CCDM and Open Source Applications

in Context of Implementing Cadastre in Iceland

Master of Science thesis: Tryggvi Már Ingvarsson

Professor: prof. dr. ir. P.J.M. van Oosterom (Delft University of Technology)
Supervisors: prof. mr. J.W.J. Besemer (Delft University of Technology)
drs. C.W. Quak (Delft University of Technology)

Delft, November 2005

Section GIS Technology
Geodetic Engineering
Faculty of Civil Engineering and Geosciences
Delft University of Technology

TABLE OF CONTENTS

PREFACE

A great deal of effort has been put into this essay during the last few months. The official starting date of the research was in March 2005 and the intended final date was to be in mid September 2005. The reason for the two months delay can be found in several side projects implemented along the way. Between now and then: two courses of total 8 ECTS were undertaken; a week long trip to Iceland with a Dutch football team was planned and executed in July; and participation in organising and hosting a week long Icelandic cultural festival in Rotterdam in September. Despite all this the research was still carried out, little by little, resulting in the report that is now in your hands.

There are many people to thank for both support and input into this project. Starting with my instructors, Wilko Quak has been invaluable with our frequent brainstorm meetings, Peter van Oosterom has been an ocean of knowledge and enthusiasm while Jaap Besemer has brought in valuable comments. Outside the formal supervision Marian de Vries has been of great help with technical issues, Sizi Zlatanova assisted in designing the UML models, Jaap Zevenbergen commented on topics of the Icelandic land registration, Elfriede Fendel translated the summary of this report to Dutch, Theo Tijssen administrated the workstation at OTB and Edward Verbree shared his enthusiasm on open-source applications. In addition to OTB there are also many other people to thank. I would like to thank in particular: Tom Barry, Ásta Sólveig Andrésdóttir and Haukur Ingibergsson at the Land Registry in Iceland; Björk Ingimundardóttir at the National Archives; Örn Arnar Ingólfsson, Daði Björnsson and everyone else at my previuos workplace Loftmyndir ltd. Moreover I thank all my friends for moral support and understanding, including Hafliði, Garðar, Marcel, Anton, Walter, Martijn, Vladimir, Osana and everyone at Ariston'80 football club. Finally but not least, my fiancée Björt and our cheerful dog Yara for taking care of me during this final stage of my MSc study. Many thanks and I hope you enjoy the reading.

Delft, 24 November 2005

Tryggvi Már Ingvarsson

SUMMARY

Keywords: cadastral registration; Iceland; CCDM; SDBMS; topology; Open-Source.

The English politician Benjamin Disraeli once said: "The best way to become acquainted with a subject is to write a book about it." This is to a large extent the reason for undertaking this research. It covers various topics and aspects related to cadastral registration taking several viewpoints. The centre of attention is the cadastral registration in Iceland and how it can be enhanced by adding spatial delimitation of land parcels into it. Such a project is huge to develop and many options and factors have to be considered like:

- modelling the relevant classes, attributes and relationships ;
- choose or design appropriate spatial model;
- storage of spatial data;
- spatial access methods;
- cadastral (spatial) transactions;
- system architecture; and
- choice of software and hardware solutions.

While trying to briefly cover and discuss all of these aspects the research focuses: in one hand on detailed modelling of cadastres; while on the other hand it examines, by implementing a prototype, the applicability of open-source geo-applications to cadastral registration, taking into considerations the factors listed above. This is reflected in the research question:

In what way can the Core Cadastral Domain Model (CCDM) and open-source software benefit the development of cadastral registration in Iceland?

To answer this question several methods were employed. Literature was studied, experts on land registration in Iceland and on the CCDM questioned, reverse engineering applied on the digital land registry database and practical experience gained through implementing generalised cadastral prototype.

This, along with general findings and conclusion is discussed and summarised in the text below.

Although land registration has a long tradition in Iceland, spatial mapping of parcels has been more or less neglected by the authorities. Presently the system compromises sophisticated and digital registration of parcels, identified with analytical id but without spatial extent. The parcels are related with persons through their registered ownership rights. This is the role of the governmental operated digital 'land registry database' (LRD) in a nutshell.

Initially the municipalities in Iceland are responsible to establish land parcels as real property in the LRD. With the larger municipalities maintaining own local land information systems collecting the extent of all parcels within their administrative area, connecting them to corresponding unique land identification in LRD. This is a step in the direction of cadastres, but locally implemented with local preconditions. Making these diverse local cadastral repositories internally incomparable in terms of what is registered, attributes, quality and metadata etcetera.

Not until recently there has been some movement to accomplish spatial delimitation of parcels by the government with the Land Registry in Iceland (LRI) actively searching for ideas and solutions. This is where input of the core cadastral domain model (CCDM), as developed by the cooperative initiative of experts at Delft University of Technology, the International institute for geo-information science and earth observation in Enschede (ITC) and the International federation of surveyors (FIG), could serve of great importance.

The CCDM was first brought forward in 2002 on FIG conference in Washington, United States. The idea is to employ the concept of model driven architecture (MDA) to come up with a UML model defining the core classes and attributes of cadastral registration in the world. Thereby providing shared ontology for: data exchange; discussion; and development within the cadastral domain. Its objectives are listed in Lemmen et al (2003, p.1):

> A standardized core cadastral domain model, covering land registration and cadastre in a broad sense (multipurpose cadastre), will serve at least two important goals:
>
> 1. avoid reinventing and re-implementing the same functionality over and over again, but provide a extensible basis for efficient and effective cadastral system development based on a model driven architecture, and
> 2. enable involved parties, both within one country and between different countries, to communicate based on the shared ontology implied by the model.

Observed the innovation of CCDM has a stated potential to prevent the development of cadastres of reinventing the wheel again, e.g. in the case of Iceland when modelling recording of spatial extent of parcel. This would be possible if the current model of the LRD could be fitted within, or extended from, the CCDM, creating an Icelandic cadastral model (ICM), encompassing both registration of land rights and geometric extent of parcels. Thereby creating an opportunity of using the spatial delimitation described in the geographic part of CCDM as an example for spatial recording in the ICM.

After modelling the LRD this research concluded that the CCDM cannot simply be extended to comprise LRD and create an ICM, but it is however not far from it. In conclusion several recommendations are put forward to refine the CCDM in Icelandic perspective.

The principal contribution of CCDM to the cadastral development in Iceland, apart from the declaration of the core classes and ontology, is its healthy model driven approach. Up to now has the development of the LRD been bottom-up approach employing external readymade solutions, more or less data-driven. By following the CCDM methodology the designed solution for the Icelandic cadastral registration can be more top-down and knowledge-driven, transparent and better structured.

This research however concluded that apart from obvious benefits of the CCDM initiative it has its flaws in its approach. Trying to be universal it contains classes to also accompany nomadic rights to land, independent of if it is in Rwanda, Iceland or Nepal, resulting with growing complexity. This can especially be observed by comparing the different versions of the CCDM, whereas each version tries to be more universal covering than the preceding was.

It is suggested here that the developers focus the development of CCDM on homogenous cultural areas, like within the European Union. With this experience the model can be extended and refined later to other regions. It is also controversial objective to be in both, paradigm and a denominator of cadastral systems, as CCDM appears to be. A recommendation would be to segregate these two objectives, creating two models where one looks conceptually to the future (creating paradigm) while the other encompasses and creates common ground for present cadastral registration systems (more technically oriented and functioning as a denominator).

This report also covers the implementation of a cadastral prototype emphasising on recording the spatial delimitation of parcels according to the proposed ICM model. Therefore open-source geo-software toolkits were experimented along with elaborating on spatial modelling and possible system architecture.

The ideal spatial model for two-dimensional cadastral registration is concluded to be topological, consisting of nodes (monuments), edges (boundaries) and faces (partitioning parcels). Using function developed in the SDBMS these topological primitives can be stored with the corresponding geometry computed on demand. This was realised in the prototype whereas topology was constructed in PostgreSQL/ PostGIS SDBMS.

Ideal system architecture to facilitate the organisational setup of the cadastral registration in Iceland was found optimised in centrally stored SDBMS, updated remotely by responsible municipalities. It is further stated here that direct database connection is at present time superior to OGC transactional web feature services (WFS-T) but the development of WFS-T technology could quickly alter this. The benefit of using WFS-T is that it allows system architecture of heterogeneous desktop clients, while direct database connection is more subjected to homogeneous solutions. Especially when using open-source SDBMS that is not widely supported by proprietary GIS developers.

Finally, the case study showed that the potential contribution of open-source software in the implementation of cadastral registration in Iceland can be diverse. It argues that several open-source software available today are a serious candidate and a real choice when developing spatial enabled cadastral registration. Examples of applications are e.g. PostgreSQL with PostGIS to store spatial data in SDBMS; MapServer on Apache HTTP server to share spatial data in a web environment (WMS & WFS); GeoServer to enable WFS-T to spatial data; and finally uDig as a desktop client that can be extended with diverse customised functionalities by accessing and editing its open-source code.

Echoing the research question the main conclusion is that CCDM and open source geo-applications can be of invaluable benefit to the development of cadastral registration in Iceland, contributing in diverse ways as detailed in the report.

SAMENVATTING

Trefwoorden: kadastrale registratie; Ijsland; CCDM; SDBMS; topologie; Open Source.

De Engelse politicus Benjamin Disraeli merkte eens op "The best way to become acquainted with a subject is to write a book about it" (De beste manier om vertrouwd te raken met een onderwerp is door er een boek over te schrijven). Dit is in belangrijke mate de reden om dit onderzoek uit te voeren. Centraal staat de kadastrale registratie in IJsland en hoe deze verbeterd kan worden door het toevoegen van ruimtelijke begrenzingen aan percelen.
De uitvoering van een dergelijk project vraagt om een grote inspanning. Bovendien dienen vele mogelijkheden en factoren in ogenschouw te worden genomen, zoals:

- het modelleren van de relevante klassen, kenmerken en relaties inzake kadastrale registratie;
- het kiezen of ontwerpen van een geschikt ruimtelijk model;
- het opslaan van ruimtelijke gegevens;
- kadastrale (ruimtelijke) transacties;
- systeemarchitectuur, en tenslotte
- het kiezen van software- and hardware-oplossingen.

De poging om al deze aspecten af te dekken en te bediscussiëren richt zich enerzijds op het meer in detail modelleren van kadasters. Anderzijds ligt het accent op het – door middel van het implementeren van een prototype – toepassen van open-source geo-applicaties bij de kadastrale registratie, waarbij met bovengenoemde factoren rekening wordt gehouden. Dit resulteert in de volgende onderzoeksvraag:

> Op welke wijze kan de ontwikkeling van de kadastrale geografische informatie in IJsland profiteren van het 'Core Cadastral Domain Model' (CCDM, het kern kadastraal domein model) en open-source software?

Om deze vraag te beantwoorden zijn verschillende methoden aangewend:

- literatuuronderzoek;
- bevraging van experts op het gebied van grondregistratie in IJsland en van experts op het gebied van het Core Cadastral Domain Model (CCDM);
- reverse engineering toegepast op de digitale vastgoedregistratie database;
- verwerving van praktijkervaring door middel van het implementeren van een gegeneraliseerd kadastraal prototype.

Bovengenoemde methoden worden samen met de algemene bevindingen en conclusies in onderstaande tekst bediscussieerd en samengevat.

Alhoewel de vastgoedrechtenregistratie in IJsland een lange traditie kent, hebben de autoriteiten de kadastrale geografische informatie min of meer veronachtzaamd. Momenteel bestaat het systeem uit een geperfectioneerde en digitale registratie van percelen, die op basis van een analytische id maar zonder ruimtelijke extensie worden geïdentificeerd en die door middel van het hun geregistreerde eigendomsrecht gerelateerd worden aan personen. Dit is in het kort de rol van de bij de overheid in gebruik zijnde digitale 'vastgoedrechtenregistratie database' ('land registry database' (LRD)).

In eerste instantie zijn de gemeenten in IJsland verantwoordelijk voor het inbrengen van de eigendomsrechten op de percelen in het LRD. De grotere gemeenten houden hun eigen lokale vastgoedinformatiesystemen bij, waarbij de gegevens omtrent de grootte van alle percelen in hun administratieve gebied worden verzameld en er een koppeling plaatsvindt met de corresponderende unieke vastgoedobjectidentificatie in het LRD. Dit is een stap in de richting van een kadaster, maar wel lokaal geïmplementeerd en op basis van plaatselijke condities. Dit heeft tot gevolg dat deze verschillende lokale kadastrale informatiecentra intern onvergelijkbaar zijn voor wat betreft hetgeen geregistreerd is (zoals attributen, kwaliteit, metadata, etc.).

Pas nu is er een zekere tendens bij de regering waarneembaar om de ruimtelijke begrenzing van percelen bij de centrale grondregistratie in IJsland ('Land Registry in IJsland' (LRI)) onder te brengen. Daarbij wordt actief gezocht naar ideeën en oplossingen. Hierbij is het 'Core Cadastral Domain Model', zoals ontwikkeld als een gezamenlijk initiatief van experts van de Technische Universiteit Delft, het ITC in Enschede en de International Federation of Surveyors (FIG), van groot belang.

Het CCDM werd tijdens de FIG conferentie in Washington, USA in 2002 geïntroduceerd. Het idee is om op basis van het concept van een modelgestuurde architectuur een UML model te creëren, waarbij de kernklassen en -attributen van de kadastrale registratie in de wereld worden gedefinieerd. Hiervoor wordt een gedeelde ontologie (voor gegevensuitwisseling, discussie en systeemontwikkeling) binnen het kadastrale domein verschaft. De doelstellingen worden opgesomd in Lemmen et al (2003, blz. 1):

> Een gestandaardiseerd kern kadaster domein model, dat de administratieve (juridische) vastgoedrechten en de kadastrale geografische informatie in brede zin omvat (meerdoelenkadaster) zal tenminste twee belangrijke doelen dienen:
> 1. het voorkomen van het steeds opnieuw uitvinden en herimplementeren van dezelfde functionaliteit. In plaats daarvan wordt voorzien in een uitbreidbare basis voor een efficiënte en effectieve ontwikkeling van het kadastraal systeem gebaseerd op een modelgestuurde architectuur, en
> 2. het voor betrokken partijen mogelijk maken, zowel binnen een land als tussen verschillende landen, te communiceren op basis van de uit het model voortvloeiende gedeelde ontologie.

De innovatie van het CCDM biedt potentieel om te voorkomen dat de bij de ontwikkeling van een kadaster iedere keer het wiel opnieuw uitgevonden wordt, zoals bijvoorbeeld in het geval van IJsland bij het modelleren van de vastlegging van een ruimtelijke extensie van een perceel in de totale procedure. Dit zou mogelijk zijn als het huidige model van de LRD ingepast kan worden in of afgeleid zou kunnen worden van het CCDM, waardoor er een 'Icelandic Cadastral Model (ICM)' gecreëerd wordt dat zowel de vastgoedrechten als de geometrie van de percelen omvat. Hierdoor wordt een mogelijkheid gecreëerd om gebruik te maken van de ruimtelijke begrenzing, zoals beschreven in het geografische deel van het CCDM als een voorbeeld voor ruimtelijke opslag in het ICM.

Na het modelleren van het LRD wordt in dit onderzoek duidelijk dat het CCDM niet eenvoudig weg uitgebreid kan worden om het LRD te omvatten en om zo een ICM te creëren, maar het komt wel dicht in de buurt. Er wordt een aantal aanbevelingen gedaan om het CCDM te verfijnen naar de IJslandse behoeften.

De belangrijkste bijdrage van het CCDM aan de kadastrale ontwikkeling in IJsland is – behalve de kernklassen en de ontologie – de gezonde modelgestuurde benadering. Tot nu toe kenmerkt de ontwikkeling van het LRD zich door een bottom-up benadering, waarbij externe kant en klare oplossingen worden toegepast, die min of meer vanuit de gegevens worden gestuurd. Door het volgen van de CCDM methodologie kan de te ontwerpen oplossing voor de IJslandse kadastrale registratie een meer top-down en kennisgestuurde, alsmede een transparant en beter gestructureerd karakter krijgen.

In dit onderzoek wordt echter wel duidelijk dat het CCDM initiatief behalve onmiskenbaar voordelen ook zwakke plekken vertoont. Omdat het een universele benadering nastreeft bevat het klassen die bijv. ook nomadische rechten op de grond begeleiden, onafhankelijk of het nu om Rwanda, IJsland of Nepal gaat. Dit heeft een toenemende complexiteit tot gevolg. Dit wordt vooral duidelijk als de verschillende versies van het CCDM worden vergeleken, omdat elke versie meer universeel dekkend probeert te zijn dan de vorige.

Voorgesteld wordt dan ook dat de ontwikkelaars zich richten op de ontwikkeling van een CCDM voor homogene, culturele gebieden, zoals binnen de Europese Unie. Op basis van de opgedane ervaring kan het model worden uitgebreid en vervolgens verfijnd voor andere regio's. Ook lijkt er bij het CCDM sprake te zijn van een controverse tussen de doelen. Het verdient dan ook aanbeveling een scheiding tussen de twee doelen te maken, waarbij twee modellen worden gecreëerd. Het ene model kijkt op een conceptuele manier naar de toekomst (het creëren van een paradigma), terwijl het andere model een gemeenschappelijke grondslag voor huidige kadastrale registratiesystemen bevat en creëert (meer technisch georiënteerd en werkend in voorschrijvende zin).

Deze scriptie beschrijft tevens de implementatie van een kadastraal prototype, waarbij de nadruk ligt op het opslaan van de ruimtelijke begrenzing van percelen op basis van het voorgestelde ICM model. Hiervoor zijn open-source geo-software toolkits gebruikt, waarbij ook geëxperimenteerd is met het in detail uitwerken van ruimtelijke modellen en een mogelijke systeemarchitectuur.

Geconcludeerd wordt dat het ideale ruimtelijk model voor het kadaster een topologisch model is, bestaande uit knopen (hoekpuntmarkeringen), verbindingen (grenzen) en vlakken (percelen). Door gebruik te maken van de in SDBMS ontwikkelde functionaliteit kunnen deze topologische primitieven opgeslagen worden, waarbij de bijbehorende geometrie naar behoefte berekend kan worden. Dit is gerealiseerd door in het prototype de topologische structuur in PostgreSQL/PostGIS SDBMS te construeren.

Een ideale systeemarchitectuur om de organisatorische structuur van de kadastrale registratie in IJsland te vergemakkelijken is gevonden in het optimaliseren van de centrale opslag in het SDBMS, die op afstand door de verantwoordelijke gemeenten bijgewerkt wordt. Aangegeven is dat de directe database verbinding in de gebruikte open-source geo-software toolkits momenteel beter werkt dan de OGC 'Transactional Web Feature Services (WFS-T)', maar dat dit dankzij de voortschrijdende technologie dit snel kan veranderen. Het voordeel van het gebruik van WFS-T is dat het de systeemarchitectuur van heterogene desktop clients toestaat, terwijl een directe database verbinding meer op homogene oplossingen is gericht. Dit geldt vooral wanneer gebruik gemaakt wordt van open-source SDBMS, dat niet op grote schaal door commerciële GIS ontwikkelaars wordt ondersteund.

Tenslotte geeft de case studie aan dat de potentiële bijdrage van open-source software bij de implementatie van de kadastrale registratie in IJsland kan variëren. Gesteld wordt dat verschillende van de tegenwoordig beschikbare open-source software pakketten serieus genomen dienen te worden en een reële mogelijkheid bieden voor de ontwikkeling van een ruimtelijke kadastrale registratie. Voorbeelden zijn o.a. PostgreSQL met PostGIS om ruimtelijke gegevens in de SDBMS op te slaan; MapServer op de Apache HTTP server om ruimtelijke gegevens in een webomgeving te kunnen delen; GeoServer om WFS-T voor ruimtelijke gegevens toegankelijk te maken; en tenslotte uDIG als desktop client die uitgebreid kan worden met diverse op maat gemaakte functionaliteiten voor de toegang tot en het redigeren van de open source code.

Terugkomend op de onderzoeksvraag kan gesteld worden dat de voornaamste conclusie is dat het CCDM en open-source software van onschatbare waarde kunnen voor de ontwikkeling van de kadastrale registratie in IJsland en op verschillende manieren aan deze ontwikkeling kunnen bijdragen, zoals gedetailleerd in deze scriptie is weergegeven.

LIST OF EXAMPLES

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

Abbreviation	**Explanation**
ACID	Atomicity, consistency, isolation, durability
BLOB	Binary large objects
BSL	Berkeley software licence
CEN	European Committee for Standardization ::
CEUS	Journal on 'Computers, Environment and Urban Systems'
CCDM	Core cadastral domain model
CGI	Common Gateway Interface
DDL	Data definition language
DBMS	Database management system
DML	Data manipulation language
FIG	International Federation of Surveyors
GIS	Geographic information system
GML	Geographic markup language
GPL	GNU General public license
GPS	Global positioning system
ICM	Icelandic cadastral model
ISO	International organization for standardization
ITC	International institute for geo-information science and earth observation, Enschede, The Netherlands
JTS	Java topology suit
LAN	Local area network
LGPL	GNU Lesser general public license
LIS	Land information system
LRD	Land registry database (Landskrá fasteigna)
LRI	Land registry of Iceland
LUKR	Land information system for Reykjavík administrative area
MDA	Model-driven architecture
NLS	National land survey of Iceland
OGC	Open geospatial consortium
OO-DBMS	Object-oriented database management system
OR-DBMS	Object-relational database management system
OTB	Research institute for housing, urban and mobility studies at TU Delft
PRA	Public road administration in Iceland
RALA	Agricultural research institute in Iceland
RDBMS	Relationa database management system

Abbreviation	**Explanation**
SDBMS	Spatial database management system
SQL	Standard query language
uDig	User-friendly desktop Internet GIS
UML	Unified model language
WFS	Web-feature services
WFS-T	Transactional web-feature service
WMS	Web-map services
XML	eXtensible markup language

1 INTRODUCTION

The best way to become acquainted with a subject is to write a book about it.
Benjamin Disraeli
British politician (1804 – 1881)

Currently there is no national cadastre available in Iceland defining the spatial extent of parcels as registered in the digital land registry database (LRD). In recent years there have been movements towards accomplishing this. Some municipalities have assumed the responsibility and integrated spatial land ownership data into their own local land information systems, but voluntarily and with their own premises. However, spatial delimitation of all parcels in Iceland is in the making with the Land registry in Iceland (LRI) actively looking for options and solutions. One option is to employ the 'Core Cadastral Domain Model' (CCDM), as proposed by the initiative of experts at the Delft University of Technology, the International institute for geo-information science and earth observations in Enschede and the 'International Federation of Surveyors' (FIG), as a framework to build on. This depends very much on if present land registration in Iceland, which is already in place, fits the model.

It is a prerequisite when modelling a cadastral registration which reflects its law system. Additional premise growing in importance is transparency and accessibility of information. Both need to be considered when designing Icelandic cadastral model (ICM) based on CCDM and cadastre already in place.

The question of transparency and accessibility does further not only apply to local usage but also external usage that will increase parallel with heightening international cooperation and globalisation. With the European Union trying to establish the single market within its boundaries it will probably not be uncommon that a person living in The Netherlands to buy a real property in another member state like Germany, Belgium or possibly Iceland. Some might also want to insure their properties or raise mortgage by companies in their own country. For this to materialise the companies need assurance of the reliability of information regarding the real property in stake. Again, the development of the CCDM is a valuable contribution for standardisation in the cadastral domain.

CCDM can function as a unifier between two cadastral systems by defining the core classes of registration that both systems can be extended of. Query from one system would according to the idea be translated to this mutual core before being returned to the target system in a format it understands. In the target system the query is extended generating results travelling the same route back to the querying system. A bank in The Netherlands, estimating property holdings of one of its clients can thus send a query to foreign cadastral service and retrieve the information by using CCDM.

Additional use of the CCDM, apart from standardising the cadastral domain, is the example it sets for general development of cadastral systems. Countries with its cadastral system in transition can here look at the model as a core to found their own cadastral model on (Lemmen et al 2005).

Is standardisation of the cadastral domain enough to facilitate cross system transactions and queries? To certain extend it is, as long there are homogenous users and applications. With increased heterogeneity the standardisation of data exchange, especially spatial data, becomes important. The CCDM for instance complies to ISO-19107 standard on spatial schema regarding both input and output of spatial data, using GML. It is thus required of local desktop GIS to be able to support this format.

Commercial spatial applications have however up to this date been slow and sometimes reluctant to integrate many of the spatial standards and specifications proposed by ISO and Open Geospatial Consortium (OGC), emphasising on own solutions and innovations (Vries & Oosterom 2005). Software developed under so-called open source licenses have on the contrary been active in this field of development but yet, not much research have been carried out on the applicability of these software on real problems. Companies and organisations still prefer the solutions offered by proprietary GIS developers, as "…what kind of lunatic would entrust their business to a product built by people in their spare time?"[1]

This research was structured and carried out with the information stated above in mind. It discusses the theory and implementation of cadastres, explores land recording in Iceland, the innovation of CCDM and how it can benefit the cadastral development in Iceland. It further introduces open-source spatial applications as alternative to conventional proprietary GIS and shares practical experience of developing generalised cadastral prototype of topological structured parcels, stored using open-source tools.

Next chapter discusses the fundamentals of spatial database management systems (SDBMS) with accent on cadastral registration creating the theoretical and practical background for the implementation of the prototype. Followed is the third chapter, which discusses the CCDM version 4 as presented in Cairo, spring 2005. It explains its main objectives and goals, while also covering principal criticism, both external and from within this research.

Chapter 4 deals with the current land registration practices in Iceland. Covering aspects like: geographical and historical context; notation of what is real property in Iceland and what rights are subjected to it; the organisational setup; transaction procedures; the land registry database; and finally documenting the availability of spatial data to use in future cadastral registration.

Followed is chapter 5 going more into technical implementation of a cadastre and how the CCDM can benefit the current land registry model to realise an ICM covering both registration of real property rights and spatial extent of parcels. The chapter compares the requirements for the ICM to the CCDM and puts forward recommendations for future development.

Open-source and its geo-applications is the subject matter of the sixth chapter. It defines what is meant with open-source, if it can be utilised in commercial purposes, examples of utilisation and how it could benefit cadastral development.

Chapter 7 shares practical experiment of implementing cadastral prototype by using open-source software. It starts with system architecture of the idealised system, before going in more detail into spatial modelling and technical implementation. The last two sections elaborate on potential transaction procedure and also evaluate different system architectures in perspective of ICM.

Conventionally the report ends with summarising main conclusions in chapter eight, with the research question revisited, recommendations brought forward, contribution of work assessed. The report ends with a general discussion and author's experience.

---

[1] This question was raised on PostGIS message board in 2002 (PostGIS 2005, [8])

The findings of this project are diverse. The main conclusion is that present land registration in Iceland complies in general to the CCDM as presented in Lemmen et al (2005), with only few exceptions. The primary one is the need for clear distinction between non-planar and planar partitioning real properties in the model to facilitate separate ownership of buildings and the land they reside on. Other finding in this category includes: the introduction of textual parcels; inclusion of movable properties; and, segregation between registration of positive and negative rights.

This research also concludes that software tool-kits belonging to open-source are becoming a real alternative to proprietary GIS systems. Actually this research proofs that it is possible to implement complete cadastral system only based on open-source products, including servers of spatial data, spatial database management system and clients both for editing and viewing. System architecture is maybe; not as stable as offered by conventional GIS vendors; not as well documented; and with some security flaws (everybody can access the source code), but it can perform sophisticated requests and supports in general better the spatial standards set by ISO or OGC.

Finally, this research cannot offer a total solution to implement spatial delimitation of parcel within the Icelandic cadastral registration system. It however proposes alternatives and considerations that can benefit the development in general. It is for instance argued that parcels should be mapped topologically with nodes (monuments), edges (referring to start and end monuments with optional interpolated intermediate points) and faces (realised polygons) as the topological primitives. Facilitate sporadic development of the system with incremented quality upgrade.

This project has made a contribution in several aspects. English documentation of the Icelandic land registration is scarce and not detailed enough. Foreign scholars can use this report to get familiar to the cadastral situation in Iceland.

It has further influenced the newest version of the CCDM that now includes some of the recommendations put forward here. It moreover shows the potential of open-source software available at present, with topology implemented successfully in PostGIS and with growing interest in open-source solutions among researches at OTB. Finally it has probably contributed a lot in the preparation of the author for his new job, where he becomes part of a team responsible of developing and establishing cadastre in Iceland. This latter has caused that the scope of this research has widened somewhat from its initial objectives as is further elaborated in section 8.4: *Discussion and experience*.

1.1 Research objectives, questions and methods

Initial objectives of this research were simple as laid out in thesis planning (Ingvarsson 2005). With time and research however, they field of interest has widened and consequently the overall research has become bigger and more complex. The objective of this report is to:

- research present land recording in Iceland and construct an Icelandic cadastral model;
- compare it to the CCDM (v4) and see how it fits to the non-geographical part of it;
- use the geographical part of CCDM to add a spatial dimension to the Icelandic cadastral model;
- study the technical implementation of cadastres in general and elaborate on the development of recording spatial extent of parcels in Iceland;
- implement a limited prototype using only open-source software tools, emphasising on parcels registration/mapping;

- analyse the applicability of open-source both in general and to cadastral projects; and,
- document both practical experience of modelling, in context to Icelandic cadastral model and CCDM, and implementing prototype.

These objectives are reflected in the main research question:

In what way can the Core Cadastral Domain Model (CCDM) and open-source software benefit the development of cadastral registration in Iceland?

This main question can be subdivided into several sub-questions:

a) How cadastral systems are technically implemented and what are the main characteristics of cadastral database management systems and transactions?

b) What is CCDM and how can it be adjusted to fit the Icelandic cadastral model? What refinements are needed on the CCDM to complete this task?

c) What is the history of land- and cadastral registration in Iceland, what cadastral data is already available and what are the present cadastral procedures?

d) What would be ideal system architecture when adding cadastral spatial delimitation to present land registration in Iceland?

e) What is open-source, what are its applications and how can the implementation of cadastre in Iceland benefit from it?

f) What is the experience of this research and how can it be utilised for the implementation of cadastre in Iceland and the development of a universal CCDM? What recommendations are relevant in this context?

Sub-questions a) to d) have been answered by in depth desk research and by questioning/interviewing experts in the relevant field of interest. Sub-question e) was partially answered by doing research on the Internet, and partially by downloading diverse open-source software and experimenting available functions with data at hand. Finally a generalised cadastral prototype was constructed to get grip of using open-source geo-applications in practice obeying to the proposed ICM in earlier chapters. The experience of conducting this research along is finally used to answer question f).

The tools used to carry out this research were manifold, with the objective to use mainly free or open-source licensed software whenever possible. In this context e.g. IrfanView was used to edit images, SQL Manager 2005 Lite for PostgreSQL editing, while the community versions of XML Spy (freeware) and MagicDraw was used to view and manipulate respectively XML/GML and UML diagrams. Some proprietary software where though used and experimented with like: ArcGIS and GeoMedia to manipulate spatial data and Macromedia Freehand to draw figures. The report is written in Microsoft Word using Garamond 12pt for main text, with UML and SQL notation expressed with Courier New.

References are cited using author surnames and year of publication. In the case of more than three authors, only the first one is cited with the addition of et al. This should correspond to the list of detailed reference found in the bibliography. Web references are cited employing brackets with a number in, e.g. (PostGIS 2005, [1]), pointing to corresponding web reference in bibliography. This is done to facilitate many URL of same origin. Finally notations like no.73/1997 are used throughout the report when citing Icelandic law (2005, [1]), indicating law number and year of issue.

The models presented in the report are based on Unified Model Language (UML) notation. More information can be accessed in Brown (2002): 'An introduction to object-oriented analysis: objects and UML in plain English' and a short but much to the point UML introduction in Stoter (2004). In brief it is good to acknowledge that the design of UML model goes through three phases: conceptual model (1), logic model (2) and to become physical model (3). A conceptual model emphasises the broad picture, what classes exist and how the relationship is between them. It does not necessarily cover all attributes of the classes and relevant constraints, as this is the purpose of the logical model and this is the phase where the database structure is designed. Data model can be categorised into relational, object-relational and object oriented models. Finally the physical model phase is where the data model is translated in accordance to system architecture, defining e.g. storage, indexing, query processes and access paths. These phases are reflected in the models presented in the report.

1.2 Considerations and limitations

As far as known here, limited research has been carried out in this field in Iceland. Tom Barry, geo-expert at LRI has being researching various implementation possibilities for nationwide cadastre in Iceland and National land survey (NLS) has furthermore done some experiments in collaboration with LRI. These projects have brought verification on the need to establish spatial mapping of parcels parallel to land registration in Iceland (Barry 2005; Sigurbjarnarson 2000).

Listed below are limitations that this project came across or were set to restrict the scope of it:

- No exhaustive model was found documenting the land registration in Iceland and therefore is the model presented in the report based on interpretation and reverse-engineering of diverse references like: public law; regulations; user's guides to the land registry clients; land registry web-interface; and interviews. The model should deliver the message how land registration is arranged in Iceland but not necessarily reflect it precisely in the way that it could replace it without any problems.
- No technical security matters will be considered in the research or as a part of the solution, e.g. security of transactions, though this subject should be regarded of high priority when designing the real cadastre. The solution sought is most likely not so different from those used in other systems, e.g. banking.
- Physical model will be restricted to designing architecture to set up prototype, but performance and reliability issues will be neglected to large extent.

1.3 Use of terms and concepts

The exact definition of the term cadastre does not exist. It ranges from being very wide, like representing registration of owners, rights and parcels to very narrow as given be Henssen, only covering the spatial delimitation of parcels. International federation of surveyors (FIG) for example look at cadastre as synonym for what Henssen refers as land recording (1995, [1]):

> Cadastre is normally a parcel based, and up-to-date land information system containing a record of interests in land (e.g. rights, restrictions and responsibilities). 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.

This difference in ontology of terms and concepts complicates the overall discussion of development of cadastres and a clear stance has to be taken from the start of this report. This research uses the concept cadastral registration and cadastre as synonym for land recording. Land recording comprises: land registration, referring only to registration of rights, responsibilities and restrictions; and, cadastral spatial data, storing the geometrical extent and delimitation of the main cadastral entity: parcel.

2 CADASTRAL SDBMS

Statement four in Cadastre 2014 claims that future and modern cadastres will be maintained in a highly digital environment contrary to the old-fashioned paper and pencil cadastres (Kaufmann & Steudler 1998, p.22). This chapter is intended to give overview of the technical concepts and solutions for digital cadastres. Databases have emerged as the solution to manage cadastral data with the chapter coverage consequently focusing on spatial database management systems (SDBMS) and its potential, from storage of spatial data to transactions and data exchange.

This chapter starts with section 2.1 giving general overview of what is DBMS before emphasising more on the fundamentals of SDBMS in section 2.2. Modelling spatial objects is the subject matter of section 2.3, while section 2.4 covers general aspects of system architecture and maintenance of cadastral SDBMS. SDBMS transactions, consistency checks and concurrency control are the topics discussed in section 2.5, before concluding this chapter with remarks section 2.6

2.1 DBMS in a nutshell

DBMS are one of the great evolutions that the computer revolution has brought to the modern societies. Wherever we look we see DBMS in action, when shopping, withdrawing money from bank automat or looking up a telephone number on the Internet.

With DBMS huge quantities of data can be stored at one place and queried with simple methods. Here is good to keep in mind that data is not equal to information with the latter regarded as processed data with added value and DBMS are effective tool to oversee such processes. The definition of DBMS, as given by Rigaux et al (2002, p. 4) is:

> …a collection of software that manages the database structure and controls access to data stored in a database. .

DBMS are furthermore ideally thought of facilitating several processes, such as (Rigaux et al. 2002, p. 4):

- Defining a database; that is specifying data types, structures and constraints.
- Constructing a database; that is storing the data itself into persistent storage.
- Manipulating a database.
- Querying a database to retrieve specific data.
- Updating a database (changing values).

To do this most DBMS uses 'Structured Query Language' (SQL) that was initially developed by IBM but has since become the standard query language for relational DBMS (RDBMS), the most common used form of DBMS.

RDBMS are regarded as effective tool to store and manipulate simple data that is organised in tabular form using unique keys to join or relate different tables together. However, as RDBMS only stores simple data types, alternative solution had to be invented to handle complex data types like spatial data. Here was the solution found in object-oriented approach and the innovation of object-oriented DBMS (OO-DBMS). This technology has however not gained as much popularity as first expected, but guided the development of object-relational DBMS (OR-DBMS) that somewhat combines the functionalities of the two approaches (Shekhar & Chawla 2003).

The following subchapters will go more in depth to SQL and the elements of DBMS. However, only the object-based model (vector) is discussed but not the field-based model (raster) that does usually not directly apply to the implementation of cadastral systems.

2.1.1 SQL

SQL can be regarded as composition of several separate components. Two of the most important are 'Data Definition Language' (DDL) and 'Data Manipulation Language' (DML) (Shekhar & Chawla 2003; Rigaux et al. 2002).

DDL is used to define for an instance the data, tables, constrains and association. **CREATE**, **ALTER** and **DROP** are examples of DDL statements as illustrated below:

```
CREATE TABLE person (person_id char(10) PRIMARY KEY, name
  varchar(50));
ALTER TABLE person ADD COLUMN address varchar(255);
DROP TABLE person;
```

DML is in contrary used to access and edit data in a database and perform operation like **INSERT**, **UPDATE**, **DELETE**, and **SELECT**. Thus, manipulating the data already defined by DDL. Examples of DML statements in PostgreSQL are e.g.:

```
INSERT INTO person (person_id, name, address)
  VALUES ('0409773259', 'Tryggvi Már', 'Den Haag');

UPDATE person SET name = 'Tryggvi Már Ingvarsson'
  WHERE person_id = '0409774449' ;

SELECT * FROM person;
SELECT name,
    char_length(name) AS name_length,
    address,
    char_length(address) AS address_length
  FROM person;

DELETE FROM person WHERE person_id = '0409774449';
```

Example 1 shows the results of one of the **SELECT** statements above.

Example 1: In select statement, temporal columns can be created that display the results of specific functions.

```
dbms=# SELECT name,char_length(name) AS name_length,address,char_length(address) AS
address_length FROM person;

         name          | name_length |  address  | address_length
-----------------------+-------------+-----------+----------------
 Tryggvi Már Ingvarsson |          22 | Den Haag  |              8
 (1 rows)
```

2.1.2 Conventional DBMS data types

There are several primitive data types typically provided by default in most databases, although their naming can vary. Table 1 provides overview of database data types, its description and naming comparison between two popular databases provided under open-source license, MySQL and PostgreSQL.

Table 1: List of common default data types in DBMS (PostgreSQL 2005; MySQL 2005).

Data type name	**PostgreSQL**	**MySQL**	**Description**
Integer	int, int4	int	-2,147,483,647 to 2,147,483,647
Small Integer	int2	smallint	-32,768 to 32,767
Big Integer	int8	bigint	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
Float / Real	float, float4	float	32 bit floating point
Double	float8	double	64 bit floating point
Auto-incrementing number	serial	auto_increment	4 bit integer
Character	char	char	fixed-length character string (0-255)
Varying character	varchar	Varchar	variable-length character string (0-255)
Character string	text	Text	variable-length character string (0- 65,535)
Boolean	boolean, bool	boolean, bool	zero = false / non-zero=true
Time	time	Time	time of day
Date	date	Date	calendar date (year, month, day)
Year		Year	4-digit format allowing (0, 1901-2155)
Timestamp	timestamp	Timestamp	date and time
Blob (Binary large objects)	bytea	Blob	binary data ("byte array")

2.2 Spatial DBMS

Spatial data types are not naturally provided by RDBMS and necessity to perform considerable work to store simple spatial object in such databases. Hence, spatial data types are thought as complex while the data types listed in Table 1 is considered of simple type. There are limited possibilities to bypass this in RDBMS as the set of data types are restricted, and there are no geometric data types like point, line or polygon defined by standard. Example 2 illustrates how polygon geometry can be expressed in RDBMS. Special program would be needed to visualise this geometry.

Example 2: Topology as represented in relational DBMS.

```
Relation: land_parcel
land_id |      land_name      |   land_owner    | boundaries
---------+-------------------+----------------+----------
 100     | Sellafield         | John Smith      | 1010
 101     | Chernobyl          | Bill Bob        | 1020
 102     | Three Mile Island  | Claire Hepburn  | 1030
(3 rows)
```

```
Relation: land_extent
boundary_id | edge_id
------------+---------
 1010        | A
 1010        | B
 1010        | C
 1010        | D
(4 rows)
```

```
Relation: points
point_id | x_coor | y_coor
---------+--------+--------
 10001    |      0 |       0
 10002    |      0 |       2
 10003    |      2 |       2
 10004    |      2 |       0
(4 row)
```

```
Relation: edges
edge_id | point_id
---------+--------
 A       | 10001
 A       | 10002
 B       | 10002
 B       | 10003
 C       | 10003
 C       | 10004
 D       | 10004
 D       | 10001
(8 rows)
```

The way that the land parcel is defined in Example 2 is however not very convenient in practice. It is hard to implement constrains to maintain consistency of the data and there are no readymade functions or operators available in the RDBMS that can be applied to the data to calculate area, lengths, adjacency etcetera. These drawbacks are the reason for the popularity of object-oriented approaches when storing spatial data in spatial DBMS (SDBMS).

The object-oriented paradigm offers more relaxation on defining data types compared to RDBMS, giving way to the possibility of customised spatial types with own specific function and behaviour. The initial paradigm gave reasons to expect that OO-DBMS would completely overtake RDBMS but that has not been the reality. Shekar & Chawla (2003, p.8) suggest two principal reasons for that:

- the market adoption of OO-DBMS have been limited, restraining much needed capacity for bettering the approach, and consequently causing GIS users to look for other more stable solutions; and
- the SQL is language Franca in most databases and highly incorporated in the relational database model.

SQL is a declarative language in the way that it focuses more on desired results of user specification than the means of production. As a consequence has OO-DBMS not overtaken RDBMS but rather the former adjusted to the latter, resulting in the so-called object-relational databases, OR-DBMS. Example 3 illustrates how simple polygon can be stored in an OR-DBMS. Here is an object, a polygon, stored as a column variable within the traditional RDBMS.

Example 3: Geometric objects stored in a OR-DBMS.

```
Relation: Parcel
parcel_id |    parcel_name    |  parcel_owner  |        parcel_geometry
----------+-------------------+----------------+------------------------------
 100      | Sellafield        | John Smith     | POLYGON((0 0,0 2,2 2,2 0,0 0))
 101      | Chernobyl         | Bill Bob       | POLYGON((0 2,0 4,2 4,2 2,0 2))
 102      | Three Mile Island | Claire Hepburn | POLYGON((2 0,2 4,4 4,4 0,2 0))
(3 rows)
```

As stated earlier, the object-oriented approach loosens the restriction on data types in OR-DBMS creating the opportunity to define user specific data types. Example 4 illustrates how to define 'rectangle data type' in PostgreSQL. Functionality of this type is further enhanced with the SQL3 standard that brings RDBMS further in the object-oriented direction (Shekhar & Chawla 2003). OR-DBMS offer own (or integration of) programming language to make even more sophisticated functions than supported by SQL.

Example 4: Use of customised defined data type in PostgreSQL OR-DBMS using SQL.

```
--creates data type rectangle with two sides: x & y and two simple functions
CREATE TYPE rectangle AS (x_side int4, y_side int4);

CREATE FUNCTION rect_area(rectangle) RETURNS int4 AS $$
       SELECT $1.x_side * $1.y_side;
       $$ LANGUAGE SQL;

CREATE FUNCTION rect_perimeter(rectangle) RETURNS int4 AS $$
       SELECT (2 * $1.x_side) + (2 * $1.y_side);
       $$ LANGUAGE SQL;

--creates table schema: rectangles and populates it
CREATE TABLE rectangles (id serial, rect rectangle);
INSERT INTO rectangles (rect) VALUES ((2,5));
INSERT INTO rectangles (rect) VALUES ((1,3));
INSERT INTO rectangles (rect) VALUES ((4,4));
INSERT INTO rectangles (rect) VALUES ((3,7));

--selects and temporarily calculates and creates columns for display
SELECT id, rect, rect_area(rect), rect_perimeter(rect) FROM rectangles;

Relation: Rectangles
id  | rect  | rect_area | rect_perimeter
----+-------+-----------+----------------
  1 | (2,5) |        10 |             14
  2 | (1,3) |         3 |              8
  3 | (4,4) |        16 |             16
  4 | (3,7) |        21 |             20
(4 rows)
```

With SQL3 a support for object-oriented structures is brought into the SQL language giving alternatives for (Manola & Sutherland 1997, [1]):

- user-defined types (abstract data types, named row types, and distinct types);
- type constructors for row types and reference types;
- type constructors for collection types (sets, lists, and multisets);
- user-defined functions and procedures; [and]
- support for large objects (BLOBs and CLOBs)[2].

2.2.1 DBMS advantages over file system

There are other ways to store and retrieve spatial information than offered by SDBMS. For example in file format like e.g. provided with ESRI shape file (SHP) and Drawing eXchange Format (DXF) used in CAD systems. However, the concept of SDBMS offer an advantage that no other file based solution offers, that is (Shekhar & Chawla 2003):

- storing large amounts of diverse data structured and associated in a way that is hard to implement in file systems;
- to process complex set-based queries on the data with near instant results (e.g. routing); and,
- provide concurrency control, like e.g. locking mechanism and consistency checks, enabling multi-users.

Given this, SDBMS offers much sophisticated solution, depending on software of course, for GIS projects than the classic GIS applications. GIS software vendors have though more and more adapted to the concept of SDBMS and many can connect to SDBMS, acting as an interface to the underlying data.

[2] CLOB is here 'character large object' used over large file of characters stored as part of a database record.

2.2.2 Spatial data types

Through the years there has been disagreement on how to model geographic features to spatial objects in databases. For instance it depends on what OR-DBMS solution is used, which spatial data types are available to use. However, the OpenGeospatial Consortium (OGC) specification from 1999 on simple features established some consensus on this matter by creating a paradigm for database vendors to follow when managing two-dimensional spatial data. See Figure 1.

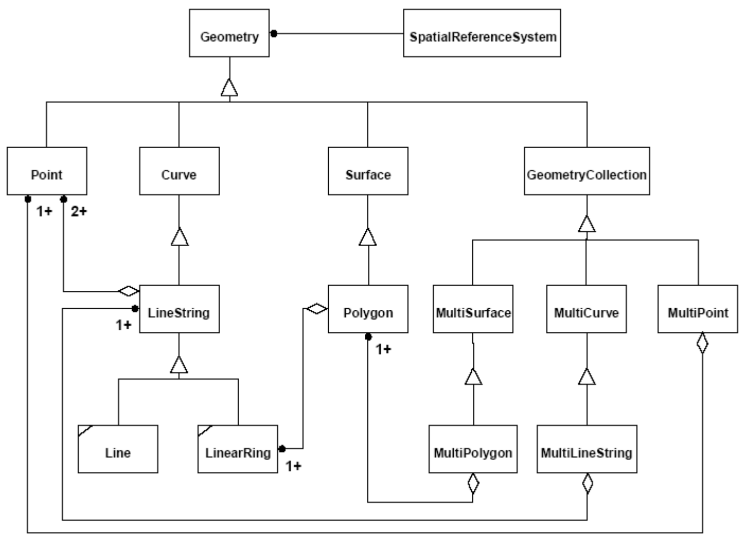


Figure 1: Geometry class hierarchy as presented in the OGC Simple Feature Specification (image: 1999, p.2-2).

The simple feature model has since been adapted and developed further by the international standardisation authorities ISO, e.g. in ISO/TC211-19107: 'Geographic Information Spatial Schema' (ISO 2003).

The discussion on spatial standards will start by introducing the OGC simple feature specification that is currently 'the standard in action' in most spatial enabled databases. Finally, the development of ISO/TC211-19107 standard is described from author's viewpoint, which is expected to replace the OGC specification in implementation of geometric features, in not so distant future.

Briefly, the OGC model can be defined as collections spatial objects, referred as **`Geometry`**, associated to spatial extent in space, defined in spatial reference system.

The **`Geometry`** class is abstract, with four subclasses considered as geometric primitives: **`Point`**, **`Curve`**, **`Surface`** and **`GeometryCollection`**. Except **`Point`**, these classes are also abstract in the sense that they are here to provide methods to their subclasses and some freedom for extending the model in the future.

`Point` represents a 0-dimensional object located in space. Linear interpolations of at least two **`Points`** are required to create a 1-dimensional **`LineString`** that is specialization of class **`Curve`**. Although not defined here, **`Arc`** or **`Spline`** could also be considered member of **`Curve`** as is defined in ISO/TC211-19107.

The simplest **`LineString`** is a straight **`Line`** segment that only connects two **`Points`**. If the **`LineString`** is simple (i.e. does not cross itself) and closed with it's begin and end points connected, it is considered to be **`LinearRing`**. At least one **`LinearRing`** is necessary to define the exterior boundary of a 2-dimensional **`Polygon`**, a specialization of the geometric primitive **`Surface`**. Extra **`LinearRings`** define interior boundaries of the **`Polygon`**. Again, other classes than **`Polygon`** can be considered as specialization of **`Surface`**, e.g. **`TriangulatedSurface`** as presented in ISO/TC211-19107.

A collection of one ore more **`Points`**, **`LineStrings`** or **`Polygons`** can be stored in respectively: **`MultiPoint`**, **`MultiLinestring`** or **`MultiPolygon`**. There of **`MultiLineString`** and **`MultiPolygon`** are specializations of **`MultiCurve`** and **`MultiSurface`** respectively, with everything a specialization of the geometric primitive **`GeometricCollection`**. To understand this fully an example is needed. An island can be represented by **`Polygon`**. A group of islands comprising an administrative unit could be stored in a **`MultiPolygon`**. However when mapping the complete set of administrative units, i.e. several groups of islands, a **`MultiSurface`** becomes useful (Kresse & Fadaie 2004; OGC 1999; Shekhar & Chawla 2003). More detailed definition on spatial data types of the simple feature model is given in: *Appendix A: Spatial Data Types.*

The scope of object definitions increases considerable with the introduction of the ISO/TC211-19107 standard: 'Geographic information – Spatial schema' (ISO 2003). Especially by including three-dimensional features based on **`GM_Solid`** as part of the model. There are though some familiarities to the OGC specification on simple features. See Figure 2.

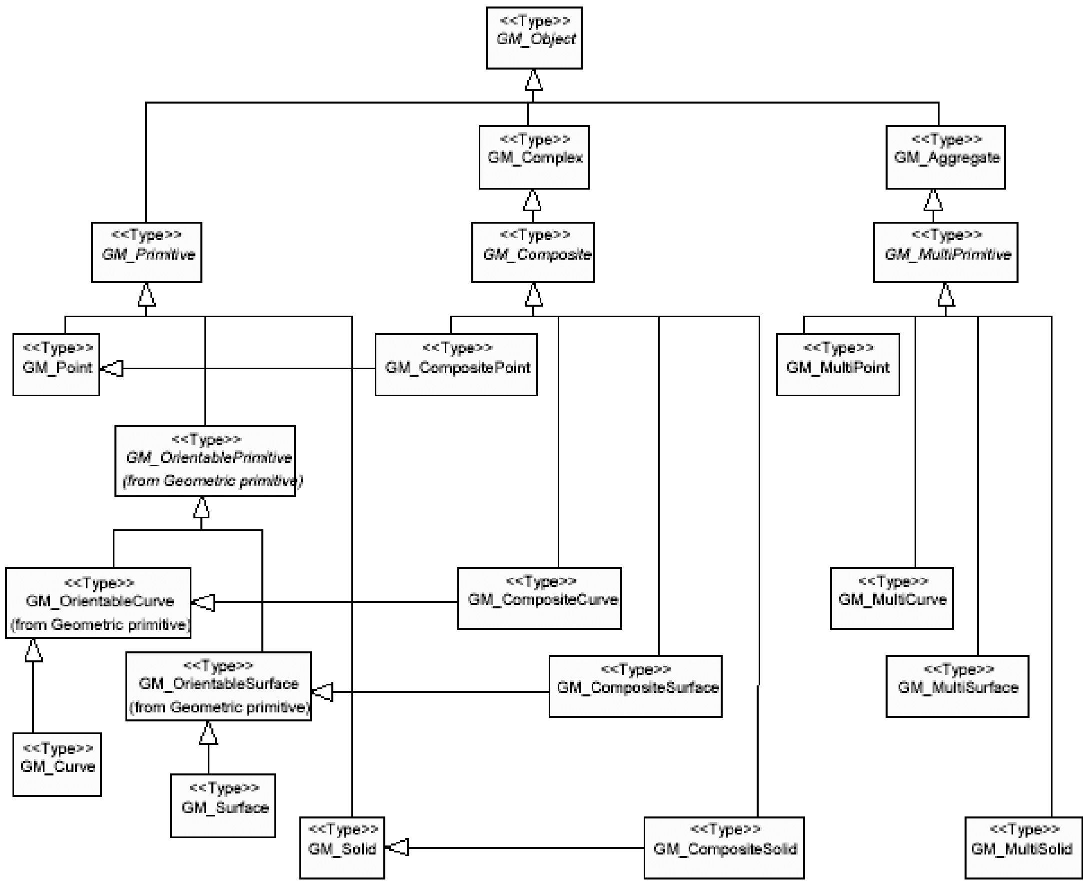


Figure 2: Geometric basic classes with specialisation relation (image: ISO 2003, p.24).

The **`Geometry`** class from OGC specification has been replaced by **`GM_Object`** that has three subclasses: **`GM_Primitives`**, **`GM_Complex`** and **`GM_Aggregate`**. As specialization of **`GM_Primitives`**: **`GM_Point`**, **`GM_Curve`** and **`GM_Surface`** can be recognized as similar as **`Point`**, **`Curve`** and **`Surface`** in the OGC specification, though orientation has been added.

`GM_Complex` stores sequence of **`GM_Primitives`** e.g. as in **`GM_CompositeCurve`** that stores curves in the way "that each curve (except the first) starts at the end point of the previous curve in the sequence" (ISO 2003, p.5).

`GM_Aggregate`, finally, functions similarly as **`GeometricCollection`** in the OGC specification. One interesting fact is though that the spatial extent is not associated with root class, **`GM_Object`**, but to **`GM_Primitives`**. This indicates that **`GM_Complex`** and **`GM_Aggregate`** are more like feature containers than primitives, with spatial reference dependence on the class **`GM_Primitive`**.

By looking to one of the specializations of **`GM_Primitives`** one can observe the increased complexity in the ISO feature model from the OGC specification. Figure 3 shows for an instance that **`LineString`** is not longer the only subset of a **`Curve`** but one of many, e.g. **`ArcString`** and **`SplineCurve`**.

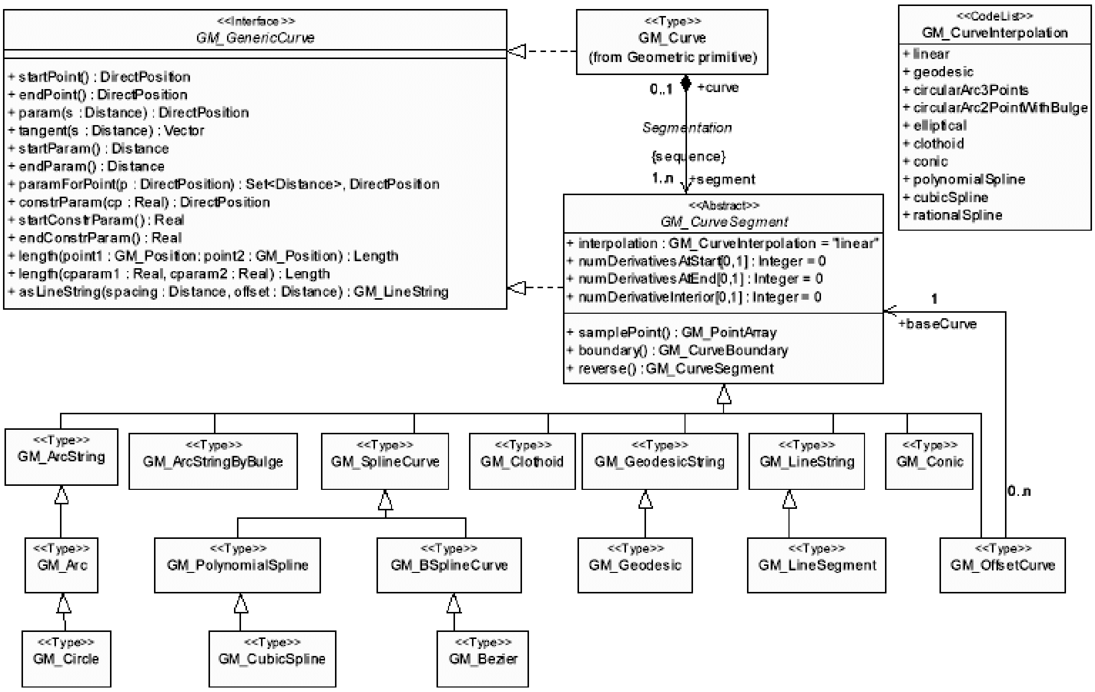


Figure 3: Curve segment classes (image: ISO 2003, p.50).

2.2.3 Relationships between spatial objects

2-dimensional objects, as polygons, lines and points, can have 512 binary relationships between each other depending on the intersection of interior (°), boundary (б) and exterior (¯). This can be described with the nine-intersection matrix where the topological relationship between two geometries, surfaces A and B are compared (Figure 4).

$$R(A,B)=\begin{pmatrix} A^{\circ}\cap B^{\circ} & A^{\circ}\cap \partial B & A^{\circ}\cap B^{-} \\ \partial A\cap B^{\circ} & \partial A\cap \partial B & \partial A\cap B^{-} \\ A^{-}\cap B^{\circ} & A^{-}\cap \partial B & A^{-}\cap B^{-} \end{pmatrix}$$

Figure 4: The dimensionally extended nine-intersection matrix (image: Shekar & Chawla 2003, p.28).

By observing the output of this matrix, eight relations can be realised describing the intersection of two surfaces: *disjoint*, *meet (thouches)*, *overlap*, *equal*, *contains*, *inside*, *covers* and *covered by*. Figure 5 illustrates this. The matrix can similarly be used over relationships between spatial types of different kind and dimension, e.g. line versus line, point versus surface, point versus line, and line versus surface etcetera (Shekar & Chawla 2003, p. 28-30).

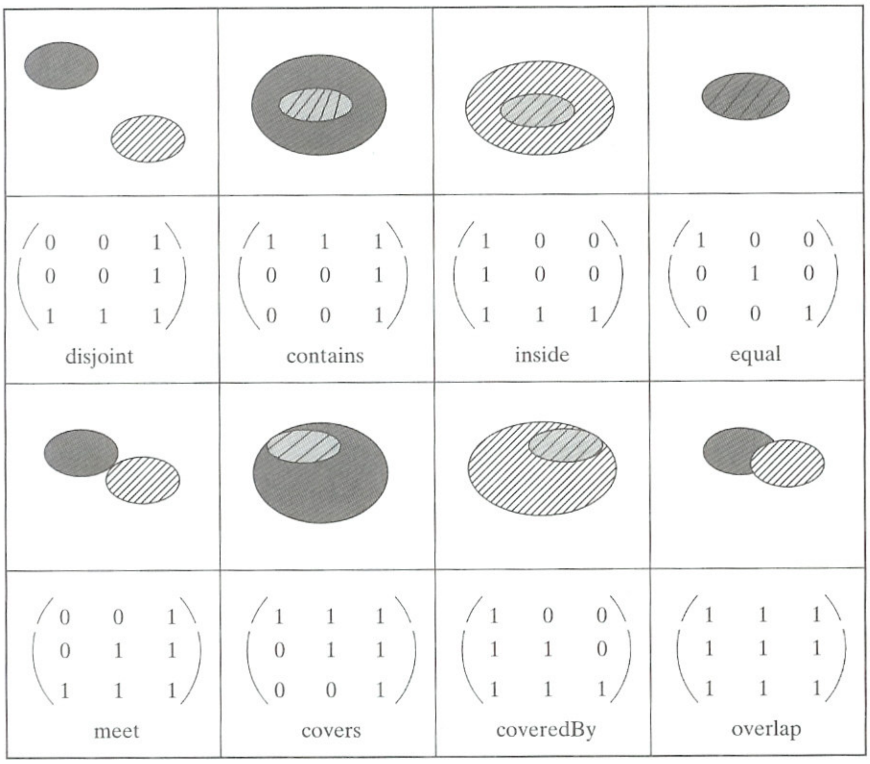


Figure 5: Illustration of the nine-intersection model (image: Shekar & Chawla 2003, p.30).

The topological relationships presented in the nine-intersection matrix can be used in SQL queries along with several non-topological operations, like:

- Euclidian distance(geometry, geometry): Double
- Direction (geometry, geometry): String, e.g. north, east, south, west, left or right.
- Length (curve): Double
- Area (surface): Double
- Perimeter (surface): Double

This is not an exhaustive list, as the number of topological and non-topological operations for spatial data is enormous, especially given the fact that only two-dimensional geometries have been covered here. Three-dimensional relationship of geometries is still very much a topic of research. Next section will cover how this can be used to construct spatial queries.

2.2.4 Spatial queries

SQL queries on spatial data are not so different from those on non-spatial data, only with spatial data types, functions and operators added.

There are two types of spatial queries: static query and dynamic query. A static query only observes the spatial objects and returns a result without affecting the objects queried. Example of static query is for example: measure area of surface returning a double. Dynamic queries are different from static in the way that they affect the data it self. Examples are e.g.: merge, split, rotate, resize and copy (Shekhar & Chawla 2003).

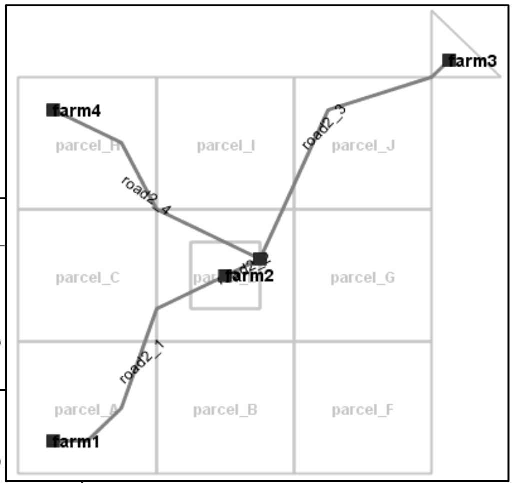
Three examples of spatial queries are shown here, all static, using data as shown in Example 5.

Example 5: Data used in the spatial queries examples.

```
  name | geometry
-------+----------------------
 farm1 | MULTIPOINT(1 1)
 farm2 | MULTIPOINT(6 6,7 6.5)
 farm3 | MULTIPOINT(12.5 12.5)
 farm4 | MULTIPOINT(1 11)
(4 rows)
```

```
  name   | geometry
---------+--------------------------------------
 road2_1 | LINESTRING(1 1,2 1,3 2,4 5,6 6)
 road2_2 | LINESTRING(6 6,7 6.5)
 road2_4 | LINESTRING(7 6.5,4 8,3 10,1 11)
 road2_3 | LINESTRING(7 6.5,9 11,12 12,12.5 12.5)
(4 rows)
```

```
  name    | geometry
----------+------------------------------------
 parcel_A | MULTIPOLYGON(((0 0,0 4,4 4,4 0,0 0)))
 parcel_B | MULTIPOLYGON(((4 0,4 4,8 4,8 0,4 0)))
 parcel_C | MULTIPOLYGON(((0 4,0 8,4 8,4 4,0 4)))
 parcel_D | MULTIPOLYGON(((4 4,4 8,8 8,8 4,4 4),
   (5   5,5 7,7 7,7 5,5 5)))
 parcel_E | MULTIPOLYGON(((5 5,5 7,7 7,7 5,5 5)))
 parcel_F | MULTIPOLYGON(((8 0,8 4,12 4,12 0,8 0)))
 parcel_G | MULTIPOLYGON(((8 4,8 8,12 8,12 4,8 4)))
 parcel_H | MULTIPOLYGON(((0 8,0 12,4 12,4 8,0 8)))
 parcel_I | MULTIPOLYGON(((4 8,4 12,8 12,8 8,4 8)))
 parcel_J | MULTIPOLYGON(((8 8,8 12,12 12,12 8,8 8)),
   ((12 12,12 14,14 12,12 12)))
(10 rows)
```



Notice that parcel_D has an enclave, defined as parcel_E. Also that parcel_J is composed of two touching parcels (a square and a triangle).

Example 6: Calculates the total length of the road network:

```
SELECT sum(length(geometry)) FROM roads;

       Sum
-----------------
 25.5506444522934
(1 row)
```

Example 7: Observes which parcels road2_1 crosses.

```
SELECT r.name,p.name FROM roads r,parcels p
WHERE crosses(r.geometry,p.geometry)
AND r.name = 'road2_1';

  name   |   name
---------+----------
 road2_1 | parcel_A
 road2_1 | parcel_C
 road2_1 | parcel_D
 road2_1 | parcel_E
(4 rows)
```

Example 8: Lists all neighbours of particular parcel along with their area size.

```
SELECT p.name, count(p1.name) AS num_neighbours, area(p.geometry) AS size
       FROM parcels p, parcels p1
       WHERE Touches(p.geometry,p1.geometry) AND p.name != p1.name
       GROUP BY p.name, p.geometry
       ORDER BY num_neighbours DESC;

   name   | num_neighbours | size
----------+----------------+------
 parcel_D |              9 |   12
 parcel_C |              5 |   16
 parcel_B |              5 |   16
 parcel_I |              5 |   16
 parcel_G |              5 |   16
 parcel_F |              3 |   16
 parcel_A |              3 |   16
 parcel_H |              3 |   16
 parcel_J |              3 |   18
 parcel_E |              1 |    4
(10 rows)
```

2.2.5 Spatial indexing methods

Accessing and indexing spatial data stored in a database is more complex than accessing textual data, though both methodologies are often theoretically based on the so-called B-tree indexing method. Indexing is important aspect of SDBMS, to speed up queries and in locking mechanisms.

The simplest form of index is frequently found in hardcopy books with indexes on last pages to guide readers to specific content they seek. It simple connects terms and words to pages in the book. In databases storing textual data this implemented similarly through unique identifier assigned to specific entry. The identifiers are then categorised in logical groups of similar value, e.g. according to the B-tree method. A definition of B-tree is (ITTIA 2005, [1]):

> B-tree is a fast data-indexing method that organizes the index into a multi-level set of nodes. Each node contains a sorted array of key values (the indexed data). Two important properties of a B-tree are that all nodes are at least half-full and that the tree is always balanced (that is, an identical number of nodes must be read in order to locate all keys at any given level in the tree). A well-organized B-tree will have only three or four levels.

B-tree indexing is useful when retrieving spatial data based on its attribute values. Different approach is however needed when using spatial queries, e.g. window or point queries. Example of B-tree index structure is shown in Figure 6.

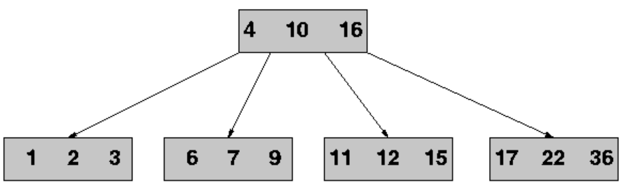


Figure 6: B-tree hierarchical structure (image: Author).

The problem with querying spatial data is that a common query, like a point query, would need to compare and check the point location with the geometry of every object in the database, which is both time and memory consuming if the database is large. Spatial indexing was developed to resolve this.

Common to some spatial indexing methods is to store approximation of geometry in the so-called minimum-bounding box, sometimes referred as an envelope. Spatial query is furthermore considered as a two-step process, filter step and refinement step.

The filter step compares the spatial query to the envelopes of the objects in the database and filter those out that do not intersect with the query. This saves a lot of processing time as the geometry of the envelopes is likely to be much simpler to process than the object geometry. The refinement step processes thereafter the spatial query on the filtered objects, where as some are dropped out, where geometry does not intersect with the spatial query though its envelope did. Finally the correct objects are returned as a result of the spatial query.

The latter step of refinement is rather uniform in all spatial indexing methods. It is however the filtering-step that differentiates between different methodologies, as they are conventionally categorised into either 'space-driven' or 'data-driven' approaches (Rigaux et al. 2002).

In a space-driven approach the 2-dimensional planar space is partitioned into number of rectangles that are independent of the objects they serve. The objects are then mapped to cells according to geometric criteria that differ somewhat considering what method is used. Most popular space-driven methods are named 'grid-file', 'linear-quadtree' and 'z-ordering tree'.

A data-driven approach on the other hand, focuses on the objects and in partitioning them into appropriate/logical groups considering number and distribution in space. Most popular data-driven methods is the 'R-tree' and its reformed versions 'R+tree' and 'R* tree' (Rigaux et al. 2002; see also Guttman 1984).

The structure of R-tree (Figure 7) is similar to B-tree introduced earlier and is characterised by several properties (Rigaux et al. 2002, p.238):

1. For all nodes in the tree (except the root), the number of entries is between *m* and *M*, where *m* ϵ[0, *M*/2]
2. For each entry (*dr*, *nodeid*) in a non-leaf node *N*, *dr* is the directory rectangle of a child node of *N*, whose page address is *nodeid.*
3. For each leaf entry (*mbb*,*oid*), *mbb* is the minimal bounding box of the spatial component of the object stored at address *oid.*
4. The root has at least two entries (unless it is a leaf).
5. All leaves are at the same level.

With the addition of (Shekhar & Chawla 2003, p.99-100):

6. All *mbb* have sides parallel to the axis of a global coordinate system.

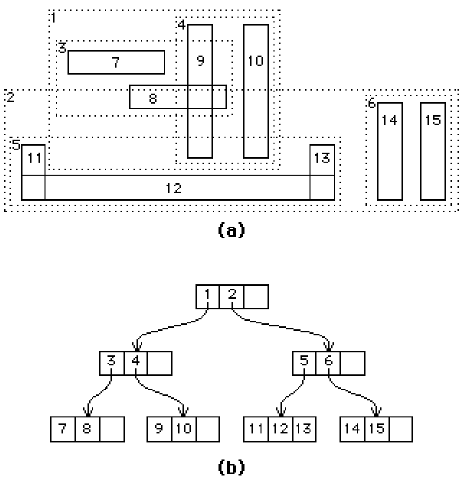


Figure 7: R-Tree (image: Rubin 1994, [1])

2.3 Modelling spatial objects

Up to now the focus has been on how to store individual objects in a database and how to query them, but not on the presentation of collection of objects. How object of the same collection interrelate and respond to queries and editing.

This section goes into more detail into spatial modelling, thus how to model features like parcels or cadastral boundaries to database data types. Two models are discussed, spaghetti and topological and how they can be realised in a SDBMS.

2.3.1 Spaghetti model

The notion of 'spaghetti model', indicating no explcit structure, is used if the geometry of spatial features in SDBMS is described completely independent and irrespective of other features in the database, without relationship. Relationships like adjacency, within, outside etcetera, between separate geometries are therefore calculated on demand.

The term 'spaghetti data', or raw data, is often used over data stored in accordance to the spaghetti model. The main characteristics of such data are possible overlapping geometries and dangling lines. This is evident when representing land parcels as spaghetti polygons, whereas each boundary has to be stored twice, and the same corner monument stored at least three times, in different polygons. This creates problem of tracking boundary measurements. Also if geometric data is of different quality, adjacent land parcels can either overlap or be disjoint, but not touching as would be correct.

Primary advantage of storing spatial data according to the spaghetti model is simplicity. Especially when manipulating the data: inserting, deleting or updating, as relational constraints are limited to none-existing.

However, the disadvantages are numerous. Like was stated before, by storing every land parcel as a polygon entails that every parcel boundary is stored twice and every corner monument at least three times. There is consequently difficult to prevent inconsistency and complex queries like: 'what land parcels share boundaries with the queried parcel?', are extremely cumbersome.

The main sources for spaghetti data come from diverse data generation, like: manually digitisation of plans and maps, and; automatic scanning and vectorisation of simple drawings (Bernhardsen 1999; Rigaux et al 2002).

2.3.2 Topological model

> The topological model is one in which the connections and relationships between objects are described independently of their coordinates; their topology remains fixed as geometry is stretched and bent (Bernhardsen 1999, p.60).

A topological model is a planar version of the network model where the planar subdivision delimitates adjacent polygons, optionally linked to actual geographic objects. The two-dimensional domain is thus made of number of directed edges, each connected with a start and end node, compromising a segment with a face (the topological representative of polygon) on its right and left side. Faces are not explicitly stored but derived from surrounding segments. This prevents redundant registration of storing polygon boundary of adjacent areas twice, as is valid for the spaghetti model. Figure 8 presents a diagram of two-dimensional topological model as developed by Molenaar (1998).

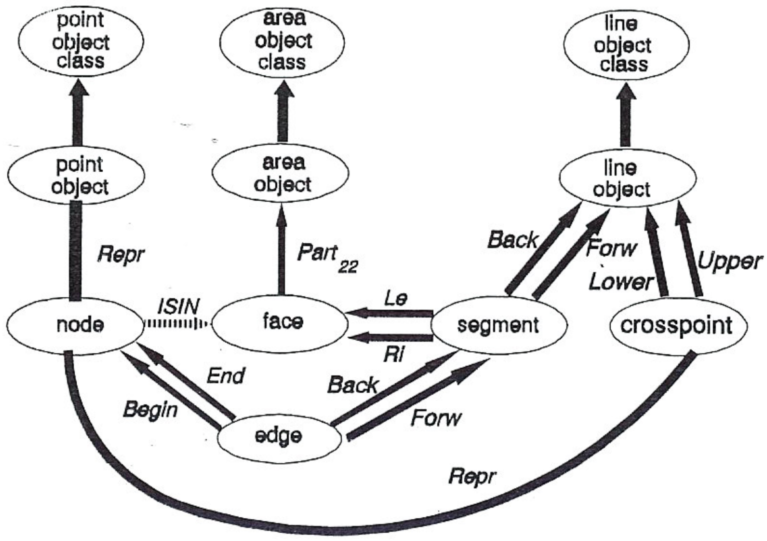


Figure 8: An extended diagram for a topological model of two-dimiensional objects (image: Molenaar 1998, p.101)

Considering cadastres, the nodes represent corner monuments; edges and segments are the boundaries between adjacent parcels, and faces represent the parcels themselves. Crosspoints, as presented in Figure 8, are however not relevant here as they are intended to distinguish between lines that cross but do not actually intersect. For an instance river and road (Quak et al 2003; Molenaar 1998).

The objects of interest in the topological model are here (Molenaar 1998, Rigaux et al 2002):

- Point: x (real), y (real)
- Node: point, [related edges]
- Edge/Segment: node-start, node-end, [sequence of intermediate points], left-polygon, right-polygon
- Face: [sequence of edges]

There are system rules common in the topological model as listed by Worboys (1995, p.194):

- Every directed edge has exactly one start node.
- Every node must be must be either start or end node or both, of at least one edge.
- Every face is bounded by at least one directed edge.
- Edges can only intersect at nodes.
- Every directed edge has exactly one face on its left and right.
- Every face must be the left or right area or both of at least one directed edge.

As the geometry of land parcels is not explicitly stored in a topological model, diverse methodologies have been invented to extract polygon information from set of edges and nodes. One method is referred as the *winged-edge structure*, which is i.e. employed by the Dutch cadastre. Other methods discussed her are *double-connected-edge-list, left-right realisation* and *wheel topology*.

Edge and face are the geometric primitives in the winged-edge method whereas nodes are left out and the geometry therefore directly realised from edges. The attributes of a boundary table, storing edges include:

- object_id: unique identifier for boundaries
- geo_polyline: sequence of points forming the geometry of the boundary
- fl_line_id: first boundary connected on left seen from the start of the directed boundary
- ll_line_id: last boundary connected on left seen from the start of the directed boundary
- fr_line_id: first boundary connected on right seen from the start of the directed boundary
- lr_line_id: last boundary connected on right seen from the start of the directed boundary
- l_parcel: parcel located on the left side of the directed line
- r_parcel: parcel located on the right side of the directed line

Parcels are derived from the boundaries by storing one of the surrounding boundaries as an attribute in the parcel table, using algorithm to extract the remaining exterior and interior boundaries of the parcel. The schema for the parcel table according to the winged-edge structure includes:

- object_id: unique identifier for parcels
- parcel_id: parcel reference identification
- exterior_id: reference to one of the surroundings boundary stored in the boundary table
- interior1_id: reference to one of the boundaries of first exterior, stored in boundary table

If there are more interiors than one, an additional 'parcelover' table is used. The parcelover table has schema as follows:

- object_id: reference to relevant parcel
- interior1_id: reference to one of the boundaries of second exterior
- interior2_id: reference to one of the boundaries of third exterior
- …
- interior10_id: reference to one of the boundaries of eleventh exterior

To generate geometry for a parcel, the algorithm starts with reading the exterior boundary referred and by going counter clockwise, systematically reverts the directed edges (the polygon needs to be on the left side of the edge, if not stored that way it can be reverted) so each of them points to the start of next one. The algorithm discovers the orientation of the edges by looking at how the edge is connected to other edges. If an edge is reverted in the process it is indicated by assigning negative reference to it. Interior boundaries of enclaves are however realised in reverse direction to exteriors, thus clockwise.

Seen from the start of the directed edge, four attributes are stored: first line left and right, and last line left and right. Figure 9 illustrates this.

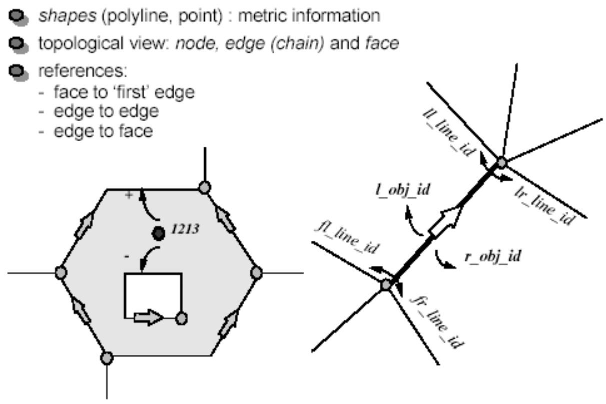


Figure 9: Winged-edge topological model as used by the Dutch Kadaster (image: Quak et al 2003, p. 2).

The procedure can be described as follows. Imagine that we have a land parcel that only has exterior boundaries, like parcel_A shown in Example 9 and Figure 10.

Example 9: Parcel table structured according to the wing-edge method.

```
TEST=# SELECT * FROM parcel ORDER BY object_id;
 object_id |  parcel   | exterior_id | interior1_id
-----------+----------+-------------+-------------
 001       | parcel_A | -1101       | 0
 002       | parcel_B | 3533        | 2088
 003       | parcel_C | 8612        | 0
(3 rows)
```

Parcel A, with the identification '001' points only to exterior boundaries with the identification '1101'. By selecting all boundaries from the 'boundary' table that have parcel A, either on left or right side it is possible to generate closed linear ring with necessary and correct ordered coordinates to construct a polygon. See Example 10.

Example 10: Results when all boundaries with parcel 001 either on right or left sides are selected. The negative value is assigned if the linked boundary line as opposite direction to the reference.

```
TEST=# SELECT object_id, fl_line_id,fr_line_id,ll_line_id,lr_line_id,l_parcel,
       r_parcel FROM boundary WHERE l_parcel='001'OR r_parcel='001';

 object_id | fl_line_id | fr_line_id | ll_line_id | lr_line_id | l_parcel | r_parcel
-----------+------------+------------+------------+------------+----------+---------
 1101      | 1212       | 2202       | 1414       | -4242      | 131      | 001
 2202      | 1101       | 8118       | -3303      | 6113       | 001      | 303
 3303      | 2343       | 4404       | 1398       | -2202      | 181      | 001
 4404      | 3303       | 3443       | -1101      | 4242       | 001      | 635
(4 rows)
```

The procedure is as follows:

1. Boundary 1101 is considered clockwise and the first boundary connected to it in counter-clockwise direction, also bounding parcel A, must be first line on right (hence that parcel A is on right side of boundary 1101) or line 2202. Hence if line 1101 would be reversed counter-clockwise (-1101) we would go for last line left.
2. From now on we look for last line left. For 2202 that is line –3303.
3. 3303 has to be dealt with in the same manner as 1101 as it orientation is clockwise. –3303 last line left is thus number 3303 first line right or line 4404.
4. The last line left of 4404 is 1101 and the linear ring is completed: -1101, 2202, -3303 and 4404.

Special function has to be developed to return the parcel geometry of related edges. Same procedure is followed for the interiors but clockwise (Quak et al 2003).

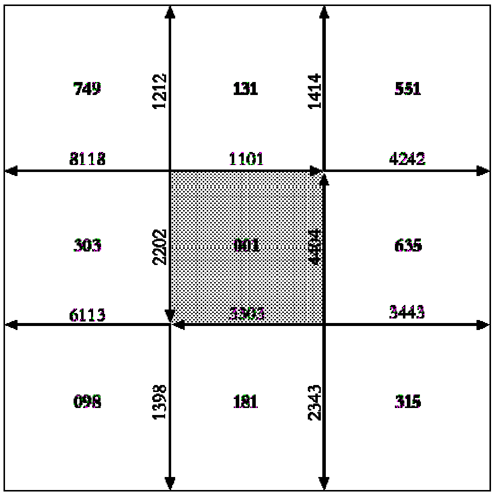


Figure 10: Parcel geometry used in examples with annotated faces and edges (image: Author).

Similar function to the winged-edge method is the 'Double-connected-edge-list' (Worboys 1995) whereas only two references are stored about connected edges as illustrated in Figure 11:

- the previous edge that also shares left face of referred edge, and
- next arc that also share right face with referred edge.

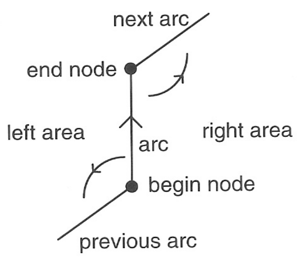


Figure 11: The edge reference in Double-connected-edge-list method (image: Worboys 1995, p.197).

To generate polygons based on the double-connected-edge-list method one has to employ the following algorithm:

1. Choose a face and one bounding edge to start with;
2. If referred face is to the left of the edge then choose 'previous listed' edge, else 'next listed' edge;
3. Repeat step 2 on the output edge until original edge is reached, and thus the ring completed.

Using the example given in Figure 10 and starting with edge 1101 to find the boundaries of face 001:

- Face 001 is to right of 1101 giving next-listed edge: 4404
- Face 001 is to left of 4404 giving previous-listed edge: 3303
- Face 001 is to right of 3303 giving next-listed edge: 2202
- Face 001 is to left of 2202 giving previous-listed edge: 1101
- Ring completed and polygon generated.

Another method to realise geometry from set of edges and related to the winged-edge method is simply to derive *left-right* information assigned to the boundary lines by using more 'intelligent' procedures as discussed be Quak et al (2003). Instead of storing all the necessary information on neighbouring lines in the boundary attribute table, a more sophisticated algorithm is carried out on the data.

First step is to retrieve all boundaries that belong to the referred parcel, using left or right side, relationship. To begin with all edges are stored in a variable called 'graph'. End points are furthermore extracted for further research and stored as nodes with information of connected edges.

The second step is to check each node and find those that are connected to exactly two edges. Then these nodes are compared against each other and those that share common edges glued correctly together (directed) and put back into the 'graph' container.

This procedure is repeated until a linear ring is found that is temporarily stored in the 'rings' container, while the algorithm finishes with the remaining data in the graph container. If the graph container is completely emptied the procedure is regarded successful, otherwise an error message is delivered.

The final step is to compare linear rings found in the ring container and perform necessary consistency checks on them. The procedure could be as follows:

1. Find the largest linear ring and assign as a exterior boundary of the polygon, remaining linear rings are assigned as interior boundaries
2. Check if other linear rings are contained by the exterior and if they overlap each other
3. Check all linear rings if they touch themselves or are not-simple which is both invalid geometry

When this is finished a valid polygon should have been generated. This methodology developed at TU Delft works according to Wilko Quak only on polygons, not on collection of polygon stored as multi-polygon. Summarising this method, only key references are stored but more processing is needed, potentially affecting performances.

Finally to mention is the so-called 'Wheel Topology' where a reference to each edge, bordering a face is stored as an attribute of a parcel (Oosterom et al 2002). Here are the interior edges stored in an array and valid polygon realised by algorithm. Not much was however found documented on this method.

All the methods listed here have their pros and cons which depend on the purpose of usage, the nature of the data, desired results and performances.

2.4 System architecture and maintenance

There are two main setup and procedures to maintain and update spatial data. One is to store the data in central database and update it either locally, where it is stored, or distributed, where the clients are located. The other way is to store the data in a federated database, where number of databases are combined in a way that they look and feel as one integral, maintained either locally or distributed.

2.4.1 Distributed/federated databases

The term 'distributed DBMS' is rather vague as it is often simultaneously used over the <u>horizontal distribution</u> of data in spatially detached 'partial databases', where the data is clustered e.g. according to spatial extent, and <u>vertical distribution</u> where diverse layers of data are combined.

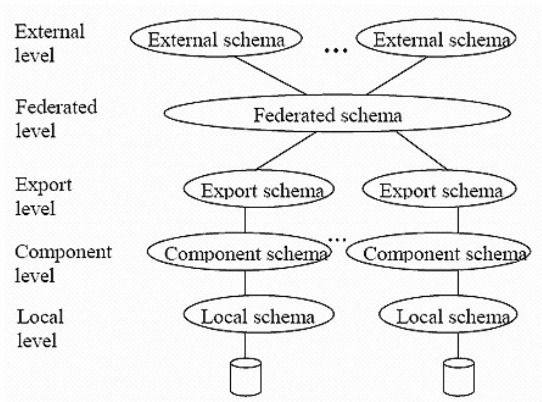
Horizontal distribution indicates that the complete dataset cannot be explored unless federating the relevant database to look and fell as one integral one. An example would be a cadastre that was maintained in number of separate non-overlapping cadastral districts. By federating the databases it could be accessed and explored, as it was one integral system.

Vertical distribution is in fact not real federation of partial databases but more combining related but independent and complete databases. An example is a cadastral database that joins separate databases like parcel-, land register- and public DBMS and represents as one integral SDBMS (also relevant is vertical and horizontal fragmentation of databases).

According to Ramakrishnan & Gehrke (2001) the federation of distributed or partial database should appear transparent, that is containing following two properties:

- Distributed data independency: refer to that users do not have to indicate any location or source attributes in queries as the federated databases appear as one integral one.
- Distributed transaction atomicity: refer that users have the same control for write transaction that access and update data in distributed databases as it was performed locally. Furthermore that the transaction is maintained atomic, that is, changes persist if completed, otherwise discarded.

This can be accomplished by intensive modelling of the diverse levels of the federated database where each separate level has its own schema, used to transfer the data up and down the hierarchy. Figure 12 at left illustrates this. At the bottom is the *local level* where the *local schema* resides. This is the data model of each individual database. Above is the *component level*, which translates the local schema into *component schema*, that is, according to a common model. If the local schema is already modelled in a common model the component level is unnecessary.



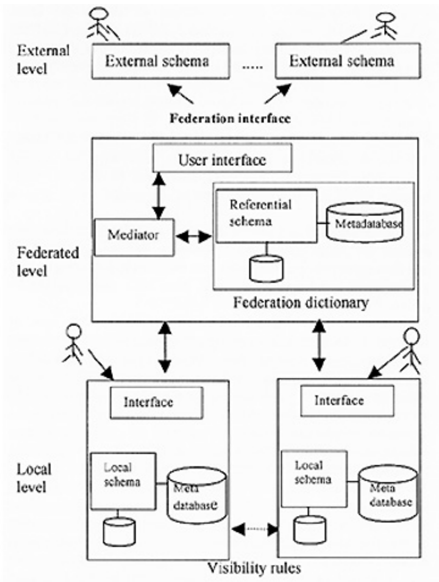


Figure 12: Two approaches describing the way partial databases are federated (image: Benchikha et al 2001, p.282-283).

On the *export level* are the database parts extracted that are relevant for the federated database before being merged at the *federate level* in a *federate schema*. *External level* and *external schema* are finally thought as an access point for users and provide relevant information for them to exploit the federated database (Benchikha et al 2001). Figure 12 right illustrates this further. At the bottom level are local databases maintained locally. The double arrows indicate the translation process from local/partial, through component and export level, to federate database. Using diverse software employing external schema it is possible for external users to access and view the federated database.

2.4.2 Characteristics and properties of federation

Looking at the federated databases, several characteristics can be observed: *homogeneity-heterogeneity*, *autonomy* and *distribution* (Tuladhar et al 2005).

Homogeneous refers to that each of the distributed databases employs the same DBMS software, data model and architecture so translation of data is unnecessary within the distributed system. Heterogeneous distributed systems contrary consist of several DBMS, each potentially employing own software, data model and architecture. Federating such a system is thus more complex as the data has to be translated to shared format before being served as a content of one integral database. Geography MarkUp Language (GML) as specified by the OGC is intended to serve this role for exchange of spatial data.

Two main types of heterogeneity are observed. DBMS heterogeneity that is due to differences of DBMS, like e.g. differences in data model, data structure, constraints and query language. Secondly is semantic heterogeneity, which is caused by semantic and ontological differences of DBMS modellers. This is e.g. differences in hierarchies, classes, geometry and attributes.

Autonomy of federated DBMS can further be grouped into three types:

- <u>design autonomy</u>: is how autonomous each DBMS component is to select its own design. Increased design autonomy leads to increased heterogeneity of the federated system.
- <u>execution autonomy</u>: is how autonomous each DBMS is to execute internal operation without interference from external operations.
- <u>association autonomy</u>: is how autonomous each DBMS component is to restrict the share/association to the other databases, both temporarily and permanently, and if it is allow to be part component of other federated systems.

Distribution characteristics refer to the system architecture of the federated system. In a conventional DBMS there is a direct communications between a user and a database where as the DBMS resides on user's computer or within a local area network (LAN). In a distributed DBMS exist multiple independent databases that by some kind of intermediate level are federated in such way that they appear as one integral. The distributed databases can in fact reside on the one and the same computer, or on multiple computers connected together through either LAN or wide-area-network (WAN).

The benefits of distribution are diverse. For instance it enables separate data updaters to be spread spatially over large area, each working in own DBMS, with the product collected and shared in federated database and accessible as such to external data users.

Also beneficial is to construct so-called parallel-distributed setup that spreads the burden between different servers. Here the spatial data would not necessarily be stored in spatial clusters but randomly stored on different databases. Query on three adjacent parcels in a cadastre could thus cause the response from three different database servers (Shekhar & Chawla 2003).

2.4.3 Distributed architectures

Ramakrishnan & Gehrke (2001) argue for that there exist three different types of distributed DBMS architectures. Namely: Client-Server System; Collaborating Server System; and Middleware System (Figure 13).

Client-server systems have gained much popularity with the introduction of the web and bettered communications. Its popularity is foremost gained by its explicitly, simplicity and usability. At the backend there is a server that connects to the databases and services their content to a network which number of clients is connected to. The clients (front end) can be diverse software that receives data from servers and delivers to users through customised interfaces. Therefore can data stored as raw text, number and binary codes in a database, be viewed by user as nicely drawn map with user friendly editor to manipulate the data.

The main drawback of client-server distributed system is that it cannot easily fulfil the condition of distributed data independency as defined earlier. It is maybe possible to develop a client that translates spatial queries into sub-queries for relevant underlying databases and returns appropriate results. It is however difficult when submitting complex query string that overlaps the databases, as the query has to be run separately on each database. The clients also get fairly complicated if they need to be capable of constructing sub-queries to be executed in different databases and then piecing them together again. In a way they would then be starting to act as servers themselves.

This problem is approached with the collaborative server system architecture (Figure 13c). Here is the client level skipped and the user interacts directly through his interface to a local server, which furthermore is a component of larger network of servers all connected to own database. Each server can only commit transactions on its local database. For accessing external data it is capable to structure and direct sub-queries to the other databases, as a result of user query, and piece together again for the user to interpret.

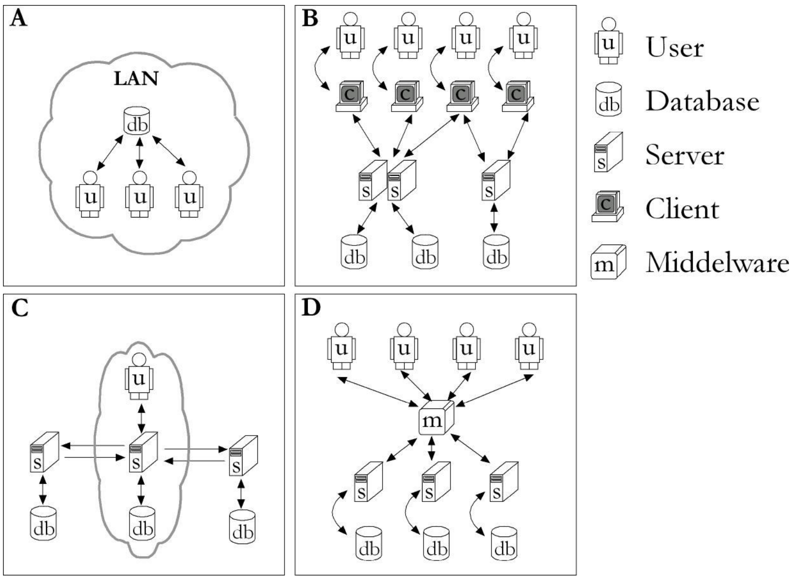


Figure 13: (a) conventional relation between database and users, (b) the Client-Server System, (c) Collaborating Server System, and (d) Middleware System (image: Author).

As an example we could imagine that there were four cadastral districts in a country each operating their own database, but served federated in accordance to collaborative server system architecture. Now a user on one office makes a query on its data that includes parcels stored in the other databases. The procedure could be as follows:

1. User makes a query to its server (I): 'give me all parcels in the country larger than 4 ha'
2. Server I receives the query and by inspecting the query discovers its scope and as a results constructs a sub-query that is directed to the three other database servers;
3. Each of the servers runs the corresponding query on their databases and returns the results to server I.
4. Server I receives the results, pieces it together and returns to user, as it was his own.

Finally to mention is the middleware system architecture that releases the servers of this burden of constructing sub-queries and directing them to other servers. A middleware mirrors the underlying servers along with performing a lot of work that otherwise would be committed at the server side. This is could be for example locking mechanisms, data constraints and validation, logging and etcetera. Figure 13d illustrates this, as users cannot connect to the databases without through the customised middleware that directs theirs queries and transactions to corresponding databases.

2.5 SDBMS transactions, consistency checks and concurrency control

Transactions are a vital part of every DBMS system whereas the content of the database is modified. In a very simple SDBMS with only one user, designing transaction procedure is not of that much importance. But with increased complexity, multiple users, and even federated SDBMS the urgency of having sophisticated transaction system is crucial to preserve the consistency and to enable concurrency.

This sub-chapter starts with discussing DBMS transactions in general before taking the pole of what exactly is meant with cadastral transactions. Followed is section 2.5.2 describing diverse consistency checks needed to preserve the quality of the system, before heading to spatial locking mechanism in section 2.5.3.

It is important to acknowledge that more sophisticated solutions call on more complexity of completing data transactions.

2.5.1 Cadastral transactions

Transactions in DBMS are referred as when the content of the database is manipulated using SQL statements like **`INSERT`**, **`UPDATE`** and **`DELETE`** statements. The SQL statements can be freestanding transaction or grouped together using **`BEGIN`** and **`COMMIT`** transaction calls, meaning that all SQL statements between are executed as single transaction. The properties of successful transaction procedure to comply with have been identified as (Hanssen 2003, p.2):

- Atomicity. Ensures that either all operations of a transaction complete successfully or all of its effects are absent.
- Consistency. Ensures that a transaction maps the database from one consistent state to another consistent one, i.e., database consistency.
- Isolation. Ensures that no transactions ever view any partial effects of other transactions, even though they execute concurrently, i.e., transaction consistency.
- Durability. Ensures that changes (committed transactions) to a database are persistent even when the system crashes.

Often referred as the acronym ACID, these properties are condition for concurrent transaction to be executed. Concurrency control and locking mechanisms are further discussed in section 2.5.3.

Cadastral (parcel) transactions handling geometric data in SDBMS are of similar nature and subjected to the same properties as other SDBMS transactions. The three basics types as listed in Brentjens are (2004):

- regular transaction: where ownership of a delimited parcel changes hands with no geometric change;
- merging parcel transaction: where two or more adjacent parcels are owned by the same person and merged to one new; and,
- splitting parcel transaction; where owner of a parcel decides to sell part or parts of his parcel that is consequently split into two or more separate parcels.

In an incomplete cadastral system additional action can be observed:

- adding parcel: parcel are brought sporadically/systematically into the system. Most of them already established in a land registration database.
- updating parcel: manipulating either boundaries or attributes of parcel, improving its quality.

Fundamental difference is between spatial and administrative cadastral transactions in the meaning that 'merging- or splitting parcel transaction' does always require that both spatial extent of parcels and ownership information have to be revised, which is not the case in 'regular transactions'. Where the spatial information of parcels in sufficient there is no need of resurveying the parcel before it simply changes owners.

Finally the two cadastral actions mentioned are solely matter of spatial cadastral procedures and interfere thus not with the information in the land registry except when parcels are brought into the system and assigned unique identification. In Dale & McLaughlin (1999) this is referred as adjudication.

2.5.2 Consistency checks

Depending on the spatial model employed, different methods can be used to preserve consistency of cadastral datasets. This could e.g. be using constraints on spatial features applied before a feature is created or modified. Examples of this for topologically constructed cadastral database are constraints like:

- boundaries (edges) are not allowed to cross themselves or other boundaries;
- dangling boundaries are not allowed (thus monument with relationship to only one boundary); and
- parcels (faces) cover complete domain leaving no space for voids indicating wrong topological relationships stored in edges.

More general constraints on geographic data could e.g. be set on roads not intersecting with buildings, or on maximum distance that two features can be within of each other. Building permits are for instance generally not permitted to spatially intersect or be within specific distance of roads (Brentjens 2004).

2.5.3 Concurrency control in cadastral systems

It is important for a cadastral system, especially if update remotely or using distributed architecture, to use sophisticated concurrency controls. Here is the notion of locking mechanisms something more than usually is referred to in regular database concurrency discussions (e.g. like preventing double booking of hotel room or seat on a airplane). Hence the transaction of spatial data does itself not consume so much time (solely computing matter), it is the construction of the transaction query that takes time. If any of the presupposition used in the query changes while constructing it, the transaction cannot be completed in accordance to the principles of ACID.

It is thus necessary that objects in cadastral databases subjected to manipulation be locked before the construction of the transaction query takes place, with this occurring on application level through user check-in and check–out, but not database level like traditional locking (Oosterom & Lemmen 2001). With subjected, is implied that not only directly edited objects should be locked, but also neighbouring objects that can be indirectly affected (Cheng et al 2004).

Several methods have been developed to perform locking mechanism on spatial data. Per feature lock emphasises on individual features. In a spaghetti model, editing one object results only with lock on relevant feature, as topological relationships are not preserved. In topological model this is however much more complex. See Figure 14 for example of per feature locking.

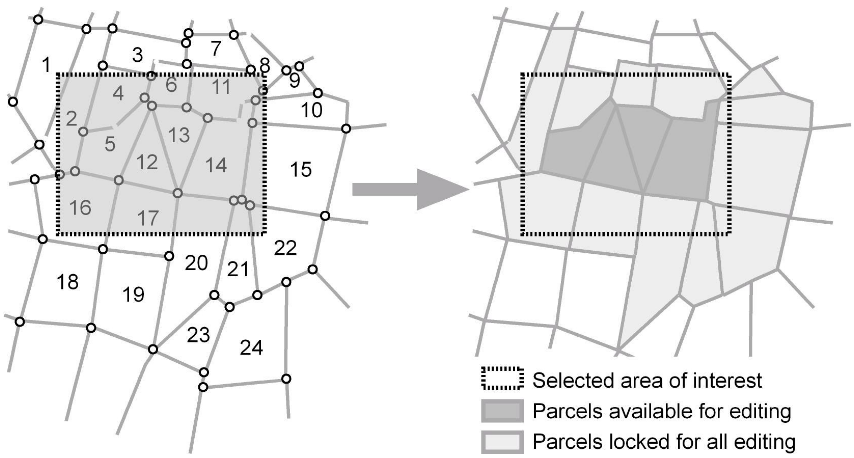


Figure 14 Example of per feature locking mechanism (image: Author).

Another way of locking spatial features is partial locking as described in Oosterom & Lemmen, where every object completely inside a selection rectangle is locked, but object overlapping it only partially locked, in the sense that (2001, p.518):

> …the coordinates of the line crossing the boundary of the work area are not allowed to change. Together with the fact that that the rectangle work areas can never overlap, this implies that the other changes to the edges and faces that cross the borders of the two nearby work areas are additional and can be merged in the database.

Choose of appropriate locking mechanism depends much on application software and logging possibilities it offers (check-ins and -outs). Alternative here is the use of middleware, a mediator that could manage this by assigning work areas and lock by request.

2.6 Remarks

It can be observed from the content of this chapter that the possibilities to design and technically implement a cadastral spatial registration are many. Several factors need consideration when storing cadastral data within a SDBMS. First of all is how the features are mapped to database objects. Even though this should be rather definite in terms of cadastral spatial registration (parcels are of course naturally represented by polygons) it is not. Parcel is composite of parcel boundaries and parcel boundaries are made up from set of measurements. If the system is to be able to keep track of all relevant information needed to construct a parcel, a simple polygon objects is simply not enough.

It depends on the viewpoint how cadastres can been observed. An economist sees parcel in fiscal context, as a base to found land taxation, whereas a lawyer sees the legal consequence of a parcel, but is less concerned with exact delimitation except in cases of boundaries disputes. A surveyor is here less interested in the parcel but puts the accent on the boundaries and survey points. Depending on the viewpoint preferred a cadastre is often referred as either parcel or boundary focused, reflecting the land registration law of the administrative unit it serves (province / country).

Here is the role of the geo-magician,[3] the designer of the cadastral SDBMS, to unify approaches in a solution that can satisfy the need of every group. Implementing topological model does this. With employing the topological primitives: node, edge, face and solids, a system can be designed that is in a way both parcel and boundary focused. With the traditional parcel realised from set of boundaries that further represent ordered sequence of survey points.

Also discussed in this chapter was system architectures and how maintenance of a SDBMS is not longer limited to local computer or network, but can be spread geographically, federated, accessed, queried and edited remotely as integral system. Complex system-design calls further for complex solutions in term of integrity tests and concurrency control to enable consistent and simultaneous transactions of several users.

Technically and communicational progress enables that SDBMS architecture for cadastral databases can presently be organised in such way that suits the organisational setup the most. Therefore the 'cadastral system-design' environment is more and more becoming knowledge-driven, where imagination sets the limit of innovations, instead of technology-driven, where user had to adjust their needs to the technology available each time. This creates e.g. the possibility to design a federated system of heterogeneous cadastral SDBMS, accessed as one integral. The technology is in place, but correlation of ontology, data exchange and representation is needed.

Next chapter discussed the innovation of the 'Core Cadastral Domain Model' that is intended to define core classes, attributes and relationships of cadastral registration to enable cross-system interoperability and shared ontology.

[3] Educated experts in GIS and SDBMS are in desperate need for a recognised title. Today they are either referred as GIS-experts, technical geographers, geomaticians and geoinformation engineers etcetera.

3 CORE CADASTRAL DOMAIN MODEL

In a more and more complex world the importance of standardisation is growing. Not only to simplify communication within specific domains, but also to generate paradigms and thus preventing that the 'cycle' is reinvented over and over again.

The development of cadastral registration is one of those things that have been regarded very cultural dependant. Many cadastral systems have their roots far back in history and matured in correspondence with both political and social, needs and resources. This gives the reason for the variety of cadastral systems that exist in the world. While the Dutch have highly developed, accurate and complex spatial cadastre, the Icelanders have maintained simple land registration, merely to serve fiscal purposes, without storing spatial extent of real properties. Most countries have in fact their own characteristic land recording system that differentiates for an instance on what is registered (deeds, titles), how it is registered (e.g. fixed- or general boundaries) and where it is maintained (on local or national level) (Zevenbergen 2002). This gives way for a jungle of systems and information that are hard to interpret or compare against each other. It moreover does not explicitly set an example for other countries to follow in their development.

Several contributions have been to the standardisation of the cadastral domain as discussed by Oosterrom and Lemmen (2003), on international level, country level and by private companies. None of these contributions has however attempted to generate a model that would be in either example to follow or 'domain language' to facilitate exchange of information between different systems.

The Core Cadastral Domain Model (CCDM) was with this in mind brought up at FIG congress in Washington 2002 by Professor Peter van Oosterom (TU Delft, The Netherlands) and Christiaan Lemmen (ITC, The Netherlands), embracing the fundamental concepts of most cadastral systems. Its main purpose is to (Lemmen et al. 2003):

- Enable effective and efficient implementation of flexible (and generic) cadastral information systems based on a model driven architecture… and
- Provide the 'common ground' for data exchange between different systems in the cadastral domain.

The following chapter deals with the CCDM, its goals, structure and potential. It starts with describing the main objectives along giving general overview on the ideology it braces it with. Followed, is the model presented along with describing in detail the main aspects of it. Finally are several remarks put forward, especially that concern the future potential and criticism of CCDM as observed from literature.

The discussion here is based on the 'Cairo version' of the model, presented on a FIG conference in April 2005, but newer version of it has been published in Oosterom and Lemmen (2005) in proceedings of 'United Nations Human Settlements Program' expert group meeting in Moscow in October 2005. Finally, unpublished is a version of it in CEUS journal presumably next year (Oosterom et al 2005).

3.1 Objectives of CCDM

The CCDM is defined with the widely used Unified Modelling Language (UML) employing the idea of model driven architecture approaches, stating that everything can be modelled to UML diagrams describing exact relationships between different objects, their role and flow of information. This facilitates "portability, cross-platform interoperability, platform independence, domain specificity and productivity" of systems (Oosterrom & Lemmen 2003, p.14).

Main objectivities of the CCDM are stated in the opening sentences of Lemmen et al. (2003, p.1):

> A standardized core cadastral domain model, covering land registration and cadastre in a broad sense (multipurpose cadastre), will serve at least two important goals:
> 1. avoid reinventing and re-implementing the same functionality over and over again, but provide a extensible basis for efficient and effective cadastral system development based on a model driven architecture, and
> 2. enable involved parties, both within one country and between different countries, to communicate based on the shared ontology implied by the model.

Based on these premises every cadastre in the world should be included and regarded as extension of the CCDM, being a specialisation of the core model. Furthermore it was thought that countries without cadastral registration could benefit from using the core to develop their own specific system.

Using this model driven approach every cadastral system should consequently be compared through this mutual core, the CCDM. This is for instance illustrated in Hess & de Vries (2004) where the Dutch and the Greek cadastres are queried simultaneously through a 'Query Translator' that along with CCDM functions as an intermediate between the two systems.

This approach could later be extended to include all cadastral systems from e.g. the countries of the European Union, to enable one European cadastral service, a service that would facilitate cross-border transactions with real properties. This is an important aspect, in times of heightening international cooperation, establishment of single market within European Union and consequently different environment of real estates transactions. Countries still developing their cadastral registration system could use the CCDM to sharpen their approach to the registration (model driven using UML) while providing core classes that the simply could be picked according to suitability.

3.2 Components

Land recording can be described as registering the association between three primary classes, seeing that person is related to land through the rights that he holds. Land is on other hand subjected to right, thus there must at least one right be associated to proprietary land parcel, allowing multiple ownership in accordance do share hold. See Figure 15.

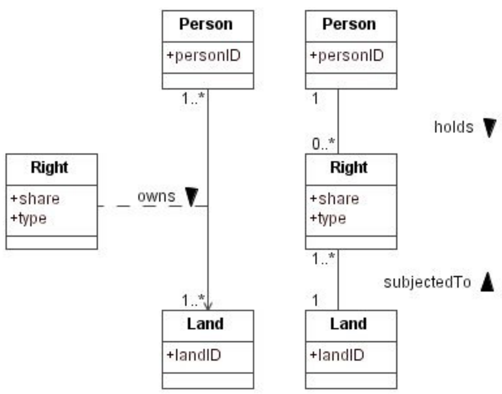


Figure 15: Two versions of expressing the UML relationship between person and land. The right one is preferred as it allows the same person to have multiple rights in the same property (image: Author).

The CCDM is in accordance split into three main sections:

- person section, defining how persons are registered, both natural and non-natural;
- legal section, handling the rights and restrictions that are subjected to ownership of real estates; and
- geographic section, covering parcel registration and geographic extent.

Together the person and legal sections make up what is referred as land registration as defined in section 1.3, while geographic section represents spatial delimitation. The following subchapters present and discuss each section briefly, before the complete model is presented in chapter 3.3. This summary on the CCDM is based on the Cairo version published in Lemmen et al (2005), with assistance of older versions, namely Oosterom et al (2004) and Lemmen et al (2003).

3.2.1 Person section

The administrative part of the CCDM encapsulates the classes that are related to public administration like maintaining a public registry, address repository and legal certification of individuals and companies. This is illustrated in a diagram presented with Figure 16.

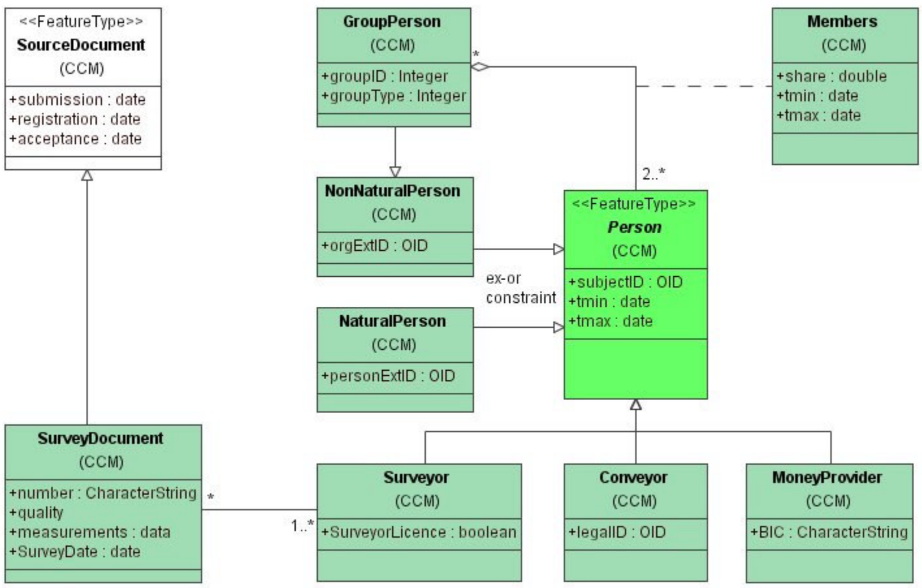


Figure 16: The person part of the CCDM (image: Author).

`Person` is here the centre of attention as that is the class which links subjects like human beings or companies to property objects through rights or restrictions. There are two principal specialisations of **`Person`**: **`NaturalPerson`** and **`NonNaturalPerson`**, inheriting **`Person`** attributes: **`subjectID`**, **`tmin`** and **`tmax`**. **`Person`** is here defined either natural or non-natural with unique identifier, contrary to **`Person`** other specialisations **`Conveyor`**, **`Surveyor`** or **`MoneyProvider`** that are not assigned unique identifier as they do not create specific instances of the **`Person`** class. This is because a **`Surveyor`** is a **`NaturalPerson`** but **`NaturalPerson`** is not necessarily a **`Surveyor`**. Thus, the **`subjectID`** of **`Conveyor`**, **`Surveyor`** or **`MoneyProvider`** always has to match either **`NaturalPerson`** or **`NonNaturalPerson`**.

GroupPerson intends to represent communities, co-operations and diverse entities that make up social units or structures. An example is e.g. nomadic tribes that ignore modern individualism but live collective life, grazing the land. Here the tribe can be considered as instance of the **GroupPerson** whereas each member of the tribe is automatically assigned share of the total.

Some attribute names presented in the diagram can sound unfamiliar and in need of brief definition. **BIC** is a 'bank identifier code' used internationally to identify financial institutions.

OrgExtID stands for 'organisation external identification' and has the purpose of being a link between the CCDM and some kind of company repository. Similarly, **personExtID**, stands for 'person external identification' and provides the link to person repository.

Finally, the variable **OID** represent 'object identifier', an unique identification assigned for objects, e.g. like 'social security number' assigned to natural persons in many countries.

3.2.2 Legal section

In the legal section the value of the link between **RealEstateObject** and **Person** is assigned with an UML link attribute to the class **RRR** (Rights-or-Restriction-or-Responsibility). **RRR** has several important attributes.

share is intended to register the amount of the **RRR** instance that **Person** has in **RealEstateObject**. The **type** attribute has **codelist** as value that lists all possible types of **RRR**. **tmin** and **tmax**, furthermore indicate the temporal begin and end of **RRR**. Examples of **RRR** types are e.g. 'freehold' and 'leasehold'. **RRR** has also several subclasses: **Appurtenance**, **Encumbrance** and **Obligation** that all serve as further specialisation of **RRR**.

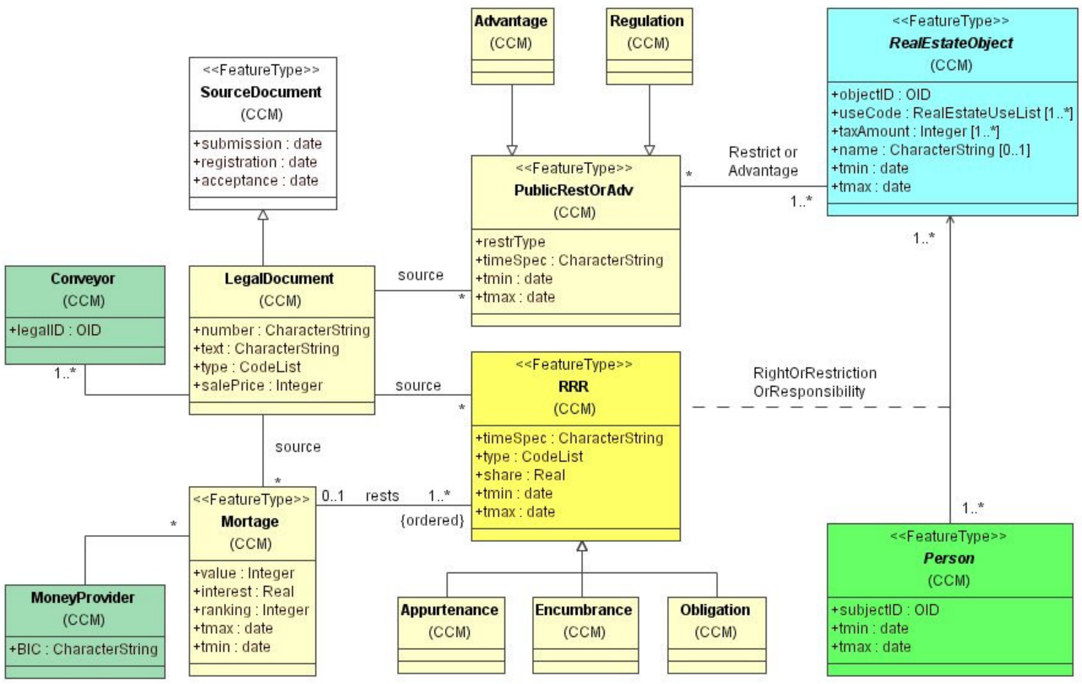


Figure 17: The legal part of the CCDM (image: Author).

Mortgage, provided by **MoneyProvider** (lender like bank) is possibly associated to **RRR** and both refer to a **LegalDocument** conveyed by **Conveyor**.

In most jurisdictions there exist also possibilities that properties are affected by negative right, a restriction or advance set by public laws or regulations. Hence, the **PublicRestOrAdv** reflects how public law influence usage of **RealEstateObject**, while property law deal with **RRR**. A **PublicRestOrAdv** can for instance be restriction like preservation zones made because of natural or historical relevance, or advance like a building permit as indicated in physical planning. Like **RRR** the reference document for **PublicRestOrAdv** is found in a **LegalDocument**.

3.2.3 Geographic section

The geographic section is here presented in two parts. The 'Parcel registry' part shows the relationships that the class **RealEstateObject** has to diverse subclasses. The 'Geometry/topology' part shows how the spatial dimension is recorded using **SurveyPoint** and corresponding topology and geometric realisations.

Parcel registry

The core of a cadastral registration is the real estate object that is represented in the CCDM with the **RealEstateObject** class illustrated in Figure 18. As an abstract class, no instances can be created of the **RealEstateObject**, but only at subclass level. **NonGeoRealEstate** is e.g. intended for properties that cannot be defined geometrically or have not yet been surveyed. Examples of this are e.g. diverse rights that can be separated from land ownership, for example fishing rights in a lake or a river. Similarly are the classes **AppartmentUnit**, **VolumeProperty** and **RestrictionArea** specialisation of **RealEstateObject**, each registering delimited shapes that real estate objects can take.

All the specialisation of **RealEstateObject** mentioned until now have in common that they do not partition or cover completely the spatial domain. That does however the role of **PartitionParcel**.

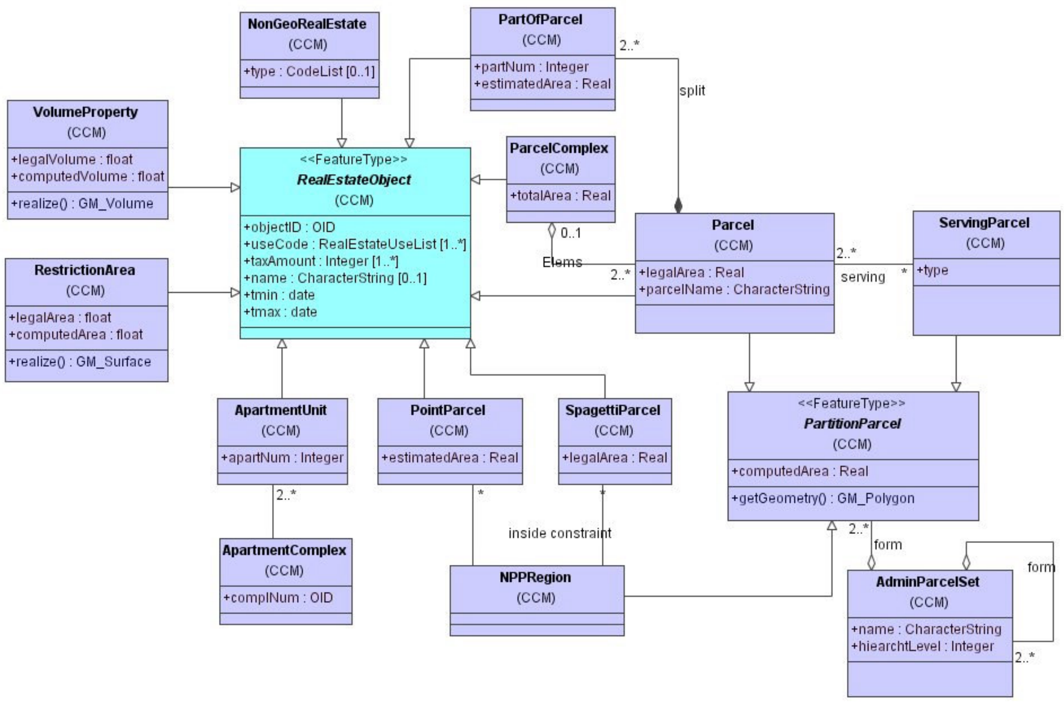


Figure 18: CCDM parcel registry (image: Author).

Parcel along with **NPPRegion** (Non-Planar Partition Region) and **ServingParcel** partition completely the planar spatial domain and together make up the abstract class **PartitionParcel**. It is noticeable that **Parcel** is specialisation of both **RealEstateObject** and **PartitionParcel** that could create implementation difficulties as some program languages and databases do not allow multiple inheritances.

The role of the **NPPRegion** is to include all **PointParcel** and **SpagettiParcel** used to represent spatial locations of parcels that do not have properly registered boundaries. For example, an address location can be used as the initial source for a **PointParcel**. Based on new data the boundaries could later be indicated by spaghetti data and the parcel moved to the **SpagettiParcel** class. Finally, when the parcel's boundaries have been properly measured it is upgraded to be a proper parcel and does not longer belong to the **NPPRegion**.

ServingParcel compromises two or more **Parcel** and belongs to (servers) them. Thus, is not directly registered to **Person** and therefore not a specialisation of **RealEstateObject**. An example of **ServingParcel** is e.g. playground or parking area that is jointly used by several parcels.

The **Parcel** class is the most valued unit of registering land ownership. It is the formal unit of land ownership and subjected to transactions. In this context, both **PartOfParcel** and **ParcelComplex** can serve as intermediate state of **Parcel** while splitting or merging land parcels. **ParcelComplex** could further represent set of spatially disjoint parcels with some RRR attached to it. Figure 19 illustrates how **PartOfParcel** serves in a splitting procedure with a UML state-diagram.

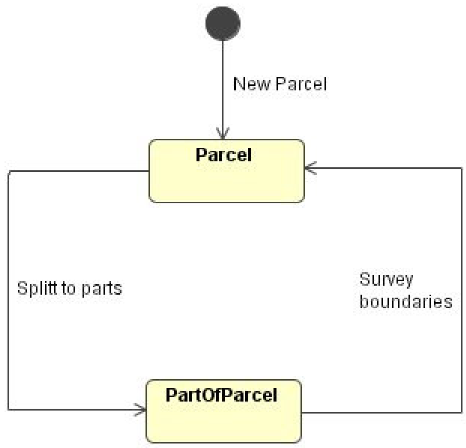


Figure 19: UML state diagram that illustrates how PartitionParcel is split. If a part of a parcel is o be sold, the parcel is split into PartOfParcel that get the status of regular Parcel when their boundary have been surveyed (image: Lemmen et al 2003, p.9).

Geometry/topology

The diagram presented in Figure 20 shows how spatial dimension is added to the classes already introduced in the parcel registry part. The basic element here is the **SurveyPoint**, used by other classes to derive geometry in accordance to ISO/TC211-19107 standard on 'Geographic Information – Spatial Schema'. A good overview of this standard can be obtained in Kresse & Fadaie (2004).

Survey points are obtained from 'survey document' and the association is that there can exist multiple survey points per document. By using these measured survey points classes like: **RestrictionArea** and **VolumeProperty** can realise its geometry. For instance, at minimum three survey points are needed to construct two-dimensional **RestrictionArea**, or at least four points to construct **VolumeProperty**. Zero to many survey points can indicate **`ApartmentComplex`**, used to represent multiple **`ApartmentUnit`**, and it is optional if and how it should be represented. Similarly, **`SpagettiParcel`** is indicated with at least three points and **`PointParcel`** with one.

`ParcelBoundary` is a combination of two or more **`SurveyPoint`**, and has direct geometrical association to instance of the topological **`TP_Edge`**. **`TP_Edge`** is defined as two end nodes referred here as **`TP_Node`** with direct association to **`SurveyPoint`**. **`TP_Edge`** has furthermore optional number of intermediate points derived from **`SurveyPoint`**.

Finally, the geometry of **`PartitionParcel`** is constructed from at least one **`ParcelBoundary`**, while each **`ParcelBoundary`** has relationship to at least two **`PartitionParcel`**. This topological relationship is preserved between **`TP_Edge`** and **`TP_Face`** whereas the geometry of **`PartitionParcel`** is realised directly from **`TP_Face`**.

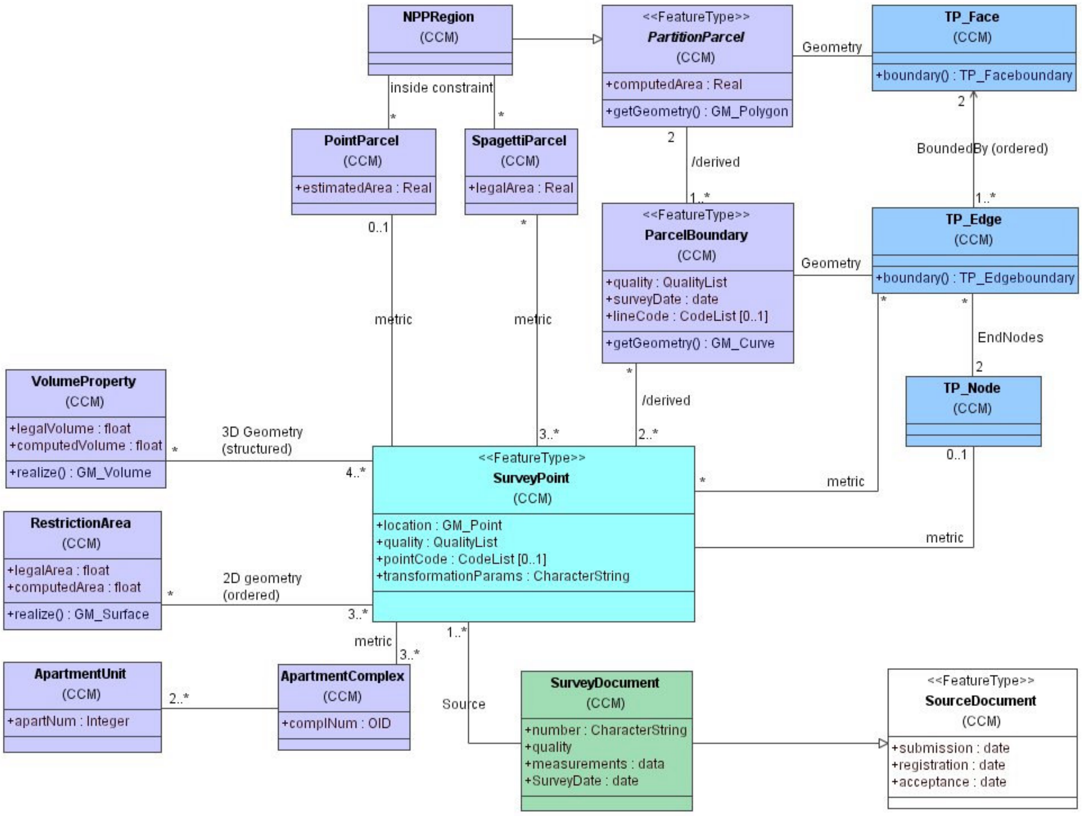


Figure 20: CCDM geometry/topology part (image: Author).

3.3 Representation of the complete CCDM

Figure 21 and Figure 22 represent the complete CCDM as introduced by Lemmen et al (2005) in Cairo 16-21 April 2005. Looking at all sections together gives more comprehensive view of the model than displaying isolated parts, even though it might seem a bit more complex.

Definition of classes of the CCDM are presented in section 3.4 to accompany the diagrams.

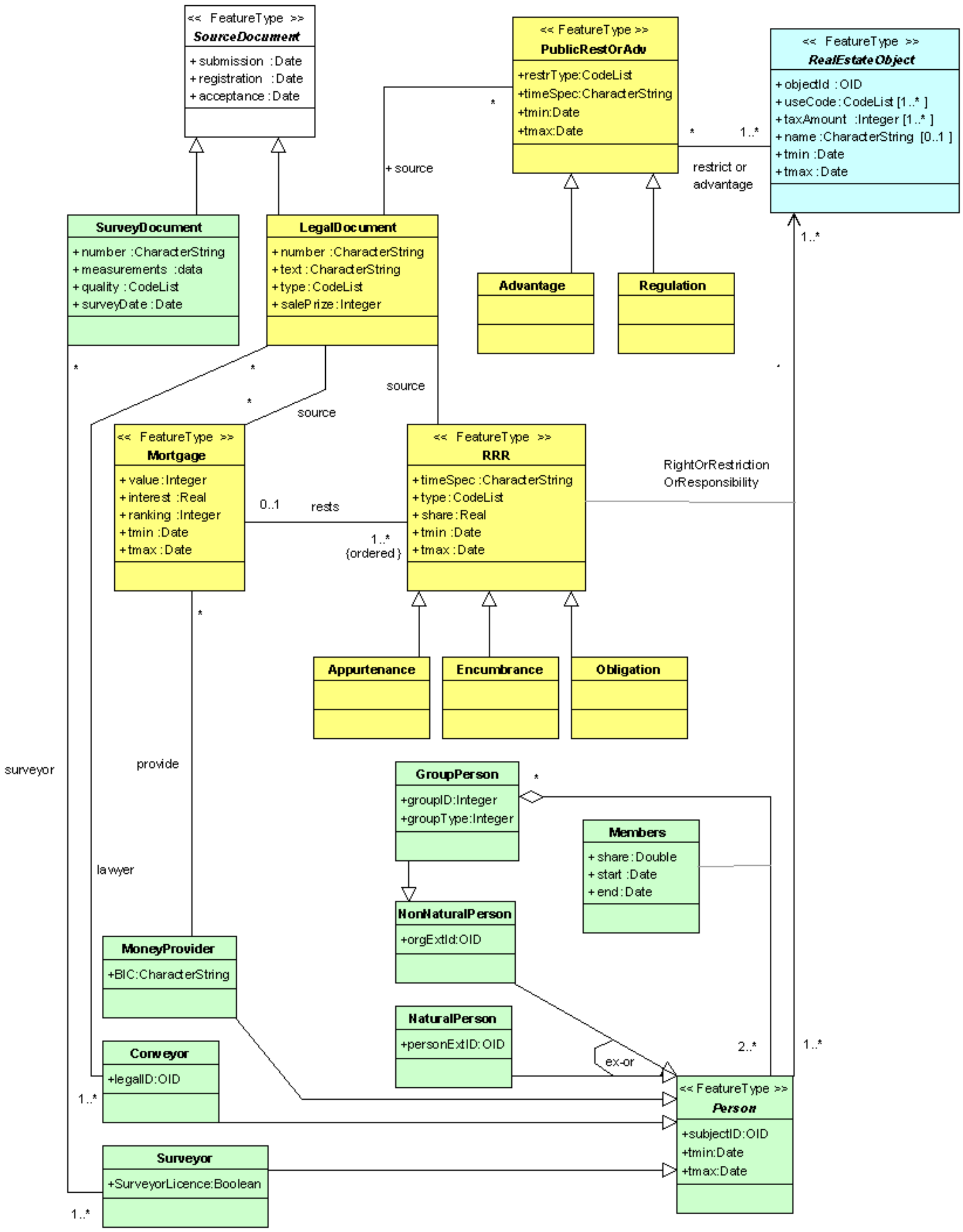


Figure 21: The administrative/legal side of the CCDM (image: Lemmen et al 2005).

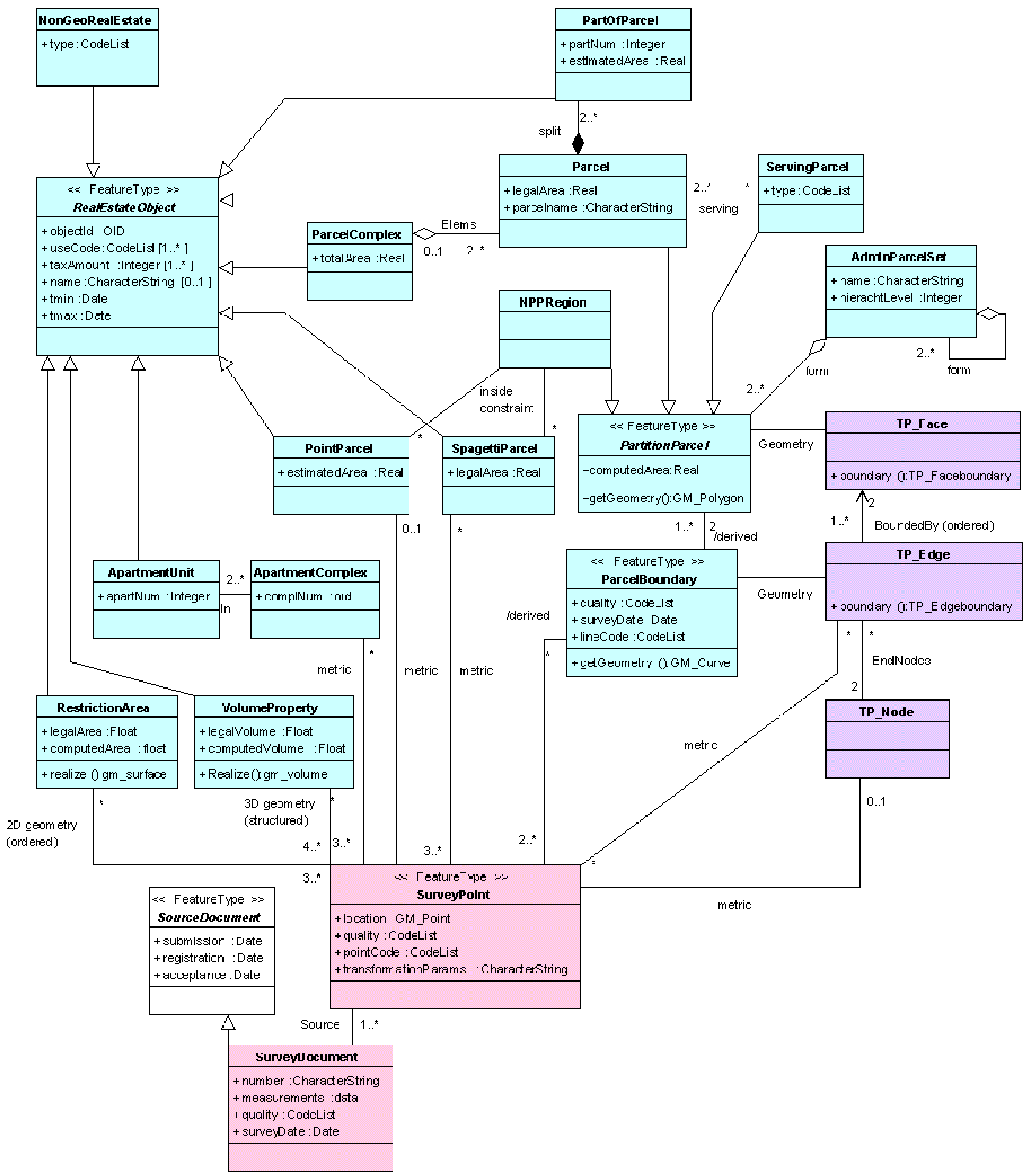


Figure 22: The geographic side of the CCDM (image: Lemmen et al 2005).

3.4 Definition of classes

The list of definition shown below was compiled by using diverse publication of the CCDM (Lemmen et al. 2003; Lemmen et al. 2005; Oosterom et al. 2004) along with a bit of 'googling'.

AdminParcelSet Aggregation of many parcels that form an administrative unit, e.g. municipality, county, province and election district.

Advantage E.g. building permit, the right to build a house on a certain parcel.

ApartmentComplex Composition of several **`AppartmentUnits`** that form one complex.

AppartmentUnit	Building unit in **`AppartmentComplex`** that can be associated to person through right, being an object for registration.
Appurtenance	Something that is connected to a property in such way that it has to be transferred along it. Examples are e.g. diverse land use rights.
CodeList	States possible alternatives for an attribute and restricts in that way the values that can be assigned.
Conveyor	A person that is licensed to convey (notarise) legal documents.
Encumbrance	A claim, line charge, attached to and binding real property.
GroupPerson	Person that is an aggregation of many persons, each with own share. This is meant to encapsulate diverse nomadic rights in the CCDM.
LegalDocument	A document that is conveyanced and among others serves as a proof of evidence.
Members	Represents the association value between **`Person`** and **`GroupPerson`**.
MoneyProvider	Bank or other money lending institutions.
Mortgage	A legal instrument that creates a lien upon real estate securing the payment of a specific debt.
NaturalPerson	Human being.
NonGeoRealEstate	Reserverd for real estates that do not have fixed geometry like e.g. land-use rights (fishing, hunting) or when geometry is unknown.
NonNaturalPerson	Is a non-human, often referred as legal entity, examples are company or organisation.
NPPRegion	Stands for Non-Planar-Partitioning Region and is used parallel with **`PartitionParcel`** and **`ServingParcel`** to partition the complete planar domain of landownership. Within can **`NPPRegion`** parcels can be represented with spaghetti data or points.
Obligation	Duty or responsibility that comes with the property.
Parcel	Fundamental entity in cadastral systems. This class inherits both **`RealEstateObject`** and **`PartitionParcel`**, connecting the parcel registration with spatial delimitation.
ParcelBoundary	Between two parcels is at least one parcel boundary. It has the geometry **`GM_Curve`** as defined in ISO/TC211-19107 and 1:1 association with **`TP_Edge`**.
ParcelComplex	Aggregation of two or more parcels having a shared **`RRR`** attached.

PartionParcel	Partitions the complete planar land domain into non-overlapping parcels.
PartOfParcel	Parcels can composite of several **`PartOfParcel`**. Represents the different parts after parcel-split. Temporary intermediate level when splitting a parcel into two or more.
Person	Is either natural or non-natural and has unique identifier.
PointParcel	Single coordinate pair to locate parcel when complete geometric extent is unknown. **`PointParcel`** is always within **`NPPRegion`**.
PublicRestOrAdv	Public decision can influence a property, either restricting its exploitation because of restrictive regulation, or an advantage e.g. like building permit. Hence, associated with the immovable property not the **`Person`**.
RealEstateObject	**`RealEstateObject`** is the centre of the CCDM. It is an object subjected to cadastral registration, bridging legal/administrative part of the CCDM with the geographic part.
Regulation	Restricts use of a property e.g. because of preservation (cultural or natural).
RestrictionArea	Is meant to register right and consequently restriction that can be to land use because of e.g. utilities or preservations.
RRR	Right, restrictions and responsibilities. This class is subjected to private laws opposite to the **`PublicRestOrAdv`** class that relies on public law.
ServingParcel	Is associated to at least two parcels and services them in a way by providing e.g. common playground or parking. Serving parcels belong not directly to persons but other parcels.
SourceDocument	This is a super class to **`LegalDocument`** and **`SurveyDocuments`** and assigns commonly important attributes to both classes.
SpaghettiParcel	Parcel that is made up with often inconsistent data not storing any topology. **`SpaghettiParcel`** is always within **`NPPRegion`**.
SurveyDocument	Survey points are published in a survey document that is made by a surveyor.
Surveyor	Person that is licensed to carry out a legal survey.
SurveyPoint	Point surveyed in the field. This class assigns diverse measurement parameters to the survey, like quality, projection etc.

TP_Edge	Topological edge as defined in ISO/TC 211-19107. Edge = [node-start, node-end, left-face, right-face, <intermediate points>]
TP_Face	Topological face as defined by ISO/TC211-19107. Face = <edges>
TP_Node	Topological node as defined by ISO/211-19107. Node = point {x , y}
VolumeProperty	A thee-dimensional property that does not fit into the conventional planar registration of cadastres. Example of **`VolumeProperty`** is e.g. underground tunnel or a building property built on a bridge (does not have claim to the land beneath the bridge).

3.5 Potential and experience

The potential of the CCDM is widely regarded large, but still there is little experience of its utilisation. It is still in development and will need to got through a of further discussion and criticism before it finally can be regarded as a universal conceptual core for international / cross-platform comparison of cadastres, similar to other ISO, FIG, CEN standardisation procedures.

This section deals with the experience of CCDM so far, along with discussing its main criticism and future development.

3.5.1 Experience

Hess & de Vries, developed in their work, a query translator, based on the idea of CCDM. The objective was to build a translator that could translate queries between different cadastral systems, in their case, the Dutch and Greek cadastres. The ultimate goal was that a Dutch user could construct a query, valid for the Dutch system, and execute on the Greek system. The query translator would then translate the query appropriately for the Greek system, execute it, retrieve the results and translate it back to the Dutch system. The Dutch user would then receive results similarly as he were querying his own system. In their conclusions they observe among others (Hess & de Vries 2004, p.11):

> By reformulating queries from the Dutch into the Greek cadastral system via the core cadastral model, we demonstrated that data can be exchanged between different information systems which have no direct links and no common historical background but which are only extensions of a common core model.

This supports the proposed functionality of the CCDM and that it complies with its initial objectives. However, Hess & de Vries further made following remarks (p.11-12:.

> …national models can extend the core model in very different ways. Thus, it might be the case that data is not available on the same level of detail in both cadastral systems.

> …the differences in the abstraction level, i.e. the core model is more conceptual and the national domain models are closer to the technical implementation, lead to problems during the identification of mapping relations and the rewriting of queries.

> …the Query Translator offers no translation for those parts of the national model which have no corresponding part identified in the core cadastral model.

This is part of the criticism that the development of CCDM has received.

3.5.2 Criticism

As the CCDM is in development and thus subjected to constant revision, external remarks have been stimulated to better the model. This is e.g. reflected in the 'Cairo version' of the CCDM (Lemmen et al 2005), which is a revision from prior version based on a workshop in standardisation in the cadastral domain, Bamberg, Germany in December 2004.

One of the main criticisms of the CCDM approach from the Bramberg meeting is found in Kaufmann (2004, p.1) where he states:

> The conceptual background of the Core Cadastral Domain Model at the moment needs complex objects to be able to create a correct real property based model. This approach tends to be complex.

And (2004, p.8):

> …The core cadastral domain model initiative, trying to model existing occurrences of cadastres, is [thus] confronted in every step with new questions.

Professor Erik Stubkjær (2003) from Aalborg University has moreover pointed out that CCDM does not treat transactions and spatial frame reference sufficiently, which he thinks is of significant relevance. The counter argument to this is that spatial reference system should be considered belonging to different repository than cadastral registration. Consequently this raises the question of the correctness of including person registration within the CCDM.

3.5.3 Future development

The CCDM is of such nature that constant development and refinement of it are necessary through out its lifetime. While this research was undertaken a new version of it was represented in (Oosterom & Lemmen 2005). There it has taken considerable changes from the one presented in this paper, i.e. based on interaction between author and the developers of the CCDM and recommendations from ISO/TC211. *Appendix B: New version of the CCDM* presents the newest version of the CCDM diagrams as presented in Moscow, October 2005.

3.6 Remarks

The innovative of CCDM is much-needed contribution to the discussion and general development of cadastral systems. It has an ambitious future vision and could benefit both developed countries, for cross-platform interoperability, and developing countries, giving them core to build on own sophisticated cadastral system, avoiding reinventing the wheel again.

It is however the view of this paper that it is hard to fulfil both its objectives of setting an example for effective cadastral system development and create shared ontology to enable cross-platform interoperability. In this context the two objectives could be segregated in implementation. Argument for this is that shared common aspects of some cadastral systems do not necessarily offer the optimum solution when developing others.

Covering common aspects of cadastral registration all over the world could increase the complexity of the model to such an extent that it will not be easily interpretable. Consequently the question can be asked why complex universal applicable CCDM, covering various heterogeneous indigenous variations like nomadic rights, when such model does not yet exist for Europe, with more homogenous legal and cultural background? It is the view here that separating the implementation of 'cadastral system paradigm' and CCDM could:

- stimulate discussion on a cadastral paradigm much like Cadastre 2014, but using more explicit tools like UML to express its future vision and objectives; and
- give the CCDM more room to evolve and grow technically, to become universal applicable as a common denominator of diverse cadastral system.

Another recommendation of this paper is to focus more explicit on modelling relations of the core packages and present them as such, which further could serve as units for comparison. Vladimir Strončék from Slovakia has in his studies on the CCDM mapped and categorised classes to separate core packages as shown in Figure 23. This way each package can be studied, discussed and compared separately from the others in more efficient way than now visualised in the CCDM.

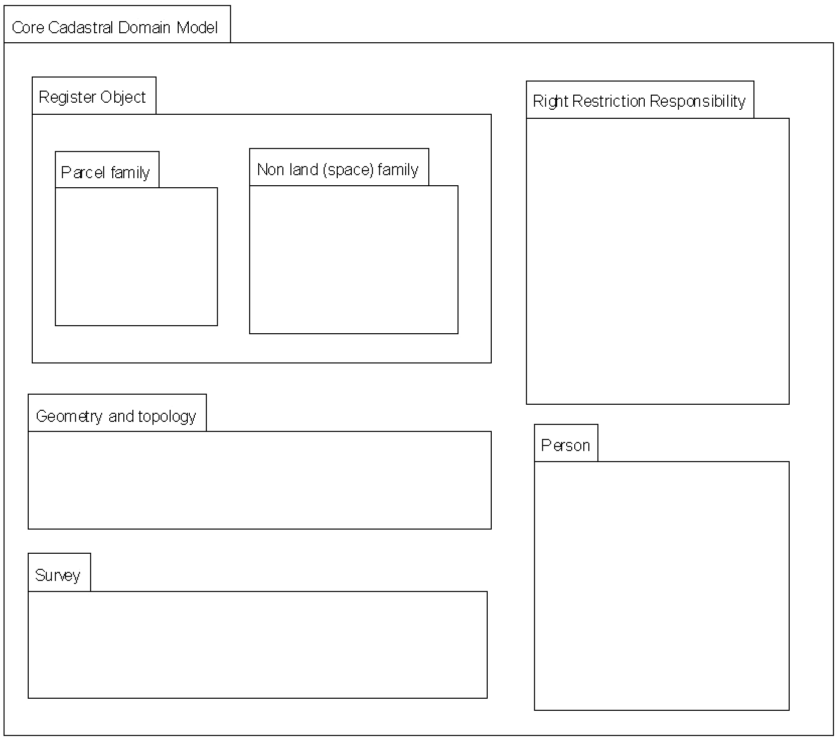


Figure 23: The core packages of CCDM (image: Strončék 2005).

4 CADASTRAL REGISTRATION IN ICELAND

No exhaustive coverage does exist in English on the organisation of land recording in Iceland. Not much has even been written in Icelandic, except articles appearing in the annual report of the Land Registry of Iceland (LRI), most notable Matthíasson (2003).

This chapter is intended to provide overview of the present cadastral registration in Iceland. It starts with placing the cadastral registration in historical and geographical context with section 4.1. Followed is section 4.2 that describes how real properties are defined and treated in Iceland, before moving to section 4.3, which describes the institutional setup and organisation of land recording. Section 4.4 covers cadastral transactions in Iceland discussed in coherence with Stubkjær (2002 & 2003). The data model of the Icelandic land registry database (LRD) is discussed and derived diagram presented in section 4.5, explaining classes, relationships and most important attributes. Section 4.6 looks at the availability of spatial data that can be utilised to establish spatial delimitation of all parcels in Iceland. Section 4.7 contributes to the future vision of cadastral registration in Iceland before ending this with remarks in section 4.8.

4.1 Geographical and historical context

A volcanic island, Iceland is placed on the middle of the Mid-Atlantic ridge, northerly between the divergent North American and Eurasian tectonic plates.

The landscape is quite mountainous, with more than three quarters of total area above 200 meters elevation, not very arable because of unfavourable climate. Glaciers, sands and lava fields further restrict the habitable lowland, and in some places (especially in the East and Westfjords) only a narrow strip is between a mountain and a coast. In total only around 15% of Iceland is arable.

Approximately 300 thousand people live in Iceland, whereas the majority lives in the southwest corner, within 60 km from the capital Reykjavík. The population in this area counts for 74.4% of total population in Iceland (FAI 2004; Hagstofan 2005; LMÍ 2004).

The tradition of land recording is old in Iceland and can be traced as far back as to the period when the country was initially settled in late 9th century. Well known is the 'Book of settlement' published in the early 12th century that describes the initial delimitation of land parcels in Iceland. In fact present land ownership in Iceland is at large extent based on this book (at least theoretically), e.g. as the root of title, extent and the location of legal boundaries (Landnáma 1986).

Formal registration of rights to land started in 1096 with the tithe laws promoted by Bishop Gissur Ísleifson. These laws needed public inventory for property and land valuation as the tax was calculated from both moveable and immovable properties values. The procedures of land registration developed and matured next centuries with the registry maintained mainly by representatives of local parishes[4].

In 1976 the appraisal segment became role of one public institution, 'The Land Registry of Iceland' (LRI). 25 years later, with the emergence of advanced information and communication technology, the institution also started to maintain one digital centralised <u>land registry database</u> (LRD) named *Landskrá Fasteigna* (Guðmundsson 2000).

[4] Municipalities have at large extent replaced parishes as an element in administrative subdivision in Iceland during extensive merges of parishes in recent decades.

Cadastral mapping does not have a long tradition in Iceland, mainly because of shortage of sufficient topographic maps. The first laws regarding cadastral mapping were set in 1914/35, which regulated surveying and mapping of lots in Reykjavík. Similar law were set for Akureyri in 1951/16 and literally for the whole country in 1997/73 (Matthíasson 2003). The main shortcoming here is that every administrative unit (municipality / parish) is responsible to separately implement this, on it own premises and initiative, resulting with incomplete and heterogeneous registration. To complicate things even further, the Ministry of Agriculture has obligations to preserve diverse information on agricultural land, e.g. rural ownership boundaries; and because of lack of available cadastral data several other public institutions have mapped ownership boundaries and maintained separately for own purposes.

A recent proposal for new bill is to overcome these shortcomings, by establishing one nationwide cadastre in Iceland (Alþingi 2004). Although the legal framework has been sketched somewhat in the proposal there are lot of concerns yet to be resolved, especially regarding technical implementation.

4.2 Notation of real property and real property rights

The concept of property does not have explicit universal meaning as its definition varies greatly between or even within different fields of professions (e.g. law, economics and engineering).

In general the term refers to the relationship that a person can have to a specific object. An object can be categorised in many ways depending on legal environment and physical appearance. For example a property can be immovable (land or house, hence real estate) or movable (ship, caravan, computer), tangible (article, plan) or intangible (idea, usage), and subjected to private, public or collective (common) ownership (Dale & McLaughlin 1999; Matthíasson 2003).

The following discussion is restricted to immovable properties in Iceland, often referred as real property (Icelandic: 'fasteign') and the diverse rights that it can be subjected to.

4.2.1 Real property

A real property is defined in Iceland law on real property appraisal and registration no.6/2001, 3rd article, as a "…delimited part of land, along with organic and inorganic components, enclosed rights and those structures that are permanently placed there."

This clause continues that every real property, its components and individual structures should be registered as specific units in the LRD. These units are categorised as:

- land parcel, that can because of distinct ownership- or use rights, exploitation, distinctions or legal boundaries, be counted as independent unit;
- structures connected to land;
- separate apartment estates in apartment house;
- parts of structures in the case of specific usage;
- cultivation;
- natural resources (corresponding to the Icelandic concept: hlunnindi); and
- other rights related to properties.

Icelandic law further allows separate ownership of land and building as long as permanent rights there between are generated and notarised. For instance, city municipalities commonly own most land within their urban zone, but make long-term leasehold contracts to residents that consequently are registered owners of the building they live in, but not the land it resides on (Matthíasson 2003).

Examples exist in Iceland of real properties that do not have permanent right registered to land, but they are scarce. In some cases properties are e.g. somewhere between to be regarded as movable or immovable, e.g. stationary caravan or temporal working facilities. Thus there is question if the property is immovable and if it should be registered as such (Matthíasson 2003). Moreover, some buildings that where built prior the Planning and Building Act no.73/1999 have not necessarily registered land rights (delimited building lot) and municipality officers have even found unregistered summerhouses far and wide in the rural of Iceland.

Finally, land ownership is categorised into three classes in Iceland: private ownership where either natural or legal persons own land; public ownership where land is owned by the central or local government acting as a legal person; and, collective ownership where land is not subjected to ownership at all although it can be utilised under custody of the central government as set in law no.58/1998.

4.2.2 Real property rights

It is the role of county officers in Iceland to register property rights into the LRD. Prior to the Land Registry Database each county had their registrar book, in Icelandic *Þinglýsingarbók*, similar to the German *Grundbuch* where all rights to real properties were registered. The content of these books has now more or less been modernised by moving it into the LRD as a part of the 'registrar section'.

Only notarisation of documents embracing splitting land parcels and sale contracts of apartments in condominiums are stipulated by laws (no.6/2001 Art.17; no.26/1994 Art. 17). However, it creates decisive legal security for property owners and leaseholders to notarise contract while money lenders like banks and governmental institutions request that mortgages are always notarised (Andrésdóttir 2005).

In this context it is interesting to point out that Icelandic land registration belongs to the category of deed registration systems (Andrésdóttir 2005), contrary to what is widely reported in cadastral questionnaires and reports that state that title registration is practiced in Iceland. The reason for this misunderstanding is partially due to translation as the Icelandic language does not make distinction between deed and title, and both concepts are translated as 'afsal'.

Persons can hold multiple types of right to a real property in Iceland. Generally these rights are categorised into two subgroups: direct property rights like freehold and indirect property rights that are further divided into several groups (Matthíasson 2003).

Direct property right in Iceland is a full ownership of real property, similar to what is referred as 'fee simple / absolute right / freehold ' in British land administration terminology. It is the greatest interest that a person can have in real property, is without time limitation, and is freely transferable and inheritable (Dale & McLaughlin 1999).

Indirect property rights are segmented in Iceland to seven different categories according to Matthíasson (2003):

- <u>A right to use and exploit</u> (Icelandic: afnotaréttur): Either extensive right to use and exploit land or property, completely owned by another person, for specific period. Most common is leasehold. Or temporal restricted exploitation right of another's property like e.g. for grazing, fishing, access etcetera.
- <u>Profit à prendre</u> or easement (Icelandic: ítak): Limited ownership right that a person (or property) can have in another property. This is e.g. the exclusive right to exploit natural resources like whale drifts, grazing rights, hunting and logging.

This is old tradition that legislator is trying to abolish, e.g. with laws no.113/1952 that allows that properties can get rid of “profit à pendre” by compensating financially for it.

- Mortgage – Collateral (Icelandic: veðréttur): This is indirect ownership right as it allows the property owner to give in exchange for a loan, a right to the creditor to requisite if the debtor does not keep to his undertakings.
- Toll obligations (Icelandic: afgjaldsskyldur): This is negligible phenomenon, which means that the owner or a leaseholder of a property must pay toll of the property usage to authorities or rightful claimant.
- Hold right (Icelandic: haldsréttur): Is a right that possessor has sometimes to hold his possess of a property until specific payment has been realised.
- Pre-emption and purchasing right (Icelandic: forkaupsréttur og kaupréttur): Pre-emption right is the right to buy a property under specific conditions, if the holder decides to sell and in some cases when change of usage. Purchasing right is the right to buy a property independent of the decision of the holder to sell or not.
- Internment right (Icelandic: réttur samkvæmt kyrrsetningu): This is preliminary right to limit the usage or provision right, which would alter the value of the property. Creditors apply this to insure their collateral in terms like when debtors go bankrupt.

4.3 Actors, administrative and institutional setup

By briefly observing the land administration setup, the following parties can be found as actors (either passive or active) in implementing and maintaining cadastral information in Iceland:

- Central government (Ministries of: Finance, -Justice, -Environment, -Social Affairs, and –Agriculture);
- Governmental institutions (Registration and Valuation Office, Planning Agency, National Land Survey, Agricultural Research Institute, Environment and Food Agency, Public Road Constructions etcetera)
- Jurisdictions and county registrars (Supreme- and regional courts, and county offices)
- Local authorities (Municipalities, Planning- and construction officer, special municipality departments)
- Private surveyors
- Consultancies, services and providers of geoinformation (Cartography, GIS, orthophotos, satellite imagery, digital terrain models)
- Software vendors
- Real property owners
- Tax payers

There are three main administrative levels maintaining cadastral registration in Iceland: central government, counties and local authorities. Sections 4.3.1-4.3.3 introduces each of these level, whereas section 4.4 goes in more detail into the transaction procedure.

4.3.1 Central government

The LRI, which has the legal role to operate the LRD, is mainly concerned about textual and numerical facts like property name, property identification, estimated value and size. Up to now, no geographical information is maintained in the LRD and it is mainly maintained for fiscal purposes with real property taxes an important source of income for many local governments in Iceland. Example of LRD record is found in *Appendix C: Example of a Record in LRD.*

The institution, governed by the Ministry of Finance, is operated by governmental grant, but with cost recovery program, as it charges levies for access and transactions in the land registry. Major rise was in the income of the institution in 2001 when new laws on property registration and valuation became a fact. This law moved large of the responsibilities formerly carried by the municipalities to LRI and into the LRD. The municipalities became of course firm client of information causing this surge in revenue. The Ministry of Agriculture has the legal obligation of registering all farmland in Iceland (FMR 2005).

4.3.2 Counties

There are in total 25 counties in Iceland, governed by the Ministry of Justice and each headed by a <u>county officer</u>, which also has the role of head registrar within his administrative area.

The role of the county registrar is to store legal documents like ownership of real properties (deeds), mortgages, pre-emption right etcetera. It is quite variable what is registered in each occasion for land parcels. Plats provided by the municipalities are mostly used for urban areas. In rural areas boundary description or boundary maps are approved by relevant landowners and overseen by the municipal planning- and construction officer.

4.3.3 Local authorities

The municipalities belong to the Ministry of Social Affairs. There are 101 municipalities in Iceland (1.1.2004) and according to Laws on structures and planning, No.73/1997, they are obliged to maintain boundary information of lots, except farmland, in their administrative area. In every municipality in Iceland there has to be position(s) that supervise(s) legality of all constructions, spatial planning and land partitioning. In the smallest municipalities, this can be one and same person, referred as *planning- and construction officer*. Sometimes one man is responsible as planning- and construction officer for several municipalities or the role is outsourced to consultants. This is especially if a municipality is not financially capable to operate full position. Larger municipalities usually have separate planning- and construction officers, and the largest municipalities run specific departments to serve the same purpose.

The role of these officers is miscellaneous. Their main task in context to land recording is to keep track of all construction and land activity within their area and supervise the making of necessary documents, e.g. boundary maps. In urban area, they often maintain so-called plats and even operate sophisticated land information systems.

4.4 Transaction and registration procedures

If someone wants to sell or buy real property in Iceland the common practice is that these persons (legal or natural) find each other by using real estate agency. At each real estate agency there must be at least one licensed agent that conventionally takes care of making the contract, finalise mortgaging and submit necessary information to the registrar at the county office. It is obligatory to use licensed real estate agent when transferring ownership of a real property except in law cases where barristers or solicitor can conclude the matter (no.99/2004). Thus serve the licensed real estate agents in Iceland similar role as licensed notary in many countries, like the Netherlands. The country registrar though completes the notarisation procedure in Iceland.

Table 2: Procedure and actors when subdividing land parcel in Iceland.

Context

- Owner (seller) splits up a parcel before selling it to another person (buyer).

Active actors

- Seller, municipality officer, county registrar, Land Registry of Iceland

Passive actors

- Buyer, neighbours, holder of specific rights in the unit, bank/mortgagors and diverse institutions (natural and historical preservation, planning and agriculture).

Trigger

- Seller approaches a municipality officer

Sub-activities

Subdividing parcel:

1. Seller approaches planning- and construction officer of his municipality and asks him to create new land identifications for the new parcels he wants to split up from his land and sell.
2. Seller appoints a certified person that makes a plat of the new parcel showing the boundary of the original parcel along with delimiting new parcels. This is referred as ‘specific plan’ (Ice. ‘Deiliskipulag’). It often also shows building lot (associated to building permit), access to lot, connection to utilities etc. If boundaries have not been measured and registered as such between neighbouring parcels, the neighbours have to give their consent for legality of the plan.
3. The municipal officer obtains unique land id for the new parcel, creates a property document to establish it as a real property in the land registry and pre-registers it in the land registry database with a temporal status. In a property document the following information are registered:
    a. Name of the parcels as confirmed by relevant authorities.
    b. Land ID of the established parcels (can be many established at one in a document)
    c. Land ID of original land
    d. Plat confirmed by relevant planning authorities (the municipality). The area measure of the original parcel has to be registered both before and after the splitting. Also the area measure of the new parcels.
    e. Real estate ID of parent parcel (corresponds to its land ID).
    f. Name and identification number of owner of the parcels. This document registers the original owner as the owner of all parcels; change of ownership has different procedure.
4. The new property document is sent digitally to LRI for inspection and coordination. They are responsible for appraising the value of the new parcels. Completed it is forwarded digitally to the county registrar.
5. The property document along with plan is sent directly to the county registrar from the municipal officer, which registers owner and appraisal value to the parcels and establishes the new property in the land registry according to the property document. Finally it updates the Land Registry Database.
6. The plat is archived at municipal office and county archive. Some municipalities do integrate the plats into own land information system (LIS / GIS).
7. The textual information of the changed ownership (who owns, what, monetary values) can be accessed digitally at the land register database operated by the Land Registry in Iceland.

Transaction

1. Seller and buyer are brought together by certified real-estate agent that is responsible for making all legal documents, along with checking potential mortgages, easements etc. that can be resting on the real property.
2. The real estate agent prepares a deed, listing all necessary information with the signature of consent from both parties.
3. The deed is sent to county registrar for notarisation.
4. County registrar registers the transaction in the notary book and updates the land registry database.

Related Activities

- None

When all necessary documents have been provided to the legal registrar, it finalises the transaction of the real property by registering or altering the rights in the register book together with updating corresponding information in the LRD (no.39/1978).

Table 2 describes the transaction process when a land parcel is sub-divided and sold in Iceland, borrowing template as used by Stubkjær (2002). Some terms used there, like land ID and real estate ID are discussed in more detail in following sub-chapter.

4.5 The land registry database (LRD)

New laws on "property appraisal and registration" No.6/2001 established the implementation of uniform digital land registry for whole Iceland that integrated property records, house records, property valuations and notary books into one central database, the already mentioned LRD. Moreover, it is the legal role of LRI to maintain the database in cooperation with relevant local authorities like county registrars (act as notaries) and municipalities (Guðmundsson 2000; Matthíasson 2003).

The LRD comprises of four sections that all have their specific contribution to the whole (Alþingi 2004a; No.6/2001 & No.81/2004):

- *Base-section* is where properties are distinguished and given unique identification numbers like *landID*, *addressID* and *realEstate*ID. This is where the land registration models resides and consequently forms the heart of the land registration as the other sections points to here;
- *Construction-section* preserves general information concerning buildings and their general or particular usage. This is for example blue print of a building;
- *Property-section* contains appraisal papers along with estimated property market valuation and building costs;
- *Registrar-section* of the land registration embraces information about mortgages, collaterals and encumbrances along with other things written in the notary books; and finally;

Together these sections make up the LRD, stored centrally in an Informix relational database at the LRI. The client-server approach is employed as users responsible for manipulating the content of the database connect remotely to it through special client interface provided by the LRI. With login and password the database can also be viewed through LRI website (http://www2.fmr.is/landskra) and at the same site is also a restricted public version available for free. A new feature available when exploring the free version of the database is the availability of viewing orthophoto of the selected property. This is provided through cooperation of LRI and private photogrammetry firm Loftmyndir ltd. that maintains a separate spatial referenced address- and orthophoto repository.

The connection between spatial delimitation of parcels and land registration would be through the base section as it is where the registration model resides. If the proposal on implementing nationwide cadastral mapping as a part of the LRD materialises it could create yet another section that could be similarly documented:

- *Cadastral-section* preservers the delimitation of land parcels and other real properties in Iceland.

4.5.1 LRD data model

According to best knowledge there does not exist any explicit formal data model describing registration in LRD. After enquiring LRI, two entity-relationship diagrams[5] where found, but with restricted usability (see Figure 24 & Figure 25). The restricted usability is because they are authorless; without detailed explanation of classes and attributes; contradict; and, with some of the symbolisation, not standard for entity-relationship models. Actual table definition from the database was further not available.

With this in mind it was decided to make own model that could describe the present Icelandic land registration in more illustrative way by using UML notation as is employed for the CCDM.

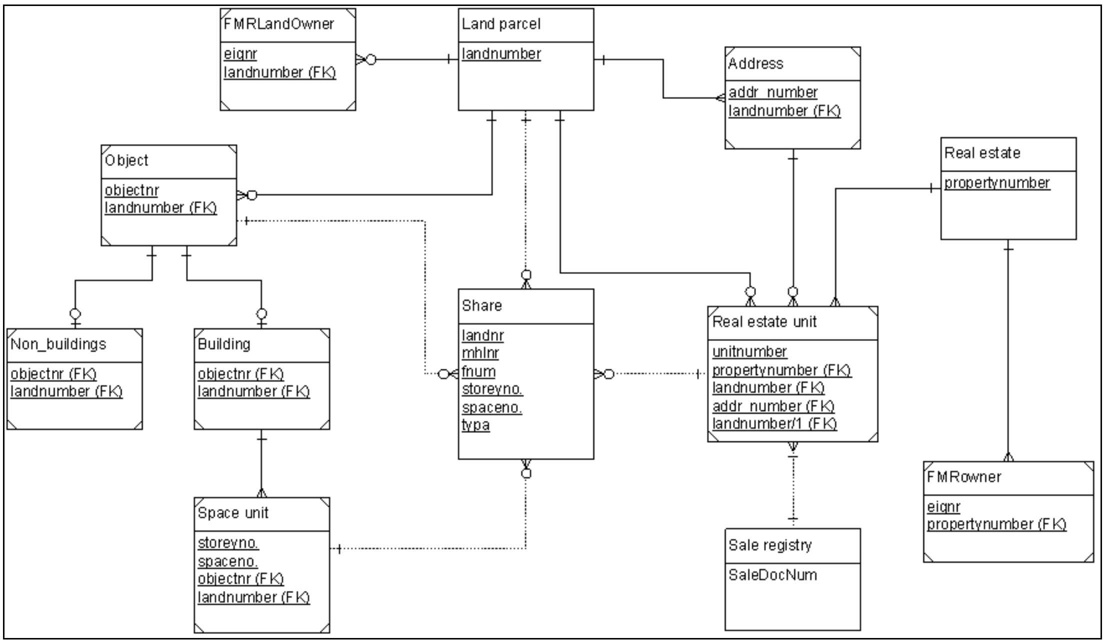


Figure 24: Model for the Land Registry Database I (image: Barry 2005).

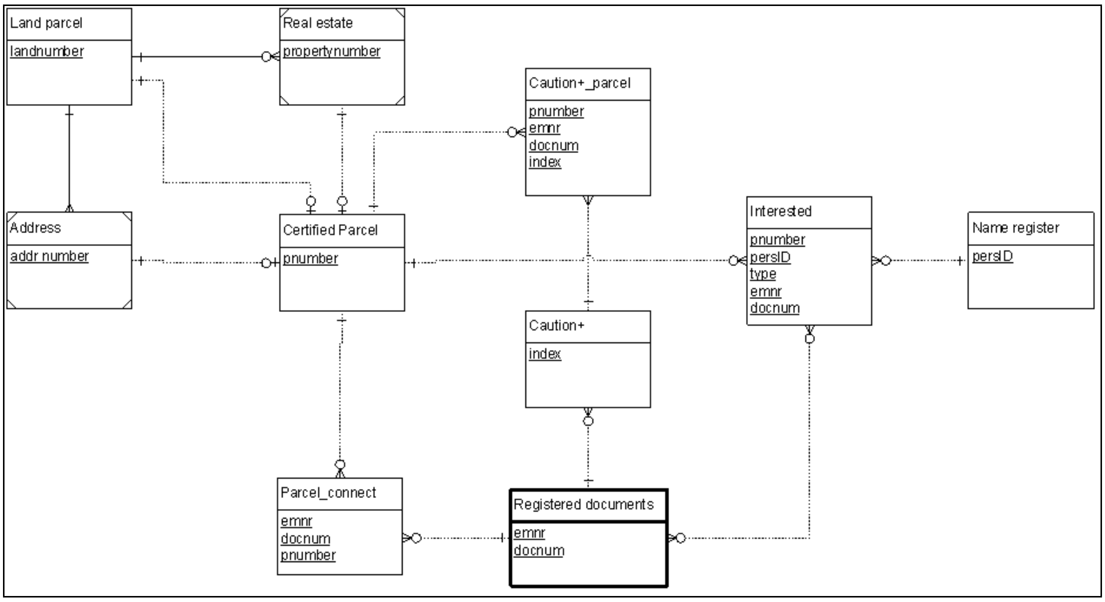


Figure 25: Model for the Land Registry Database II (image: Barry 2005).

[5] ER-model is the traditional method to design and model RDBMS. It is however limited to express objects and therefore is UML preferred in OR-DBMS design.

4.5.2 Reverse-engineering of LRD data model

It is not an easy task to come up with UML diagram of the LRD data model with out having access to the actual database that it is stored in, or other similar sources of information. Therefore, after gaining special view access to LRD it was decided to do a reverse engineer on the information hoping it could lead to usable data model.

It was decided to design the model in UML instead of creating ER-diagram as is conventional for RDBMS to enable comparison with the CCDM and facilitate inclusion of spatial data types.

It should be stated here at the beginning of this discussion that the model is only intended to reflect the LRD as observed by author. It will never be detailed enough to cover all classes and attributes, but hopefully be able to give an idea how land is registered in Iceland and what relationship it has with other immovable properties. The primary objective of doing this is to have something to compare to the CCDM and form a foundation for further discussion on its applicability for land recording in Icelandic.

4.5.3 Mapping of classes

To start with it was necessary to map relevant classes and attributes. The classes that were observed by exploring LRD are as followed:

- **`Address`**: This class is outside the LRD database but still employed much within it. Every address name in Iceland has a unique address identification, which the LRD uses to refer to, instead of street name, number and postal code. This information is maintained by LRI in an address repository.
- **`Person`**: either legal or natural as maintained in the National Register of Persons operated by Statistics Iceland (www.hagstofan.is). Join between the two databases is through unique person identification number (10 digits).
- **`RightsAndRestrictions`**: Both direct and indirect rights are stored in the LRD although only the former is accessible through the LRD interface. Direct right can be of two different kinds: right to a land or right to a real estate, with the latter embracing all real properties except landownership.
- **`LandParcel`**: Has unique land identification (6 digits) as allocated by the LRI. A land parcel can contain several real estate complexes. Land parcel has a unique address given by address identifier.
- **`RealEstate`**: Represents the aggregation of different real estate units that together form a real estate object for registration. The real estate identification is the same as the real estate unit identification of the main unit. Refer to Example 11.

  It is noteworthy to mention that the relationship between a real estate and a land parcel is many-to-many. Thus, a real estate can theoretically be spread over several parcels. Another thing is that if there are no real estate unit existing, because of one or other reason (as real estate unit id has not yet been established), an address id can be used temporally as real estate identifier until a proper real estate unit id has been issued.

Example 11: Example of composite urban real estate. Appendix C: Example of a Record in LRD, contains similar example for parcel registration if rural farm parcel.

Real estate complex composites of:

- three-storey apartment building with four apartments; and,
- separate garages for each of the apartments.

Real estate composites of:

- two-storey, second floor apartment to left;
- cellar storage room; and
- separate garage.

As a part of the same complex, the apartment and the cellar classify as two real estate sub-units, merged as one real estate unit. The garage however gets separate real estate unit, as it is a part of separate real estate complex. Together these two real estate units are registered as an integral real estate object using the real estate unit identification of the apartment/cellar.

Description	Complex	Sub-Unit	Unit	Real Estate Unit ID	Real Estate ID
2nd floor apartment	01	0201	0201	110-1105	110-1105
2nd floor apartment	01	0301	0201	110-1105	110-1105
Cellar	01	0002	0201	110-1105	110-1105
Garage	02	0101	0101	110-1106	110-1105

Appraisal complex number (complex): identifies a real estate complex within the same parcel.
Appraisal sub-unit number (sub-unit): identifies functional space in a building, with first two letters a storey number, and last two letters the location of space in a storey, counting from left to right.
Appraisal unit number (unit): identifies spaces that belong together within a complex. The rule is that the sub-unit number of the main space (where the main entrance is) is used as the unit number.

- `RealEstateComplex` (or even better `RealEstateComposite`): A house, apartment building, separate garage, diverse natural resources etcetera, in other words: everything that logically is appraised together is referred as a real estate complex. A real estate complex can have an address which is the same as the address of the parcel it is placed on, if there is only one complex. When there are more complexes on the same parcel, with own address, each gets own address identification. Refer to Example 12. As unique identifier for complexes a special number is used, comprised of municipality number, district number and lot/parcel number in addition to the appraisal complex number. Here referred as location number. `RealEstateComplex` should be completely contained by single land parcel. There are though some exceptions of them crossing boundaries but that should be regarded as invalid relationship.
- `RealEstateUnit`: Is a unit within a real estate with unique real estate unit identification (7 digits). A unit contains one to several sub-units that are stored and retrieved from the appraisal table. Real estate unit types are not restricted to buildings, as all kind of rights or usage can be registered as such, e.g. fishing right in a lake or a river, seal hunting, exploitation of a bird cliff, accumulation of eider down and cultivation etcetera.

Example 12: Use of address identifications in Iceland.

Three different real estate complexes are placed within delimited boundaries of a parcel. The address name and identification of these could be as follows:

Type	Address	AddressId
Appartment building	Mekelweg 7	1301215
Appartment building	Mekelweg 9	1301216
Appartment building	Mekelweg 11	1301217
Parcel	Mekelweg 7-11	1301218

- `RealEstateSubUnit`: A sub-unit is registered in an appraisal table with an appraisal sub-unit number and represents a unit of functional space. A two-storey real estate unit is for instance recorded as two separate sub-units in the appraisal table. See Example 11 and Table 3.
- `AppraisalTable`: For every real estate complex is made special appraisal table as represented in Table 3.

Table 3: Example of an appraisal table for real estate in Reykjavík.

Address: Reykás 22, Reykjavík
AppraisalComplexNum: 01

LocationID	Sub-Unit	Con.	Description	Bphase	Area	Cubic	CeilingH	Balcony	Updated	RealEstate-UnitID
0000-01-4383303-01	0000		Shared space	7	80,10	0,00	0,00	0,00	22/05/1994	-
0000-01-4383303-01	0001	0001	Apartement	7	78,60	0,00	0,00	0,00	22/05/1994	204-6380
0000-01-4383303-01	0002	0002	Apartement	7	78,60	0,00	0,00	0,00	22/05/1994	204-6381
0000-01-4383303-01	0100		Shared space	7	18,50	0,00	0,00	0,00	22/05/1994	-
0000-01-4383303-01	0101	0101	Apartement	7	113,90	0,00	0,00	9,80	22/05/1994	204-6382
0000-01-4383303-01	0102	0102	Apartement	7	113,90	0,00	0,00	9,80	22/05/1994	204-6383
0000-01-4383303-01	0200		Shared space	7	13,50	0,00	0,00	0,00	22/05/1994	-
0000-01-4383303-01	0201	0201	Apartement	7	116,40	0,00	0,00	9,80	22/05/1994	204-6384
0000-01-4383303-01	0202	0202	Apartement	7	116,40	0,00	0,00	9,80	22/05/1994	204-6385
0000-01-4383303-01	0301	0201	Part of apartment	7	36,50	0,00	0,00	0,00	22/05/1994	204-6384
0000-01-4383303-01	0302	0202	Part of apartment	7	36,50	0,00	0,00	0,00	22/05/1994	204-6385

Other classes could not easily be observed through the web interface of LRD but by exploring diverse resources like handbooks for county registrars (FMR 2003) and the laws on properties among others, several other classes were discovered, e.g.:

- `RealEstateAgent`: Has the responsibility to prepare documents for notarization, like sale contract. Also often mediator between money provider and a buyer in case of mortgaging the real property. Can be liable for incorrectness of notarized sale contract because of negligence or intentional wrongdoing (Andrésdóttir 2005).
- `MoneyProvider`: Provides the money by obtaining collateral in the real property.
- `Mortgage`: Instrument used when a borrower gives a lender a lien in a property as to secure the loan.
- `LegalDocument`: Deed, sale contract, mortgage or just anything that is notarized by a county registrar and thus gaining legal status.

4.5.4 The LRD diagram

The diagram presenting the observed LRD data model is displayed in Figure 26. There are several things in it worth more detailed discussion.

Information on land and structures are maintained together. A real property is either regarded as:

- planar partitioning (land parcels);
- non-planar partitioning (e.g. structures, cultivation, natural resource).

Separate ownership is allowed between these two groups in a way that an owner of a building does not necessarily have to be the owner of underlying land parcel. In these circumstances is a permanent relationship arranged between the land parcel and the building with special long-term lot contract. In Reykjavík for instance is a tradition to make these contracts to 75 years. When the contract expires the ownership of the lot is commonly transferred to the house owner (Matthíasson 2003).

Consequence of this is that land has a bit complex relationship to real estate objects. As the diagram indicates the relation ship between **`LandParcelObject`** and **`RealEstateComplex`** is one-to-many. Resulting in that **`RealEstateComplex`** cannot overlap two separate land parcels. The relationship between land and **`RealEstate`** is however many-to-many. Examples are apartment buildings placed on one parcel (each entrance counting as separate complex) with garages on other parcel. Here the garages are appraised separately from the apartments that they belong to, but registered together as integral **`RealEstateObject`** (this example refers to Fellsmúli 17-19 in Reykjavík).

In LRD the **`RightAndRestriction`** class is not declared as as a link attribute between **`Person`** and **`Object`** but as a separate intermediate class. This allows a person to hold multiple rights in the same land parcel, something that is not possible with the notation in the Cairo CCDM model. The developers of the CCDM have also identified this, with the newest version of the model notating like here.

Spatial delimitation of parcels is completely ignored in the LRD and the only references to parcels boundaries are some qualitative statements. The spatial extent and boundaries of parcels is of great importance as it gives among others:

- creditors more security for the money they lend;
- registrars more certainty of what they are notarising;
- better ground for land taxation and management;
- municipalities better data to work with and saves double work; and,
- owners a tool to secure boundaries and avoid costly disputes.

Many diverse actors, both private and public, currently gather the spatial extent of parcels. There is no central access point to access this data and much of it is only available on hard copy. The following section goes in detail of the availability of cadastral data in Iceland before the discussing the making of Icelandic cadastral model (ICM) encompassing both registration of rights and spatial delimitation.

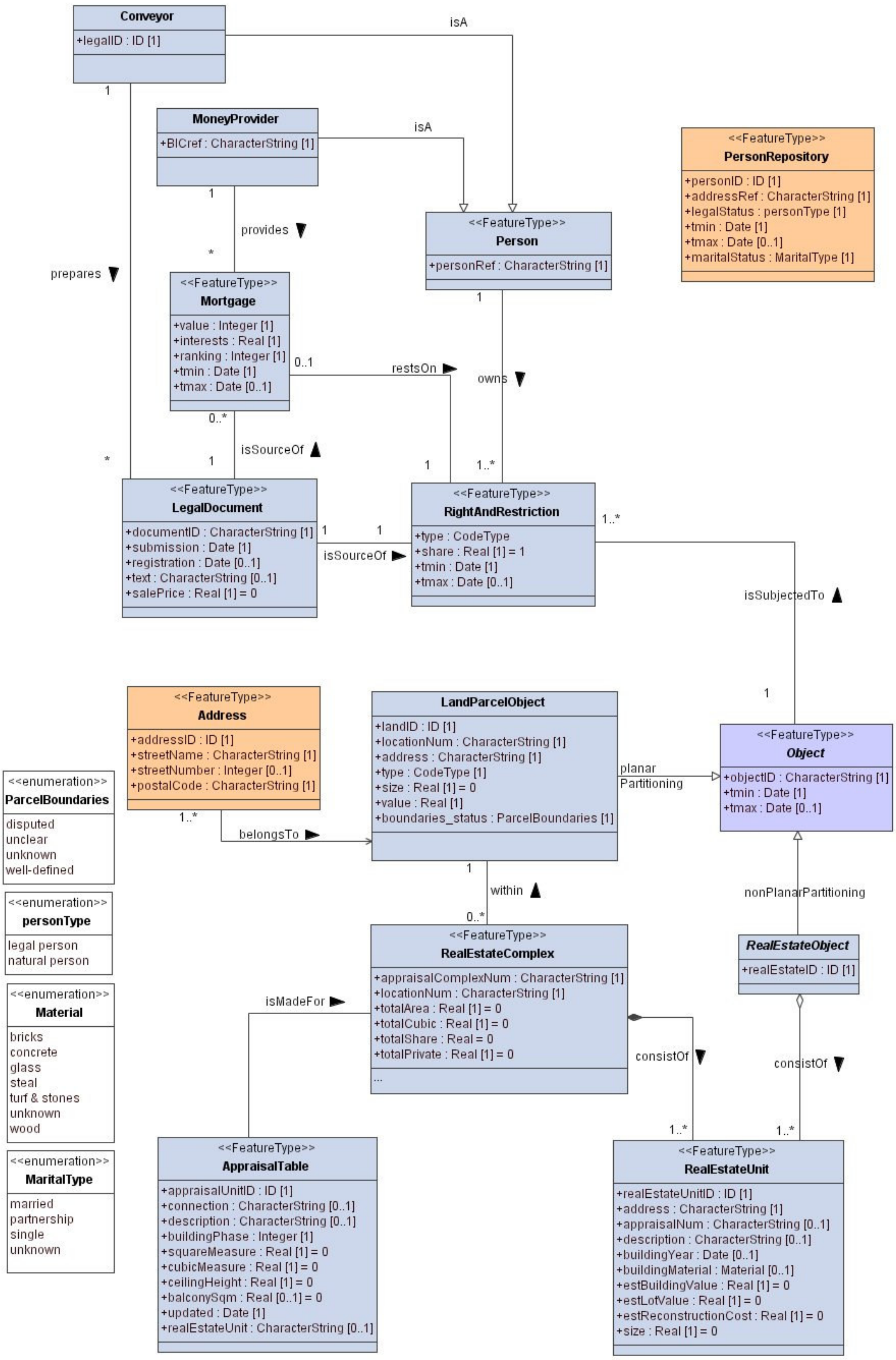


Figure 26: The land registry database model as observed by author (image: Author).

4.6 Availability of spatial data to use in cadastral SDBMS

As discussed earlier there does not exist one central place in Iceland to access cadastral maps. There are many diverse municipalities, institutions and private parties gathering data and often without consulting each other. The cadastral maps also take diverse shapes in regard to format, coverage, structure, accuracy and attributes assigned. This causes inconsistencies in the whole registration, making comparison or integration of separate systems difficult.

Around 90.000 parcels are in Iceland with approximately 10% rural. The enumeration on next pages is an attempt to "map" present availability of spatial data to use to map these parcels, both type and purpose. It is based on several references, e.g. Nytjaland (2005), Óbyggðanefnd (2005), Sigurbjarnarson (2000) and own experiment in as a consultant for landowners in delimitating and notarising legal boundaries. Special thanks are to Daði Björnsson, a geographer at Loftmyndir ltd., and Tom Barry at LRI, who provided additional information and recommendation on this subject.

4.6.1 County offices

The county offices preserve notarised documents stating the address, size and at times the location of legal boundaries. For urban parcel only the address and size of the parcel is notarised with a reference to an urban/municipality plan for exact location of boundaries. For rural areas, textual boundary descriptions dating from late 19th century are most common though maps or list of coordinates have grown in popularity in recent times.

These boundary descriptions can be highly ambiguous as they use local place names, landscape and manmade features to locate boundary monuments.

First of all is Icelandic landscape very dynamic in nature, where weathering can erode and change the appearance of land features in a short time.

Secondly, Iceland has got through massive cultural and social changes in last century influencing land use, whereas most of the boundary descriptions were composed in late 19th century. This is especially due to rapid development of urban settlement, more intensive cultivation and draining of land.

Finally, the relationship between man and land is constantly weakening. The generations that grew up in the farming society of early 20th century, in close relationship to land and familiar with using local place names, are little by little passing away. Younger generations are less familiar with place names, especially as they use different methods to locate themselves with the evolution of new feature to refer to (addresses/structures, road names, irrigation ditches, etcetera), modern maps and GPS technologies.

An example of typical Icelandic boundary description is presented below and clearly shows how confusing they can be to unfamiliar observers and how highly they are depended on subjective interpretation. This description was made in 1884 for the farm Arnarbæli, in Grímsnes area, South-Iceland. The translation was done by the author and relevant place names/monuments are underlined (Sigurbjarnarson 2000, [1]).

> Corner monument: The ruin close to Heiðrimakelda-kill, south of Oddholtsmúla-mull; from there is a line of sight, west to Héðinslækjabotnar-depression, wherefrom the boundary follows first the creek Héðinslækur-creek, and thereafter Höskuldslækur-creek to the river Hvítá-river. To east of the above mentioned ruin decides Heiðrimakelda-kill the boundaries south to Þverkelda-kill conflux that runs from Galtatjörn-pond; wherefrom this conflux to a kill mouth in the pond that runs south and south-west and this kill decides all the way to Rauðkollsflóð-mire; then a straight line (not line of sight) to a tussock on Markholt-hill, thereof to the start of Markarkelda-kill, which then decides the boundaries to Hvítá-river, that bounds to already mentioned Höskuldslækur-creek.

4.6.2 National archives

The National Archives preserves most boundary descriptions that are also kept at the county offices. It also preservers old documents the predecessors of current descriptions. This is an important source material to solve boundary disputes, when searching for the best root of title and location of legal boundaries.

4.6.3 Municipalities

Most municipalities keep records of urban-lots and their spatial extent within their administrative area for their own purposes and according to legislative obligation.

Reykjavík is the best example operating its own land information system (LIS), named after the Icelandic abbreviation *LUKR*. Other large municipalities operate also digital LIS in diverse forms and extent. Common to all these systems is emphasises on especially urban parcels in addition to other geo-information (e.g. topography, utilities etcetera). Rural parcels are typically not included or maintained in such LIS.

LIS is in most cases too expensive for rural and less populated municipalities that rather lean on analogue file archiving and notarised documents at the county offices.

General spatial plan implemented at municipality level include also parcel boundaries, but have the shortcomings that the origin (reference) and accuracy of the boundaries is often unknown, or the boundaries are only approximated. The spatial reliability of these sources is consequently uncertain. The municipalities themselves in cooperation with the 'National Planning Agency' implement these plans.

Specific plan is another type of planning that landowners need to implement in ordinance to the general plan made by the initiative of the municipalities. Those plans often show legal boundaries and can be quite accurate source of information for that purpose. These plans can be accessed at the municipality offices.

4.6.4 Ministry of agriculture

The Ministry of Agriculture is according to laws obliged to supervise all farmland and farmsteads. As there is no cadastre existing, covering the whole country, the ministry has consequently problem to fulfil its role of supervision. To address this problem, the 'Agricultural Research Institute' (RALA) has been working on a project called 'Nytjaland', to collect the legal boundaries of all farms in Iceland. RALA has now drawn boundaries for 7967 farms (as April 2005), whereas 4781 of these have been identified and assigned a land identifier as employed by LRI. Further 274 consist of land classified as either commons or disposed parcels, while 2822 have not been assigned an identification at all (Barry 2005; Nytjaland 2005). See Figure 27. The future responsibility of maintaining the Nytjaland project has now been transferred to the GIS section of LRI.

The boundaries drawn by Nytjaland are mostly based on satellite images with 15-meter ground resolution. The strategy is to visit farmers, or establish gatherings, where landowners point out the boundary of their own farmstead and even also those that they are familiar with.

Additional boundaries have also been obtained from several external sources (mainly institutions stated in next section) or collected with different methods (e.g. GPS, aerial photographs and maps). The accuracy of the boundaries in Nytjaland database is therefore overall rather low.

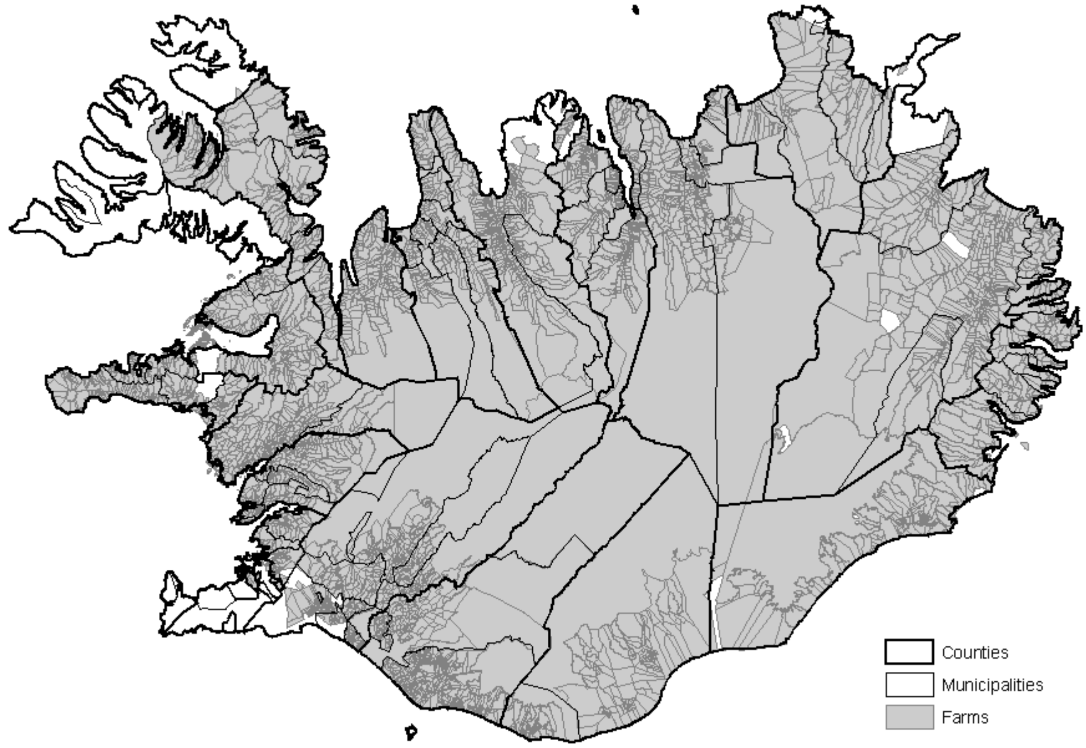


Figure 27: Boundaries of farms (indicated in grey) that have been sampled and located by Nytjaland in April 2005 (image: Author).

The collection is also based on the "best" knowledge of the farmer, but not on legal documents leading to additional errors in the location of boundaries. A formal consent of landowners is not sought and Nytjaland states that the boundaries kept in the database have no legal relevance, not today or in the future. They were primarily collected to help the Ministry to fulfil its legal role of overseeing agricultural activities and land use on farm land (Nytjaland 2005).

4.6.5 Diverse institutions

Several institutions collect ownership boundaries as a part of their activities or special projects.

National Land Survey of Iceland (NLS) has assembled boundaries in connection with general mapping procedures and in collaboration projects with LRI. NLS also has the role of maintaining administrative-, jurisdictional- and election district boundaries in Iceland. Another important role of NLS is to operate and maintain the national geographic reference system in Iceland. Currently Isnet93 datum is the standard datum but will be replaced by Isnet04 in not so distant future (LMÍ 2005).

Land Registry of Iceland (LRI) has carried through several minor experimental projects in spatial delimitation of parcels. Currently they have two positions with the role of consulting cadastral mapping in municipalities along with carrying out researches for the implementation of a nationwide cadastre.

LRI also works in collaboration with NLS and other governmental institutes in developing approach to nationwide cadastre in Iceland, along with being involved in collecting data from various sources (government institutes, municipalities, private sources) and developing methods to integrate these data (Barry 2005).

Óbyggðanefnd is a committee established by the government to orderly determine the boundaries between 'property-land' and 'collective-land'. The difference is that property-land is a land subjected to either private or public ownership, while collective-land belongs to the people in the country, not subjected to either private or public ownership with explotation controlled by central government. Collective-land is mostly found on the unpopulated highland, deserts and glaciers.

Public Roads Administration (PRA) due to planning, legal procedures and construction of public roads the PRA needs to collect information regarding legal boundaries. The extent is however limited to an area neighbouring the road system.

State Energy Authorities and Landsvirkjun are concerned about the construction of hydropower facilities, reservoirs and electrical transmission lines. In context to these practices, the institutions have to collect boundary information to be able to minimise design costs, efficient resource management and to compensate relevant landowners that are affected by their activities.

Environment and Food Agency is not concerned about parcel boundaries but restriction to land uses. E.g. the spatial extent of preserved nature areas and national parks. To accomplish this the agency publishes regularly list of areas and places that are subjected to restriction of land use (UST 2005).

National Museum of Iceland has similar role as the Environment and Food Agency in restricting land use in protected sites and areas, though here it concerns culture and heritage instead of nature.

Icelandic Institute of Natural History has through the years collected and published parcel boundaries parallel with extensive vegetation mapping of Iceland. These boundaries references are included in the 'Nytjaland' project.

Other institutions worth mentioning as potential source for legal boundaries, especially in rural areas, are e.g. utility companies and regional forestry projects.

4.6.6 Private companies and organisations

Various private consultants collect information of ownership boundaries when working explicitly for clients or just as a side product of diverse projects. Although this information is digital it is not easily tangible because it is preserved in separate projects and with haphazard accuracy, detail, map scale etcetera.

The private company Loftmyndir ltd. has here special contribution to proposed cadastre, as it is presently developing an address database of all buildings in Iceland now counting 60.000 addresses (Loftmyndir 2005). This project could eventually be valuable to simple locating parcels (e.g. as a contribution to a *point cadastre*).

Another source for cadastral information is among private organisation, especially 'Skógræktarfélag Íslands', an association of local forestry societies in Iceland.

4.6.7 Individuals

Finally to mention are diverse individuals that have on their own incentive collected legal boundaries in Iceland. First and foremost to mention is the former professor of geography in the University of Iceland, Gylfi Már Guðbergsson (1936-1998) that dedicated his lifework to collecting legal boundaries in Iceland. A large part of his collection is now parts of the Nytjaland project while his inheritor preserves some additional. Many students in geography were also influenced to collect boundaries as a part of their studies.

4.7 Future vision

LRD is still that new in Iceland that neither all legal documents nor real properties are yet included in the registry. According to recently proposed bill (Alþingi 2004-2005) the target to finish current development of the LRD during the period 2005-2008. This does however not take into account any integration of spatial data into the database.

Concerning recording spatial extent of parcels there is nothing decisive yet (Barry 2005). It is the current aim to review the current status of geographical information within LRI and simultaneously LRD, by making a policy document that explores available options and recommends a viable strategy. It is important that this strategy tackles how this data is integrated with existing data structures at LRI by integrating spatial data into its overall data model. The making of this policy document is scheduled in begin of the year 2006.

The ultimate goal is that the LRD will provide one-stop access to information concerning land ownership in Iceland, maintaining information like (Barry 2005, e-mail):

- Land parcels, location (co-ordinates)
- Constructional specifications, building units, material etc.
- Valuation data
- Land tenure and real estate title [note: deeds]
- Mortgage listing
- Graphical representation of boundaries (digital maps)
- Scanned registration documents

To fulfil this goal LRI has taken the responsibilities of maintaining the data already gathered by the Nytjaland project. This data, even though its positional quality is rather low, can serve as basis for further development and with time be upgraded.

Another idea that has been brought to discussion is to include registration of movable objects in the LRD. Though it sounds surprising to register objects like land, buildings, cars and airplanes in the same register it could proof convenient for the authorities. After all, at a certain abstraction level this is about registering right between a subject and an object, no matter if the property is movable or not (Ingibergsson 2005).

4.8 Remarks

This chapter has covered how present land registration in Iceland is organised, modelled and the availability of spatial data usable in cadastral development. It can be concluded that the general framework of land registration is relatively good and adding spatial dimension would not revolt current system, simply extend it.

Another finding is that the relationship between land and structures is more complex than e.g. is in the Netherlands. An owner of a building does not necessarily have to be the owner of the parcel, as long permanent rights are established between the structure and the land. What is considered a permanent right is however not defined clearly, but long-lease contract ranging over several decades is generally thought as one.

Finally it can be stated that there exist some spatial data to found delimitation of parcels in Iceland on. This data is though heterogeneous in both nature and quality. Special approach is needed so this data can be used as start of cadastral spatial registration, allowing the quality to be upgraded incrementally during the evolution of combined cadastral registration of textual and spatial data in Iceland.

Next chapter discusses the making of Icelandic cadastral model, questioning if it could be made simply by fitting LRD into the CCDM, using the spatial classes of the latter.

5 MAKING AN ICM

Is it possible to use the CCDM as a basis for the integration of spatial information into LRD? By this, develop an Icelandic cadastral model (ICM), which is simply extension of the CCDM? What are the prerequisites? Do there exist conceptual differences between the two models (CCDM and LRD) resulting with that ICM cannot simply be extended from CCDM without adjustments? Answering these questions is the objective of this chapter.

The chapter starts with describing the main premises that govern the integration of spatial data into the LRD. Based on these findings a desired spatial model is proposed which complies with the CCDM. This results with a data model that will be referred as ICM. Section 5.3 presents the ICM and briefly describes its organisation of classes and how to interpret it before moving to section 5.4, which compares the two models. There is presented the conclusion of this research that the ICM cannot simply be extended from the CCDM without adjustments. Remarks in section 5.5 conclude this chapter.

5.1 Premises of implementation

There are several premises that have to be considered before integrating spatial data into LRD and develop an extensive ICM. Iceland is a large island, with the relatively few inhabitants and most activities concentrated in one corner of the country. Less inhabited rural areas, heaths, deserts, wastelands and glaciers cover the rest.

This is also one of the reasons why Iceland is so scarcely mapped in sufficient map scale (only 1:50.000 is available of the whole country) to support cadastral mapping. Thus, present cadastral registration depends on surveying either on ground or by utilising remote sensing techniques.

Another thing that this influences is the financial capability to support spatial delimitation of parcels. Although the southwest corner of Iceland is densely populated with roughly 75% of Iceland total population within 60 km of Reykjavík, the rest of the country is not. Causing huge expenses if the whole country is to be mapped at once with the same accuracy that neither governments (local or central) nor individuals are prepared to pay for.

5.1.1 Gradual update available cadastral data

First of all is the cadastral data available in Iceland to integrate. As discussed earlier the positional quality of the data already gathered can be questioned. According to Ingvarsson (2004) were in August 2004, total 88.001 registered land parcels in Iceland. These parcels are further split into 80.637 lots, and 7.364 of larger parcels like farms. In Reykjavík alone there are between 15-16.000 parcels (Hallgrímsson 2004).[6]

However when investigating the availability of spatial data representing these parcels it was found that the Nytjaland project has approaching 4000 parcels mapped with land id, mostly farm parcels, while the LIS of Reykjavík has around 14.000 parcels mapped with land id. It can be estimated that merely 30-40.000 parcels in Iceland have mapped extent by summing up the diverse local or public LIS existing in Iceland (Barry 2005). This is the data that is needed as a start of the development of nationwide coverage of spatial cadastral registration. Data that in almost all aspects is heterogeneous, produced in different GIS systems, has different set of attributes, is of different quality and origin and not necessarily equipped with sufficient metadata. It is therefore important that the

[6] According to Loftmyndir (2005) the current number of parcels in Iceland is closer to 91.600.

chosen spatial model makes it capable to trace correctly the origin of the data so that this data can be gradually upgraded in quality.

With around 2/3 of the parcels in Iceland without any reference to boundary information at all other means have to be employed. Loftmyndir ltd. has developed an address database connecting addresses and land id to around 60.000 parcels in Iceland (Loftmyndir 2004).[7] Here the extent of the parcel is unknown but approximate location can be derived from the address location. This data can be very useful as a temporally step in the development of cadastral database covering the whole country.

Still there are 30.000 parcels without any boundary information. However, the Icelandic registration system is organised in such way that every parcel registered in the LRD has a location identification number that can be used to approximate its location within a municipality or a 'mother'-parcel'.

In the above discussion it can be observed that different methods can be used to represent geographical location of parcels in Iceland. This could be defined as:

1. parcels measured with sufficient enough positional accuracy to be qualified to have fixed boundaries (**`Parcel`**). This is especially important within urban areas where parcels are small and land values high;
2. parcels with general boundaries where only approximate location of the boundaries is recorded (**`Parcel`**). The parcel belonging to this category can also consist partially of fixed boundaries;
3. parcels with not complete boundaries or no at all, and represented by a point and or a spaghetti polygon (**`PointParcel`** / **`SpaghettiParcel`**); and,
4. parcels with no spatial reference at all (**`TextParcel`**).

Parcel with fixed boundaries has should be sufficiently surveyed up to some predetermined standard as defined by law or regulation. General parcels have general boundaries, meaning that only approximate location of the boundaries is known and mapped. Only by exploring in the field, the exact location of the general boundaries can be observed. Typically are these kind of boundaries are utilised in rural areas where landscape, like hills or depressions, forests, rivers, ditches and dikes provide natural boundary evidence (Dale & McLaughlin 1999).

By looking at these four categorise we can observe two axes that control we can approach parcels and their spatial representation. One axis is the quality of location, ranging from being none, general where parcels have established boundaries of insufficient quality, to fixed where the parcel has all its boundaries with sufficient quality measure.

The other axe represents the quality of the feature presentation. At one end is NULL, thus no feature representation. Then is point representation, mixture of points and lines, polygons until the other end is reached where the parcel has all of its boundaries, forming a valid topological face.

The objective is of course that in the end of the development all parcels in Iceland to have fixed boundaries managed in topological structure. To be able to do this gradually, the prime requirement for the system is to handle exhaustive registration of surveys, and consequently the quality, of boundaries. This can only be achieved sufficiently with using topological spatial model as discussed in section 5.2.

By employing the CCDM (Cairo) notion of non-planar partition region with only little adjustments this can be achieved. In the CCDM the concept of Non-Planar-

[7] It can be estimated that all parcels that have boundary reference in Iceland are also included in Loftmyndir's database.

Partitioning Region (**`NPPRegion`**) was introduced. It is meant to use over areas where parcels have not yet been identified or properly brought into cadastral system and is of great benefit to the ICM. Within **`NPPRegion`** all kind of data can exist, like inconsistent agglomeration of features like points, lines and polygons (spaghetti data) indicating the extent of a parcel.

In Iceland however, the existence of parcels with no geometry at all, requires another option for **`NPPRegion`**, a **`TextParcel`**. **`TextParcel`** is here defined as a textual source of information with limited spatial reference. The worst-case scenario is a lost parcel with no or very little spatial reference. Location identification number should though indicate in what municipality the parcel is and approximately where it is within it. Then, parcels can also have special boundary description as was described in section 4.6.1.

Together this arrangement results with the class **`LandParcelObject`** as was described in section 4.5 (the class becomes abstract) having four specialisations in the ICM:

- **`TextParcel`**: Used when location of a parcel is unknown
- **`PointParcel`**: Used when location is known but the extent not. A point parcel can be supported with spaghetti lines or topological edges of already formally mapped parcel.
- **`SpaghettiParcel`**: Is used when raw data describing the extent of a parcel is initially imported into the system. Eventually this will become a parcel.
- **`Parcel`**: Is the formal way of registering parcels. A parcel has defined topological boundaries. Can be either general boundary, whereas the geometric features only indicate the approximate location of the actual boundaries. Or, fixed, when the actual boundaries have been surveyed with sufficient and predetermined accuracy.

As Figure 28 illustrates the ICM model employs also the concept of **`PartitionParcel`** as a way to partition the two-dimensional domain.

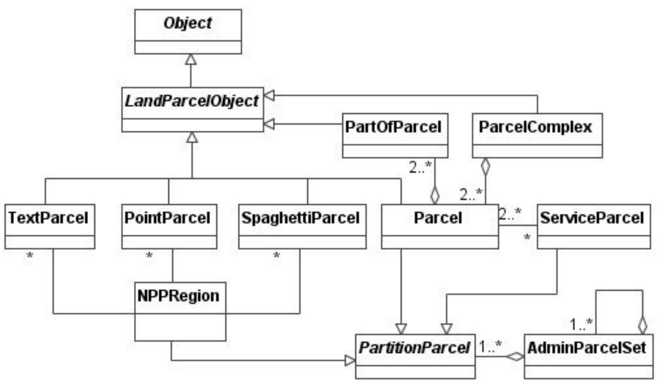


Figure 28: Conceptual model for parcel hierarchy as suggested for ICM (image: Author).

Two-dimensional, partitioning of land into **`PartitionParcel`** indicates that there can neither be overlap nor void between parcels. If we think of this in hierarchy we could imagine that the Iceland along with its territorial waters complete one parcel in a partitioning of the complete earth globe. We can subdivide the Iceland parcel further into two parcels on next hierarchical level: land and sea (hence land parcels vs. sea parcel). Land can further be subdivided into regions/districts used for election or jurisdiction purposes. One region contains at least one county that furthermore contains at least one municipality. Every municipality can finally be subdivided into land parcels.

This is what is meant with land partitioning, thus creating non-overlapping basic units (parcel) that can be aggregated to other units higher up in the hierarchy. Notice e.g. the **`AdminParcelSet`** class in Figure 28 and its relationship with **`PartitionParcel`** and itself.

The class **`ServingParcel`** is included in the ICM, as it can serve in registration of areas that are not subjected to direct ownership of persons, but still subjected to other parcels. Example of this is the interior of larger lakes, which we can refer to as 'lake-parcel'.

The Icelandic law (no. 15/1923 & no. 76/1970) state that ownership only extends maximum 115 meters from a waterside of a parcel. Exceeding these 115 meters is the interior of a lake, referred as a common, and can only be exploited by the owners of the adjacent land parcels. Here can the definition of **`ServingParcel`** become useful, thus the lake parcel can be defined as joining interest of adjacent land parcels. This is comparable to the 'joint property cadastral unit' as defined in the Norwegian cadastre (Mjøs 2002). Other uses of **`ServingParcel`** can be in urban areas where several parcels have interest in one 'common parcel' that services the others. Example could be a playground situated in the centre of several privately owned parcels.

5.1.2 Expenses

Developing complete and up-to-date cadastre system covering all parcels in Iceland is an extremely expensive project. However, this can be accomplished with sporadic cadastral mapping and gradual improvements avoiding enormous start-up cost. Law could stipulate that a parcel needed to have sufficiently surveyed and registered boundaries as prerequisites for transfer of ownership. The cadastral system would then gradually build up with time, whereas existing sources of cadastral boundaries could be used as temporarily replacement until the property would change ownership next time.

Some land parcels are though rarely transferred. These are e.g. parcels owned by legal entities like local or central government, and firms. Other procedures need to apply for these parcels and they brought more systematically into the system (as discussed in Dale & McLaughlin 1999).

Developing the cadastre as explained in the previous section gives the possibility to keep overhead costs very low. With a well-defined and structured cadastral SDBMS the development should be more or less automatic, but could take long time to be completed. One thing to keep in mind is, that with the sporadic approach the overhead costs is minimised while the cost per parcel is maximised. Contrary to the systematic approach where costs per parcel can be minimised by simultaneously survey and adjudicate many at once. This obstacle can though be met by mixing the two approaches, with landowners, or even whole municipalities, joining forces in surveying many spatially related parcels together, e.g. when reviewing municipality plan.[8]

5.1.3 Dynamic boundaries

The extremely dynamics and diverseness in Icelandic landscape are unique compared to many other countries. Ten percent of the country is covered with glaciers, with the glacier edge often used as a boundary between collective-land and property-land. In warm periods the glaciers retread, as they have been doing for last decades, and then advance in cold periods. From the glaciers stream glaciers-rivers that are highly fluctuate in flow. In the springtime, they are full of smelting water, while their runway is

[8] "Municipal plan: A development plan for a specific municipality expressing the local authority's policy regarding land use, transportation and service systems, environmental matters and the development of settlement in the municipality during a period of not less than 12 years" (No 73/1997 see translated English version Skipulag 2005, [1])

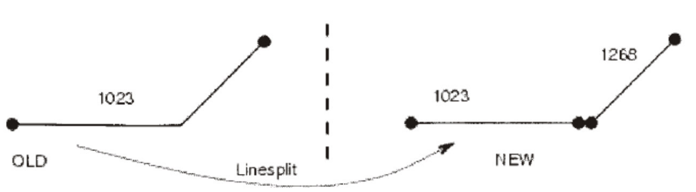
almost empty in the wintertime. Additional factors can cause the water flow to multiply, as was the reality in the volcanic eruption in Grímsvötn, Vatnajökull glacier 1996. When Skeiðará river temporarily grew from the 110 m^3/s average flow to 55.000 m^3/s, eroding everything in its way (OS 1996).

Volcanic eruptions with parallel lava flow have also been critical factor in changing Icelandic landscape in recent decades. In one volcanic eruption the coastline of south-Iceland moved 4 km southwards creating a small sandy spit referred as Kötlutangi. Since then the force of Atlantic sea wave has eroded this spit almost completely and transferred the sand eastwards on the coast of Iceland (Steinþórsson 2001). Volcanic eruption alone can also have critical effects on the Icelandic landscape as is apparent from the eruption of Laki 1783-1785 when around 565 square kilometres of lava flowed from the fissure destroying around 30 farms.

Finally to mention is the position of Iceland on the divergent boundaries of two tectonic plates: the *Eurasian-plate* and the *North-American-plate*. These two plates are moving away from each other, causing incorrectness in the national geodetic reference system. Though this divergence is at average quite slow, only few centimetres per year, then can sudden shift in divergence accompany volcanic activity as was witnessed in Krafla eruptions 1975-1984 when a local rifting totalled 7 meters (USGS 1999).

Summed up together, this causes that different approach to boundary registration has to chosen than presently is in most other countries. Important is that the system will be capable of including temporal aspects of cadastral boundaries and that it can make distinction between static (fixed) and dynamic boundaries. Coastline or glacier edge is an example of dynamic boundary that has to be systematically (or periodically) updated. Recent verdict on the boundaries between collective and property land in southeast Iceland confirms that some boundaries should according to law be treated dynamic in their nature (Héraðsdómur 2005).

Oosterom & Lemmen (2001) propose a way to include temporal aspects. By integrating timestamp in the registration of boundaries the temporal aspect, its dynamicity, can be observed. When a boundary is established it gets a unique id and a time value registered as **`t_min`** attribute. When it is updated, the id of the original boundary feature gets its **`t_min`** added to its identification number and another timestamp registered as its **`t_max`** attribute. Then a new boundary feature is created with the **`t_min`** equal to **`t_max`** of the original feature and same id as before update. Similar procedure is when a boundary feature is deleted, except no other feature is created. See Figure 29. This enables to integrate the element of time into boundary registration, dynamic boundaries like coastline or glacier edges can be update regularly and the overall development of boundaries can be monitored through time.



Object_id	Shape	Quality	T_min	T_max
1023	(0,0), (4,0), (6,2)	1	12 jan	20 feb
1023	(0,0), (4,0)	1	20 feb	14 apr
1268	(4,0), (6,2)	1	20 feb	–
1023	(0,0), (4,0)	2	14 apr	–

Figure 29: Temporal aspects dealt with in a SDBMS (image: Oosterom & Lemmen 2001, p. 516-517)

5.2 Choosing spatial model for ICM

The choice of appropriate spatial model is crucial in the design of cadastral database. It is moreover important that this decision is made in the start of the procedure as it can be decisive for the design of system architecture, choice of software, and enables that data can be gathered correctly simultaneously while developing the system architecture.

As was covered in 2.3 there exist various spatial models, with the rawest being the so-called 'spaghetti model', to more complex models like topological model. A complete topological model does not store any spatial data it self, but refers to spatial entities that do so. Thus has edge or face no geometry. In other words, the topology is preserved even though the geometry is stretched on all sides, as the accent is put on the relationship between objects. Hence, topological model is also referred as rubber sheet model.

In spaghetti model however, the objects are stored with their geometry without explicitly preserving relationship between objects. If the geometry of e.g. one object is altered, the relationship that it has to other object can also change.

An intermediate solution is somehow mixture of spaghetti and topological model. This is e.g. when edges are primitives and their relation to other edges is obtained through shared geometry, where either start or end meets another edge. This is e.g. the case in the model proposed by Quak et al (2003). Thus with altering the geometry of either end or a start of a line, its relationship to other lines changes simultaneously, resulting with corresponding parcels failing to materialise as the function cannot connect the lines.

5.2.1 Reasoning for choice

Because of all its drawbacks the spaghetti model is not considered as appropriate spatial model for implementing nationwide cadastre in Iceland. However, because of it simplicity it is suitable to use for data that is for some reason not capable to be expressed with topology. For instance, raw data, yet to be processed into the topological model, or incomplete data with missing boundaries, unknown relationships and vague quality. Here is the **`NPPRegion`** class discussed in section 5.1.1 very useful.

As observed from text there also exist many variations of topological spatial models, each with its own pros and cons. With many of them using edges, not employing nodes as their topological primitive. This is a drawback as it causes redundancy of registration, whereas the endpoints of boundaries are registered at least three times and possibly of from as many different surveys. Hampering extensive registration of source and quality information (metadata) of individual points (end points especially).

This problem can be overcome by using a standard, like every point in a boundary has to be measured and registered within specific quality measure. However, in a young system in transition, this is not a desirable solution. The solution employed needs to accept bad quality boundaries until new one are measured that replace the older ones. The system thus gradually upgrades as new surveys are done. Using nodes as a topological primitive offers here the possibility to restrain redundancy of point registration and surveying. Therefore there are number of reasons why the cadastral model should employ topological data structure as presented in Stoter (2004, p.130)[9]:

- The approach allows calculations on correctness of topology after updates.
- It opens the possibility to relate attributes to the boundaries between parcels, e.g. date of survey, name of person locating the boundary, etc.
- If each parcel would be represented in the DBMS by a closed polygon, it would be complicated to represent the basic object of cadastral surveying: one boundary between two neighbour parcels.

[9] This is in fact summary of text presented in Oosterom et al 2001.

- Closed polygon representation would lead to double (or triple or even more) storage of all coordinates (except the territorial boundary), which complicates data management in a substantial way.
- Closed polygon representation can result in the introduction of gaps and overlaps between parcels, which is not related to reality.

It can thus be concluded that the ICM should be topologically structured, but the question still remains what kind of topological model should be employed.

5.2.2 The topological model for Iceland

The proposed solution here for Iceland is to employ nodes extensively as the topological primitive of ICM. Instead of the word 'node' is the more descriptive word 'monument' preferred. The definition of monument is here as follows. A monument:

- is related to at least two boundaries, and:
  - if related to three or more boundaries the term corner-monument also applies,
  - if related to only two boundaries merely connects two different surveys of the same boundary, a situation that can occur after merging parcels;
- has explicit relationship with point geometry;
- is maintained separately from other components of a boundary; and
- complies with the common topological rules presented in Worboys (1995), see section 2.3.2.

A boundary is composed of two monuments, functioning as start- and end nodes. Between these nodes can be from zero to many intermediate points, stored as ordered (a sequence) multipoint geometry. See Figure 30.

However, as database can have problem with keeping the ordred sequence of points in a multipoint geometry correct, linestring geometry can be candidate for substitution. The drawback of using linestring is though that it does not offer the option of a boundary with only two monuments, or two monuments plus one intermediate point, as linestring alone must at least have two points defined. A linestring that is zero length (with identical points) is not valid according to the semantics of **`GM_Curve`** defined in ISO 19107- Spatial Schema (2003, p.43).

Multipoint on the other hand is defined as a collection of point geometries and this collection can be set empty. Here the drawback is however to create correct and valid line by aggregating the start node, multipoint geometry and end node together. The sequence of points in the multipoint collection can mess up when updated and it is more difficult to create constrains that restrain intersection of derived lines (making them invalid).

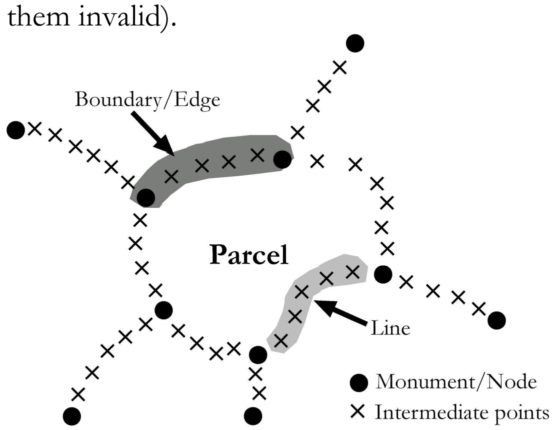


Figure 30: Proposed spatial topology for ICM (image: Author).

A parcel, or more specifically **`PartitionParcel`**, is here a counterpart for the topological primitive face. The spatial extent of a parcel is not explicitly stored but realised through its boundaries. A parcel can have from one up to multiple boundaries that can be aggregated to multiple linear rings, where the largest linear ring represents the exterior boundary. Other linear rings represent then the interior boundaries (the enclaves) and at the same time the exterior boundaries of yet another parcel. How this is exactly implemented in practice depends largely on the database involved and its availability of spatial functions and/or possibility of customised ones (support to procedural language like e.g. Pl/pgSQL, C, Perl and Python).

Function to realise polygon geometry like described in Quak et al (2003) can e.g. be utilised little bit altered, when storing nodes as a topological primitive. By adding 'boundarize' step in the 'polygonize' procedure, the topology can be preserved, even though node ends are modified. This is explained in the implementation of the prototype, section 7.2.

5.3 The proposed ICM model

The main alteration of the LRD diagram with the ICM is the addition of geometric registration of parcels. For this purpose was specially looked to the example set by CCDM and objectives of the Cadastre 2014, which embraces several statements on how cadastres should be implemented in the year 2014 (Kaufmann & Steudler 1998).

Figure 31 presents the geometric part of the proposed ICM but the administrative and legal parts are kept the same as presented in the LRD model presented in Figure 26. The connection to the CCDM is obvious, but there are still some fundamental differences between the two models. To start with is the abstract class **`Object`** that is referred to in the CCDM as **`RealEstateObject`**. In the ICM the **`Object`** class is more general and only adds the temporal aspect to its subclasses, **`RealEstateObject`** and **`LandParcelObject`**, both abstract.

Following sections go into more detail of the classes modelled in Figure 31 and where it is attempted to explain relationships between different classes and their purpose. Furthermore is given explanation of the absence of those classes that were not included from CCDM.

It should be noted that the ICM is not, and appears not to be, a complete data model for the land recording in Iceland, as it emphasises more on the conceptual modelling, and then especially the geographical part of it.

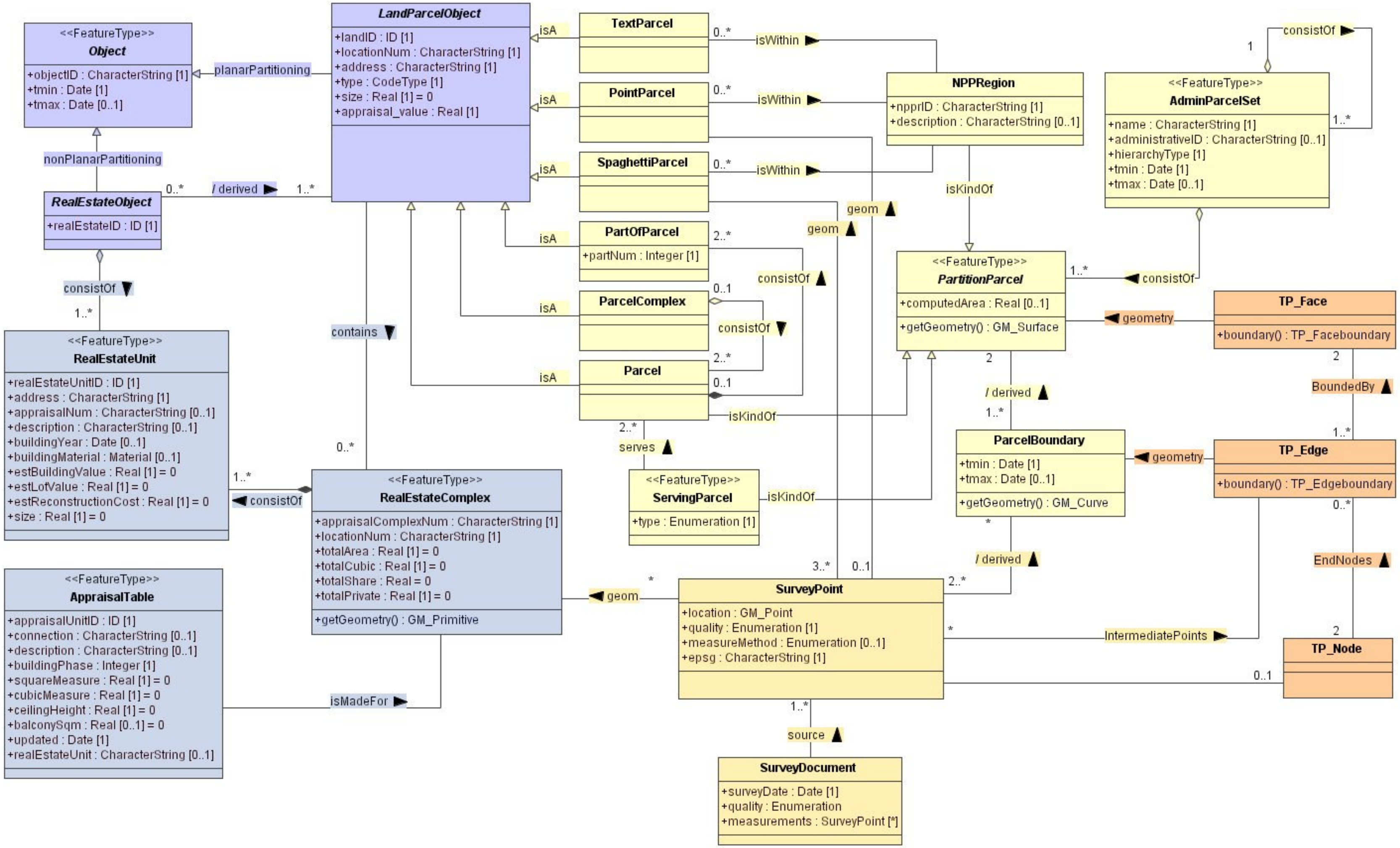


Figure 31: The proposed geographic part of the Icelandic Cadastral Model (ICM) (image: Auhor).

5.3.1 Planar partitioning classes

LandParcelObject is a land partitioning class and thus planar partitioning in the case of two-dimensional registration. No instance of its subclasses can overlap other instances, as land ownership cannot overlap another land ownership, except additional measures like three-dimensional registration is employed. Three-dimensional registration is however not supported in current version of the ICM model, but can easily be added later on if required as separate class.

Parcel along with **NPPRegion** and **ServingParcel** makeup the **PartitionParcel** class that partitions all land in a way that a point on land (or sea) is always associated to exactly one instance of **PartionParcel**.

Set of **PartitionParcel** classes make up an instance of **AdminParcelSet** like e.g. municipality area. An **AdminParcelSet** can further be source for higher-level hierarchal administrative areas. Counties e.g. *consist of* one to several municipalities that again *consist of* number of **PartitionParcels**.

PartitionParcel furthermore provides the link between Parcels and geometry, which is realised through the use of topological spatial model, complying with the ISO 19107 – Spatial Schema, employing classes: **TP_Node**, **TP_Edge**, **TP_Face** and the source for geometric primitive: **SurveyPoint**.

SurveyPoint is pictured in different colour because of its association with **SurveyDocument** and its administrative/legal role as a source of information. See Figure 32.

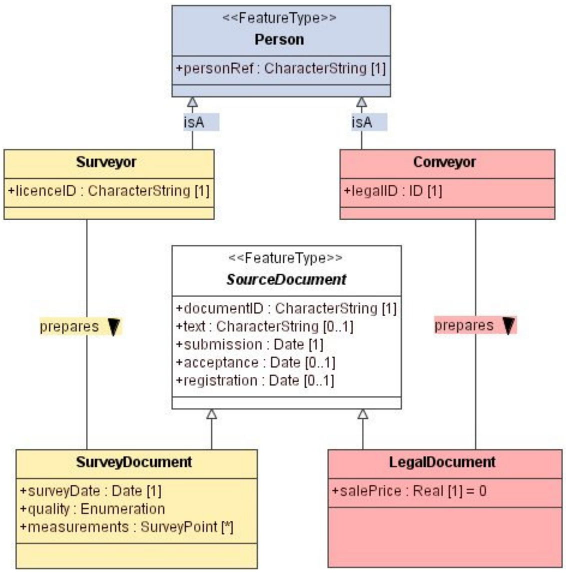


Figure 32: The circle completed. Through SurveyDocument, Surveyor the Person class is again reached.

TP_Node corresponds to what was defined as 'monument' in section 5.2.2. There is zero to one **TP_Node** associated to every **SurveyPoint**, indicating that a **TP_Node** has always a **SurveyPoint**, but **SurveyPoint** is not necessarily a **TP_Node**.

To construct a **TP_Edge** at least two instances of **TP_Node** are needed, with all intermediate points acquired from **SurveyPoint**. **TP_Edge** is here the geometry source for **ParcelBoundary**, which only has derived association with **SurveyPoint**. Indicating that geometry for **ParcelBoundary** is not explicitly stored but derived through other means, i.e. by means of topology.

TP_Face consist of at least one instance of **TP_Edge** that refer to at least one exterior ring and number of interiors, often called enclaves. The way to realise **TP_Face** from

TP_Edge is not defined here, as the options are multiple. ISO-19107 standard on spatial schema is not yet widely realised in practice so other measures have to be used. The prototype discussed in chapter 7, illustrates only one way to implement topology in SDBMS, but other methods are described in section 2.3.2. **TP_Face** has furthermore, one to one relationship with a **PartitionParcel**, which it provides the geometry for. Ideologically, we can think of **PartitionParcel** as number of **ParcelBoundaries**, but this relationship is not explicitly stored, but derived through the topology as was the case between **SurveyPoint** and **ParcelBoundary**.

5.3.2 Non-partitioning classes

RealEstateObject is the primary class of the non-partitioning classes. What is meant here with non-partitioning classes is that the real estate objects partition not the two-dimensional planar domain like ownership of land. It can however be part of land that has special value or contribution, like cultivation of land is appraised to monetary value in Iceland and a piece of land can consequently be registered as such twice in the LRD: as separate **RealEstateUnit** in a **RealEstateObject** and as a part of the complete land parcel under the **LandParcelObject**. Together **RealEstateObject** and **LandParcelObject** can form an integral **Object** for registration using **objectID** (taking the value of **landID**), which is usual the case of farms in Iceland as they are considered as one integral object subjected to registration.

RealEstateUnit that comprise **RealEstateObject** can be of diverse nature like observed above. The most common are buildings, but also registered and appraised are cultivation, hunting right (birds, seal, reindeer), fishing right (salmon or trout) grazing right, eider down production and sea driftwood to name just a few. The relationship between **RealEstateObject** and **LandParcelObject** is null-to-many real estates are located on one-to-many **LandParcelObject**. This relationship is however not explicitly stored in the database. In the same sense that **RealEstateObject** is a composite of several **RealEstateUnits**, **RealEstateComplex** is an aggregate of **RealEstateUnits**. These classes have already be defined and discussed in section 4.5. The relationship that **RealEstateComplex** (and thus also **RealEstateUnit**) has with **LandParcelObject** is many to one. Thus, a **RealEstateComplex** must always be within the boundaries of a parcel. The association between buildings and land is derived from this relation.

Another important thing here is that the owner of a land parcel and a real estate as defined here, do not have to be the same person, even though the legislation assumes so. This is actually the fact in most urban areas where the municipalities are the registered owner of the lots while diverse persons own the buildings (Matthíasson 2003). Causing that non-partitioning property like **RealEstateObjects** automatically creates restriction for the planar partitioning **LandParcelObject**.

Not all **RealEstateObjects** have spatial relevance (like hunting rights) and can thus not be expressed with geometric features like point, line, polygon or solid. The spatial extent of buildings is however useful in many applications and because it can be registered separately from land has consequently large relevance for the implementation of ICM. How to register buildings is not dealt with in this research but vague connection to geometry left in the ICM to enable spatial reference. The presupposition is that only instances of **RealEstateComplex** are worth mapping as they show the extent of houses or complete apartment buildings, but not individual apartments, as if the extent of **RealEstateUnit** instances would be mapped (which are better mapped within a **RealEstateComplex** along with shared space as experienced in condominiums).

The association between **`RealEstateComplex`** and **`SurveyPoint`** is optional. A building can thus be indicated with a point (address point), polygon (a blueprint) or solid (3D-outline), or just not at all (Zlatanova 2004). Google Earth has a nice example how building registration can be implemented using simple three-dimensional solids shown in Figure 33.

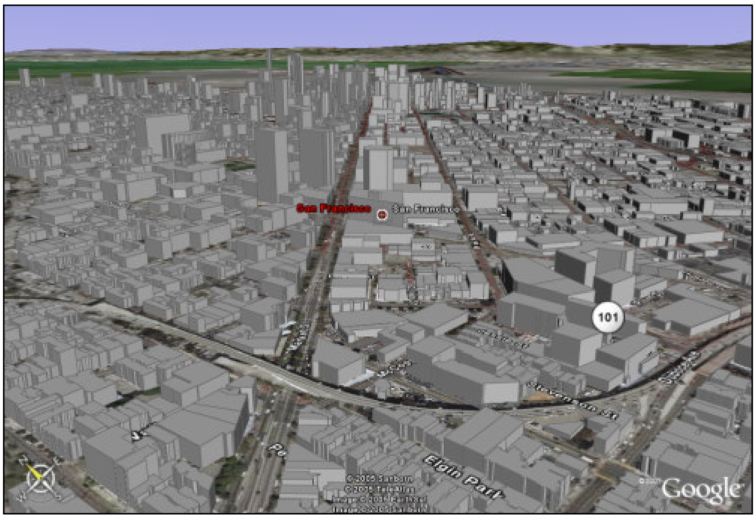


Figure 33: Centre of San Francisco as observed in Google Earth (image: http://earth.google.com).

With buildings included as an entity in the spatial cadastre, the potential usage of the system increases even further. Benefiting field of professions like architects (building and landscape), physical planning, civil engineers (constructions), surveyors (relative positioning), disaster management and insurance companies along with obvious benefits for fiscal and legal cadastral purposes. The implementation of three-dimensional cadastres has e.g. been researched by Stoter (2004). Figure 34 shows the modelled link between **`RealEstateComplex`** in more detail than was expressed with the ICM model in Figure 31.

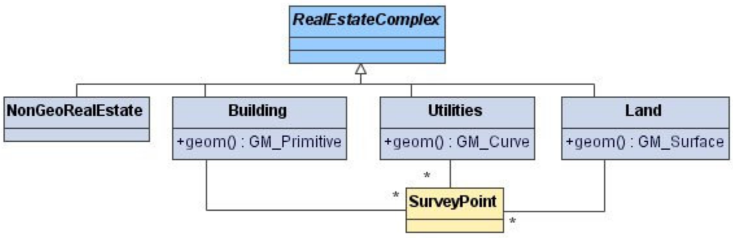


Figure 34: Specialisations of RealEstateComplex and association with SurveyPoint (image: Author).

5.3.3 Discussion

Mapping buildings in a cadastre truly enhances the usability of the cadastre SDBMS for all kind of applications. Alone, the registration of parcels as proposed in this paper is according what is referred as multi-purpose cadastre: "…record of interests in land, encompassing both the nature and extent of these interests" (FIG 1999, [1]).

It is the view of this research that only positive rights to properties should be registered in the ICM with negative rights like public restrictions maintained in different repository, but highly integrated in the ICM. Enquire on a parcel of land in the ICM should still, always giver complete legal situation of that parcel (both positive and negative) by combining the information from these diverse repositories. An example of public restriction is e.g. preservation areas related to historical, natural or geological significance. This differs somehow to the CCDM which defines the **RestrictionArea** class as a way to register both public restrictions and restrictions due to e.g. utilities or transportation, to the **CCDM.RealEstateObject** (in ICM simply **Object**).

The view of this research is however that the class **RestrictionArea** cannot be specialisation of the **Object** class, which should always be subjected to positive right. It can on the other hand be associated to **Object** through **PublicRestriction** as presented in Figure 35. **Object** can be subjected from zero to several **PublicRestriction** while **PublicRestriction** is always related to at least one **Object**.

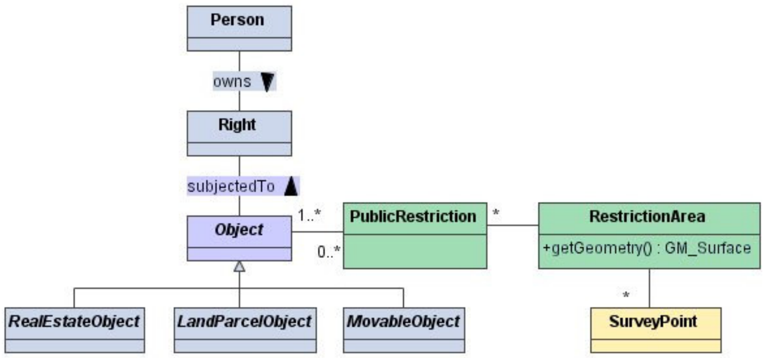


Figure 35: PublicRestriction as proposed for the ICM model (image: Author).

Restrictions due to utilities and transportation should be registered depending on if they are public or private. Public roads are e.g. defined in laws in such way that automatically create restriction to surrounding parcels. E.g. houses cannot be built within specific distance from a centreline of a road. The land under the road itself is however expropriated and ought to be registered as an ownership of the Public Road Administration. Separate parcels for roads are though not widely available and only recently was started to establish specific road parcels in the LRD (Barry 2005). An idea here would be to register roads as structure with permanent right to land instead of creating road parcel. This will be discussed in more detail later.

Private utilities like fibre optic cables, electricity lines or water pipes should on the other hand be registered simply as other real estate units with consequent restrictions (encumbrances) on use of land parcel. Recall Figure 34 where the class **Utilities** is presented here as one of the possible specialisation classes of **RealEstateComplex** class.

One noticeable class in Figure 35 and not introduced before is the **MoveableObject** class. To this class belong moveable properties like cars, motorcycles, airplanes, boats and ships to name just a few. These properties are just as immovable properties (**LandParcelObject** and **RealEstateObject**) associate to person through right and thus convenient to store in the same registry. **PublicRestrictions** do also affect **MoveableObjects** as e.g. large part of the interior of Iceland is closed for automobile traffic seasonally each year to preserve nature. The **MovableObject** class is not a present reality but more like a goal that could be obtained in the future. There is no reason why moveable properties should not be part of the ICM, which maybe should then rather be referred as Icelandic Registration Model. Figure 36 presents the conceptual version of this model, emphasising again on registration of **Object** only.

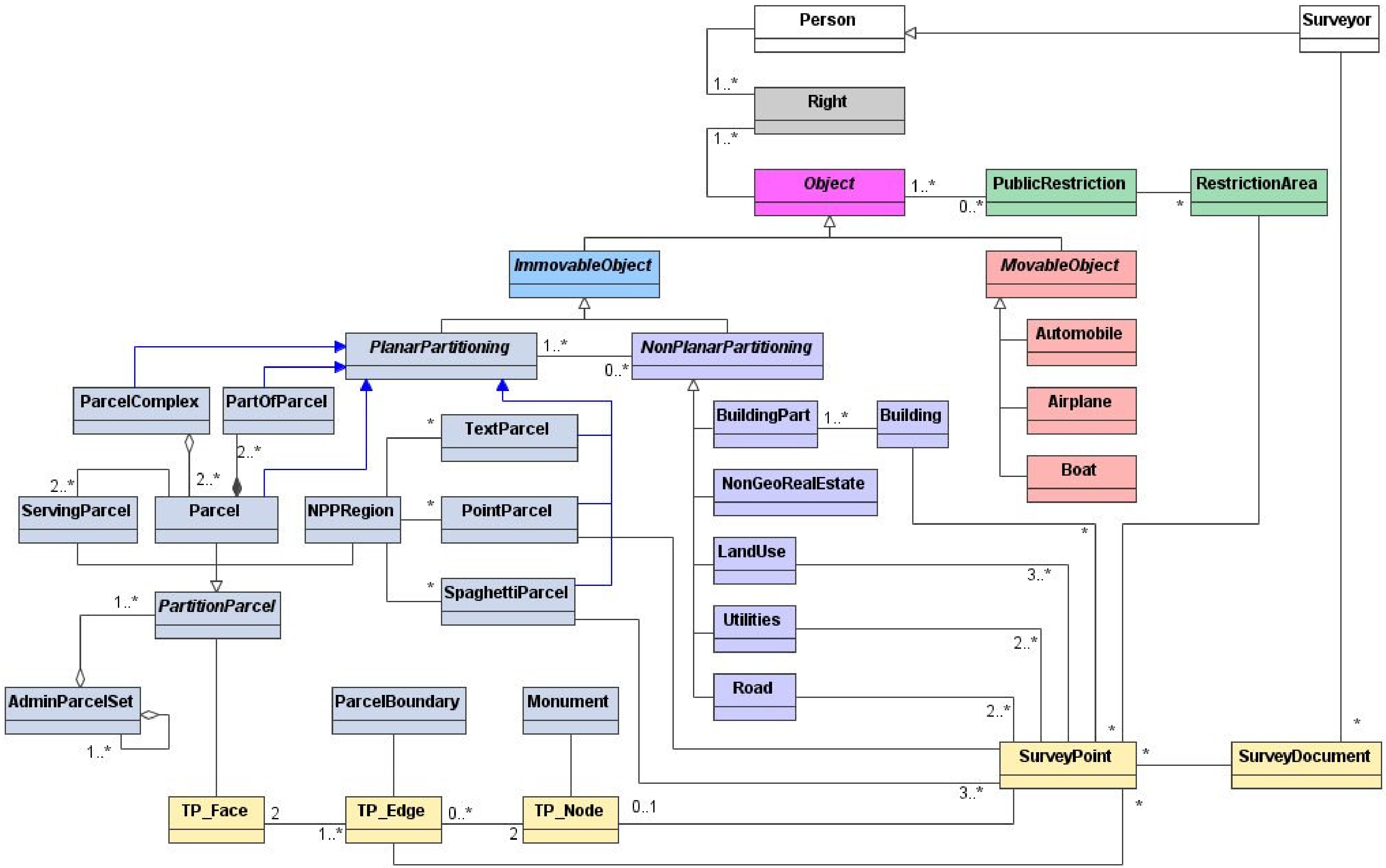


Figure 36: The conceptual and little bit idealised version of the ICM emphasising on registration of objects only (image: Author).

5.4 Evaluating the ICM

There are several things that have to be considered when evaluating the ICM as proposed in this research. In principal this is:

- how well does ICM fit to the LRD currently in use?
- what changes does imply or recommend to current registration procedures?
- how does to fit to the CCDM?

Each of these questions is answered in following sections.

5.4.1 ICM vs. LRD

How well does it represent current reality in Iceland and implementation of the LRD? The proposed ICM fits generally well to the LRD as far as the model presented in Figure 26 corresponds to the actual reality. There are parts of the ICM model that are already defined and in place by the LRD, other that are new and can be recognised from the CCDM. The most notable change between LRD and ICM, not originated form CCDM, is the abstraction and increased importance of **`RealEstateComplex`** class and the introduction of movable properties as a part of the registration.

5.4.2 ICM in practice

What things need special attention, revision or call on administrative alteration in the implementation of ICM?

It is important to define the role of relevant authorities and institutions in the ICM. It is the view of this research that negative rights should not be part of the ICM or the responsibility of the LRI that mainly should emphasise on positive rights.[10] Different institution should on the other hand maintain negative rights in separate repositories, in cooperation and perhaps under coordination of the LRI as a part of a multi-purpose cadastre, also referred as land information system. As an example the:

- Environment and food agency would be responsible for maintaining database listing all preserved areas in Iceland in accordance to law no.44/1995. Along with preservation areas the database could map restriction (could be time dependant or even seasonal) that can be on land use regarding to e.g. bird life or unique vegetation.
- The national museum would be responsible to map all preserved historical relics along gathering information about other relics in accordance to law no.107 2001. Ideally there would be a database locating all declared relics where construction are completely forbidden while also listing other relics that might affect land use to some extent. Constructions are e.g. forbidden closer than 100 m to legally defined relics (no.107/2001) without a prior investigation of an archaeologist (no.106/2000).
- The Iceland State Electricity (Rarik) could be stipulated to maintain national repository over transmission lines.
- The public road administration would be responsible for repository of all public roads in Iceland in accordance to law no.45/1994.
- Etcetera.

Of course these institutions are already doing the above mentioned to some extent but the problem is that it is nowhere centrally integrated and viewable as such. This information has to be integrated and viewed along parcel information stored in the ICM

---

[10] The difference between is that negative rights proscribe actions while positive rights prescribe actions.

to give complete overview of all interests and restriction that can rest on a parcel of land.

The question if public roads should occupy specially registered parcels has to be answered before the implementation of cadastre in Iceland in accordance to ICM can materialise. Today the law state that a land, where planned road is supposed to intersect, should be expropriated and the landowner should be paid compensation relative to the magnitude of damage/loss (no.45/1994).

Although this has been implemented intensively in Iceland since the invasion of the automobile, the Public Road Administration has rarely completed the registration procedure of the claimed land, hence there are few road parcels established in LRD with land id. Consequently are many landowners, in accordance to notarized documents, rightful owners of land occupied by roads, even though the Public Road Administration expropriated it several years back. Figure 55 in *Appendix D: Example of Boundary Declaration*, illustrates this clearly, whereas the road in the middle of picture is declared within the notarised boundaries. No comments where made of relevant authorities during the notarisation procedure, even though the road intersecting the parcel is categorised as public main road (stofnvegur 76-8).

Making roads as specific object in LRD creates several problems that will be hard to solve without changing the current law and procedures:

- Splitting up valid parcels requires one of the following: new land ids, that the uniqueness of land ids to be discarded or different geometric approach (use of multipolygon instead of polygons). In the ICM prototype a new parcel id was defined with land id as attribute, allowing **`LandParcelObject`** to be composition of several parcels (**`ParcelComplex`**). This could however be source of data inconsistencies in practice.
- Creates large parcels (road parcels) that are associated with many regular parcels, causing problems like of concurrent registration and consistency (a road parcel can accidentally extent between administrative areas that should not be allowed).
- Creates enormous work of registering all road parcels that consequently affects all neighbouring and/or intersecting parcels.
- Redundancy, as the influence area (buffer zone with restriction of land use) of roads affects land use of neighbouring parcels that has also to be mapped as public restriction.

This paper recommends that following measure to be taken:

- That public road will not necessarily occupy special road parcel but will be registered positively as non-partitioning real estate, in similar way as other constructions like buildings with permanent right to land. Public restriction outside the space that the road occupies could be derived from public road repository.
- That complete expropriation of land for roads will be part of the history and replaced with long-term seizure of delimited space, compensated as such. If the road drops out of national plan for public roads, the land should be returned to former owner. Current law state that when roads are moved within a parcel the old road parcel can be returned to its former owner as a part of the compensation for the new expropriated road parcel.
- That the Public Road Administration keeps repository of all public roads in the country, where information like road type and centreline among other

information are stored. By retrieving those information the complete legal restriction that the road has to its intersecting parcels, can be obtained.

5.4.3 ICM vs. CCDM

Finally, how does ICM fit to the CCDM? Actually it fits more or less to the objectives and content of the CCDM with only few exceptions. It is argued here, and expressed in the ICM, that immovable properties should be categorised either as planar partitioning like land parcels (**`LandParcelObject`**), or as non-partitioning as all other real estates are (**`RealEstateObject`**), except land parcels. Also, by associating **`Right`** to **`Object`** instead to **`RealEstateObject`** the possibility is created to register moveable properties (**`MoveableObject`**) in the same repository as the cadastre.

Another thing to consider is the association between land and buildings, between partitioning and non-partitioning objects. According to the objectives of the Cadastre 2014 the relation there between should not be stored explicitly but be realised on demand through the geometry of the relevant objects (referred there as land objects). This is a paradigm (as Cadastre 2014 truly is) and does not reflect the current situation, at least not in Iceland. It is essential to be able to locate buildings and parcels together through direct associations, e.g. because of administrative purposes. In a system where neither sufficient maps are available of parcels or buildings, the "compute on-demand" is unrealistic.

Finally to mention here is the inclusion of **`TextParcel`** in the ICM, which differs somewhat from the **`NonGeoRealEstate`** that it is originated from. **`NonGeoRealEstate`** is here thought only as a non-planar partition object that because of its physical nature cannot be clearly expressed (mapped) with geographic feature. Examples of this are e.g. rights to exploit particular natural resource, to hunt and fish. Conventionally these kinds of properties are intertwined with ownership of land parcels, thus the owner has the right to exploit the natural resources within his ownership boundaries. Today however, advantages like this are appraised as separate property and can possibly be excluded from the land parcels that it originally belonged to, as has been the experience in Scandinavia. Where seller can exclude and retain the fishing right when transferring ownership of a parcel (Lemmen et al 2003). This experience has though not exactly been shared in Iceland. It is however possibility that a land parcel at one place can own a fishing right in a lake residing in a far away parcel, usually commons.

The **`TextParcel`** on the other hand is part of the planar partitioning objects and an innovation of this research to accommodate all the parcels in Iceland that do not have mapped boundaries, but only textual description. This textual description can actually be minimal, or even none, resulting in worst case scenario with 'a lost parcel'.

5.5 Remarks

It is the conclusion of this discussion that extending the LRD with the suggestions included in the ICM model is a realistic foundation for the start of spatial cadastral registration in Iceland. The ICM model is in general identical to the CCDM as proposed by FIG (Lemmen et al 2005), except few minor things, which this paper has suggested to be added into the model.

Much has to be taken into consideration when extending the cadastral registration in Iceland. One of these things is registration of features like ownership of apartments, roads and utilities. It is traditional that land under roads is expropriated and notarised as separate land parcel (large objects and tedious in a SDBMS), utilities as parcel easements (creates difficulties when the parcel is split, as who inherits the easement) and apartments as apartment complexes (further divided with additional registration creating vertical division of ownership).

This complies with the notion of real property as land and anything permanently fixed to it like structures, improvements and appurtenances. Carefully observed it can be noted that this definition is very two-dimensional and limited, as it does not assume for ownership of space that can horizontally overlap other spaces. Creating difficulties in registration and consequent evolution of approaches like researched in Stoter (2004).

By altering the point of view and consequently the legislation, a real property could however be conceptually defined as anything permanently fixed in space. Looking at real properties as three-dimensional enables us to use different approaches for registration and mapping. Instead of land being expropriated under road, a space could be expropriated, registering the road simply as structure with permanent right to land, similarly as is presently valid for buildings in Iceland. Calculation of compensation to landowner would then be based on the negative impacts of the road restraining the potential to utilise the land for other purposes. Similarly holds for utilities, apartments and to some extent buildings (those that reside on land not owned by the owner of the building).

Registration and mapping of parcels would on the other hand remain planar as it is important that land ownership is not allowed to overlap, creating clear bases for transaction and administrative subdivision/hierarchy of land. Structures would thus occupy a space, subtracted from the total space reserved within a parcel as indicates in the carrot theory, and create negative right of the future land disposals. Landowner could therefore not dispose land conflicting with structures already in place.

Considering that road parcels have not been widely established in the LRD it is recommended here that this proposal will be taken into considerations with the potential legal affects, concerning e.g. security of ownership, carefully researched.

Before discussing the potential implementation of ICM prototype it is considered essential to give brief overview of open-source software tools in chapter 6. The software covered there where later used to construct the ICM prototype as described in chapter 7.

6 OPEN-SOURCE AND ITS APPLICATIONS

An evolving trend in software development is the so-called *open-source* licensed applications as a counterpart to the more commercial *closed-source* and proprietary developed products. This chapter starts with defining what open-source licences are before covering the commercial usage of open-source products (6.2). Section 6.3 discusses geo software tools provided under open-source, with section 6.4 covering their potential for cadastral applications. Finally section 6.5 concludes the chapter.

6.1 Open-source by definition

The definition of open-source is a group of licenses obeying to the open source protocol that states that the source code of a computer software should be made available free of charge, for both modification and redistribution (OpenSource 2005). The open source protocols assume for:

1. Free distribution of the software;
2. Availability of source code;
3. Allow modification and derived works to be redistributed;
4. Integrity of the author's source code, to separate initial code from addition and refinements;
5. No discrimination against persons or groups, e.g. because of political view or intention usage;
6. No discrimination against fields of endeavour as the software should be allowed to be used commercially in business;
7. Distribution of license to those that get it second hand (in the case of redistribution);
8. License must not be specialised to a product or specific product vendor;
9. License must not restrict other software, e.g. commercial and *open source* should be allowed to work together if desired;
10. License must be technological-neutral, thus not favour one vendor above another

Examples of open source licenses are like Berkeley Software Licence (BSL), Common Public Licence, GNU General Public Licence (GPL), GNU Lesser General Public Licence (LGPL), to name just a few.

Of these mentioned licenses the GPL is far the most used open source licence. The difference between them can be manifold, e.g. does LGPL allow the software in demand to be bundled with other software with different open source licence or even proprietary, which is not the case with GPL. The main difference between GPL and BSL is that BSL allows derived work to be redistributed as proprietary software, which is not the case under the GPL licence (Wikipedia, [2] & [3]).

There is little conceptual difference between open source licences and those that are developed under the *free software movement* (should not be confused with freeware that is free but closed source) licences that emphasise more on giving the user freedom. Some examples are found where open source software cannot be categorised as free software and vice verse (Wikipedia 2005, [4] & [5]). E.g. some open source software licences allow third party to modify a code and redistribute and charge as its own.

6.2 Commercial usage

The idea of free software is often related in peoples mind to limitations of usage or exploitation. This applies e.g. to freeware, shareware or demo software where: usage is restricted to personal uses; people are required to buy the software if they enjoy using it (appealing to normal conscience); has restricted usability compared to a full-version edition which is charged for; or is restricted to limited trial period with some of the functionality disabled. None of this however applies for open source licensed software, as it would violate the set of protocols mentioned before.

Open-source software is meant to be free of charge and with full functionality, independent of user or intended usage. Freely available to governmental institutions fulfilling their tasks and duties, and private companies with commercial purposes. Moreover it is possible to modify and customise the source code of the software to adjust or extend its functionality for the proposed tasks.

Common question is here, how does the development of open source software pay if there is no direct income from creating them? As the openness of the source code along with united effort of the open source community to develop it, it is regarded that better results can be achieved compared to develop the software being developed within a commercial corporation. Revenue, or even cost avoidance, is thus created by different means, like (Wikipedia 2005, [5]):

- offer the software for free but charge for installation and support (common among Linux distributors);
- making the main software available as open source, creating demand for related products, add-ons and services that are sold and licensed separately (e.g. OpenOffice);
- cooperating in developing new product that benefits all participants and by this sharing development costs and omit commercial products. (e.g. Apache).

The question if open-source applications are suitable for cadastral purposes can be viewed and thus answered from different angles. Firstly, are open-source applications more vulnerable in context of security, performance and reliability, which is so much required of cadastral systems? Secondly, are open source applications really capable to maintain sophisticated topological SDBMS for a cadastre and perform relevant cadastral transactions?

To answer the first question is out of the scope of this essay, as it would require extensive research and testing on the open-source applications, but this is worthy topic for future researchers. People ask themselves: “what kind of lunatic would entrust their business to a product built by people in their spare time?” and this has to be answered. Interesting would be to know who are using open-source applications for real applications, what is their experience, drawbacks and advantages. Why choosing open-source software over commercial available counterpart?

The second question is partially answered later in this chapter with the implementation of the ICM prototype. The general conclusion is that open-source applications currently provide the tools to implement ‘open-source cadastre database’ although refinements on user interface, documentation and sometimes software stability is often needed. The availability of open-source software for geo-applications is the topic of next section.

6.3 Open-source geo-applications

It is characteristic of open-source software that they reflect international standards, specifications and trends, better and sooner than their commercial counterparts. Spatial features and methods are in principle in accordance to OGC Simple Feature Model developing to the more advanced ISO-19107 Spatial Schema. They support GML as export, offer WMS, WFS and even WFS-T services (see e.g. comparison between ArcIMS and MapServer in Anderson 2005).

The variety of available solutions is vast and the discussion below will only reveal the tip of the iceberg of open-source solutions currently available. Not only are new software added regularly while other drop out of development, but new versions of older software can enhance their functionality to such an extent making them incommensurable to predecessors. For more information the main gateway to open-source GIS applications is www.freegis.org.

This section will provide brief overview of the open-source applications that could benefit the implementation of cadastral database. The discussion will start with spatial databases and GIS applications, before moving to spatial data servers, ending with clients and interfaces.

6.3.1 SDBMS and GIS applications

Currently exist two main options for open-source licensed spatial databases, PostGIS / PostgreSQL and MySQL. PostGIS on PostgreSQL is the most popular open-source spatial database solution. PostrgreSQL is actually an OR-DBMS with only limited support for spatial types. However, as the object factor enables freedom for additional customised data types and functions to be declared, the ground was made for the development of PostGIS (PostGIS 2005, [1]):

> PostGIS adds support for geographic objects to the PostgreSQL object-relational database. In effect, PostGIS "spatially enables" the PostgreSQL server, allowing it to be used as a backend spatial database for geographic information systems (GIS), much like ESRI's SDE or Oracle's Spatial extension.

PostGIS is further based on the open-source application program interface referred as Java Topology Suit (JTS) that provides (JTS 2005, [1]):

- an implementation of the spatial data model defined in the OGC Simple Features Specification for SQL;
- a complete, consistent, implementation of fundamental 2D spatial algorithms including binary predicates (such as touch and overlap) and spatial analysis methods (such as intersection and buffer)
- an explicit precision model, with algorithms that gracefully handle situations that result in dimensional collapse.
- robust implementations of key computational geometric operations
- I/O in Well-Known Text format

Powerful procedural programming language, PL/pgSQL is integrated in the PostgreSQL database making it easy to create customised functions and triggers. It also supports variety of other procedural languages as e.g. C++, Python and Perl (PostgreSQL 2005).

MySQL on the other hand supports spatial data types as defined in OGC Simple Features by default without needing additional module like PostGIS. This is however a new feature in MySQL and still in development. No recent comparison to PostgreSQL / PostGIS could be found, but mid-year 2003 comparison by Refraction Research, the developers of PostGIS, indicated that the spatial capabilities of MySQL was somewhat faulty but promising (PostGIS 2005, [3]). Comparing the recent MySQL 5.1 with PostGIS documentation reveals that not much difference is between the two systems, implementation wise, as both are quite loyal to the OGC Simple Feature model. Some additional functions like 'polygonize' or 'makeline', available in PostGIS were however not found in MySQL.

Only one open-source GIS application is available that is comparable to the GIS paradigm set by commercial developers like ESRI, Intergraph and MapInfo. GRASS, an acronym for 'Geographic Resources Analysis Support System', is described as "…a raster/vector GIS, image processing system, and graphics production system" (GRASS 2005, [1]). Its spatial data management capabilities are further stated to include: spatial analysis; map generation; data visualization (2D, 2.5D and 3D); data generation through modelling (list of simulation models); link to DBMS (PostgreSQL, others via ODBC); and, data storage.

The main advantage of GRASS over other GIS systems, beside obvious capital savings of buying licenses, is its interoperability to other open-source products like PostGIS/PostgreSQL. The main disadvantage is on the other hand its complexity and dependency on Unix/Linux and Unix like environment, making it difficult for Windows users adapt to it, let alone compile it.

6.3.2 Spatial data servers

There are several open-source programs capable to access and share spatial data over the Internet, both to thin client like simple web browsers and thicker clients like desktop GIS systems. The most widely used is MapServer developed by the University of Minnesota, but more sophisticated one is GeoServer. The functionality of these two products overlap somewhat, though MapServer can be regarded as closer to the client, while GeoServer is more server biased. This will be explained here below.

MapServer is not a stand-alone server as it depends on being integrated with a web services as e.g. the open-source Apache HTTP Server. It can access data from almost all spatial database vendors along supporting number of vector (e.g. SHP, DGN, DXF) and raster (e.g. GeoTiff/Tiff, Jpeg) formats. Through a user defined map file, which arranges data inputs and outputs, the organisation of layers and display is decided. The content of MapServer map file can be accessed with a normal web browser without any plug-ins or extensions, as the program uses common gateway interface (CGI) technology to interact with client queries and produce viewable HMTL pages, only consisting of HTML code and images. A CGI enabled HTML template has however to be provided by server. MapServer map file can also act as OGC WMS or WFS services (delivering GML), and is in such a very powerful application for sharing data (Mapserver 2005 [1]).

GeoServer on the other hand does not need to be integrated with a web service like Apache as it: "…aspires to be the Apache of spatial data sharing, by providing an open source, freely available implementation of the Open Geospatial Consortium's (OGC) Web Feature Service (WFS) and Web Map Service (WMS) specifications" (GeoServer 2005, [1]). In the program interface a connections to data sources are defined, along with defining capabilities and registering metadata of the services provided. The great advantage of GeoServer is that it supports transactional WFS (WFS-T), making way that remote client can update the data at data source. Another advantage, but still in development, is the data validation possibility where data consistency can be preserved, when many, and often irresponsible clients are manipulating the data. This could e.g. be rule like "lines are not allowed to overlap".

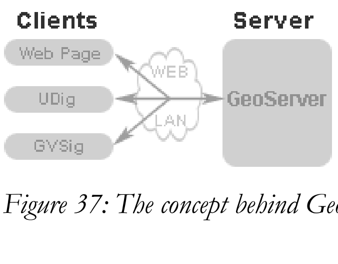
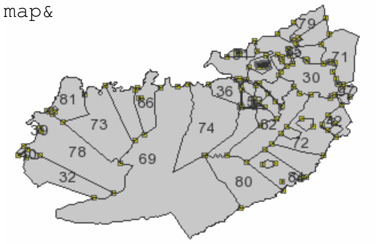


Figure 37: The concept behind GeoServer (image: 2005, [1]).

Figure 37 illustrates the possibilities how to access data served by GeoServer. Simplest, but most troublesome, is to enter request in a web-browser address bar referring to desired data layers, extent, styling etcetera. When requesting WMS services an image is returned (Example 13), but GML document in the case of WFS. See *Appendix E: GML Output from GeoServer* for an example of WFS output. GML can be visualised in a web interface using the approach of scalable vector graphics (SVG).

Example 13: WMS request that returns simple GIF image.

```
http://130.161.233.64/cgi-bin/mapserv.exe?map=C:/tmp/test.map&
  REQUEST=GetMap&FORMAT=image/gif&
  WIDTH=325&
  HEIGHT=225&
  LAYERS=parcels,boundaries,monuments&
  SRS=EPSG:3057&
  BBOX=354000,426000,367000,437000&
  SERVICE=WMS&
  VERSION=1.1.0
```



6.3.3 Clients and interfaces

Open-source clients like uDig or MapBuilder can also be used to access and manipulate, if WFS-T, data provided with GeoServer, while whole of range of both open and closed-source clients can access and display data serviced by WMS and WFS, as it is becoming an application standard.

The variety and availability of open-source clients is enormous, contributing diversely to the field of spatial data sharing and updating. Roughly these clients can be categorised according to several attributes, e.g.:

- desktop or web-based;
- platform (Windows, Linux, Mac);
- support OGC specifications (WMS or WFS);
- accessing and/or editing vector and raster data; and,
- layer management (specify layers);

Table 4: Comparison of several open-source clients available. The table was constructed (24.10.05).

Client	**Interface**	**Windows Installer**	**Vector input**	**Raster input**	**Database connection**	**GML output**	**WMS client**	**WFS client**	**WFS-T client**
Gaia 2.0.4	Desktop	Yes	No	No	No	2.0 & 3.0	1.0.0, 1.1.0 & 1.1.1	1.0.0	No
GvSig	Desktop	Yes	unknown	unknown	unkown	unknown	Yes	Expected	No
JUMP 1.1.2	Desktop	Yes	Yes (SHP, GML2)	No	Yes (PostGIS up to v7.4)	2.0	Yes	No	No
MapBender 2.1.0	Web	x	x	x	x	x	Yes	Yes	No
MapBuilder 0.4	Web	x	x	x	x	x	Yes	Yes	Yes
MapServer 4.6.1	Web	x	Yes (OGR), only view	Yes (Gdal), only view	Yes, PostGis & Oracle	Yes 2.0	1.0.0, 1.1.0 & 1.1.1	1.0.0	No
PrimaGIS 0.4.0	Web	x	x	x	x	x	Yes	Yes	No
QGIS 0.7.3	Desktop	Yes	Yes (SHP, TAB, GML)	Yes (e.g. GeoTiff and Jpeg 2000)	Yes (PostGIS)	No	No	No	No
Udig 1.0.5	Desktop	Yes	Yes (SHP)	Yes (Tiff)	Yes (e.g. PostGIS, Oracle)	No	1.0.0, 1.1.0 & 1.1.1	1.0.0	Yes

Table 4 is an attempt to compare the capabilities of diverse clients. Each of the desktop clients was downloaded and experimented with. The web-based clients where less explored, as it can be quite complex and thus time consuming to do so. Instead was available documentation utilised.

6.4 Open-source and cadastral projects

There are several projects that use open-source applications in one way or another in a cadastral SDBMS. Most commonly this is related to the public data access, thus using open-source solutions to publish data on the Internet.

The canton Solothurn in Switzerland has for instance used MapServer since 2001 to share their data to public and other governmental institutions. In 2002 it was decided to manage the data in PostGIS/PostgreSQL after comparing the available solutions like ArcSDE and OracleSpatial. Finally they use commercial GIS products to update data using e.g. ArcGIS (PostGIS 2005, [5]). See Figure 38 and Figure 39.

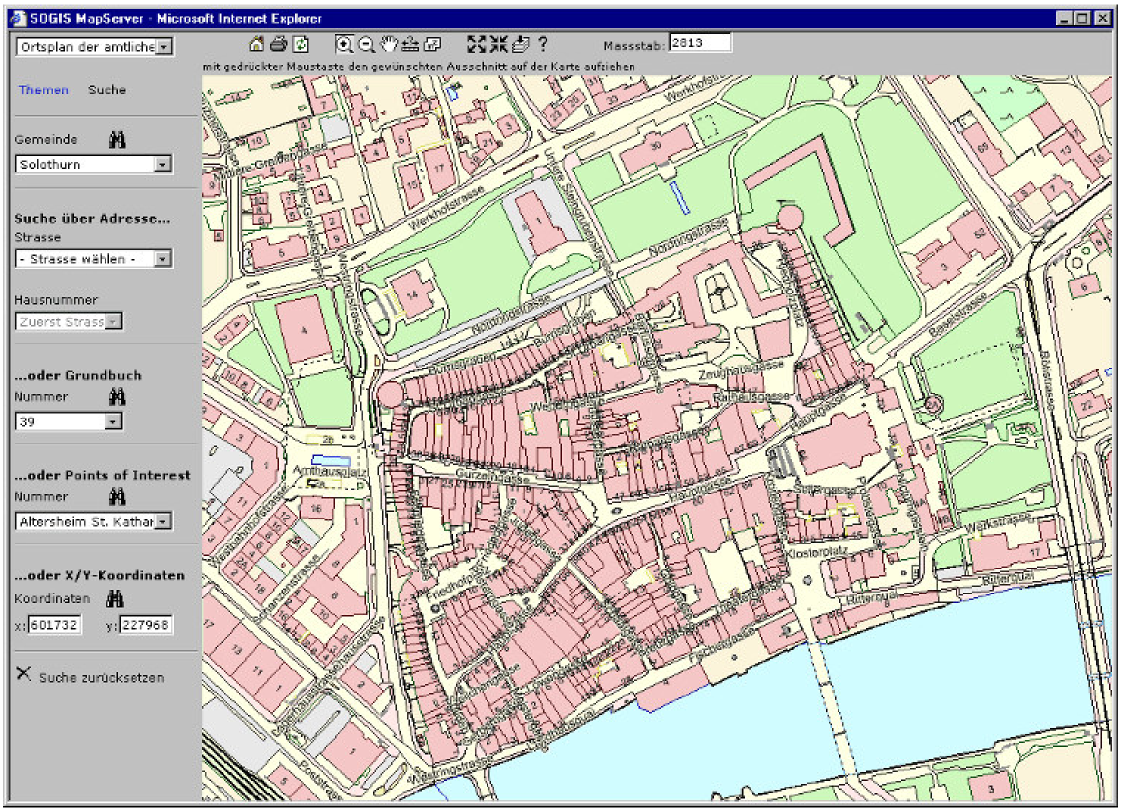


Figure 38: MapServer / PostGIS (image: PostGIS 2005, [5]).

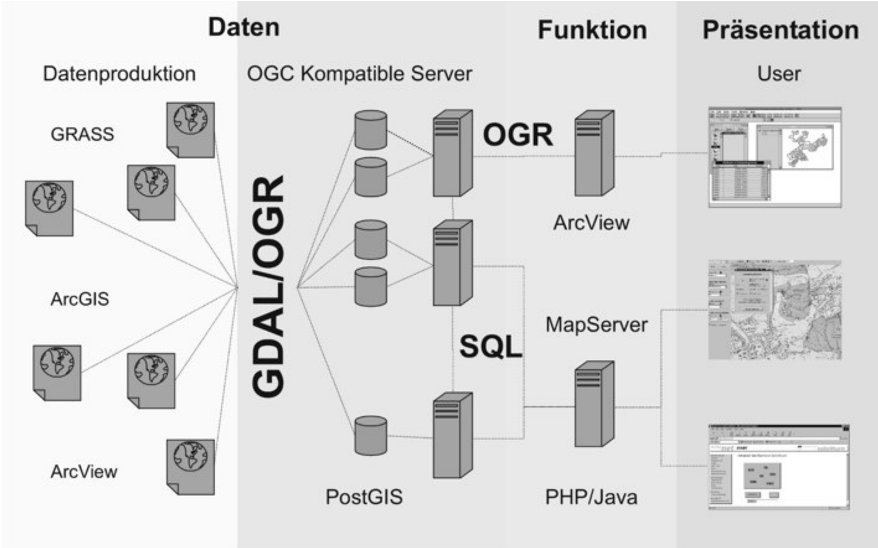


Figure 39: System architecture at Kanton Solothurn (image: PostGIS 2005, [5]).

Fulton County in Georgia US provides data access to its cadastral data in similar way as Solothurn canton in Switzerland. They however go one-step further using open-source applications by using Jump and uDig parallel with ESRI products as the main mapping tools (PostGIS 2005, [6]). See Figure 40.

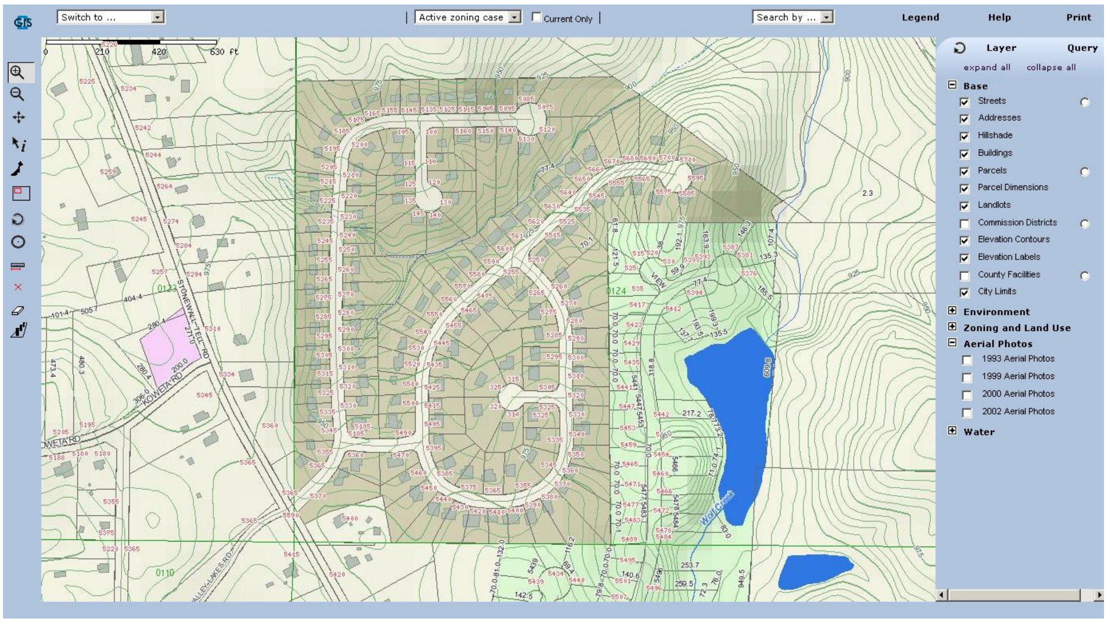


Figure 40: Fulton County in Georgia US provides web-access to diverse land information by employing PHP enhanced Mapserver (image: http://wms.co.fulton.ga.us)

The department of neighbourhood development at the city of Boston maintains diverse information in their open-source PostGIS on PostgreSQL database. For example: city owned properties, parcels, owner addresses and building footprints (PostGIS 2005, [7]).

The experience so far further encourages uses of open-source applications and this latest example provides verification that even a large developed city is prepared to store and manipulate its spatial data within an open-source SDBMS.

Recent project sponsored by IDABC[11], belonging to the European Commissions, further proofs that. The Bulgarian city Kardjali is part of IDABS project, referred as 'Support to e-government initiatives at local level through free and open source software in South East Europe'. There states: "With open source solutions, the city above all wants to save costs avoiding paying licences for certain software, increase the level of security and raise the level of IT skills of the employees" (IDABC 2005, [1]). Cadastral services are one of the information that is dealt with in the project.

Another noteworthy project supported by IDABC but unrelated to cadastral developments is 'Dutch use open source for Geoservices in public works' (IDABC 2003, [1]). The objective of the Geoservices project committed by RWS/AGI was:

> … setting up a central web based infrastructure to make geo information available directly from the source. In doing so, the use of the open standards of ISO and OGC was a given, whereas the use of open source software to realise the OGC web architecture was a possible choice. In short, open standards were the starting point, open source software was a preference.

The project turned up to be huge success. Among findings were (IDABC 2003, [1]):

- The initial expectations for open source were not as high, but the quality of the open source applications and the procedure of open source communities were positively surprising.
- The response speed of the open source community is impressive. It was very high, and usually it needed less time for bugfixes from the open source community than from regular software suppliers. Moreover, the community makes the problem more transparent, which means a better notion of the time/effort needed for possible solutions can be obtained. The transparency of problems was experienced as positive.
- Supplier dependency does decrease. The position in negotiations with closed [-source] software suppliers is much stronger.

6.5 Remarks

The above discussion clearly indicates that open-source applications are becoming a realistic option for institutions and companies to develop and maintain SDBMS, let alone cadastral project.

Along with further development of spatial standards in line with ISO and OGC Specification it can be predicted that the future will bring open-source and closed-source applications even nearer to each other than can be observed at present. The evolvement and use of open-source software creates an obstacle to continuing monopoly of dominant commercial GIS developers and could force them more to follow international guidelines and standards in their software development. It at least encourages them to do better because of two equal alternatives the market tends to choose the less expensive one.

A prototype was implemented within this research whereas simple parcel registry was built up and exploited using only open-source applications. This is the subject of next chapter.

[11] **I**nteroperable **D**elivery of European eGovernment Services to public **A**dministrations, **B**usinesses and **C**itizens.

7 CASE STUDY - IMPLEMENTING ICM PROTOTYPE

The objective with implementing an ICM prototype is primarily to gain knowledge of technical implementation of recording the spatial extent of parcels plus experiencing the applicability and potential of open-source application on real problems. Answering questions like what factors have to be considered when designing system architecture for cadastral spatial data and how to transform conceptual/logical model to physical one, fitting the system in use? Moreover, what practical experience can be obtained by using open-source applications e.g. in terms of stability, performance, applicability, and documentation?

In forehand several premises were defined relevant to the implementation:

- The system should be stored centrally but maintained by diverse heterogeneous and geographically distributed clients (both desktop GIS and web browsers). This is due to the organisational setup in Iceland where municipalities and counties would act as local cadastral authorities.
- Nodes and vertices create boundary-edge, the primary entity of the spatial registration, which is used to realise the geometry of parcels, the primary entity of right registration.
- Potential procedures within the open-source applications have to be identified to perform transaction that involves splitting, merging or creating new parcels.
- That the system appertain the ICM model as defined in earlier chapters.

With these goals as beacon the ICM prototype was developed with the accent on the geographic implementation of cadastral registration, but less on legal or administrative parts. Initially it was the target to make complete prototype that would consider all these aspects, but obstacles like access to real data datasets prevented this. The assumption is though that these parts: legal, administrative and geographic, are only linked together with corresponding identifiers.

Explained generally, a person is registered in separate repository with unique person id, and through that identification owns an ownership right, registered in another repository. On the other hand, land parcel is an object with recorded spatial extent, identified with a parcel id, which can be aggregated under a land id representing a land object. Ideologically and according to ICM, land id and parcel id should be identical but given the possibility that one land object could be a composition of several spatially disjoint parcels, creates the need for a separate parcel id. Example of this is for instance a parcel under uniform ownership but split by road parcel. Eliminating the need for road parcels could alter this. Concluding the registration of rights points to the relevant subject (person) and object (land), maintained separately. Therefore the prototype implemented only considers the registration and mapping of parcel and land ids, but not persons or right.

The ICM prototype employs open source application, as this was one of the initial objectives of the research to examine its applicability to store and maintain cadastral data. Despite this the following sections will attempt to be not too open source directed in its discussion, as the problems raised here are many the same, independent if the applications used are open-source or not.

First section of this chapter discusses system architecture of the prototype, with section 7.2 dealing with spatial model used and general aspects of implementation. Section 7.3 briefly discusses transactions and concurrency control available within open-source. Before concluding the chapter with remarks in 7.5 general discussion and evaluation of system architecture is presented in section 7.4.

7.1 System architecture

Open-source applications were chosen in the system architecture for the prototyope in such way that the current LRD would simply be extended to ICM, enabling spatial mapping of parcels. The general structure was considered consisting of SDBMS for storing the data, server to share the data, and client/interface for visualisation and editing. Moreover it is a premise that this architecture can be integrated in the land registration architecture already in place with the LRD. Optimum would be to integrate the cadastral spatial data in the existing LRD DBMS using its native IBM Informix, complying with the second statement of Cadastre 2014 (Kaufman & Steudler 1998, p.19). That is however not an option in this research and the spatial recording thus implemented in separate SDBMS using open-source SDBMS. PostGIS on PostgreSQL SDBMS was the chosen as the open-source solution because of its well-known status within the open-source community.

IBM however offers spatial extensions to its database products (Informix and DB2) that comply with the OGC specification on simple features and offers various spatial functions. This was not researched further but IBM spatial product specification can be accessed at IBM (2005, [1] & [2]).

Figure 41 illustrates how the overall architecture is considered. Only the cadastral section was implemented in the prototype but valuable it is useful to gain complete overview to model things in wider context.

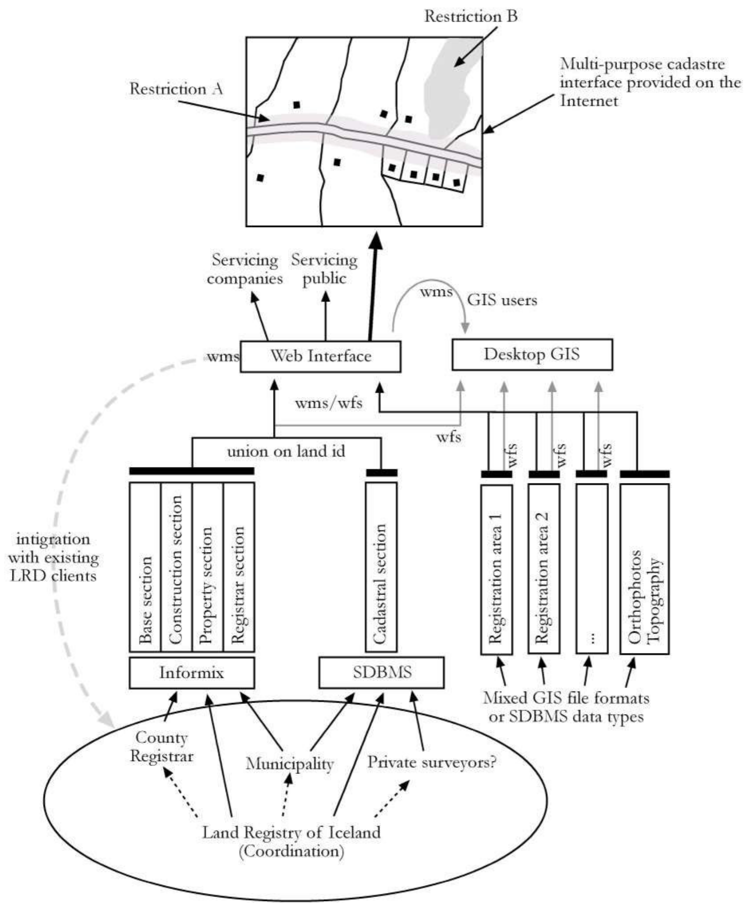


Figure 41: System architecture in context of ICM prototype showing also user's front to multi-purpose cadastral information in Iceland (image: Author).

The cadastral section is federated to other sections of the LRD. This federation can be queried, as it was integral, but is manipulated fragmented. Municipalities, private sectors[12] and LRI officers edit the cadastral section through clients using either direct database connection or WFS-T. Different modules/clients edit the other database fragments not altering current LRD procedures.

The figure also shows the potential of the proposed setup with heterogeneous data, representing diverse information regarding land tenure and restriction, integrated into one 'Multi-purpose Cadastre' covering whole Iceland. Depending on requirement the combined data or individual layers can be browsed in a web interface or integrated in remote desktop GIS application using WFS or WMS. The black filled squares above each data source represent the server of information. Using open-source software this could be GeoServer or the combination of Apache and MapServer.

By structuring things like this it is possible to combine several data sources into one application where it is possible to get detailed information regarding positive and negative rights subjected to piece of land. The approach is in harmony with recommendation that this report has brought forward on the CCDM that positive rights should be registered separately from negative rights like public restrictions to land use. It is good to bear in mind that positive rights also derives negative right to some extent in the sense that e.g. building permit on land can be regarded as property (positive right) for one while restriction (negative right) for the landowner. The negative rights implied by public restrictions are recorded and maintained outside the ICM by diverse institutions, but later integrated/federated into the whole model for display and querying in a multi-purpose cadastre interface.

Noticeable from the figure is the diversity of the origin of the information that is integrated, being databases, vector-and raster files of all sorts. This would not be possible without the contribution of the OGC specifications and most likely not without the constant contribution of the open-source community. Open-source solutions have e.g. been front-runners of the WMS/WFS evolution and far as known here, transactional WFS-T servers and clients are not yet readily available from many of the larger GIS developers.

7.1.1 Manipulating and operating the cadastral section

Due to security reasons and concurrency control direct database connection is preferred over WFS-T, to manipulate the data in the database at the client side. The latter has been researched by Brentjens (2004).

The reason for this is that WFS-T complicates user management, as it becomes more difficult to identify users without additional programs and effort. This is essential for locking mechanism and consequently concurrency control. Additionally is no security/login available in GeosServer that would otherwise act as the WFS-T server. Moreover, even though GML3 and the ISO-19107 Spatial Schema are available it is still on conceptual and development level in context of usage and utilisation, with the GML2 and the OGC Simple Feature Model predominant. The latter lacks support for topology so it can not be considered unaltered as implementation model for the prototype.

On the positive side, WFS-T will in the near future become a standard element in GIS applications and GeoServer is developing data validation possibilities that would be very helpful to maintain the consistency of input data. GML3 supports topology and the future will reveal its strengths and weaknesses in practice. The jump between GML2 and 3 is huge and therefore it will take time before software developers can adjust, with

[12] Ongoing is a debate in Iceland on licensing of surveyors.

some still coping to include GML2 in the first place. Excellent documentation on GML3 is in Lake et al (2004).

Commercial GIS software are generally not able to connect to PostGIS, which results with that only open-source GIS clients qualify as the front end to the cadastral section SDBMS. There are several clients available and easily customisable to the needs and requirements of LRI, e.g. Jump, QGIS and uDig. The applicably of each of this clients is here discussed in more detail.

Jump 1.1.2 is right away ruled out of, as it does not support connection to PostGIS based on PostgreSQL version higher than 7.4. The client is however extremely user friendly and gives a real GIS feeling when querying, styling or editing data. See Figure 42. The future of the client is unclear. The source code is being used in the new uDig client while some devoted Jump users have continued developing Jump under the name OpenJump (http://openjump.org).

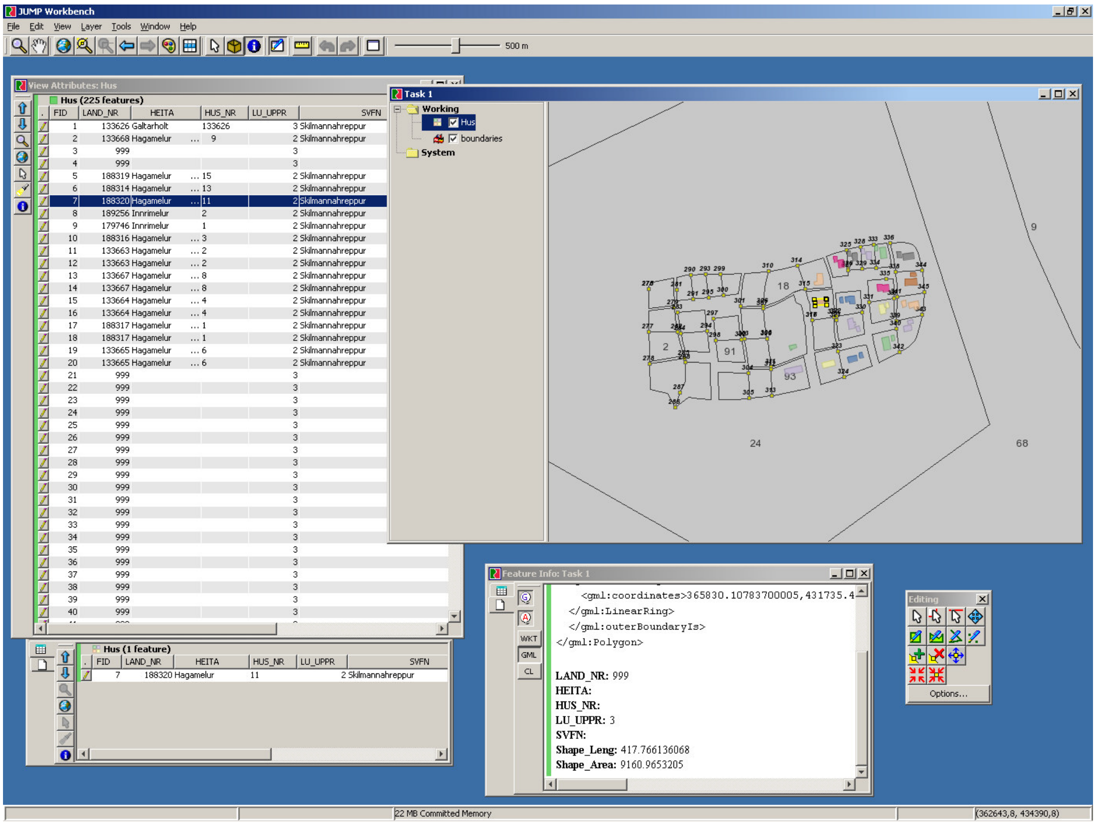


Figure 42: The Jump interface displaying parcels and buildings from Skilimannahreppur, Iceland. The parcels are retrieved as raster from Mapserver WMS connecting to PostGIS databse. The buildings are accessed from SHP file (image: Author).

QGIS 0.70 is a promising product but current version is really unstable, beside it does not support WMS or WFS as input. However, it gives a real GIS feeling with its approach to editing and viewing spatial data and hopefully will develop into something bigger and better in near future. Because of its OGC limitations it is not considered as candidate to be a front end for updating the cadastral section of LRD. However one advantage of QGIS and not shared by other similar application is the possibility to export MapServer map file of the view in use. Example of QGIS can be seen in Figure 43.

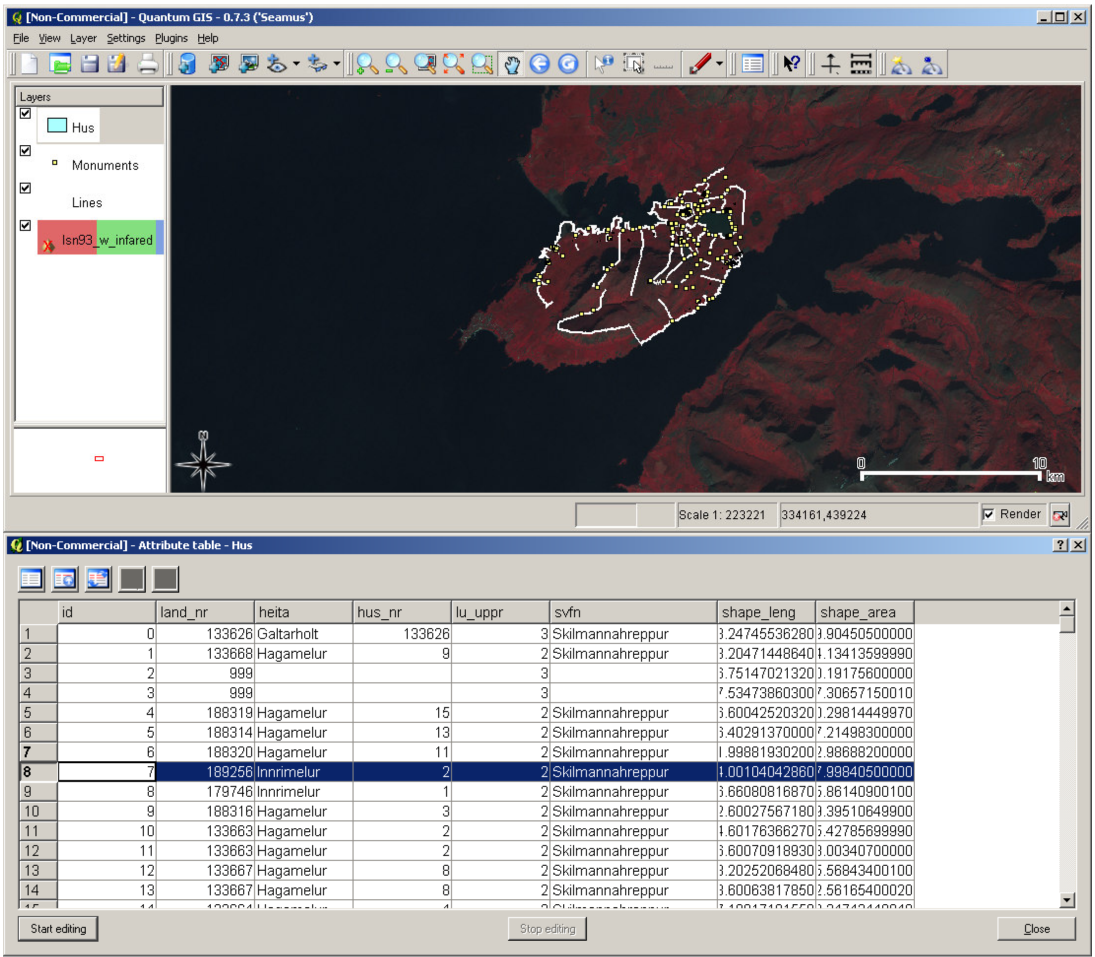


Figure 43: Quantum GIS displaying mixture of raster (geoTiff), PostGIS and vector (shp) data (image: Author).

The only real option left is the uDig 1.0.5 client, which besides its many limitations is probably the open-source desktop client currently available that can integrate the most variety of data sources. It can access several SDBMS, edit diverse vector formats, and supports WFS-T, plus being capable of viewing WMS. Its limitations are mainly shortage of functions and its raw interface. It is e.g. possible to create, update (modify or add point) or delete a linestring, but it is not possible to delete individual points already added unless doing that manually in feature properties. Another drawback is that uDig cannot display layer table or do any GIS like function like: measure distance; calculate area; create buffer zone; attribute mapping (filtering); etcetera. Moreover can it be incredible slow and computer memory consuming. See snapshot of uDig interface in Figure 44.

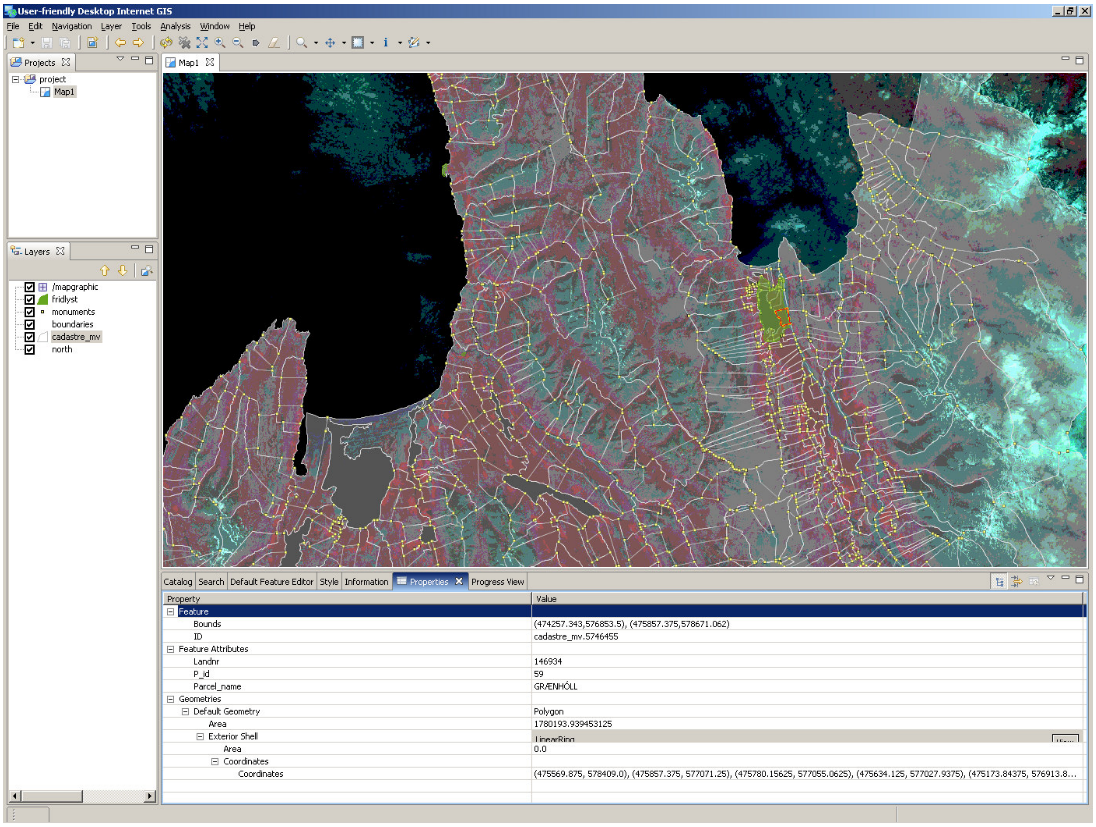


Figure 44: Snapshot of uDig displaying data from diverse sources. The aerial photograph is a WMS layer, the green preservation is a WFS layer and the parcel information is retrieved from PostGIS database (image: Author).

The main advantage of uDig apart from it ability to display and edit diverse spatial data is its ability to be extended and customised according to users wish (with some knowledge of Java). Being open-source software the source code is available for download making the possibility to create customised functions, e.g. specially intended for cadastral or surveying processes. Finally, as it is partially based on the Jump-Project, as the initial objective of the developers were to combine the functionality of the two open-source projects Jump and GeoTools. This could indicate that future versions of uDig will develop in the direction of Jump with all its GIS functionalities. Figure 45 illustrates the main objective of uDig to become cross platform desktop GIS client.

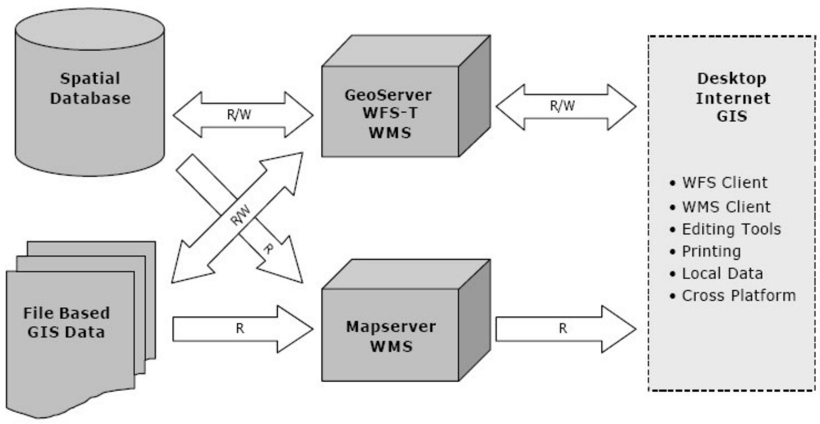


Figure 45: In uDig documentation states that "…interactive desktop access is the missing application in the open-source OpenGIS standards-based spatial infrastructures"(image: UDIG 2005).

Another advantage is how uDig transacts with PostGIS, using conventional SQL transactions. Editor can always rollback his editing until he has committed the changes. Thus the client program monitors the actions of the editor, compiles a transaction statement simultaneously as the user edits the data, and commits them all at one when the user decides to, maintaining e.g. the atomicity of the transaction as defined in the ACID transaction properties.

7.1.2 Sharing cadastral data

Open-source software provides vast opportunities to share data, either in accordance to OGC specifications, or in customised web applications. Both GeoServer and MapServer can share its data as WMS or WFS services, whereas MapServer can also provide customised maps viewable in conventional web browser using CGI to transfer requests.

MapServer is regarded here as the preferred options for sharing cadastral data in accordance to system architecture, acting as a unifier of data sources, as it supports more variety of data types than GeoServer (e.g. raster images). GeoServer is on the other hand thought preferable in the cases of institutions concerned of maintaining restriction areas, as they can use WFS-T to edit the data remotely, stored either in vector files or SDBMS. For the time being is direct connection to SDBMS preferred over WFS-T to edit data in the cadastral section but the latter is serious candidate in near future, especially as it facilitates the possibility of using heterogeneous clients.

The multi-cadastre web interface uses MapServer to integrate the cadastral data stored in LRD, plus diverse WFS and WMS layers coming from external sources. The cadastral data is also provided as WMS with MapServer for desktop GIS users to use in the same manner. They can also connect directly to one of the data source through WFS enhancing the speed by skipping the intermediate (again see Figure 41).

7.2 Spatial model and implementation of prototype

There were two topological models considered for the implementation of ICM spatial model. To start with the topological relationship between monuments and vertices was created as explicit as possible. With the system failing to perform this model was adjusted in the second attempt with the vertices integrated in the boundary table. These two experiments are described below. The step-by-step implementation of the SQL code of the second attempt is presented in *Appendix G: Implementation of Prototype*. Moreover the definitions of the corresponding tables created as a result of second attempt can be explored in *Appendix H: Table Definitions in Prototype*.

The table **`Parcels`** is used in the prototype as a replacement for CCDM **`PartitionParcel`** with corresponding specialisations: **`Parcel`**, **`ServingParcel`** and **`NPPRegion`**. It is assumed that through **`Parcel`**, **`PointParcel`**, **`SpaghettiParcel`**, **`PartOfParcel`** and **`ParcelComplex`** is a join relationship through unique land id to legal/administrative data stored in dfferent sections of the LRD.

7.2.1 First attempt: designing spatial model and organising data

To start with, a spatial model was designed that stored nodes and vertices in two separate tables, respectively **`Monuments`** and **`Lines`**. These tables where related in a boundary table, **`Boundaries`** that only referenced topological relationships like, start node (monument), end node (monument), vertices (line), parcel to left and parcel to right. The geometry of **`Boundaries`** were realised by constructing a view named **`Boundaries_view`**. The **`Boundary_view`** uses a special designed function, referred as **`getBoundaries()`** that constructs a linestring from two points and line if there is any. Finally, **`Parcels`** table was constructed to store parcel information without any spatial reference. By creating a second view, referred as **`Cadastre_view`**, the information from

`Parcels` and the geometry of **`Boundary_view`** could be used to realise polygons that could be viewed in uDig. To realise the parcel geometry a function provided by PostGIS was used referred as **`polygonize()`**. It takes linestring's as input and returns **`GeometryCollection`** with all polygons that potentially could be created with the input lines. This generated a problem, if polygon has enclaves at least two polygons are created. The largest polygon with defined holes, and further several polygons representing the geometry of each of these holes. The following query would in fact return **`GeometryCollection`** containing several polygons, representing the one with land id 111 and then all his enclaves:

```
SELECT p.oid, p.land_id, polygonize(bv.boundary)) AS geom
FROM parcels p, boundaries_v bv
WHERE p.land_id = 24
   AND (b.left_parcel = 24 OR b.right_parcel = 24)
GROUP BY  p.oid,p.land_id;
```

This made it necessary to alter the **`polygonize()`** function to return the polygon with the largest linear ring. However, the programming capability of researcher did not allow for this, resulting with different solution. By utilising PostgreSQL powerful plpgSQL procedure language, **`extractPolygon()`** function was created:

```
CREATE OR REPLACE FUNCTION extractPolygon (geom geometry)
   RETURNSgeometry AS $$
   DECLARE
     l1 float4 = 0;
     l2 float4;
     c1 int4 = 0;
     c2 int4 = numgeometries(geom);
     n int4;
   BEGIN
      WHILE c1 <= c2 LOOP
       c1 = c1 + 1;
       l2 = area(makePolygon(exteriorRing(geometryN(geom,c1))));
       IF l2 > l1 THEN
         l1 = l2;
         n = c1;
       END IF;
     END LOOP;
     RETURN geometryN(geom,n);
   END;
   $$ LANGUAGE plpgsql;
```

`extractPolygon` function takes **`GeometryCollection`** as input and scans the length of exterior ring in all geometries, which are of course only polygons. When it has reached the end of the collection it returns the polygon that has the largest area within the exterior ring, which should then return the correct polygon. The query to construct geometry of parcels is thus:

```
SELECT p.p_id, p.parcel_name, p.land_id, extractpolygon(
   polygonize(bv.geom)) AS geom, p.parcel_type, p.description,
FROM parcels p, boundaries_v bv
WHERE p.p_id <> -1 AND (bv.left_parcel = p.parcel_id OR
   bv.right_parcel = p.parcel_id)
GROUP BY p.parcel_id, p.parcel_name, p.parcel_type, p.description,
   p.land_id;
```

Several drawbacks were discovered while experimenting with the data stored in accordance to the model described above. By editing **`Monuments`** and **`Lines`** in uDig it was possible to manipulate the boundaries of existing parcels, with the appearance changing on the fly, but it was not possible to create new parcels.

The **`Boundaries`** table needed to be edited to create new boundaries and whereas uDig cannot display non-geometric data this was impossible unless doing it separately with a statement in PostgreSQL/PostGIS.

Although this solution, presented as the 'first attempt', worked with 100 parcels it totally collapsed in performance with 8.000 parcels. The intermediate level of 1.000 parcels was relatively ok, but still quite slow. However the computer where the experience was carried on was also quite slow performer. This problem could possibly be overcome with restraining the amount of data that can be viewed and thus generated each time. To implement this, envelope geometry would have to be stored for each parcel as a part of the **`Parcel`** table. The problem is however how such envelope geometry column could be maintained properly, using constraints or triggers.

Evaluating the potential of the first attempt it can be observed that this solution has potential if the user client can simultaneously edit spatial and non-spatial tables in PostGIS/PostgreSQL database. This could for instance be accomplished by extending web client like MapBuilder with PHP script, or by creating customised Java client that automatically supports and utilises the topology.

7.2.2 Second attempt: revision

After a lot of thinking and consultation, solutions were found to the obstacles met in the first attempt, updating geometry in uDig and slow performance.

Instead of storing intermediate points, represented by linestring, in separate **`Lines`** table it was found more convenient to store them as a part of the **`Boundaries`** table. Enabling boundaries and parcels to be created in uDig, not only updated. To create boundary the only thing needed is:

1. to make or find relevant monuments;
2. draw boundary line between them;
3. insert correct topological relationship in the **`Boundaries`** attribute table to monuments and parcels; and,
4. commit changes.

To create a parcel in uDig a land id is needed, established separately with a SQL statement in PostgreSQL/PostGIS interface. As soon as boundaries with correct topological relationships where formed the parcel would automatically be created.

The other obstacle met in the first attempt was the performance of the system. While it took the system only a second to realise and display 100 parcels this time, it was several seconds with 1000 parcels (round 10s) and up to hours with 10.000 parcels (round 4-5 hours) even considering exhaustive use of indexes where possible.

The main thing here is that spatial indexing is impossible on table views. Ideas to resolve this are e.g. to:

- store information of the extent of the parcel (envelope) as a part of parcel data;
- develop a cross table functional index; or,
- using materialised views.

The first solution creates the disadvantage that it could exclude polygons from query which should belong there, as they have not updated extent in parcel table. For example a rectangle window is drawn over an area of parcels for selection. Without storing explicitly information of the spatial extent of the parcels in the parcel table the query processes as follows:

1. Calculate the boundaries of all parcels by using geometry from boundary view. As the geometry in **`Boundary_view`** is also computed on demand this can be slow if the database is large.
2. Compare the window area and the parcels.
3. Select those parcels that are within the window query.
4. Return selected parcels.

By storing the envelope extent of the parcel in the parcel table the query could be optimised somewhat:

1. Compare the window query to the envelope geometry of the parcels and find those that overlap with it.
2. Query the boundary table to return all boundaries that have the selected parcels from above query on either left or right side.
3. Compute the parcel geometry from retrieved boundaries.
4. Compare the selection area to the generated parcel and select those that are completely within the window.
5. Return parcels.

The difficult here would to secure that correct envelope of each parcel would be stored in the parcel table.

The second solution is cross-table functional indexing. Functional indexing is supported in PostgreSQL but only within one table. E.g.:

```
CREATE INDEX functional_idx
  ON example_table (char_length(column1));
```

Results of query retrieving data from more than one table can however not be indexed in the same way. Thus to enable functional indexing for ICM prototype all data would be needed to be stored within the one and the same table, making this solution impossible to implement.

Finally there are materialised views, opted for the prototype. It creates performance panalty if the computer has to compute and realise all parcels on demand when the parcels geometry is browsed, especially if many users are simultaneously exploring the database (without necessarily editing it). A method to bypass this is using materialised views, functionality already offered in e.g. Oracle databases.

While the content of a 'table view' is always temporarily, a materialised view is actually a physical constructed table mirroring the SQL query of the view. When the results from a view query changes, a trigger on corresponding table is established to update relevant materialised view. Materialised views are currently not provided by default in PostgreSQL but by using pl/pgSQL it is possible to create the functionality. Jonathan Gardner (2004) describes four kinds of materialised views, categorised according to update frequency and scope, and that he has also partially developed himself in pl/pgSQL:

- <u>snapshot materialised views</u>, the table is only updated when manually refreshed. Materialised view table is in fact deleted with the content of table view, hence current snapshot, replacing it.
- <u>eager materialised views</u>, updates the table simultaneously on per row basis when underlying tables are manipulated. Using additional information like time to sort

out which rows should be updated each time. Hence the transaction triggers that the table view is scanned and the most recently updated records applied to corresponding rows in the materialised view.

- lazy materialised views, updated when the transaction commits, thus being integral part of the transaction. Yet there has not been found a way to hook a trigger inside a transaction in PostgreSQL so presently this is impossible to implement but worth future researches.
- very lazy materialised views, equivalent to 'Snapshot materialised views' except on per row basis.

The snapshot materialised view with trigger on **`INSERT`**, **`UPDATE`** and **`DELETE`** was implemented for the prototype. Eager or lazy materialised views were preferred but too complex to implement. Firstly, it is theoretically impossible to make a lazy materialised view in current version of PostgreSQL with pl/pgSQL only. Secondly because it was found difficult to update 'transaction attribute' like timestamp, on per row basis, when editing in uDig, which is a prerequisite to enable the functionality of eager materialised view. An idea for further development PostgreSQL would be to record transaction statistics for each row, enabling efficient updating of the materialised view. For instance if a node is modified, it is still necessary to update the related boundaries in the **`Boundaries`** table and re-realise the affected parcel geometries (that need to be updated in the materialised view). *Appendix F: Materialised Views in PostgreSQL* documents how materialised view were constructed in the prototype.

The advances of using materialised view over the solution employed in the first attempt are multiple: it is unnecessary to compute the geometry of the parcels each time refreshed; and the geometry can be spatially indexed (R-tree or Gist), thus further enhancing performances when viewing and querying the data.

7.2.3 Physical model

The physical model of the second attempt ICM prototype is shown in Figure 46. It displays the LRD package and its conceptual role within the ICM package. The classes in coloured in white reflect the ICM prototype, there as the PostgreSQL/PostGIS database that was constructed using sample data provided by LRI.

It can be observed that the topology presented in the physical model here differs somewhat from the conceptual one presented in the CCDM and further in the ICM. The reason for that is down to the SDBMS used. CCDM states that edges are constructed using operation referred as **`boundary()`** returning the topological **`TP_EdgeBoundary`** as defined in the ISO-19107 Spatial Schema. Faces are similarly realised returning **`TP_FaceBoundary`** (ISO 2003). PostGIS/PostgreSQL does however not support topology by default so other means have to be employed to realise this, using both views and materialised views, as has been discussed above. Conceptually the ideology is the same.

The UML notation of the diagram may look little bit different to conventional modelling methods. The reason for this is the inclusion of views and materialised view and how to visualise that in a UML diagram. The way this was implemented is under influence of Scott W. Ambler using dependencies and stereotypes to express the relationship and role of indexes, views, materialised views and keys etcetera (Ambler 2005). It should be observed that the materialised view table, **`Cadastre_mv`** in the diagram, is used when visualising and querying parcels in client software.

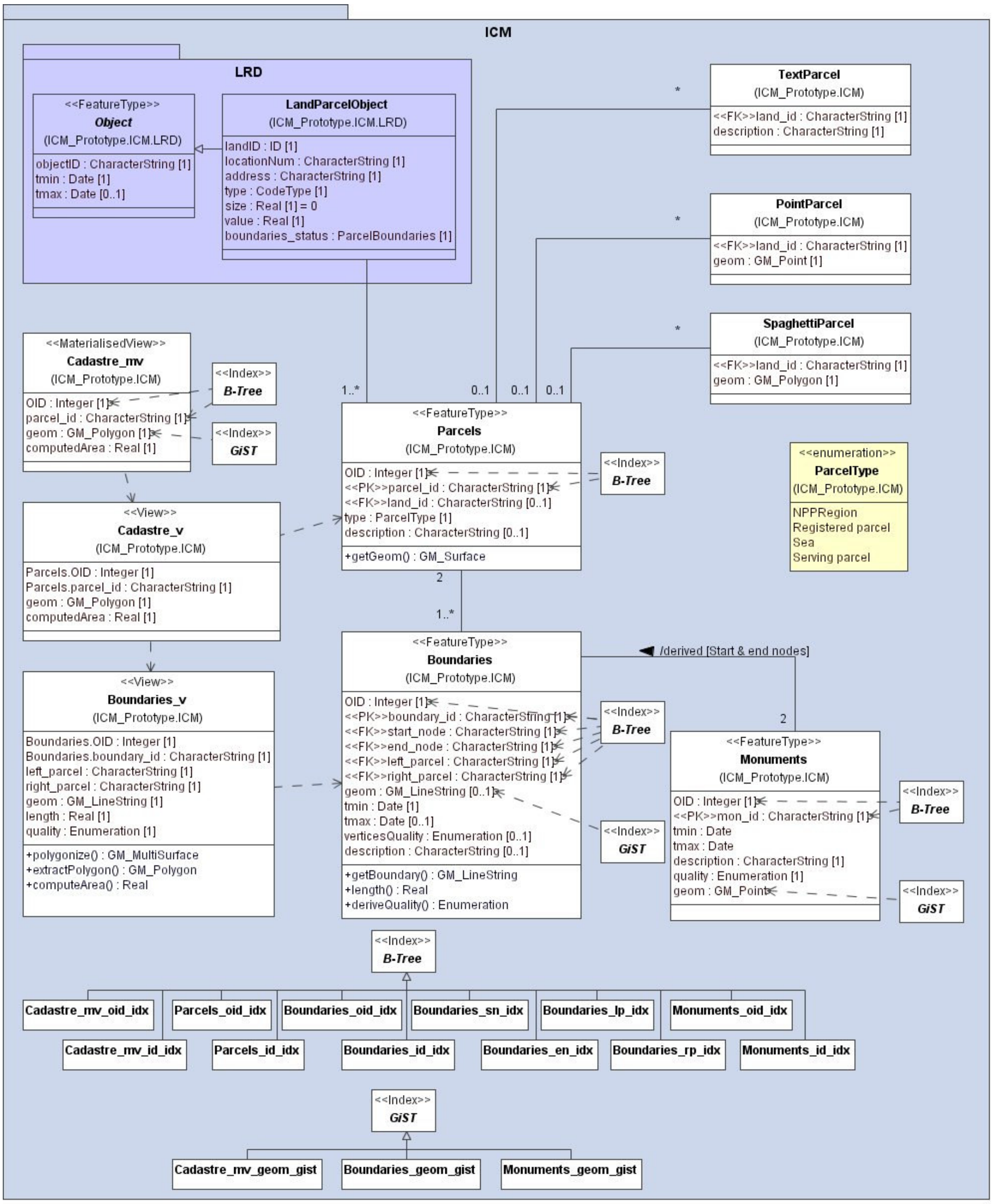


Figure 46: Physical model of the second attempt presented with the link to the LRD, within the complete ICM (image: Author).

7.3 Transactions and concurrency control

There is relatively small chance that two users simultaneously edit the same or adjacent parcels except of incidents of extremely large parcels, like the complete interior of Iceland. It would of course be unacceptable that all parcels adjacent to the interior would be locked for editing only as one of these parcels where manipulated. This could be bypassed by e.g.:

- splitting the interior into smaller parcels, which is however not a nice solution as it creates redundancy;
- permanently lock the interior and even all collective land for other editing than by administrators of the system. In fact this is a reasonable solution as these

boundaries are only obtained through court and thus not a responsibility of the municipalities to maintain.

The establishment of separate road parcels could also generate problematic parcels that similarly to the interior could affect the manipulation of many adjacent parcels. A solution to this is offered by Oosterom & Lemmen (2001) introducing partial locks as described in section 2.5.3. Getting rid off registering roads as partitioning land object, elaborated on in section 5.4.2 is also relevant in this context.

Some time and effort was spent on the issue of spatial locking with the open-source software at offer. For example were uDig tested for simultaneous editing with the earlier version prone to crash or even worse, stop communicating with the database and act independently with out an error messages, if more than one user was editing the database at once. This occurred especially if the feature edited were the same. Direct PostGIS connection proofed also to be more reliable than editing with WFS-T access but this has been refined since earlier versions of uDig.

Another issue was how to create locking mechanism that prevented two users working within the same area simultaneously. Time did not allow that concrete solution to be developed but a simple procedure, based on the idea of creating specific lock-layer that constrained manipulation of other data in the SDBMS, could be like this:

1. User logs into the SDBMS using conventional database connection.
    a. there are four layers displayed: monuments, boundaries, parcels (materialised view) and 'lock-area'.
2. User chooses lock-area for editing.
3. User draws polygon representing area that should be locked for editing
    a. Lock-area is assigned unique transaction key
    b. Lock-area cannot overlap with other lock-areas
    c. Lock-area is time dependent in the sense that if it is not unlocked within specific time it will unlock it self without changes being made
4. User makes changes to cadastral data within the defined lock-area
5. User commits changes
6. Triggers check if changes made are within an area-lock hold by committing user
    a. if true, the changes are committed and area-lock released
    b. if false, changes are rollbacked and area-lock is released.
7. transaction is completed.

Using more sophisticated and customised system than uDig it could be possible to develop more complex logging mechanism, enabling users to hold multiple transaction keys at once.

7.4 Evaluating system architecture in perspective of ICM

Ideal system architecture for cadastre has to serve the organisational setup of roles and responsibilities as much possible, while maintaining external integral appearance. Enabling the possibility for outsiders to access and enquire the cadastre, as it was unitary, but not sets (possibly roughly assembled) of independent data sources. Equally important is the accessibility of an insider to the system, to alter and update the information in a practical environment making integration of other relevant spatial information possible.

This sub-chapter discusses and evaluates the options available to realise such system architecture. The options considered here for LRI to implement spatial recording of parcels parallel to present LRD and in coherence to proposed ICM are:

- fully centralised, thus only manipulated by the employees of responsible institution LRI. Accessible for remote querying only with use of the Internet;
- centrally stored but locally maintained data, as would be the case if municipalities would be responsible for updating the geometry of parcels, while LRI would have the role of administering and coordinating the whole system;
- distributed but locally maintained system, whereas each municipality (or union of municipalities) would operate and maintain its own cadastral system. However it would be serviced as a federated database in such way that individual cross-system query would return result with the databases appearing to be integral. The role of LRI here would be to federate the databases and coordinate the whole implementation; and,
- distributed but remotely maintained system, where a user in one municipality can access and manipulate data stored on remote server/database in another municipality. This could require development of a sophisticated middleware that would route commands to relevant servers along with controlling concurrency.

When evaluating these options there are several factors that need to be taken into consideration. Next few sections discuss this categorically before concluding on the most ideal setup for implementing ICM.

7.4.1 Complexity of system and design

The more complex the design of the system is the greater is the start up cost. Experts need to be consulted, data models and transactions procedures have to be designed, and the complete time to get satisfying results can be enormous (and costly). Very simple and maybe insufficient design could however create costs in the long term because of loss of opportunities. Another related aspect is the complexity of system maintenance because very sophisticated system design tends to require highly educated and informed staff, making it even over dependant on particular individuals or experts.

7.4.2 Editing

Complexity of manipulating updating cadastral data is of high importance and much depended on client interfaces or data exchange format between client and server. Here are two options available. First is to design special client interface that user can employ to remotely access and edit the cadastral database. Given that there would only be one central cadastral database this would create a homogenous system with homogenous clients.

Benefits of such system would e.g. be consistency of registration and easier system management. The drawbacks are though costs of designing and implementing interface and that user would be using dual system. Therefore one client for registering cadastral data, and another different client for using cadastral data together with other geographical information in GIS.

The second option is to design a heterogeneous system where each user is free to use the client that he prefers, as long he support export to some common data format that can be used in data exchange. The obvious candidate for such format is geographic mark-up language (GML) as specified by OGC, especially its newest version that supports topology in spatial data. However, this has also its drawback as even though commercial GIS software will eventually be able to process GML3, the experience has shown that even with shared objectives, the GML output from diverse products, using the same reference data, is just not yet comparable (Vries & Oosterom 2005). Topology and especially customised topology, as is proposed for the implementation of ICM, is not going to make things easier.

7.4.3 Hardware and software

Choose of appropriate SDBMS is crucial at the beginning stages at the cadastral project. Many questions have to be asked and answered. Several factors can be observed, for instance from chapter 2, affecting the choice of appropriate spatial database. Among these are e.g.:

- spatial data and feature model;
- topology structure support;
- materialised views;
- consistency checks, concurrency control and locking mechanisms;
- spatial and functional indexing;
- spatial clustering;
- versioning, temporal support;
- integration of customised triggers and functions;
- reliability and performance;
- integration in system architecture; and,
- costs, documentation and support.

This is nowhere an exhaustive list of all decisive factors regarding choose of database, but those that were met while utilising the ICM prototype.

Both open- and closed-source databases have here their pros and cons. Oracle Spatial for example does not comply with the OGC Simple Feature Model as it uses its own spatial model (**`SDO_Geometry`**), while the open-source DBMS like PostGIS/PostgreSQL and MySQL do.[13] Oracle Spatial however supports materialised views and recently started to offer topology, which the open-source DBMS do not support. The open-source however offers their solution without charge. Experience like documented in IDABC (2003) further indicates quicker response time and more flexible support within the open-source community. The drawback of the open-source is though lack of responsible manufacture. If serious faults are discovered in the open-source software, e.g. when utilising it, the users have to solve it themselves, or rely on the open-source community to do so, while the commercial developer would be morally or even legally responsible to react.

To fully evaluate what database is most appropriate for each system setup needs more detailed research and comparison of options, than can be done here. An idea of future research could be evaluation of the SDBMS currently available considering the implementation of several distinct cadastral systems. Taking into account few predetermined factors like topology, costs, performances and security issues.

GIS software and database front ends to access and manipulate spatial data in SDBMS are provided both by the open- and closed-source developers. It is important when software for cadastral purposes is chosen that it fulfil all the basic needs and requirements asked for. Does it support WMS and WFS layers? Can it perform WFS-T transactions? Is it customisable to enable login access and edit in a remote SDBMS?

It depends heavily on the system architecture and SDBMS which software is most appropriate each occasion. Many commercial solution are e.g. not capable to connect to open-source SDBMS (Intergraph and ESRI do not support PostGIS by default but Safe Software FME does) and some commercial solutions can only view WMS/WFS generated by service of own origin. These integration problems could temporarily

[13] Appearantly current version of Oracle Spatial does support OGC Simple Feature Model.

prohibit heterogeneous user/client system solutions and prompt the maintainers of the cadastre to adapt all to the same software solution.

There is an enormous supply and choices of hardware, but this is important aspect of developing cadastral system. Because of lowering hardware prices with even increasing performance, this cannot however longer be accounted as decisive factor when developing local systems. The computer technology is used by default today, with only questions on appropriate equipment, configuration and tuning.

However, the more remote and distributed the solutions are required to be, the more dependence is on hardware infrastructure, like networks, bandwidths and availability. Consequently factors like speed and performances start to weigh in.

7.4.4 Appropriate setup for ICM

It can only be observed from the discussion above that to find the most optimum setup for cadastral system is like to find they correct way through a dense jungle. To evaluate what system architecture, software and hardware suit a cadastral project needs extensive cost/benefit analyses on available options and requirements. Necessary adoption to legal framework and legal responsibilities can further function as obstacle on the utilisation fo some of the solutions.

In the case of implementing the ICM prototype, the decision was to go for 'centrally stored but remotely maintained' cadastral system and with employing only open-source solutions. Centrally stored but remotely maintained system was chosen as that fits both the distributed administrative organisation of land registration in Iceland and the central approach and coordination of LRI needed to accomplish the task. The premises here are:

- Licensed surveyors are responsible of surveying parcel boundaries at the cost of parcel owner;
- Municipalities are responsible to update the cadastral section of LRD, thus bringing the surveyed boundaries in the land recording system;
- County registrars notarise contracts and formally establish/delete properties from the registry;
- LRI operates the cadastral SDBMS, maintains consistency and coordinates the action of local cadastral authorities (municipalities) and relevant parties. LRI is furthermore responsible to share the information in suitable, standardised manner, to interested individuals, companies and other public institutions, for querying or integrating in their own GIS.

This corresponds to current organisational setup of the LRD. The benefits of centrally maintained system here is to avoid large expenditures for municipalities as the local cadastral authorities (referring here only to the spatial role) and create uniformity of registration over the whole country. It is however essential to keep the updating of cadastral maps at local level. For that are three main reasons:

- not to create new expensive overhead at the LRI by using the manpower already at municipality level;
- local municipality officers are best suited to oversee boundary mapping with their local knowledge of the area; and,
- consistency of the data is managed in a centralised database.

7.5 Remarks

This chapter has in its undertaking covered a lot, among it the diverse considerations and implementation aspects of adding spatial dimension to land registration system as the LRD in Iceland truly is. The results of the hard work that went in constructing the ICM prototype cannot be pictured in a text but practical experience can be shared.

A lot of effort went into the first attempt, constructing spatial topology in PostGIS / PostgreSQL. In the beginning of the research this was unknown field of area and future topic for PostGIS, with no reference found at all, despite exhaustive search. Now there seem to be escalating interest of implementing topology support in PostGIS as will be explained later on.

Enormous time and effort went also in researching GML3 and its applicability as data exchange format for cadastral data. Data model of the ICM, modelled in MagicDraw 9 was translated with ShapeChange from Interactive Instruments GmbH to GML3 application schema with a little adjustment from Wilko Quak (see documentation for ShapeChange in Portele 2004).

No efficient way was however found to convert from and to GML3 when extracting or inserting data in the SDBMS, using its GML3 topology potential, unless writing the GML code by hand. Due to its complexity it was also considered too difficult, yet an ambitious research topic. To create procedure that by using GML Application Schema could be capable of creating GML document from SQL query on SDBMS or vice versa, as a mean to access the database. The application Snowflake developed by Snowflake Software has reputation of some potential in this direction, using Oracle, without it being researched further here.

The next few sections will discuss in more detail the practical experience gained by implementing the ICM prototype. Section 7.5.1 discusses the applicability of open-source application in context of experience, while section 7.5.2 recapitulates on the potential of the prototype it self. Finally, section 7.5.3 presents a future vision on how cadastral transaction could be constructed in not so distant future.

7.5.1 Open Source

The experience of utilising open-source products was really positive and somewhat surprising. It was surprising how frequently the open-source software were updated and enhanced with new functionalities or bugs fixed. Since uDig first release in June 2005 five updated versions have been published (1.0.1-1.0.5) and version 1.1 is just around the corner (11. November 2005). Monitoring the uDig developers' mailing list it could be observed that changes and additions were made to the software on demands from other users illustrating the flexibility of the open-source community.

Similar story can be told of the development of PostGIS, which since its release of PostGIS 1.0 in April 2005 has developed rapidly over the last few months. The development of these two above applications is actually in both cases leaded by the Canadian organisation Refraction Research. Now with the latest release of PostgreSQL (8.1), PostGIS has become component of the DBMS and expected in PostGIS 1.1 is support of topology structures that already has reached pre-alpha stage in development (18. October 2005). This topology does not seem to reflect the topology as defined in ISO-19107 Spatial Schema. Its data schema is presented as follows (PostGIS 2005, [3]):

- Node
    - node_id integer PRIMARY KEY
    - containing_face integer REFERENCES Face.face_id
    - geom geometry (a point)

- Edge
  - edge_id integer PRIMARY KEY
  - start_node integer REFERENCES Node.node_id)
  - end_node integer REFERENCES Node.node_id)
  - next_left_edge integer REFERENCES abs(Edge.edge_id)
  - next_right_edge integer REFERENCES abs(Edge.edge_id)
  - left_face integer REFERENCES Face.face_id
  - right_face integer REFERENCES Face.face_id
  - geom geometry (a linestring)
- Face
  - face_id integer PRIMARY KEY
  - mbr box2d (can be NULL)

Apart from the fact that this approach realises the geometry of the faces with 'winged-edge method' (see 2.3.2) it more or less corresponds to what has been achieved within this research. It is however recommendation of this report to emphasise future development of storing intermediate points of edges as ordered sequence of points (multipoint) but not linestring as suggested. Another thing is that this development of topology support seems to prefer envelope (minimum bounding box) as way to spatial indexing the parcels but question marks are set here on updating procedure.

Other open-source software utilised gave also positive impression of the open-source products. MapServer seems to be fully compatible to its proprietary counterparts (see *Appendix I: MapServer Configuration* for the map file used in the experiment) whereas GeoServer has extreme potential but lacks stability. However the reason behind that could be found either in GeoServer connection to PostGIS or uDig.

Concluding on the applicability of open-source software for real problem like the implementation of cadastral system, this report counter argues the truth of the old saying: "you get what you pay for." For open-source applications you pay nothing but receive a lot. The open-source software do maybe not come out of the box 'ready-to-use', but the expenses spent in buying proprietary system could be spent in customising and extending the open-source product to fit users need even better. In cases where users-need do not fit the proprietary products available, generating need to develop own customised system, open-source can proof invaluable in providing a core of functionalities.

This could e.g. be the case of uDig functioning as users front-end to a cadastral system in Iceland. It is expensive to equip all municipality officers in Iceland with expensive desktop GIS to be capable of manipulate cadastral data in accordance to responsibilities. The solution could be found with uDig providing core functionalities, with diverse customised add-ons providing other functionalities needed for cadastral purposes.

7.5.2 ICM prototype

The contribution of the prototype implemented in this research is of diverse nature. First of all it gave insight for author into SDBMS that he did not have before and made him realise that there is more distant between conceptual modelling in UML and actual implementation schema in a database. Again and again the illogic in UML diagram was discovered when experiencing the implementation in a database. Therefore it can be stated that creating a prototype is an essential tool for a modeller to have insight in the physical level of the implementation.

Secondly, this prototype contributed on the implementation of topology structure as discussed in the section above. The intention is to create a document describing this implementation, its pros and cons that could benefit the general development of topological support in PostGIS. As a participant in the open-source community it is important not only to receive discoveries from others but also occasionally contribute your own.

Finally it is thought that this prototype can contribute to the implementation of recording spatial extent of parcels as a part of cadastral registration in Iceland. It clearly shows how parcels can be registered using the topological primitives: node, edge and face, without too much expense. Concluding that complex topological system is not longer only affordable and manageable by large sophisticated institutions, as in The Netherlands with its strengths in research and development, but also in smaller and less equipped institutions as is matter of fact in Iceland. Experiments and tests indicate that it is easy to manipulate the data remotely using WFS-T or direct database connection. By designing transactions that include recording extent of land as a part of the procedure, there is optimistic feeling that sophisticated spatial recording can be established in Iceland as part of the cadastral registration. Next section gives author's vision on how this could be managed in not so distant future.

7.5.3 Future vision

In this section are two scenarios presented portraying author's vision on the future environment of the Icelandic cadastre. It is important to acknowledge that these scenarios are meant to be realistic and achievable at present time with current technology.

Scenario I. *Conventional cadastral transactions*

Imagine an employer (a client) at the city office in Reykjavík concerned with cadastral procedures within his area. To view and manipulate all spatial data in his local LIS he uses his own applications.

He connects to the cadastral database through the internet, remotely stored at LRI, and displays and queries its content like other spatial data in his LIS, though he cannot edit it. To be able to edit the data he can chose between to two possibilities, *simple/minor editing* or *complex/major editing* as explained below.[14]

<u>Simple/minor editing</u> employs the web interface provided by the SDBMS server at LRI, were the client can make simple editing like splitting and merging of parcels, relocate legal boundaries and changing attached attributes, using transactional WFS or connecting directly to the SDBMS. The total transaction could be as follows:

1. Client logs on the SDBMS server
2. Client chooses the parcels he wants to edit by drawing a polygon around area of interest or by submitting SQL query
3. Server automatically implements locking in SDBMS for outside editing that allows:
    a. only the client involved to manipulate parcels inside the chosen area, and;
    b. none to manipulate a ring of parcel around the ones

[14] Several constrains are to the examples given. First, it is not clear where land_ID would be assigned to new parcels, whether that would be the role of the client or server. Secondly, these examples only cover the cadastral part of land transactions, not administrative or land registrar.

that the client selected or edges bounding the selected parcels and the ring parcels.

4. Client performs editing and submits to the SDBMS server
5. Server receives the changes, checks consistency and topology, attaches transaction information to the edited parcels, updates SDBMS and unlocks the selected area.

However, this interface becomes cumbersome if it is required to edit large set of data, e.g. when updating large areas or whole urban districts.

<u>Complex/major editing</u> solves the need for mass manipulation of cadastral data and offers more client-oriented solution by shifting the editing from SDBMS web interface, to client's own computer and applications. Example of transaction could be as follows:

1. Client logs on the SDBMS server
2. Client chooses the parcels he wants to edit by drawing rectangle around area of interest or submitting query
3. Server automatically implements temporal locking in SDBMS for outside editing, as already explained, and offers the client to download the selected area in preferred format
4. Server extracts selected data from SDBMS into GML using ICM before converting it to selected format made available for client to download
5. Client downloads the data and performs editing using own geo-applications, e.g. provided by software vendors like ESRI, Intergraph, Bentley, AutoDesk and MapInfo
6. Client submits GML file to SDBMS that performs consistency and topology checks before attaching transaction information to it
7. Server uses the GML dataset to update the SDBMS and releases the spatial lock.

Scenario II.
Location-based services

Using GML as transaction format enables opportunities for location based services when either updating or retrieving data from SDBMS. Surveyors could opt to update or query the database, while working in the field using handheld computer (PDA) connected to mobile phone and GPS receiver. Other similar examples of usages could be:

- Employers at utility companies or diverse institutions using PDA to enquire for parcel ownership while working in the field
- Farmers to place fences along boundaries
- Emergencies like police, ambulances and fire services to effectively locate addresses when required (e.g. summerhouses)

8 CONCLUSIONS

The quotation at the start of the introduction, referred to Benjamin Disraeli, reflects to large extent the motivation of writing this report: "the best way to become acquainted with a subject is to write a book about it." The problems undertaken are manifold, ranging from conceptual modelling of CCDM and ICM, to technical implementation of SDBMS and especially the ICM prototype. Moreover, the current land registration practices in Iceland were analysed and documented and a brief introduction given to open-source geo-software. Following is a summary of the findings of each chapter.

Chapter 2 explored the theory and capabilities of current SDBMS. The options found for implementing cadastral system were vast, but the most important aspect here was how to correlate users need to a spatial model in a cadastral SDBMS. A topological model is required to combine the needs of the fiscal/legal oriented parcel focus with the more boundary focused surveyors approach.

The topological approach is in harmony to the CCDM introduced in chapter 3. The CCDM is a much needed contribution to the cadastral debate and future development. It sets an example for modelling and developing cadastral systems, while at the same time it defines the core classes of cadastral registration, enabling cross-platform interoperability. The view of this report is that this innovative is important but little bit too ambitious in its approach. Different cadastral systems extend in different ways as was experienced by Hess & Vries (2004) and if the variations of all cadastral systems are to be included in the model it would grow in complexity contrary to its objectives. An alternative solution would be to focus its development more within cultural homogeneous area, for instance Europe, before extending it to other regions. Further to make distinction between the more conceptual paradigm objective and the technical definition of core classes. Finally it is considered beneficial here to sharpen CCDM's approach by identifying the core packages, emphasising on each of their development separately.

The land registration in Iceland was discussed in chapter 4 attaining the conclusion that general framework of land registration is good, even though spatial extent of parcels is not recorded in the system. An additional finding is that the relationship between parcel and non-parcel real properties is different to that found e.g. in The Netherlands. In Iceland, land and building can be owned by different parties as long as permanent right is established between, e.g. by making long-term lease contract. Finally, the availability of spatial data to employ in the cadastral system was discussed. Much data is available, but often inconsistent and of low spatial accuracy. However, by designing the system in such a way that it could gradually be improved, it can be stated that this data is valuable input for cadastral implementation in Iceland.

The making of the ICM along with comparison to CCDM was the subject matter of chapter 5. In general it was observed that the ICM fits relatively well into the CCDM presented in Lemmen et al (2005) with only few exceptions:

- segregation of parcels, ownership that is space partitioning, and non-parcels like buildings, utilities, land use, non geometric real estates and possibly roads;
- the relationship between land and buildings is missing in the CCDM but important for the ICM where **`RealEstateComplex`** (comparable to **`AppartmentComplex`** in CCDM) is always associated with at least one parcel;
- the relationship between building complex and a unit within a building should be set one to many. Meaning that an **`ApparmentComplex`** or as referred in ICM, **`RealEstateComplex`**, can be composition of only one **`AppartmentUnit`** /

`RealEstateUnit`. Further this is composition as if we remove the unit the complex cannot exit any longer. This allows registration of separate ownership of land and buildings[15];

- introduction of `TextParcel` as representing land parcels that because of some reason do not yet have any geometry recorded. Instead these parcels can be located by textual reference like e.g. boundary description;
- introduce temporal element in registration of boundaries enabling fuzzy boundaries to follow dynamic land features like glaciers and coastlines; and,
- remove registration of negative rights implied by public law, like `RestrictionArea`, from the specialisations of `RealEstateObject`. It is argued here that this cannot be regarded as an object for cadastral registration. Currently `RestrictionArea` covers registration of negative rights like public restrictions and positive rights like in the case of utilities. This report suggested that separate `Utility` class to register the positive right is more appropriate. Restrictions that are implied by negative right only, like public restriction are better placed in separate repository with association relationship with CCDM `RealEstateObject`.

These recommendations are all included in the ICM conceptual model. Another idea put forward in chapter 5 is regarding registration of land parcels. It proposes that it would be more convenient to register roads simply as structures like buildings, with permanent right to land, instead of expropriating and creating special road parcels. By this creating `Road` class within `RealEstateObject` that attaches the owner to it via `Right` (recall Figure 36).

Chapter 6 is a prelude to chapter 7 on the implementation of ICM prototype. A research was carried out on the availability of open-source software capable of handling spatial data. Several geo-applications were spotted out, tested and analysed. It was found that open-source software is becoming a realistic option for institutions and companies to construct SDBMS, to access data and share it. The evolution of open-source is a healthy input into the GIS market, challenging the monopoly of controlling developers of proprietary GIS products.

Finally, chapter 7 documented the case study implemented in this research with practical experience shared. The highlight of this was the successful implementation of spatial topology in PostGIS and experiencing the use of open-source software tools in geo-applications. In fact complete system architecture could be constructed for cadastral registration using only open-source licensed products. Stability and reliability drawbacks could though appear as obstacles as some of the applications tested were quite unstable. At the end of the chapter a short elaboration was conducted on the ideal system architecture for the ICM. It was concluded that centrally stored SDBMS but remotely updated architecture would fit best to the organisational setup in Iceland. The municipalities would each have their remote access to the ICM cadastral database at LRI and update remotely through their clients. Using open-source SDBMS could require use of homogeneous clients, as PostGIS is not presently widely supported by proprietary GIS developers. Using WFS-T would though eliminate this need for client homogeneity, as long as the clients support the OGC specification.

[15] The names ApparmentUnit/ApparmentComplex are obviously also not descriptive enough to serve the purpose proposed here.

8.1 Research question revisited

In the introduction chapter the following question was raised:

> *In what way can the Core Cadastral Domain Model (CCDM) and open-source software benefit the development of cadastral registration in Iceland?*

The outcome of this research is that the development of the Icelandic cadastral registration can greatly benefit from the CCDM, both in general and in implementing spatial delimitation of parcels. The CCDM further offers a conceptual framework and healthy model driven architecture approach to the whole development. Employing CCDM would moreover bring the Icelandic cadastral registration on level of international cooperation of cadastral research and development, which could proof invaluable in the long run.

The new and revised version of the CCDM which is based on the experience of modelling the proposed ICM, gives greater motivation to include the CCDM in the future development of the cadastral registration in Iceland.

The role of open-source applications in that development could be diverse. The research report argues that several open-source applications available today are serious candidates in developing spatial enabled cadastral registration. Examples of applications are PostgreSQL with PostGIS to store spatial data in SDBMS; MapServer on Apache HTTP server to share spatial data in a web environment; standalone GeoServer enabling WFS-T to spatial data; and finally uDig as a desktop client that can be extended with diverse customised functionalities by accessing and editing the open source code.

Before employing open-source applications in cadastral development, few things have to be taken into consideration. Most open-source clients are not complete coming out of the box but would need additional tuning and customisation to suit in a cadastral development. Another thing is that there is in most cases no one responsible manufacture liable for the correctness and reliability of the application, and the product could even drop out of development without warning. This is because development of open-source applications is largely depended on the enthusiasm and drive of the open-source community. This results in the maxim that if open-source applications are to be used, one should handle them with certain reservation and above all be an active member of the open-source community developing the product.

8.2 Recommendations

Numbers of recommendations have already been put forward upon till this point within the report and summarised in conclusions. It is further recommended here that the future development of the Icelandic cadastral registration will employ the ideology introduced by the CCDM development and be compatible to its model. Using the model driven architecture approach would sharpen the overall development of Icelandic cadastral registration while further creating bases for automatic retrieval of DBMS DDL statements, GML/XML Application Schemas and facilitate future addition to the cadastral registration being efficiently integrated. In this context a clear models are needed covering the complete cadastral registration in Iceland, which should be produced at the initiative of LRI. These models have to address (1) the conceptual model representing the classes of registration, (2) logical model with attributes and constraints, and (3) the physical model covering technical implementation. For instance one idea elaborated in the report that needs to be discussed at a conceptual level is if roads should be registered as **`RealEstateObject`** instead of **`LandParcelObject`**. This would need an exhaustive research of legal and fiscal consequences, before deciding on the idea and modelling it on logical level.

Step by step development of adding spatial extent of parcels to the cadastral registration in Iceland, based on the experience of implementing research and prototype, is recommended as follows:

1. Make policy document which contains requirement analyses and decides on viable strategy to include spatial delimitation of parcels within the cadastral registration.
2. Design conceptual model, what classes are to be found and what should be registered?
3. Identify suitable system architecture reflecting organisational setup, addressing e.g. if the database should be central or distributed.
4. Identify and choose suitable hardware/software/interface solutions.
5. Design logical model and system architecture.
    a. Decide on spatial model.
    b. Important is the capability to include dynamic boundaries and keep track of data quality with the objective of gradually improving the system.
    c. Parallel design model of transactions and procedures.
6. Design physical model, implement the SDBMS and test its applicability.
7. Influence appropriate changes of law and regulations. This is extremely important to harmonise the whole project and assign clear roles to responsible actors.
    a. It should be stipulated that land property cannot change owners or be modified, without its boundaries being submitted/registered.
    b. It should be stipulated that all land owned by local or central government should be identified, surveyed and registered within a specific time limit.
8. Create web interface for public to access the multi-purpose cadastre where they can inspect and print out boundaries and make formal suggestions. Moreover serve the data to professional desktop GIS users with WFS.
9. Identify and integrate different sources of cadastral data already available, both of high and low quality, and import into the system.
10. Insert or update boundaries resulting of new surveys (e.g. because of transaction of real property).
11. Encourage municipalities and landowners to clean up boundaries, e.g. incident to revision of municipality planning.
12. Revision of system and procedures.

The steps indicated above are not finite, as it is possible after concluding one step, that other step has to be revisited, e.g. in the case of differences between logical and physical models (that is if the logical model is not practical in implementation). Also are some steps interdependent in the sense that system architecture cannot be chosen without considering available software solutions and vice versa (steps 3-5).

Concerning more technical issues, committing transactions with WFS-T is another field of area that needs more research, especially how to make such transaction secure and traceable. Currently there is no possibility to include logging when using GeoServer WFS-T indicating that everyone with access to the URL can visualise and manipulate the data. An idea would be to include the user information and password in the transaction request sent to the server.

To end with the following recommendation is added, although it might not be directly related to the research. In conducting this research and the experience of working with open-source software evoked the idea that open-source software should be integrated into teaching and studying geomatics. With its often raw interface open-source application is a good way for students to learn beyond what is usually covered with textbooks and could serve as a tool to gain practical experience. Recommendations are here put forward that the GIS section of OTB becomes a participant, mainly through its students, of one big spatial open-source project like uDig, MapServer, GeoServer or PostGIS. This could be in the form of testing the software, creating add-on applications or write documentations. An idea of add-on would be function in PostGIS that calculates shortest route given set of edges and nodes, using e.g. Dijkstra algorithm, or taking part in the topology support development. There are in fact number of reasons why open-source licensed software should be preferred in teaching rather than other (proprietary) software:

- they are vendor neutral (students learning their trade on one particular commercial GIS software often become vendor oriented);
- they are transparent (by studying the source code it can be explored beyond the interface);
- they are free (students can practice at home);
- they offer great deal of free support (the open-source community is really effective here);
- most of them are OGC/ISO standard oriented (like we want the GIS world to become);
- there is possibility to influence the development of the software, possible creating paradigm for other applications to follow; and,
- students work is given enhanced importance.

8.3 Contribution of work

The academic contribution of this project is diverse. In short it can be stated that:

- it has influenced the newest version of the CCDM, which now includes some of the recommendations put forward here, with the author listed as co-author;
- for the first time a detailed documentation and analyses on the Icelandic registration system appears. A large step to this direction was taken in research project autumn 2004, with the documentation here going deeper into the technical implementation;
- the report gives rare overview of the potential of open-source software available at present, with spatial topology implemented successfully in PostGIS; and finally,
- contributed in the author's successful job application to be a part of a project team which will develop and establish cadastre in Iceland. This has caused that the scope of this research has widened somewhat from its initial objectives.

8.4 Discussion and experience

There are several reasons why this research drifted somewhat from its initial objectives. First of all were the difficulties of re-engineering the LRD model, with no useable references to be found, which consequently implies that the system is developed with a bottom-up approach. Enormous time was spent trying to model and understand the LRD and several diagrams drawn, whereas every new discovery seemed revolt the preceding diagram. It however really helped to implement the prototype, as it kept the conceptual modelling down to earth and more realistic. This connection between the conceptual and implementation world I find essential because without it, it is easy to loose direction or even to create impractical solution.

This project became too large. The lesson I draw from this is the importance of properly constructing the main research question. It has to be narrow and direct. Further the fact that this question is defined right in the beginning and not altered during the implementation. The research question was initially too vague and honestly not sufficient:

> *How and in what way can "centralised-stored / remotely-maintained" approach benefit the development of nationwide cadastre in Iceland, modelled in coherence with the core cadastral domain model (CCDM) as suggested by FIG […using open-source licensed applications]?*

Looking back, this research would be structured differently if repeated. The modelling of the LRD is simply enough to fill one MSc thesis, as is the implementation of topological support in PostGIS. Because of the large scale of this research it was difficult to maintain a clear overview of content and objectives. Actually, during the final editing process around 20 pages of text were deleted, however, that was not an issue at all.

Making this research has been one big learning curve where the central objective has rightfully been served: to obtain good knowledge of cadastral registration and by that prepare myself for new tasks in my career.

BIBLIOGRAPHY

Andrésdóttir, Ásta Sólveig (2005): *Communication with e-mail October-November 2005*, lawyer at the Land Registry in Iceland.

Alþingi (2004): *Drög að frumvarpi til laga um Landeignaskrá* (e. Preliminary bill to law on cadastre), unpublished, in author possession and in Icelandic only.

Alþingi (2004-2005): *Frumvarp til laga um breytingu á lögum um skráningu og mat fasteigna, nr. 6/2001* (e. Bill altering the law on land registration and appraisal), Alþingi 131st national assembly 2004–2005.

Ambler, Scott W. (2005): *A UML Profile for Data Modeling*, Agile Data, URL (11-05): [1] http://www.agiledata.org/essays/umlDataModelingProfile.html.

Anderson, Brock (2005): A comparison of ArcIMS to MapServer, Refraction Research, URL (11-05): [1] http://ms.gis.umn.edu/community/conferences/MUM3/present/session10/msvsarcims/download.

Barry, Tom (2005). *Interviews and communications Mars-August 2005*, geo-information expert at the Land Registry in Iceland.

Benchikha, Fouzia; Boufaida, Mahmoud & Seinturier, Lionel (2001): *Integration of the Viewpoint Mechanism in Federated Databases*, Proceedings of the 2001 ACM symposium on Applied computing, Las Vegas, pages 280 – 284.

Bernhardsen, Thor (1999): *Geographic Information Systems – An Introduction*, second edition, John Wiley & Sons, New York, 372 pages.

Brown, David William (2002): *An introduction to object-oriented analysis: objects and UML in plain English*, Wiley, New York, 668 pages.

Cheng, Changxiu; Shen, Paiwei; Zhang, Mingbo & Lu; Feng (2004): *An Extended Locking Method for Geographical Database with Spatial Rules*, ICCS 2004, LNCS 3039, pages 972-980, URL (09-05): [1] http://www.lreis.ac.cn/article/2004chengcx.pdf.

Dale, Peter & McLaughlin, John (1999): *Land Administration*, Oxford University Press, Oxford, 169 pages.

FAI (2004): *Icelandic Agricultural Statistics 2004*, published by Farmer Association of Iceland in July 2004, URL (10-05): [1] http://www.bondi.is/landbunadur/wgbi.nsf/key2/hagtolur.

FIG (1995): *Statement on the cadastre*, International Federation of Surveyors, URL (10-05): [1] http://www.sli.unimelb.edu.au/fig7/cadastre.html.

FIG (1999): *The Bathurst Declaration on Land Administration for Sustainable Development*, International Federation of Surveyors Publication 21, URL (02-05): [1] http://www.fig.net/pub/figpub/pub21/figpub21.htm.

FMR (2003): *Handbók fyrir þinglýsingar* (e. Handbook for notarisation), third edition, Fasteignamat ríkisins.

FMR (2005): *Fasteignamat ríkisins – vefsíða* (e. Land registry in Iceland – webpage), URL (11-05): [1] http://www.fmr.is.

Gardner, Jonathan (2004): *Materialized Views in PostgreSQL*, URL (11-05): [1] http://jonathangardner.net/PostgreSQL/materialized_views/matviews.html.

GeoServer (2005): *The GeoServer Project– the open Internet gateway for geographic data*, URL (10-05): [1] http://geoserver.sourceforge.net/html/index.php.

GRASS (2005): *GRASS GIS - The World Leading Free Software GIS*, URL (10-05): [1] http://grass.itc.it/index.php.

Guðmundsson, Guðmundur J. (2000): *Fasteignamat á Íslandi – Frá tíundarlögum til Landskrár fasteigna* (e. Property valuation in Iceland – from tithe laws to modern land registration), appeared in 'Ársskýrsla 2000' (Year report), Fasteignamat ríkisins, URL (11-05): [1] http://www.fmr.is/lisalib/getfile.aspx?itemid=805.

Guttman, Antonin (1984): *Rtrees: a dynamic index structure for spatial searching*, in proceedings of ACM SIGMOD International Conference on Management of Data, pages 47—57.

Hagstofan (2005): *The statistical bureau of Iceland*, URL (11-05): [1] www.hagstofan.is.

Hallgrímsson, Heiðar (2004), Interview, Manager at the GIS division of the city of Reykjavík, August 2004.

Hanssen, Gjermund (2003): *Concurrency control in distributed geographical database systems*, ScanGIS'2003, The 9th Scandinavian Research Conference on Geographical Information Science, Espoo Kivenlahti, Finland, URL (09-2005): [1] http://www.scangis.org/scangis2003/papers/23.pdf.

Henssen, Jo (1995): *Basic Principles of the main cadastral systems in the world*, presentation from the workshop "From Research to Application through cooperation" in The Hague 1995, last modified 1996, URL (10-05): [1] http://www2.swisstopo.ch/fig-wg71/doc/delft_seminar/paper2.html.

Hess, Claudia & de Vries, Marian (2004): *From Models to Data: a Prototype Query Translator for the Cadastral Domain*, Joint 'FIG Commission 7' and 'COST G9' Workshop on Standardization of the Cadastral Domain, Bamberg, Germany, 9-10 December 2004.

Héraðsdómur (2005): *Verdict of district court in case nr.E-1517/2004: The Icelandic government against Sameigendafélagi Fells and Einari Birni Einarssyni*, Reykjanes district court, 42 pages.

IBM (2005): *IBM United States*, URL (11-05):
[1] http://www-306.ibm.com/software/data/spatial/;
[2] http://www-306.ibm.com/software/data/informix/blades/spatial/.

Icelandic law (2005): *Íslensk lög 1. október 2005* (e. Icelandic laws), edition 131b, URL (11-05): [1] http://althingi.is/vefur/lagasafn.html.

IDABC (2003): *Dutch use open source for Geoservices in public works*, URL (10-05): [1] http://europa.eu.int/idabc/en/document/3934/470.

IDABC (2005): *Balkan cities become e-municipalities with open source*, URL (10-05): [1] http://europa.eu.int/idabc/en/document/3920/470.

Ingibergsson, Haukur (2005): *Communications 13 July 2005*, director of the Land Registry of Iceland.

Ingvarsson, Örn (2004): *Interview August 2004*, manager of the Appraisal department at the Land Registry in Iceland.

Ingvarsson, Tryggvi Már (2005): *Technical implementation of cadastre in Iceland (centralised stored but remotely maintained – focusing on storage and transactions of geometric data)*, planning for MSc thesis, final version, unpublished, Delft University of Technology, 23 pages.

ISO (2003): *Geographic information – Spatial schema*, International Standard ISO 19107, first edition, International Organization for Standardization, Geneva, 166 pages.

ITTIA (2005): *db.* documentation*, ITTIA Embedded Database Development, URL (08-2005): [1] http://www.ittia.com/dbstar/manual/UG_Gloss.htm.

JTS (2005): *JTS Topology Suite*, URL (10-05): [1] http://www.vividsolutions.com/jts/jtshome.htm.

Kaufmann, Jürg & Steudler, Daniel (1998): *Cadastre 2014 - a vision for a future cadastral system*, the Working Group 1 of FIG Commission 7, URL: (10-05): [1] http://www.fig.net/commission7/reports/cad2014/download/cad2014_eng.pdf.

Kaufmann, Jürg (2004): *Assessment of the Core Cadastral Domain Model from a Cadastre 2014 point of view*, Joint 'FIG Commission 7' and 'COST Action G9' Workshop on Standardization in the Cadastral Domain, Bamberg, Germany, 9 and 10 December 2004, URL (10-05): [1] http://www.fig.net/commission7/bamberg_2004/papers/ts_04_01_kaufmann.pdf

Kresse, Wolfgang & Fadaie, Kian (2004): *ISO Standards for Geographic Information*, Springer-Verlag, Berlin, 322 pages.

Lake, Ron; Burggraf, David S.; Trninić, Milan; and, Rae, Laurie (2004): *Geography Mark-up Language (GML) – foundation for the geo-web*, John Wiley & Sons, Chichester (England), 388 pages.

Landnáma (1986): *Íslendingabók – Landnámabók* (e. Book of Icelanders – Book of settlement), edited by Jakob Benediktsson, Íslensk fornrit I, published by Hið Íslenzka Fornritafélag , Reykjavík.

Lemmen, Christiaan; Molen, Paul van der; Oosterom, Peter van; Ploeger, Hendrik; Quak, Wilko; Stoter, Jantien & Zevenbergen, Jaap (2003): *A Modular Standard for the Cadastral Domain.*, Dutch Cadastre, ITC Enschede and TU Delft.

Lemmen, Christiaan; Oosterom, Peter van; Zevenbergen, Jaap; Quak, Wilko & Molen, Paul van der (2005): *Further Progress in the Development of the Core Cadastral Domain Model*, FIG Working Week 2005 and GSDI-8, Cairo, Egypt April 16-21, 2005.

LMÍ (2005): *The National Land Survey of Iceland*, URL (10-05): [1] www.lmi.is/landsurvey.nsf/pages/islitolum.html.

Locke, John (1698): *Two treatises of government*. Second part, London. E.g. republished by the Cambridge University press in 1988.

Loftmyndir (2005): *Tilboð um þjónustusamning á sviði landupplýsinga og yfirtöku á rekstri og skuldbindingum Landmælinga Íslands* (e. Proposal for service contract in the field of GIS and taking over the service and responsibilities of the National Land Survey in Iceland). Loftmyndir ltd., URL (10-05): [1] http://www.loftmyndir.is/frettir/tilbod.pdf

Manola, Frank & Sutherland, Jeff (1997): *SQL3 Object Model*, National Committee for Information Technology Standards, 'Technical Committee H7: Object Model Features Matrix', edited by Frank Manola, URL (08-05): [1] http://www.objs.com/x3h7/sql3.htm.

Mapserver (2005): *MapServer Homepage*, URL (10-05):
[1] http://mapserver.gis.umn.edu/;
[2] http://mapserver.gis.umn.edu/doc46/wfs-server-howto.html.

Matthíasson, Viðar Már: (2003): *Fasteignarhugtakið* (e. The concept of real estate), appeared in 'Ársskýrsla 2003' (Year report), Fasteignamat ríkisins, URL (11-05):
[1] http://www.fmr.is/lisalib/getfile.aspx?itemid=1089.

Mjøs, Leiv Bjarte (2002): *New Cadastre in Norway*, FIG XXII International Congress, Washington, D.C. USA, April 19-26 2002, URL (10-05):
[1] http://www.fig.net/pub/fig_2002/Ts7-12/TS7_12_mjos.pdf.

Molenaar, Martien (1998): *An Introduction to the Theory of Spatial Modelling – for GIS*, Taylor & Francis Ltd, London, 246 pages.

MySQL (2005): *MySQL Reference Manual*, MySQL AB, URL (08-05):
[1] http://dev.mysql.com/doc/mysql/en/index.html.

Nytjaland (2005): *Nytjaland-Jarðabók Íslands* (e. Nytjaland – The Icelandic register of rural land parcels), project on behalf of 'The Agricultural Research Institute' in Iceland, URL (10-05): [1] http://www.nytjaland.is.

Quak, Wilko; Stoater, Jantien & Tijssen, Theo (2003): *Topology for spatial DBMS*, Delft University of Technology, Delft, 12 pages.

OGC (1999): *Open GIS Simple Features Implementation Specification for SQL 1.1*, Open GeoSpatial Consortium, URL (08-05): [1] www.opengeospatial.org.

Oosterom, Peter van & Lemmen, Christiaan (2001): *Spatial data management on a very large cadastral database*, 'Computers, Environment and Urban Systems', 25 4/5, p. 509-528.

Oosterom, Peter; Stoter, Jantien; Quak, Wilko & Zlatanova, Sisi (2002): *The Balance Between Geometry and Topology*, Delft University of Technology, 16 pages, URL (09-05):
[1] http://www.gdmc.nl/zlatanova/thesis/html/refer/ps/topo_geom02.pdf.

Oosterrom, Peter van & Lemmen, Christiaan (2003): *Towards a Standard for the Cadastral Domain*, Journal of Geospatial Engineering, Vol.5, No.1 (June 2003), p. 11-27.

Oosterom, Peter van; Lemmen, Christiaan & Molen, Paul van der (2004): *Remarks and Observations related to the further development of the Core Cadastral Domain Model*, Joint 'FIG Commission 7' and 'COST G9' Workshop on Standardization of the Cadastral Domain, Bamberg, Germany, 9-10 December 2004.

Oosterom, Peter; Lemmen, Christian; Ingvarsson, Tryggvi; van der Molen, Paul; Ploeger, Hendrik; Quak, Wilko; Stoter, Jantien & Zevenbergen, Jaap (2005): *The Core Cadastral Domain Model*, to be published in CEUS (Computer Environment and Urban Systems), available through URL (11-05): [1] http://www.sciencedirect.com.

Oosterom, Peter & Lemmen, Christian (2005): *The Core Cadastral Domain Model: A Tool for the Development of Distributed and Interoperable Cadastral Systems*, in Proceedings of UN Human Settlements Programme (UN-HABITAT) Expert Group Meeting (EGM) on Innovative Land Tools for Sustainable Urban Development, Moscow on 25-27 October 2005.

OpenSource (2005): *Open Source Initiative (OSI)*, URL (10-05):
[1] http://www.opensource.org.

OS (1996): *Skjárattan - Ársfjórðungslegt yfirlit vatnamælinga* (e. periodical overview of hydrography in Iceland), October - December 1996, URL (11-05):
[1] http://www.os.is/vatnam/aratta/skjaratta9641.html.

Óbyggðanefnd (2005): A committee established to define boundaries between property land (public or private) and peoples land (commons and wasteland) in Iceland, URL (11-05): [1] http://www.obyggdanefnd.is.

Portele, Clemens (2004): Mapping UML to GML Application Schemas, URL (11-05): [1] http://www.interactive-instruments.de/ugas/ShapeChange.pdf.

PostGIS (2005): *PostGIS Manual*, Refractions Research, URL (10-05):
[1] http://postgis.refractions.net/;
[2] http://postgis.refractions.net/docs/;
[3] http://postgis.refractions.net/support/wiki/index.php?PostgisTopology;
[4]http://postgis.refractions.net/pipermail/postgis-users/2003-June/002651.html;
[5] http://postgis.refractions.net/pipermail/postgis-users/2003-February/002038.html;
[6] http://postgis.refractions.net/pipermail/postgis-users/2005-May/007950.html;
[7] http://postgis.refractions.net/pipermail/postgis-users/2005-May/007891.html;
[8] http://postgis.refractions.net/pipermail/postgis-users/2002-March/000807.html.

PostgreSQL (2005): PostgreSQL 8.0.3 Documentation, The PostgreSQL Global Development Group, URL (10-05):
[1] http://www.postgresql.org/docs/8.0/interactive/index.html.

Ramakrishnan, Raghu & Gehrke, Johannes (2001): *Database Management Systems*, second edition, McGraw- Hill, city London-New York, 906 pages.

Rigaux, Philippe; Scholl, Michel; and Voisard Agnès (2002): *Spatial databases – with application to GIS*, Morgan Kaufmann publishers, San Francisco, 410 pages.

Rubin, Steven (1995): *Computer Aids for VLSI Design*, URL (10-05):
[1] http://www.rulabinsky.com/cavd/text/chap03-6.html.

Shekhar, Shashi & Chawla, Sanjay (2003): *Spatial Databases: A Tour*, Pearson Education Inc., New Jersey, 262 pages.

Sigurbjarnarson, Tryggvi (2000): *Landskrá fasteigna – Landeignaskrá. Tilraunaverkefni á Suðurlandi 1999-2000* (e. Land registration database – Cadastral database. Project implemented in South-Iceland 1999-2000). Notes from conference on municipality finances 2000, held in Hotel Saga in Reykjavík, 2 & 3 November 2000, URL (10-05):
[1] http://samband.skyrr.is/fundir/fjarmal00/tryggvi_sigurbjarnarson.shtml.

Skipulag (2005): *Planning and building act*, no.73/1997, English translation, URL (11-05):
[1]http://www.skipulag.is/focal/webguard.nsf/key2/planning_and_buildingact.html

Steinþórsson, Sigurður (2001): *Vísindavefurinn* (e. Science web), URL (11-05):
[1] http://visindavefur.hi.is/svar.asp?id=1775.

Stoter, Jantien E. (2004): *3D Cadastre*, The Technical University in Delft, 327 pages.

Strонček, Vladimir (2005): Unpublished material and communications September-November 2005.

Stubkjær, Erik (2002): *Modelling real property transactions*, FIG XXII International Congress, Washington, D.C. USA, April 19-26 2002, URL (10-05):
[1] http://www.fig.net/pub/fig_2002/Js14/JS14_stubkjaer.pdf.

Stubkjær, Erik (2003): *Modelling units of real property rights*, ScanGIS'2003, Espoo, Finland. June 4-6, URL (10-05): http://www.scangis.org/scangis2003/papers/30.pdf.

Tuladhar, Arbind Man; Radwan, Mostafa; Kader, Abdel & El-Ruby, Samir (2005): *Federated Data Model to Improve Accessibility of Distributed Cadastral Databases in Land*

Administration, FIG Working Week 2005 and GSDI-8, Cairo, Egypt April 16-21, 2005.

UDIG (2005): *User-friendly desktop internet GIS*, Refraction Research, URL: (11-05): [1] http://udig.refractions.net/confluence/display/UDIG/Home.

USGS (1999): *Understanding plate motions*, U.S. Geological Survey, URL (11-05): [1] http://pubs.usgs.gov/publications/text/understanding.html.

UST (2005): *Environment and food agency*, URL (10-05):
[1] www.ust; see also
[2] http://www.ust.is/Natturuvernd/landupplysingar/ for info on preserved areas.

Vries, Marian & Oosterom, Peter (2005): *The third GML relay*, in 'Geo-information Standards in Action, NCG / GIN farewell seminar Henri J.G.L. Aalders, Delft, 17 November 2004', edited by Peter J.M. Oosterom, Delft University of Technology, pages 31-40.

Wikipedia (2005): *Wikipedia, the free encyclopedia*, URL (10-5):
[1] http://en.wikipedia.org/wiki/;
[2] http://en.wikipedia.org/wiki/Gpl;
[3] http://en.wikipedia.org/wiki/LGPL;
[4] http://en.wikipedia.org/wiki/Open_source_license;
[5] http://en.wikipedia.org/wiki/Open-source_software.

Williamson, Ian (1997): *The Bogor Declaration for Cadastral Reform*, report presented to the 6th United Nations Regional Cartographic Conference for the Americas, New York, 2-6 June, 1997.

Williamson, Ian (2001): *The evolution of modern Cadastres*, in honour of Professor dr.ir. The Bogaerts, October, 2000 to February 2001.

Worboys, Michael F. (1995): *GIS: A Computing Perspective*, Taylor & Francis, London, pages 376.

Zevenbergen, Jaap (2002): *Systems of Land Registration - Aspects and Effects*, Phd research published in November 2002, TU Delft, 210 pages.

Zlatanova, Sisi; Stoter, Jantien & Quak, Wilko (2004): *Management of multiple representations in spatial DBMSs*, AGILE 2004, Heraklion, April 28 - Mai 1, pages 269-278, URL (10-05): [1] http://www.gdmc.nl/publications/2004/Management_of_representations.pdf.

APPENDIX A: SPATIAL DATA TYPES

This appendix gives overview of the data types defined in OGC Simple Feature Specification (1999). The methods described are most quite straightforward except those that look like Boolean statement but return Integer. In these cases the method returns 1 if it is true, 0 if false.

Table 5: Overview of geometry data types and methods as defined in the OGC Simple Feature Specification.

Class name	Description	Methods
Geometry	Abstract class that is the root of the geometric hierarchy.	-Dimension(): Integer -GeometryType(): String -SRID(): Integer -Envelope(): Geometry -AsText(): String -AsBinary(): Binary -IsEmpty(): Integer -IsSimple(): Integer -Boundary(): Geometry
SpatialReferenceSystem	Class that defines the spatial reference system available for geometries to be defined in.	
GeometryCollection	Collection of one or more Geometry. All geometries have to be in the same spatial reference.	-NumGeometries(): Integer -GeometryN(N:Integer): Geometry
Point	A 0-dimensional geometry representing a single location in space.	-x(): Double -y(): Double
MultiPoint	Collection of 0-dimensional geometries. Restricted to points.	
Curve	A 1-dimensional geometry usually considered as sequence of two or more points. Curve has only one subclass LineString according to this specification. Another candidate could e.g. be Arc.	-Length(): Double -StartPoint(): Point -EndPoint(): Point -IsClosed(): Integer -IsRing(): Integer
LineString	A Curve with linear interpolation between points. See also Figure 47.	-NumPoints(): Integer -PointN(N:Integer): Point
Line	A LineString with exactly two points	

LinearRing	A LineString that is both closed and simple.	
MultiCurve	1-dimensional GeometryCollection where all instances are of the type Curve.	-IsClosed(): Integer -Length(): Double
MultiLineString	A MultiCurve where all elements are of type LineString. See Figure 48.	
Surface	2-Dimensional geometric object. Defined by set of curves corresponding to exterior and 0-many interior boundaries.	-Area(): Double -Centroid(): Point -PointOnSurface(): Point
Polygon	Planar representation of Surface. See Figure 49.	-ExteriorRing(): LineString -NumInteriorRing(): Integer -InteriorRingN(N:Integer): LineString
MultiSurface	An abstract class and geometric collection of elements of type Surface, with the constrain that the interiors of any two Surface cannot intersect.	-Area(): Double -Centroid(): Point -PointOnSurface(): Point
MultiPolygon	Subclass of MultiSurface containing one or more Polygon. See Figure 50.	

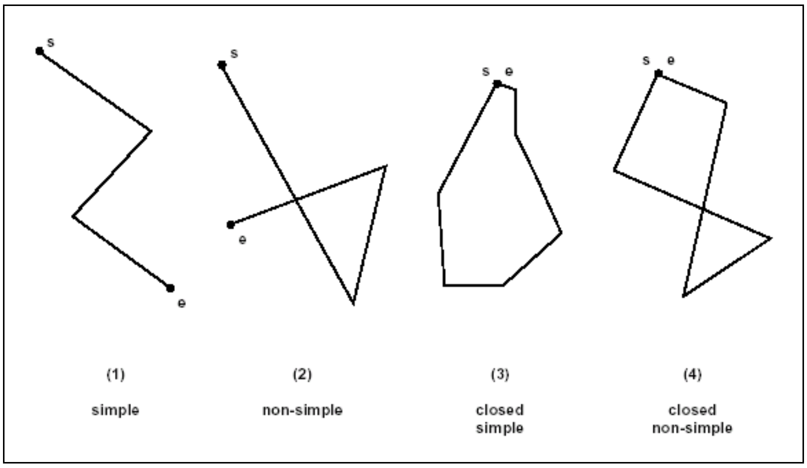


Figure 47: Different types of LineStrings (OGC 1999).

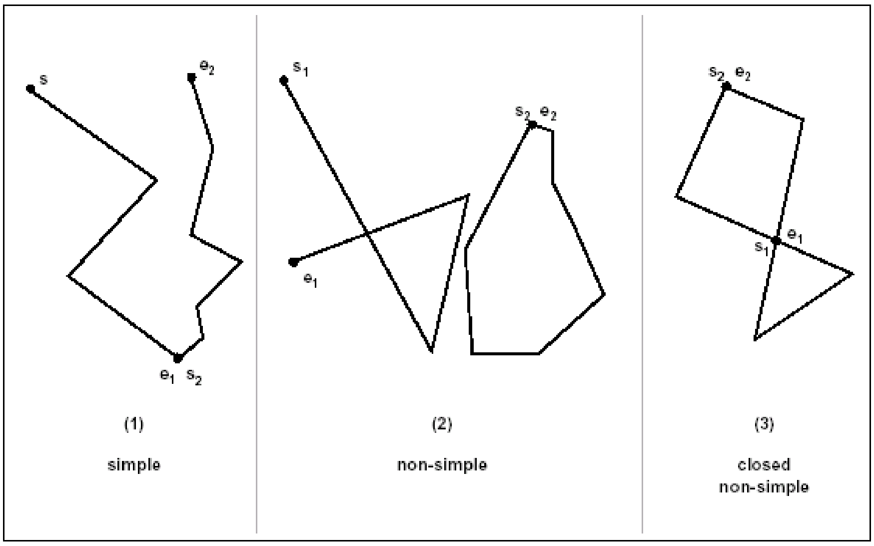


Figure 48: MultiLineString (OGC 1999).

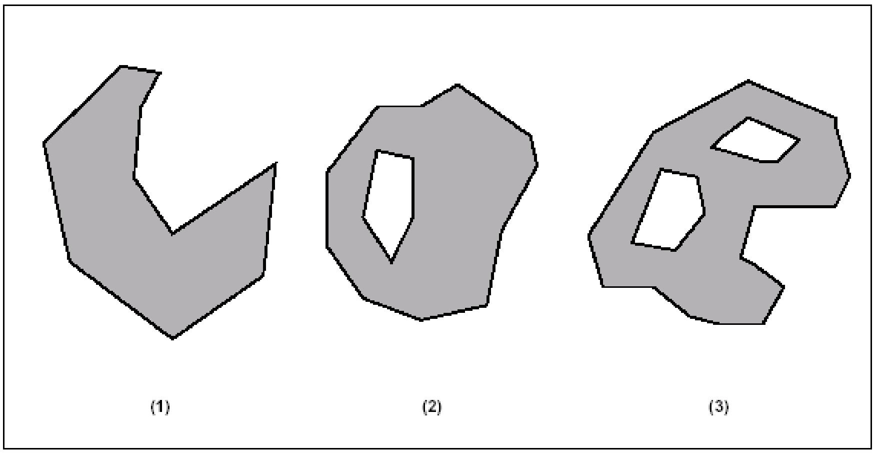


Figure 49: Examples of Polygon (OGC 1999).

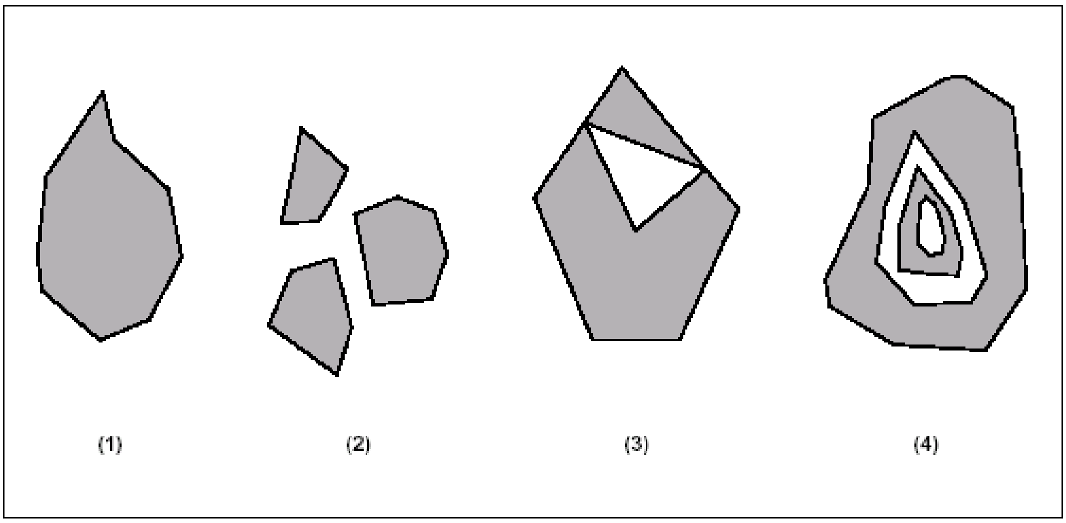


Figure 50: Examples of MultiPolygon (OGC 1999).

APPENDIX B: NEW VERSION OF THE CCDM

The diagrams in this appendix represent the newest version of the CCDM as will be published in the scientific journal CEUS (Computers Environment and Urban Systems), probably in 2006, Volume 30.

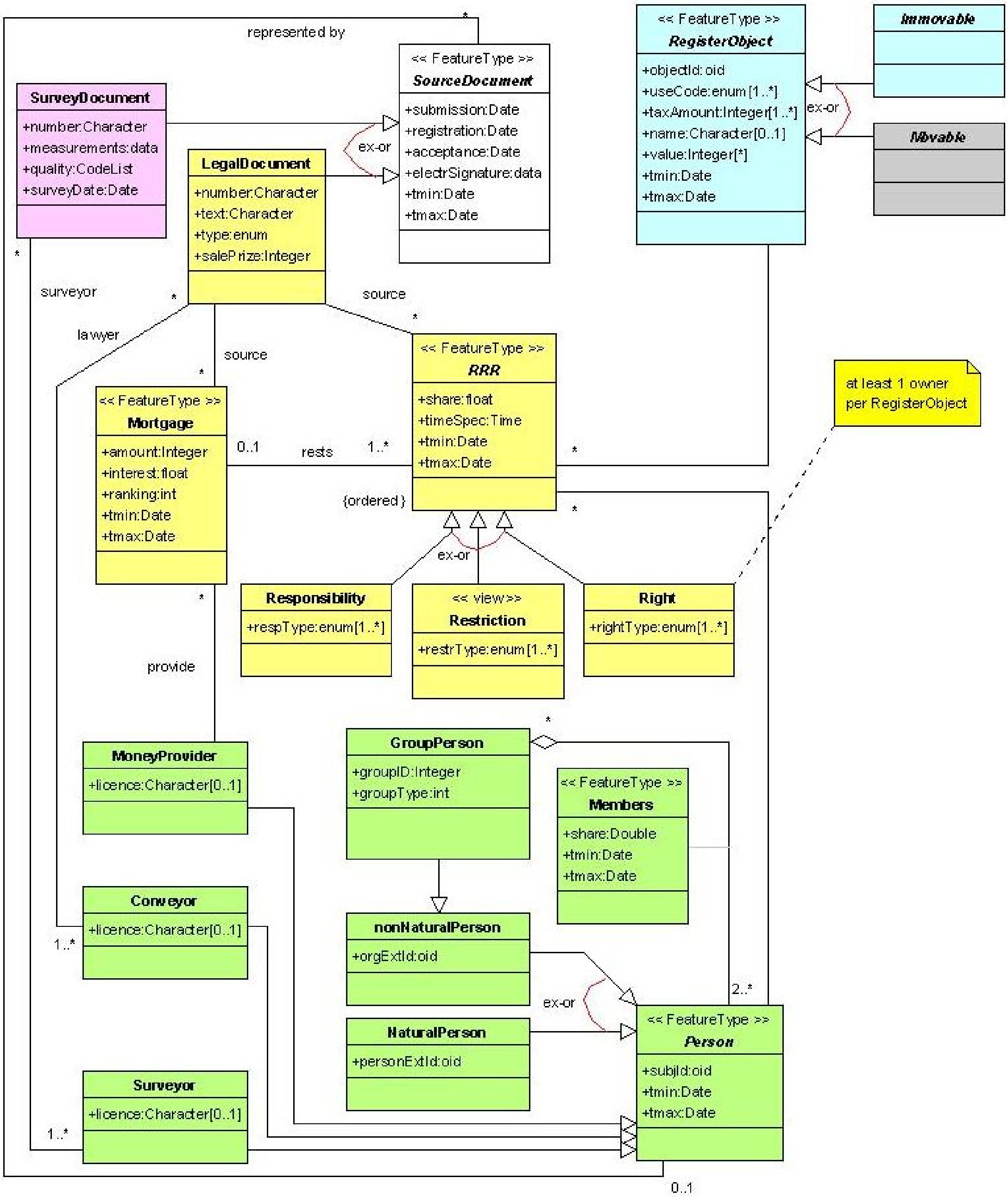


Figure 51: The legal / administrative part in the newest version of the CCDM (Oosterom et al 2005).

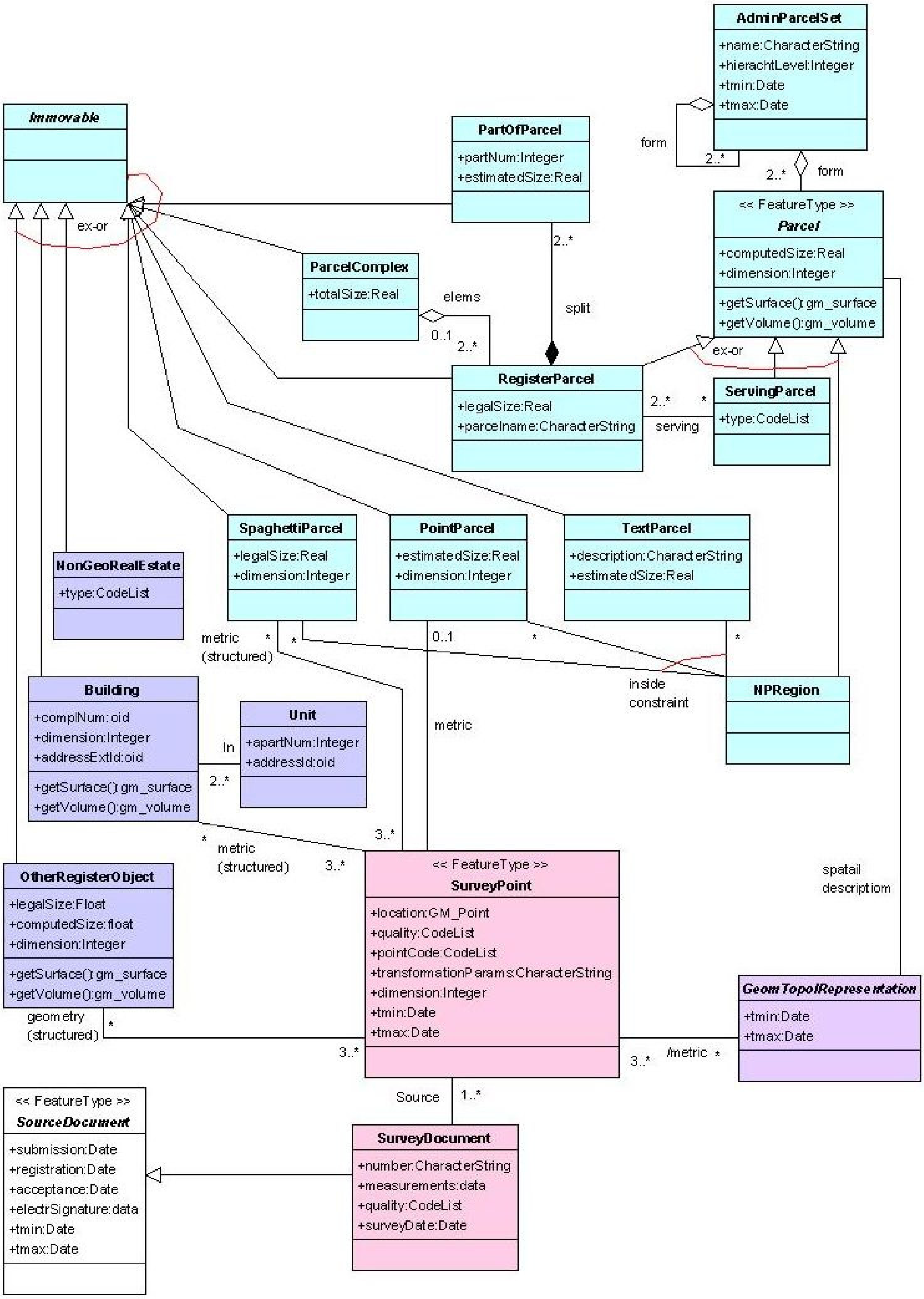


Figure 52: The geographic part in the newest version of the CCDM (Oosterom et al 2005).

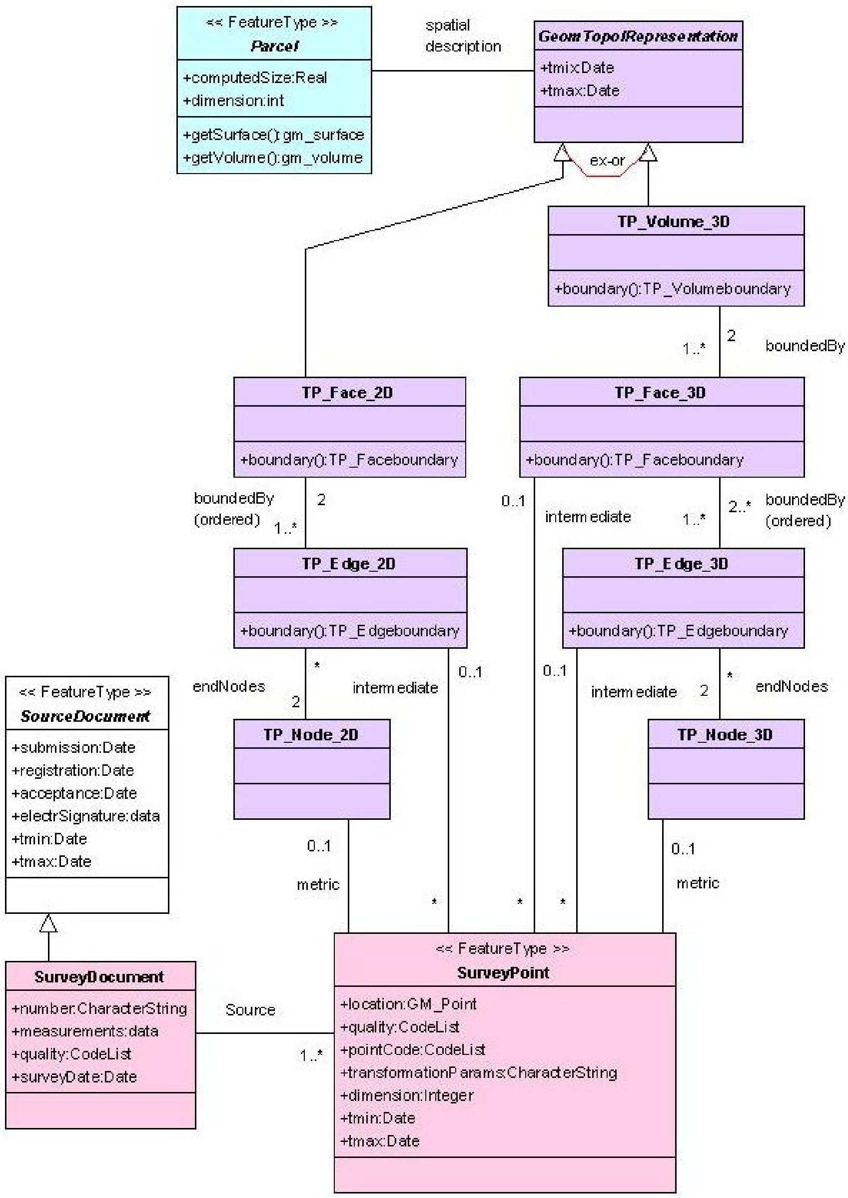


Figure 53: The topological part in the newest version of the CCDM (Oosterom et al 2005).

APPENDIX C: EXAMPLE OF A RECORD IN LRD

Table 6: Property information estracted from the web-based landregistry for the farm Hraun in Fljót, Skagafjörður, North-Iceland. In the beginning of 18th century this farm was regarded as the most valuable farm in Iceland because of natural resources.

Hraun 145889 (Skagafjörður)

Unit_ID	Indicator	Usage	Building year	Size	Value (Ex. Land)	Value (Land)	Value (Properties)	Reconstruction Value
213-9557	00	Farm land		0,0 (?)	ISK 0	ISK 167.000	ISK 167.000	ISK 0
	01	Cultivated land		28,7ha	ISK 1.421.000	ISK 0	ISK 1.421.000	ISK 0
	03	Eiderdown	Natural Resource	0,0	ISK 4.991.000	ISK 0	ISK 4.991.000	ISK 0
	27	Driftwood	Natural Resource	0,0	ISK 189.000	ISK 0	ISK 189.000	ISK 0
213-9559	02	Salmon/Trout	Natural Resource	0,0	ISK 86.000	ISK 0	ISK 86.000	ISK 0
213-9561	04 0101	Apartment	1927	153,3m²	ISK 2.251.000	ISK 0	ISK 2.251.000	ISK 13.894.000
	26 0101	Single house	1988	29,9m²	ISK 737.000	ISK 0	ISK 737.000	ISK 4.394.000
213-9562	06 0101	Cowshed	1974	45,0m²	ISK 209.000	ISK 0	ISK 209.000	ISK 1.140.000
	10 0101	Barn	1964	45,8m²	ISK 125.000	ISK 0	ISK 125.000	ISK 721.000
	14 0101	Storeroom	1908	43,0m²	ISK 136.000	ISK 0	ISK 136.000	ISK 880.000
	16 0101	Shed	1972	18,1m²	ISK 70.000	ISK 0	ISK 70.000	ISK 299.000
	20 0101	Tool room	1976	118,1m²	ISK 471.000	ISK 0	ISK 471.000	ISK 1.995.000
213-9570	21 0101	Pit	1980	112,0m²	ISK 1.592.000	ISK 0	ISK 1.592.000	ISK 5.686.000
	28	Tool room	1993	51,6m²	ISK 465.000	ISK 0	ISK 465.000	ISK 0
	29	Sheepcote	1996	372,7m²	ISK 3.618.000	ISK 0	ISK 3.618.000	ISK 12.591.000
213-9571	22 0101	Apartment	1983	169,2m²	ISK 3.747.000	ISK 0	ISK 3.747.000	ISK 23.798.000
213-9572	23 0101	Stable	1986	73,3m²	ISK 528.000	ISK 0	ISK 528.000	ISK 1.869.000
213-9573	24 0101	Storeroom	1985	35,0m²	ISK 175.000	ISK 0	ISK 175.000	ISK 622.000
213-9574	25 0101	Tool room	1977	60,0m²	ISK 140.000	ISK 0	ISK 140.000	ISK 591.000
226-8048	30 0101	Hunting lodge Skammagil	2001	13,0m²	ISK 460.000	ISK 0	ISK 460.000	ISK 1.295.000
	31 0101	Hunting lodge Aravatn	1995	17,6m²	ISK 651.000	ISK 0	ISK 651.000	ISK 1.845.000

Land_ID	Usage	Size	Locator
145889	Farm	0,0	5200-04-00024000

Figure 54: The farm Hraun in Fljót area, Skagafjörður, North-Iceland (Image in courtesy of www.islandsvefurinn.is).

APPENDIX D: EXAMPLE OF BOUNDARY DECLARATION

FYLGISKJAL NR.1

Hnitaskrá vegna þinglýsingar landamerkja

Minni Grindils

Nr.	X-Hnit	Y-Hnit
1-	496844,8	618356,2
2-	496724,1	618044,8
3-	496707,7	617690,0
4-	496591,0	617701,1
5-	496530,9	617545,9
6-	496608,7	617440,3
7-	496792,9	617274,3
8-	496798,5	617265,9
9-	496818,7	617258,5
10-	496843,5	617232,0
11-	496848,3	617231,6
12-	496867,5	617213,6
13-	496878,6	617213,2
14-	496895,8	617208,0
15-	497008,7	617336,7
16-	497237,6	617219,2
17-	497236,7	617226,0
18-	497244,3	617242,6
19-	497241,4	617251,3
20-	497246,9	617255,5
21-	497251,0	617269,7
22-	497242,9	617272,1
23-	497251,2	617293,9
24-	497245,5	617311,5
25-	497260,0	617339,4
26-	497268,1	617354,4
27-	497281,4	617358,7
28-	497287,8	617377,3
29-	497285,7	617392,4
30-	497300,6	617420,9
31-	497306,2	617422,7
32-	497315,6	617482,5
33-	497306,7	617503,3
34-	497316,8	617515,5
35-	497322,4	617538,4
36-	497316,1	617553,5
37-	497337,3	617554,3
38-	497331,1	617572,7
39-	497341,5	617572,7
40-	497340,0	617580,9
41-	497369,4	617629,0
42-	497380,6	617631,6
43-	497384,4	617642,9
44-	497405,1	617646,4
45-	497407,5	617657,6
46-	497454,1	617693,2
47-	497478,7	617723,6
48-	497473,1	617736,5
49-	497480,4	617767,1
50-	497503,2	617772,3
51-	497530,4	617796,9
52-	497524,8	617823,2
53-	497511,1	617823,2
54-	497510,3	617838,9
55-	497497,2	617872,8
56-	497515,8	617916,3

Figure 55: Example of a boundary declaration for rural land parcel (Image source: Author).

APPENDIX E: GML OUTPUT FROM GEOSERVER

The code presented in this appendix is example of GML 2.1.2 output, when querying GeoServer 1.3RC3 WFS. The data used was set of parcels from Skilimannahreppur, southwest Iceland collected by LRI. Stored in a topological way, using nodes, edges and faces the surface polygon of the parcel was realised as described in prototype. Only one featuremember is shown of 81.


```
<wfs:FeatureCollection xsi:schemaLocation="http://81.69.54.149/cgi-
bin/mapserv.exe?map=C:\temp\test.map?
http://localhost:8080/geoserver/wfs/DescribeFeatureType?typeName=cad:cadastre
http://www.opengis.net/wfs
http://localhost:8080/geoserver/data/capabilities/wfs/1.0.0/WFS-basic.xsd">
   <gml:boundedBy>
      <gml:Box srsName="http://www.opengis.net/gml/srs/epsg.xml#3057">
         <gml:coordinates decimal="." cs="," ts=" ">354400.5,426745.15625
         366282.40625,436697.03125</gml:coordinates>
      </gml:Box>
   </gml:boundedBy>
…

   <gml:featureMember>
      <gml:cadastre fid="cadastre.110546">
         <gml:land_id>76</gml:land_id>
         <gml:parcel_name>Galtarvík</gml:parcel_name>
         <gml:owner>Hörður Jónsson</gml:owner>
         <gml:geom>
            <gml:Polygon srsName="http://www.opengis.net/gml/srs/epsg.xml#3057">
               <gml:outerBoundaryIs>
                  <gml:LinearRing>
                     <gml:coordinates decimal="." cs="," ts=" ">
                        362395.18783573, 430013.18758455 362670.39083769,
                        430254.00058584 362945.59383965, 430494.81358714
                        363031.593837, 430563.59458656 364477.62483947,
                        429361.99958527 364467.90583403, 429344.87458779
                        364450.31283674,429333.12558704 364411.21783616,
                        429325.31158606 364366.25083665, 429317.49958748
                        364340.84383838, 429311.62558615 364308.42383591,
                        429137.68758797 364125.84383566, 429114.21858695
                        364094.56283606, 429082.93758735 364061.34383921,
                        429069.28158599 364030.06283961, 429049.71858427
                        364020.28183467,429020.40558671 363992.93783723,
                        428989.12458711 363953.84383473, 428983.28258672
                        363916.68783572, 428989.12458711 363885.43683467,
                        429000.8435893 363832.6548361,429030.18658541 363809.1868394,
                        429020.40558671 363793.562836, 429002.81258942
                        363707.56283865, 428910.93858451 363686.06283776,
                        428871.84358393 363668.46883615, 428840.59358719
                        363658.93683974, 428817.74958399 362395.18783573,
                        430013.18758455
                     </gml:coordinates>
                  </gml:LinearRing>
               </gml:outerBoundaryIs>
               <gml:innerBoundaryIs>
                  <gml:LinearRing>
                     <gml:coordinates decimal="." cs="," ts=" ">
                        363613.35083714, 429170.95658864 363631.74083888,
                        429145.21758523 363672.42383494, 429174.28458721
                        363643.35783728, 429214.96758952 363621.56683987,
                        429199.39958465 363602.6738369, 429185.90058753
                        363609.62283921, 429176.17458929 363613.35083714,
                        429170.95658864
                     </gml:coordinates>
                  </gml:LinearRing>
               </gml:innerBoundaryIs>
            </gml:Polygon>
         </gml:geom>
      </gml:cadastre>
   </gml:featureMember>
…
</wfs:FeatureCollection>
```

APPENDIX F: MATERIALISED VIEWS IN POSTGRESQL

The code presented in this appendix illustrates how materialised views can be implemented in PostgreSQL. The implementation is based on the innovation from Gardner (2004).

To start with is to create table to manage the materialised views in the database. The table contains the name of the materialised view, corresponding view and the timestamp of last update.

```
CREATE TABLE materialised_views (
  mv_name NAME NOT NULL PRIMARY KEY,
  v_name NAME NOT NULL,
  last_refresh TIMESTAMP
);
```

Using pl/pgsql the PostgreDQL own procedure programming three functions are defined allowing user to create, terminate or refresh materialised views.

First one is referred as create_matview(). Two variables have to be defined. The name of the view and the proposed corresponding materialised view. The function checks if the materialised view with the same name already exists before creating one.

```
CREATE OR REPLACE FUNCTION create_matview(NAME, NAME)
RETURNS VOID AS
$$
DECLARE
    matview ALIAS FOR $1;
    view_name ALIAS FOR $2;
    entry materialised_views%ROWTYPE;
BEGIN
    SELECT * INTO entry FROM materialised_views WHERE mv_name = matview;
    IF FOUND THEN
        RAISE EXCEPTION ''Materialised view ''''%'''' already exists.'',
          matview;
    END IF;
    EXECUTE ''REVOKE ALL ON '' || view_name || '' FROM PUBLIC'';
    EXECUTE ''GRANT SELECT ON '' || view_name || '' TO PUBLIC'';
    EXECUTE ''CREATE TABLE '' || matview || '' AS SELECT * FROM '' || view_name;
    EXECUTE ''REVOKE ALL ON '' || matview || '' FROM PUBLIC'';
    EXECUTE ''GRANT SELECT ON '' || matview || '' TO PUBLIC'';

    INSERT INTO materialised_views (mv_name, v_name, last_refresh)
      VALUES (matview, view_name, CURRENT_TIMESTAMP);
    RETURN;
END;
$$ LANGUAGE plpgsql;
```

It is important not only to be able to create but also to terminate the materialised views not longer wanted or when the schema of relevant view is altered. The function drops the materialised view table before deleting its entry in the materialised_views table.

```
CREATE OR REPLACE FUNCTION drop_matview(NAME) RETURNS VOID
SECURITY DEFINER AS
$$
DECLARE
    matview ALIAS FOR $1;
    entry materialised_views%ROWTYPE;
BEGIN
    SELECT * INTO entry FROM materialised_views WHERE mv_name = matview;
    IF NOT FOUND THEN
        RAISE EXCEPTION ''Materialised view % does not exist.'', matview;
    END IF;
    EXECUTE ''DROP TABLE '' || matview;
    DELETE FROM materialised_views WHERE mv_name=matview;
```

```
    RETURN;
END;
$$ LANGUAGE plpgsql;
```

Finally is the refresh function, which is quite primitive defined here, as it deletes everything from the materialised view before populating it again with a select statement on relevant view. At the end it updates the last_refresh column in the materialised_views table.

```
CREATE OR REPLACE FUNCTION refresh_matview(name) RETURNS VOID
SECURITY DEFINER
LANGUAGE plpgsql AS '
DECLARE
    matview ALIAS FOR $1;
    entry materialised_views%ROWTYPE;
BEGIN
    SELECT * INTO entry FROM materialised_views WHERE mv_name = matview;
    IF NOT FOUND THEN
        RAISE EXCEPTION ''Materialized view % does not exist.'', matview;
    END IF;
    EXECUTE ''DELETE FROM '' || matview;
    EXECUTE ''INSERT INTO '' || matview
        || '' SELECT * FROM '' || entry.v_name;
    UPDATE materialised_views
        SET last_refresh=CURRENT_TIMESTAMP
        WHERE mv_name=matview;
    RETURN;
END';
```

This topic is in desperate need for more researches, especially how it is possible to update materialised views as a part of a transaction. An alternative would be to offer timestamp to be default (by definition) stored per row indicating last update. Employing this could make it easy to define trigger in the end of a transaction that would only update in the materialised_view table the most recent changes in the database.

APPENDIX G: IMPLEMENTATION OF PROTOTYPE

The objective here was to design and implement a SDBMS capable of maintaining a cadastral data in accordance to the topological model and the Icelandic version of the CCDM, ICM. The data used in this experiment was of Skilimannahreppur municipality and provided by LRI, but of course all data can be used with only minor modification of the SQL code presented below. It only demonstrates how to create and populate cadastral database in steps, using clean data as input and node-edge-face supporting topology. Words written with Italian letters explain what is happening in every step.

It is important when creating database that stores text in Icelandic to use LATIN1 encoding as it supports Icelandic letters like "á,ð,é,í,ó,ú,ý,æ,þ,ö".

```
1.  C:\createdb -E LATIN1 cad
2.  C:\psql -d cad -f C:\Program Files\PostgreSQL\8.0\share\contrib\lwpostgis.sql
3.  C:\psql -d cad -f C:\Program
    Files\PostgreSQL\8.0\share\contrib\spatial_ref_sys.sql
```

Two shapefiles are converted to sql statement using shp2pgsql program, provided with PostGIS.

```
4.  C:\Data\SHP\fmr>shp2pgsql -s 3057 -g geom Land temp_parcels > temp_parcels.sql
5.  C:\Data\SHP\fmr>shp2pgsql -s 3057 -g geom Land_PolygonToLine temp_boundaries >
    temp_boundaries.sql
```

Next is to import the data into the cad database by using psql uploading.

```
6.  C:\Data\SHP\fmr>psql -d cad -f temp_boundaries.sql
7.  C:\Data\SHP\fmr>psql -d cad -f temp_parcels.sql
```

Enter the database

```
8.  C:\psql cad
9.  cad=#
```

Create extra columns in the temp_boundaries table

```
10. SELECT addGeometryColumn('temp_boundaries','n1',3057,'POINT',2);
11. SELECT addGeometryColumn('temp_boundaries','n2',3057,'POINT',2);
12. SELECT addGeometryColumn('temp_boundaries','body',3057,'LINESTRING',2);
13. UPDATE temp_boundaries SET n1=startpoint(geom);
14. UPDATE temp_boundaries SET n2=endpoint(geom);
```

This function strips a line of its endpoints, but only valid for lines with 4 or more points.

```
15. CREATE OR REPLACE FUNCTION getBody1(geometry) --input LINESTRING
    RETURNS geometry AS $$
      SELECT difference(
        (difference($1,(makeLine(pointN($1,(numPoints($1)-1)),endpoint($1)))))
        , (makeLine(startPoint($1),pointN($1,2)))
      );
    $$ LANGUAGE SQL;
```

This function captures lines with exactly 3 points and interpolates 2 points halfway between the middle point and the two endpoints. Then strips the line of its endpoints.

```
16. CREATE OR REPLACE FUNCTION getbody2(geometry) RETURNS geometry AS $$
    SELECT
      addPoint(
        makeLine(
          line_interpolate_point(makeline(pointN($1,1),pointN($1,2)),0.5)
          ,pointN($1,2))
        ,line_interpolate_point(makeline(pointN($1,2),pointN($1,3)),0.5),-1);
    $$ LANGUAGE sql;
```

My first plpgsql code that I use in the cadastre. Its purpose is to control which function, getBody1 or getBody2 are executed if executed at all, when extracting intermediate points from boundary geometry.

```
17. CREATE OR REPLACE FUNCTION getBody(geom geometry)
    RETURNS geometry AS –Uses plpgsql to categorise function depending on num
    points
    $$BEGIN
    IF numPoints(geom)>3 THEN
       RETURN getBody1(geom);
    ELSE
```

```
            IF numPoints(geom)=3 THEN
               RETURN getBody2(geom);
            ELSE
                IF numPoints(geom)<3 THEN
                   RETURN null;
                END IF;
            END IF;
        END IF;
        END;
        $$ LANGUAGE plpgsql;
    18. UPDATE temp_boundaries SET body=getBody(geom);
```

Create and populate the monuments table with unique end nodes as monuments.

```
19. CREATE TABLE monuments (id serial PRIMARY KEY, name varchar(50), description
    text, quality float4);
20. SELECT addGeometryColumn('monuments','geom',3057,'POINT',2);
21. CREATE INDEX monuments_oid ON monuments(oid);
22. CREATE INDEX monuments_idx ON monuments USING GIST (geom GIST_GEOMETRY_OPS);
23. INSERT INTO monuments(geom) SELECT n1 FROM temp_boundaries UNION SELECT n2
    FROM temp_boundaries;
```

Set few of the attributes to random value for the monuments table

```
24. UPDATE monuments SET quality=random();
25. UPDATE monuments SET name='unknown';
```

Create and populate lines table that contains lines in the database.

```
26. CREATE TABLE lines (id serial PRIMARY KEY,quality float4, description text);
27. SELECT AddGeometryColumn('lines','geom',3057,'LINESTRING',2);
28. CREATE INDEX lines_oid ON lines(oid);
29. CREATE INDEX lines_idx ON lines USING GIST (geom GIST_GEOMETRY_OPS);
30. INSERT INTO lines(geom) SELECT body FROM tb WHERE numPoints(body) > 1;;
31. INSERT INTO lines(id,geom) VALUES (-1, NULL);
32. UPDATE lines SET quality=random() * 2 WHERE id <> -1;
```

Here I am doing little sidestep to easy my work. I create extra columns in the temporary boundary table where I assign each boundary and end node to corresponding value in monuments or lines tables. Furthermore I assign -1 value to boundaries without a body.

```
33. ALTER TABLE temp_boundaries ADD COLUMN n1_id int4;
34. ALTER TABLE temp_boundaries ADD COLUMN n2_id int4;
35. ALTER TABLE temp_boundaries ADD COLUMN line_id int4;
36. UPDATE temp_boundaries SET n1_id = monuments.id WHERE
    equals(temp_boundaries.n1,monuments.geom);
37. UPDATE temp_boundaries SET n2_id = monuments.id WHERE
    equals(temp_boundaries.n2,monuments.geom);
38. UPDATE temp_boundaries SET line_id = lines.id WHERE
    equals(temp_boundaries.body,lines.geom);
39. UPDATE temp_boundaries SET line_id = -1 WHERE body IS NULL;
```

Create and populate parcel table with temp_parcel. Attention has to be on that the ids of parcel table reflect those used in right/left relationships of boundaries. Here 1 has to be subtracted from every gid as shp2pgsql does not allow an identification to have a 0 value so it automatically updates it. 1 has to be added to the id's so they reflect those in the left/right relationship table.

```
40. CREATE TABLE parcels (land_id serial PRIMARY KEY, parcel_name
    varchar(50),owner varchar(50));
41. INSERT INTO parcels (land_id,parcel_name,owner) SELECT gid,heita,eigandi_fa
    FROM temp_parcels;
42. INSERT INTO parcels VALUES (-1,'outside','government');
```

Create the 'boundaries' table that maintains the relationship between nodes, lines and parcels.

```
43. CREATE TABLE boundaries (
      id serial PRIMARY KEY,
      start_node int4 REFERENCES monuments(id),
      end_node int4 REFERENCES monuments(id),
      body_line int4 REFERENCES lines(id),
      left_parcel int4 REFERENCES parcels(land_id),
      right_parcel int4 REFERENCES parcels(land_id)
      );
```

Next step is to create indexes for the variables in the boundary table. No index is needed to create for the lines and the monuments id's as they are automatically indexes as a Primary Key.

```
44. CREATE INDEX boundaries_start_node_idx ON boundaries(start_node);
45. CREATE INDEX boundaries_end_node_idx ON boundaries(end_node);
```

```
46. CREATE INDEX boundaries_body_line_idx ON boundaries(body_line);
47. CREATE INDEX boundaries_left_parcel_idx ON boundaries(left_parcel);
48. CREATE INDEX boundaries_right_parcel_idx ON boundaries(right_parcel);
```

To populate the boundaries table a query has to be executed that compares the values in temp_boundaries with those in monuments and lines.

```
49. INSERT INTO boundaries(
      start_node,
      end_node,
      body_line,
      left_parcel,
      right_parcel)
    SELECT n1_id,
      n2_id,
      line_id,
      left_fid,
      right_fid
    FROM  temp_boundaries
```

Next step is to create functions that are able to extract geometry from the boundary table by using appropriate join to table lines and monuments. The problem here is that not all boundaries have a body so there has to be a IF statement to govern which function is used at time.

```
50. CREATE OR REPLACE FUNCTION getBoundaries2(geometry, geometry)
      RETURNS geometry AS $$
      SELECT makeLine($1,$2);
      $$ LANGUAGE SQL;

51. CREATE OR REPLACE FUNCTION getBoundaries3(geometry, geometry, geometry)
      RETURNS geometry AS $$
      SELECT addPoint((addPoint($3,$1,0)),$2,-1);
      $$ LANGUAGE SQL;

52. CREATE OR REPLACE FUNCTION getBoundaries(n1 geometry, n2 geometry, body
    geometry)
      RETURNS geometry AS $$
    BEGIN
    IF numPoints(body)>0 THEN
        RETURN getBoundaries3(n1,n2,body);
    ELSE
        RETURN getBoundaries2(n1,n2);
    END IF;
    END;
    $$ LANGUAGE plpgsql;
```

Below is the first view of two created.

```
53. CREATE OR REPLACE VIEW boundaries_view AS
    SELECT
     b.oid,
     b.id,
     getboundaries(m1.geom, m2.geom, l.geom) AS boundary,
     b.start_node,
     b.end_node,
     b.left_parcel,
     b.right_parcel,
     round(length(getboundaries(m1.geom, m2.geom, l.geom))) AS length
    FROM
     boundaries b,
     monuments m1,
     monuments m2,
     lines l
    WHERE
     b.start_node = m1.id AND
     b.end_node = m2.id AND
     b.body_line = l.id;
```

To be able to generate valid polygons I need a function that returns the largest polygon from the polygonize function which only delivers geometry as GEOMETRYCOLLECTION. Of course it would be more effective to edit the polygonize method but for that I need a copy of the source code and advanced knowledge of programming in C.

```
54. CREATE OR REPLACE FUNCTION extractPolygon (geom geometry)
    RETURNS geometry AS $$
    DECLARE
      l1 float4 = 0;
```

```
    l2 float4;
    c1 int4 = 0;
    c2 int4 = numgeometries(geom);
    n int4;
  BEGIN
     WHILE c1 <= c2 LOOP
      c1 = c1 + 1;
      l2 = area(MakePolygon(ExteriorRing(geometryN(geom,c1))));
      IF l2 > l1 THEN
        l1 = l2;
        n = c1;
      END IF;
    END LOOP;
    RETURN geometryN(geom,n);
  END;
  $$ LANGUAGE plpgsql;
```

By employing the function above the parcel polygons can be realised without problem. One thing to consider is the speed of the realisation. However this does not trouble the dataset used in the experiment.

55.
```
CREATE OR REPLACE VIEW cadastre AS
SELECT
 p.oid,
 p.land_id,
 p.parcel_name,
 p.owner,
 extractPolygon(polygonize(bv.boundary)) AS geom
FROM
 parcels p,
 boundaries_view bv
WHERE
  p.land_id <> -1 AND
 (bv.left_parcel = p.land_id OR
 bv.right_parcel = p.land_id)
GROUP BY
 p.oid,
 p.land_id,
 p.parcel_name,
 p.owner;
```

The database is complete.

To be able to use the data in some application, like e.g. GeoServer the geometry column of VIEW has to be explicitly added to the geometry_columns table in the PostGIS database. Thus to add the cadastre view:

```
INSERT INTO geometry_columns VALUES
('','public','cadastre','geom',2,3057,'POLYGON');
```

Still uDig has problem to read this but the output from GeoServer seems correct and can be checked by getCapabilities, and if correct structured by the WFS getFeature request:

http://localhost:8080/geoserver/wfs?request=GetCapabilities
http://localhost:8080/geoserver/wfs?request=GetFeature&typename=cad:cadastre

APPENDIX H: TABLE DEFINITIONS IN PROTOTYPE

The table definitions presented below shows the exact table definition of the prototype as was implemented in the second attempt (see sub-chapter 7.2).

Table "public.monuments"

```
     Column     |         Type          | Modifiers
----------------+-----------------------+-----------
 mon_id         | integer               | not null
 name           | character varying(50) |
 description    | text                  |
 tmin           | date                  |
 tmax           | date                  |
 quality        | integer               |
 geom           | geometry              |
 transaction_id | integer               |
Indexes:
    "monuments_pkey" PRIMARY KEY, btree (m_id)
    "monuments_geom_btree" btree (geom)
    "monuments_idx" gist (geom)
    "monuments_oid" btree (oid)
    "monuments_t_id_idx" btree (transaction_id)
Check constraints:
    "enforce_srid_geom" CHECK (srid(geom) = 3057)
    "enforce_geotype_geom" CHECK (geometrytype(geom) = 'POINT'::text OR geom IS NULL)
    "enforce_dims_geom" CHECK (ndims(geom) = 2)
Triggers:
    updatecadastremv AFTER INSERT OR DELETE OR UPDATE ON monuments FOR EACH STATEMENT
EXECUTE PROCEDURE updatecadastreview()
```

Table "public.boundaries"

```
    Column    |   Type   | Modifiers
--------------+----------+-----------
 boundary_id  | integer  | not null
 start_node   | integer  |
 end_node     | integer  |
 left_parcel  | integer  |
 right_parcel | integer  |
 tmin         | date     |
 tmax         | date     |
 description  | text     |
 quality      | integer  |
 geom         | geometry |
Indexes:
    "boundaries_pkey" PRIMARY KEY, btree (id)
    "boundaries_en" btree (end_node)
    "boundaries_geom_gist" gist (geom)
    "boundaries_lp" btree (left_parcel)
    "boundaries_oid" btree (oid)
    "boundaries_rp" btree (right_parcel)
    "boundaries_sn" btree (start_node)
Check constraints:
    "enforce_geotype_geom" CHECK (geometrytype(geom) = 'LINESTRING'::text OR geom IS
NULL)
    "enforce_dims_geom" CHECK (ndims(geom) = 2)
    "enforce_srid_geom" CHECK (srid(geom) = 3057)
Foreign-key constraints:
    "boundaries_left_parcel_fkey" FOREIGN KEY (left_parcel) REFERENCES parcels(p_id)
    "boundaries_right_parcel_fkey" FOREIGN KEY (right_parcel) REFERENCES
parcels(p_id)
    "boundaries_start_node_fkey" FOREIGN KEY (start_node) REFERENCES monuments(m_id)
    "boundaries_end_node_fkey" FOREIGN KEY (end_node) REFERENCES monuments(m_id)
Triggers:
    "updateCadstreMV_boundaries" AFTER INSERT OR DELETE OR UPDATE ON boundaries FOR
EACH STATEMENT EXECUTE PROCEDURE updatecadastreview()
```

```
Table "public.parcels"
   Column    |         Type          |    Modifiers
-------------+-----------------------+--------------------------
 parcel_id   | integer               | not null default
 parcel_name | character varying(50) |
 land_id     | integer               |
 parcel_type | integer               |
Indexes:
    "parcels_pkey" PRIMARY KEY, btree (p_id)
```

```
View "public.boundaries_v"
    Column    |       Type       | Modifiers
--------------+------------------+-----------
 oid          | oid              |
 boundary_id  | integer          |
 geom         | geometry         |
 start_node   | integer          |
 end_node     | integer          |
 left_parcel  | integer          |
 right_parcel | integer          |
 quality      | integer          |
 length       | double precision |
View definition:
   SELECT b.oid, b.boundary_id, getboundaries(m1.geom, m2.geom, b.geom) AS geom,
b.start_node, b.end_node, b.left_parcel, b.right_parcel, deriveQuality(b.quality,
m1.quality, m2.quality) AS quality, round(length(getboundaries(m1.geom, m2.geom,
b.geom))) AS length
   FROM boundaries b, monuments m1, monuments m2
   WHERE b.start_node = m1.mon_id AND b.end_node = m2.mon_id;
```

```
View "public.cadastre_v"
   Column     |         Type          | Modifiers
--------------+-----------------------+-----------
 parcel_id    | integer               |
 parcel_name  | character varying(50) |
 land_id      | integer               |
 parcel_type  | integer               |
 description  | text                  |
 geom         | geometry              |
View definition:
   SELECT p.p_id, p.parcel_name, p.land_id, extractpolygon(polygonize(bv.geom)) AS
geom, p.parcel_type, p.description,
   FROM parcels p, boundaries_v bv
   WHERE p.p_id <> -1 AND (bv.left_parcel = p.parcel_id OR bv.right_parcel =
p.parcel_id)
   GROUP BY p.parcel_id, p.parcel_name, p.parcel_type, p.description, p.land_id;
```

```
Table "public.materialised_views"
    Column    |            Type             | Modifiers
--------------+-----------------------------+-----------
 mv_name      | name                        | not null
 v_name       | name                        | not null
 last_refresh | timestamp without time zone |
Indexes:
    "materialised_views_pkey" PRIMARY KEY, btree (mv_name)
```

```
Table "public.cadastre_mv"
   Column    |         Type          | Modifiers
-------------+-----------------------+-----------
 p_id        | integer               |
 parcel_name | character varying(50) |
 land_id     | integer               |
 parcel_type | integer               |
 description | text                  |
 geom        | geometry              |
Indexes:
    "cad_gist" gist (geom)
    "p_id_idx" btree (p_id)
```

APPENDIX I: MAPSERVER CONFIGURATION

The MapServer configuration presented below creates WFS that can bee accessed (read-only) by desktop clients that support WFS as input. The map file used for this purpose is not so much different from the map files designed to simple serve data to web browser (acting as a MapServer clients see Figure 56) but few things are prerequisites.

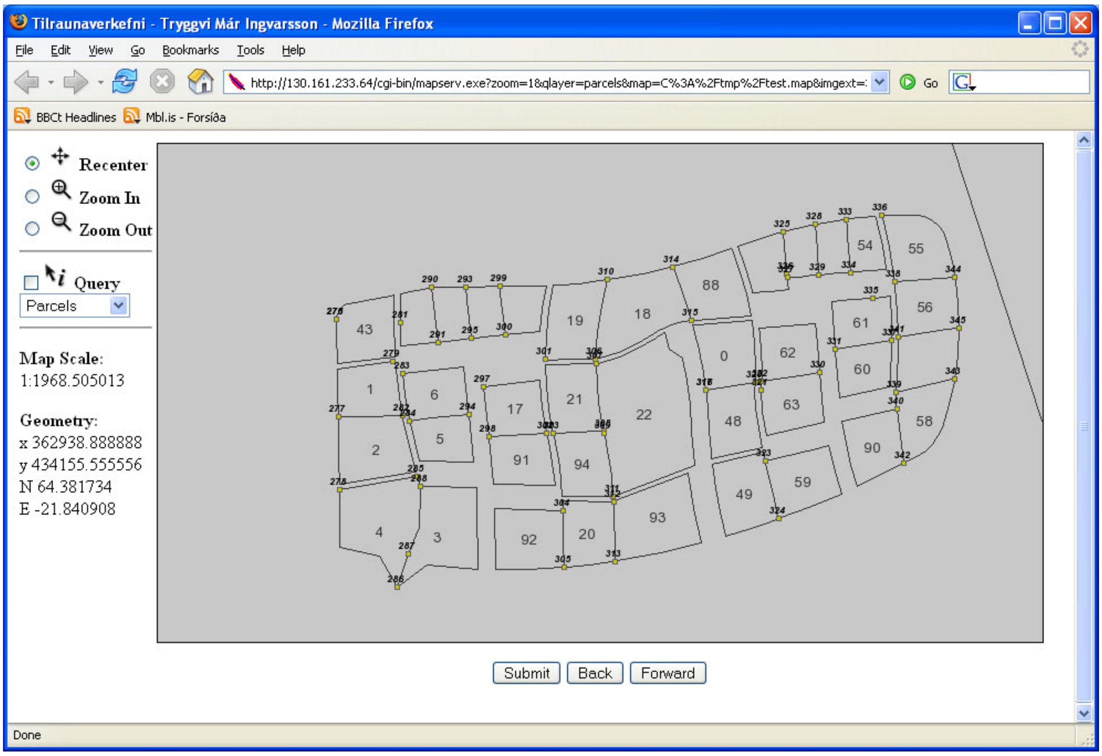


Figure 56: MapServer interface created with HTML. Its dynamicity is controlled by CGI variables.

One of these things is the use of the metadata. Briefly stated, to enhance a functional MapServer application to act as WFS one has to add metadata to it (which otherwise is not prerequisite) using 'wfs_title', 'wfs_onlinesource' and 'wfs_srs'. Moreover has the projection to be defined for the output and input layers. More detailed discussion on how to build a WFS server can be accessed at MapServer (2005, [2]).

```
MAP
  NAME iceland
  STATUS on
  UNITS meters
  DEBUG on
  IMAGETYPE png24
  EXTENT 350000 420000 370000 440000
  SIZE 800 450
  IMAGECOLOR 255 255 255
  SHAPEPATH 'C:/Data/MapServer_Data/Iceland/SHP/'
  SYMBOLSET 'C:/Data/MapServer_Data/Iceland/Symbols/symbols35.sym'
  FONTSET   'C:/Data/MapServer_Data/Iceland/Fonts/fonts.list'

  WEB
     HEADER 'C:/temp/test_header.html'
     FOOTER 'C:/temp/test_footer.html'
     TEMPLATE  'C:/temp/test.html'
     IMAGEPATH 'D:/Program Files/Apache Group/Apache2/htdocs/tmp/'
     IMAGEURL 'http://81.69.54.149/tmp/'
      METADATA
```

```
        "wfs_title" "Cadastre - Skilimannahreppur"
         "wfs_onlineresource"    "http://81.69.54.149/cgi-
        bin/mapserv.exe?map=C:\temp\test.map?"
        "wfs_srs" "EPSG:3057 EPSG:4269 EPSG:4326"
     END
  END

######## PROJECTION ########

   PROJECTION
      "init=epsg:3057"
   END

######## QUERY #############
  QUERYMAP
      SIZE 533 300
      STATUS on
      STYLE hilite
      COLOR 255 255 0
   END

######## LAYERS #############
   LAYER
      NAME "parcels"
      METADATA
         "wfs_title" "Parcels"
      END
      CONNECTIONTYPE postgis
      CONNECTION "user=Tryggvi dbname=cad password=040977 host=localhost"
      DATA "geom from (SELECT * FROM cadastre WHERE land_id <> -1) as cad USING
         UNIQUE oid USING SRID=3057"
      STATUS default
      TYPE polygon
      DUMP true
      LABELITEM land_id
      LABELCACHE on

      CLASS
         OUTLINECOLOR -1 -1 -1
      COLOR 200 200 200
         TEMPLATE 'C:/temp/test_parcels.html'
         LABEL
            COLOR 50 50 50
            TYPE TRUETYPE
            FONT arial
            SIZE 10
            ANTIALIAS TRUE
            POSITION cc
            PARTIALS FALSE
            BUFFER 4
       END
      END

      PROJECTION
         "init=epsg:3057"
      END
   END

   LAYER
      NAME "boundaries"
      METADATA
         "wfs_title" "Boundaries"
      END
      CONNECTIONTYPE postgis
      CONNECTION "user=Tryggvi dbname=cad password=040977 host=localhost"
      DATA "boundary from (SELECT * FROM boundaries_view) as bound USING UNIQUE oid
         USING SRID=3057"
      STATUS default
      TYPE line
      DUMP true
      TOLERANCE 5

      CLASS
         OUTLINECOLOR 50 50 50
         TEMPLATE 'C:/temp/test_boundaries.html'
      END
```

```
      PROJECTION
         "init=epsg:3057"
      END
   END

   LAYER
      NAME "monuments"
      METADATA
         "wfs_title" "Monuments"
      END
      CONNECTIONTYPE postgis
      CONNECTION "user=Tryggvi dbname=cad password=040977 host=localhost"
      DATA "geom from (SELECT oid,id,name,quality,description,geom FROM monuments) as
         mon USING UNIQUE oid USING SRID=3057"
      STATUS default
      TYPE point
      DUMP true
      TOLERANCE 5
      LABELITEM id
      LABELMAXSCALE 5000
      LABELCACHE on
      CLASS
         COLOR 200 200 0
         OUTLINECOLOR 100 100 100
         SYMBOL "square"
         SIZE 4
         TEMPLATE 'C:/temp/test_monuments.html'
         LABEL
            FORCE true
            COLOR 0 0 0
            TYPE TRUETYPE
            FONT arial-bold-italic
            SIZE 7
            ANTIALIAS TRUE
            POSITION uc
            PARTIALS FALSE
            BUFFER 4
       END
      END

      PROJECTION
         "init=epsg:3057"
      END
   END
END
```