded legal administrative documents.

It has been suggested in the introduction that the conceptual model as proposed for the survey plans, could also be applied to the cadastral database. Future work should explore how this can be realized: integrating the topology island from the survey plan into the topology structure of the complete cadastral database, and using the technique of “Versioned Objects” as used in ISO19152 [2]. The ultimate goal is to integrate the survey data with the cadastral map data in an integrated database [39]. Further investigation of the “two-tolerance” approach is needed to ensure that a rigorous logic can be assured.

The aim of the InfraGML project is to satisfy all of the functional requirements of LandXML, so that an expression of this conceptual schema in InfraGML should be relatively trouble-free. The appropriate sections of InfraGML are Section 6 (Survey) and Section 7 (Land Division), but the latter is not currently completed [6]. When the InfraGML standard is firmed up, an extension of this research to use that standard is strongly indicated.

**Author Contributions:** The research reported here is the work of all three authors. Rodney James Thompson wrote the proof of concept software, and all three authors contributed the text of the final paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lemmen, C.; van Oosterom, P.; Thompson, R.J.; Hespanha, J.; Uitermark, H. The Modelling of Spatial Units (Parcels) in the Land Administration Domain Model (LADM). In Proceedings of the XXIV FIG International Congress, Sydney, Australia, 11–16 April 2010.
2. ISO TC211. ISO 19152:2012 Geographic Information—Land Administration Domain Model (LADM). Available online: <https://www.iso.org/standard/51206.html> (accessed on 6 May 2017).

3. Thompson, R.J.; van Oosterom, P.; Soon, K.H.; Priebsenow, R. A Conceptual Model Supporting a Range of 3D Parcel Representations through all Stages: Data Capture, Transfer and Storage. In Proceedings of the FIG Working Week 2016, Christchurch, New Zealand, 2–6 May 2016.
4. LandXML. Available online: <http://www.landxml.org/> (accessed on 6 May 2017).
5. Scarponcini, P. InfraGML Proposal (13–121), OGC Land and Infrastructure Domain Working Group. Available online: [https://portal.opengeospatial.org/files/?artifact\\_id=56299](https://portal.opengeospatial.org/files/?artifact_id=56299) (accessed on 6 May 2017).
6. OGC InfraGML 1.0. Available online: [https://portal.opengeospatial.org/files/?artifact\\_id=72352](https://portal.opengeospatial.org/files/?artifact_id=72352) (accessed on 6 May 2017).
7. Thompson, R.; van Oosterom, P. Axiomatic Definition of Valid 3D Parcels, Potentially in a Space Partition. In Proceedings of the 2th International FIG 3D Cadastre Workshop, Delft, The Netherlands, 16–18 November 2011.
8. Thompson, R.; van Oosterom, P.; Karki, S.; Cowie, B. A Taxonomy of Spatial Units in a Mixed 2D and 3D Cadastral Database. In Proceedings of the FIG Working Week 2015, Sofia, Bulgaria, 17–21 May 2015.
9. Thompson, R.J.; van Oosterom, P.; Soon, K.H. Mixed 2D and 3D Survey Plans with Topological Encoding. In Proceedings of the 5th International FIG 3D Cadastre Workshop, Athens, Greece, 18–20 October 2016.
10. ICSM. ePlan Data Model Governance. Available online: [http://www.icsm.gov.au/eplan/ePlan\\_Governance\\_v1.1.pdf](http://www.icsm.gov.au/eplan/ePlan_Governance_v1.1.pdf) (accessed on 6 May 2017).
11. Janečka, K.; Karki, S. 3D Data Management—Overview Report. In Proceedings of the 5th International FIG 3D Cadastre Workshop, Athens, Greece, 18–20 October 2016.
12. Dimipoulou, E.; Karki, S.; Miodrag, R.; de Almeida, J.-P.D.; Griffith-Charles, C.; Thompson, R.; Ying, S.; van Oosterom, P. Initial Registration of 3D Parcels: Overview Report. In Proceedings of the 5th International FIG 3D Cadastre Workshop, Athens, Greece, 18–20 October 2016.
13. Soon, K.H.; Tan, D.; Khoo, V. Initial Design to Develop a Cadastral System that Supports Digital Cadastre, 3D and Provenance for Singapore. In Proceedings of the 5th International FIG 3D Cadastre Workshop, Athens, Greece, 18–20 October 2016.
14. Zeiss, G. A Proposal to Replace LandXML with a New Standard: InfraGML. Directions Magazine 11th Dec 2013. Available online: <http://www.directionsmag.com/entry/a-proposal-to-replace-landxml-with-a-new-standard-infragml/371892> (accessed on 6 May 2017).
15. Stoter, J.E.; van Oosterom, P. *3D Cadastre in an International Context: Legal, Organizational, and Technological Aspects*; CRC Press: Boca Raton, FL, USA, 2006; p. 344.
16. Oldfield, J.; van Oosterom, P.; Quak, W.; van der Veen, J.; Beetz, J. Can Data from BIMs be Used as Input for a 3D Cadastre? In Proceedings of the 5th International FIG 3D Cadastre Workshop, Athens, Greece, 18–20 October 2016.
17. Paulsson, J. Swedish 3D Property in an International Comparison. In Proceedings of the 3rd International FIG 3D Cadastre Workshop: Developments and Practices, Shenzhen, China, 25–26 October 2012.
18. Atazadeh, B.; Kalantari, M.; Rajabifard, A.; Ho, S.; Ngo, T. Building Information Modelling for High-rise Land Administration. *Trans. GIS* **2017**, *21*, 91–113, doi:10.1111/tgis.12199.
19. Van Oosterom, P.; Stoter, J.; Quak, W.; Zlatanova, S. The Balance between Geometry and Topology. In *Advances in Spatial Data Handling*; Richardson, D.E., van Oosterom, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; pp. 209–224.
20. Hoel, E.; Menon, S.; Morehouse, S. Building a Robust Relational Implementation of Topology. In *Lecture Notes in Computer Science: Advances in Spatial and Temporal Databases. SSTD 2003*; Hadzilacos, T., Manolopoulos, Y., Roddick, J., Theodoridis, Y., Eds.; Springer: Berlin/Heidelberg, Germany, 1993; Volume 2750, pp. 508–524.
21. Louwsma, J. H. Topology Versus Non-Topology Storage Structures. Available online: [http://www.gdmc.nl/publications/2003/Topology\\_storage\\_structures.pdf](http://www.gdmc.nl/publications/2003/Topology_storage_structures.pdf) (accessed on 6 May 2017).
22. De Hoop, S.; van Oosterom, P.; Molenaar, M. Topological Querying of Multiple Map Layers. In *Lecture Notes in Computer Science: Spatial Information Theory A Theoretical Basis for GIS. COSIT 1993*; Frank, A.U., Campari, I., Eds.; Springer: Berlin/Heidelberg, Germany, 1993; Volume 716, pp. 139–157.
23. Ledoux, H.; Gold, C. Simultaneous Storage of Primal and Dual Three-Dimensional Subdivisions. *Comput. Environ. Urban Syst.* **2007**, *31*, 393–408, doi:10.1016/j.compenvurbsys.2006.03.003.
24. Boguslawski, P.; Gold, C. Construction Operators for Modelling 3D Objects and Dual Navigation Structures. In *Lecture Notes in Geoinformation and Cartography: 3D Geoinformation Sciences*; Zlatanova, S., Lee, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; Part II, pp. 47–59.

25. Baumgart, B.G. Winged Edge Polyhedron Representation. Stanford Artificial Intelligence Project, 1972, 44, Document STAN-CS-72-320 Memo AIM-179. Available online: <http://www.dtic.mil/dtic/tr/fulltext/u2/755141.pdf> (accessed on 6 May 2017).
26. Molenaar, M. Single Valued Vector Maps—A Concept in Geographic Information Systems. *GeoInform. Syst.* **1989**, *2*, 18–27.
27. Van Oosterom, P. *Research Issues in Integrated Querying of Geometric and Thematic Cadastral Information (2)*; Technical Report; Delft University of Technology: Delft, The Netherlands, 2000.
28. DNRM Cadastral Survey Requirements v7.1, Reprint 1, Department of Natural Resources and Mines, 2016. Available online: <https://www.dnrm.qld.gov.au/?a=105601> (accessed on 6 May 2017).
29. ICSM. ePlan Protocol LandXML Mapping. Available online: <https://icsm.govspace.gov.au/files/2011/09/ePlan-Protocol-LandXML-Mapping-v2.1.pdf> (accessed on 6 May 2017).
30. Abdel Malek, K.; Yeh, H.J. Determining Intersection Curves Between Surfaces of Two Solids. *Comput. Aided Des.* **1996**, *28*, 539–549.
31. Thompson, R.; van Oosterom, P. Modelling and Validation of 3D Cadastral Objects. In *Urban and Regional Data Management: UDMS Annual 2011*; Zlatanova, S., Ledoux, H., Fendel, E., Rumor, M., Eds.; CRC Press: London, UK, 2012; pp. 7–23.
32. Thompson, R.J. Towards a Rigorous Logic for Spatial Data Representation. Ph.D. Thesis, Delft University of Technology, Delft, The Netherlands, 2007.
33. Weinrich, B.E.; Schneider, M. Use of Rational Numbers in the Design of Robust Geometric Primitives for Three Dimensional Spatial Database Systems. In Proceedings of the 13th Annual ACM Workshop on GIS, Bremen, Germany, 4–5 November 2005; pp. 163–172.
34. Lema, J.A.C. Dual Grid: A Closed Representation Space for Consistent Spatial Databases. Ph.D. Thesis, Universidade da Coruña, Coruña, Spain, 2012.
35. Lema, J.A.C.; Güting, R.H. Dual grid: A new Approach for Robust Spatial Algebra Implementation. *GeoInformatica* **2002**, *6*, 57–76.
36. Thompson, R.J.; van Oosterom, P. Connectivity in the Regular Polytope Representation. *GeoInformatica* **2011**, *15*, 223–246.
37. Belussi, A.; Migliorini, S.; Negri, M.; Pelagatti, G. Establishing Robustness of a Spatial Dataset in a Tolerance-Based Vector Model. *Trans. GIS* **2016**, doi:10.1111/tgis.12221.
38. Stolk, P.; Lemmen, C.H.J. Technical aspects of electronics conveyancing. In Proceedings of the 2nd FIG Regional Conference, Marrakech, Morocco, 2–5 December 2003.
39. Van Oosterom, P.; Lemmen, C.; Uitermark, H.; Boekelo, G.; Verkuijl, G. Land Administration Standardization with focus on Surveying and Spatial Representations. In Proceedings of the ACMS Annual Conference Survey Summit 2011, San Diego, CA, USA, 7–13 July 2011.

