FIG Working Week 2009
Surveyors Key Role in Accelerated Development
Eilat, Israel, 3-8 May 2009

UNIVERSITY of OTAGO
Te Whare Wānanga o Otāgo
NEW ZEALAND

Architecture for an Open Source Land Administration Application

Geoff HAY and G. Brent HALL
School of Surveying
University of Otago
Dunedin
New Zealand
(*Aotearoa*)

UNIVERSITY of OTAGO
Te Whare Wānanga o Otāgo
NEW ZEALAND

Overview

- Background
- Temporal Model
  - Events
  - Cross-jurisdictional schema variation
  - Schema evolution
- Instrument based application
  - Legal definition of data/process
  - Workflow process applies an instrument
- Externalized knowledge model
  - RDF/OWL
- OSCAR framework
  - Components
  - Interaction
  - Current Status

Background

- Initiative came from the Land Tenure Group of the Food and Agricultural Organization (FAO) of the UN
- Free/Libre Open Source Software (FLOSS) cadastre and land registration initiative started in 2007 with FAO Expert Group Meeting in Rome.
  - Considered that the use of FLOSS for cadastre and land registration could benefit many land administration projects, especially in developing countries.
- Free/Libre Open Source (FLOSS) Cadastral and Land Registration shell seminar, Dunedin 2008.
  - Participants from NZ, Australia, Fiji, Samoa, Vietnam, Cambodia, Nepal, the Kyrgyz Republic, Albania, Switzerland, and FAO in Rome.

Typical Issues

The state of current systems is poor…

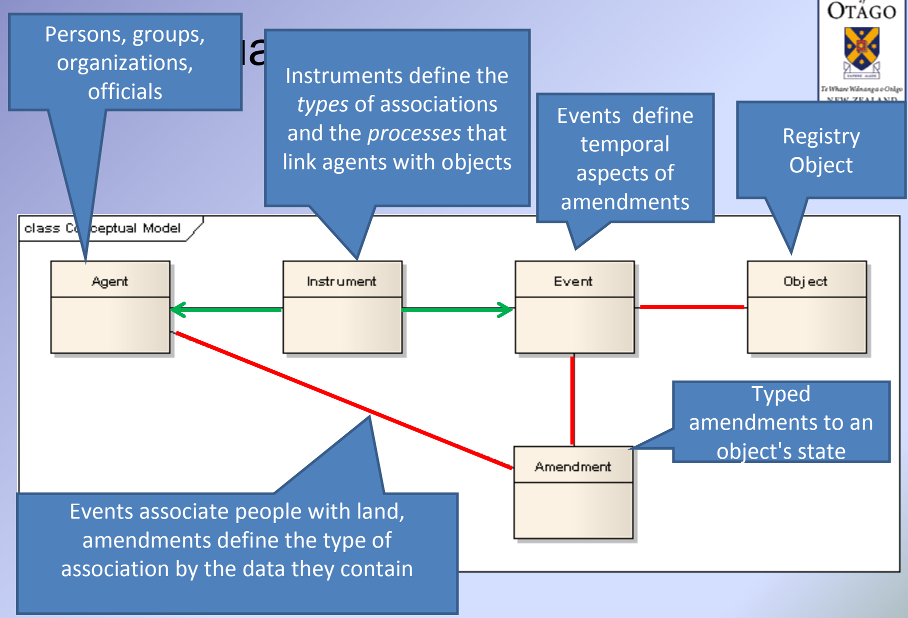
custom software is required…

hindered by…

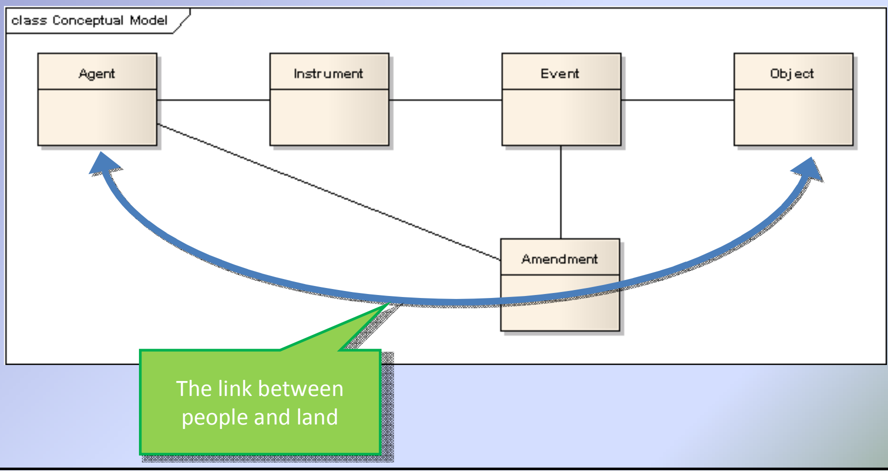
- Paper based systems
- Lost or destroyed data, incompl
- Land registration across multiple
- Cultural aspects (customary ownership, right of use, etc)
- Lack of HR especially IT
- Lack of money
- Funding processes
- Highlighted the need for computerization of land registry data and administration processes (as important as the spatial aspects)
- OSCA showed that FLOSS components (and the OS development model) can easily be used to build software for managing the spatial aspects of a cadastre
- OSCAR adds the registry capability using FLOSS components
- OSCAR also shows that it is possible to build software that can be reused across jurisdictions despite variations in schema (data models, concepts, understanding of terms, relations, associations, processes, cultural aspects, etc)

Event-based temporal model

- History for an object (e.g. land parcel, building) can be stored as a set of *events* which contain a set of *amendments* (to spatial or thematic data)
- Events (and amendments) are inserted as part of a defined process or *workflow*
- Events record the changes to objects over time. Typically changes will be at irregular intervals but may also be regular, cyclic etc.



Core Conceptual Model

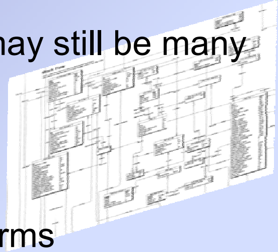


Event-based model

- For any point in time (including now), events (and associated amendments) can be summed to reproduce a *state*
- However, relational/object database schema are essentially *current state*, *static*, and must be explicitly defined before use
  - Difficult to design/implement a temporal data model that allows multiple versions of the same attribute (i.e. history)
  - A true temporal model should also allow schema evolution and emergence

Cross-jurisdictional Schema Variation

- For similar schema there may still be many differences:
  - Language
  - Concepts
  - Understanding/use of terms
  - Interactions
  - Processes
- Custom builds will not easily allow integration, sharing or cooperation
- Custom builds are costly, time consuming, and may fail



Schema Evolution (and emergence)

- Schema change over time
  - Attributes come and go,
  - types change
  - use and understanding changes
  - not all is known at the time a schema is defined
- Difficult to change/modify 'hard-coded' systems
  - Systems become 'brittle' over time
  - Costly to modify 'hard-coded' systems
  - Requires recompilation, testing, redeployment

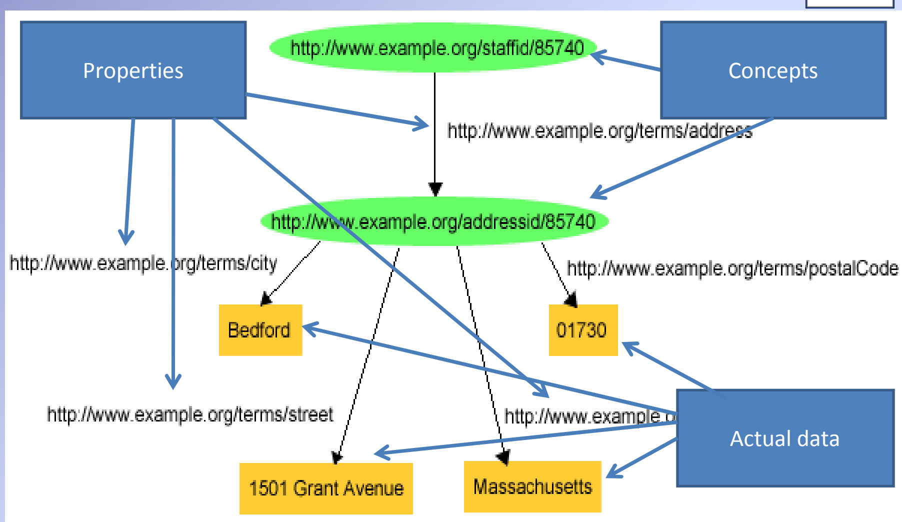
Solving the schema evolution problem also solves the variation problem

OSCAR

Addresses variation, emergence, and evolution:

- No ‘hard-coded’ data model – data models are implied as part of process definition
- Meta-data are encoded externally in a ontology description language (OWL)
- Data (amendments) are stored with ‘mark-up’ that describe its properties (RDF) – software interrogates the data
- Data are ONLY inserted (never deleted or overwritten) via defined processes
- Process definition also provides for cross-jurisdictional integration and harmonization

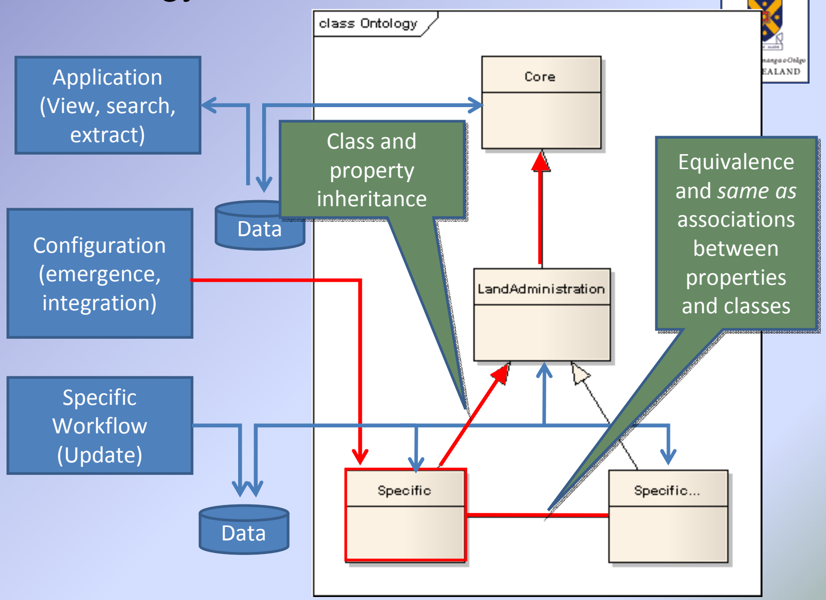
Resource Description Framework (RDF)

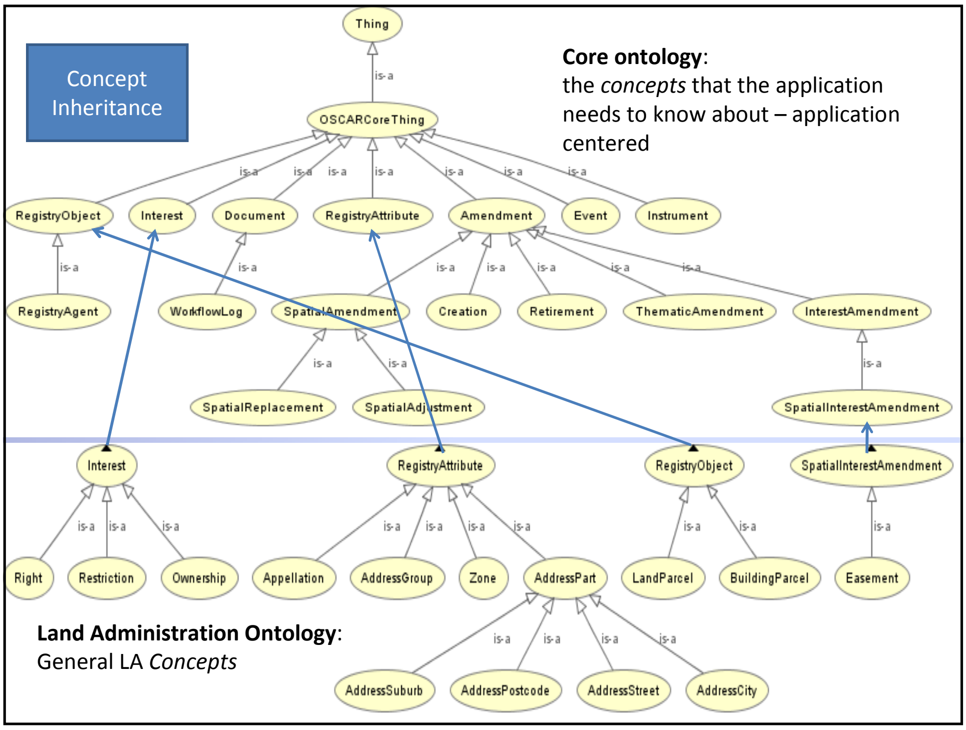


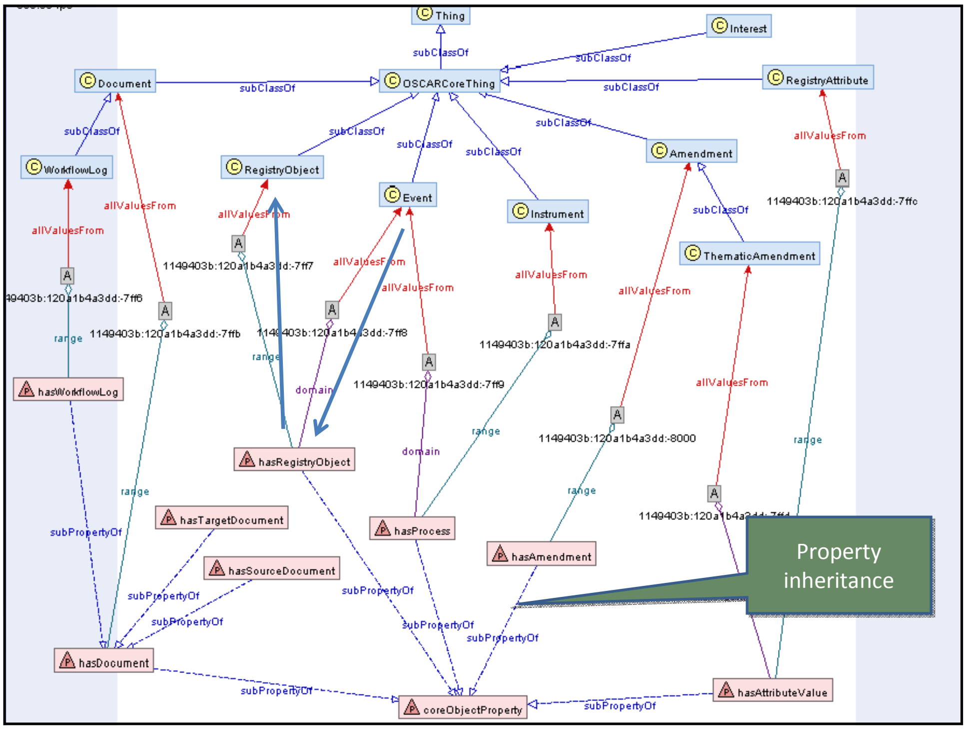
OWL Ontology

- Web Ontology Language (OWL) defines the meta-model:
  - concepts, object properties (associations) including cardinality, data type properties, annotations
  - Reasoning, inference to produce new entailments
  - RDF Schema, Dublin Core document meta-data standard also used

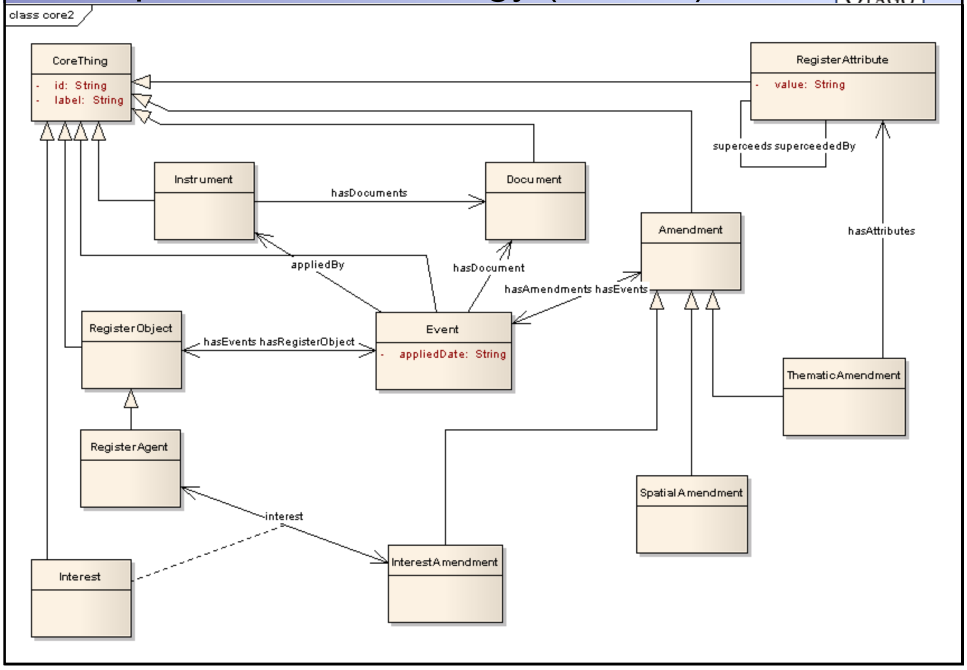
OSCAR Ontology use




Concept Inheritance
Core ontology: the concepts that the application needs to know about – application centered
Thing
OSCARCoreThing
RegistryObject
Interest
Document
RegistryAttribute
Amendment
Event
Instrument
RegistryAgent
WorkflowLog
SpatialAmendment
Creation
Retirement
ThematicAmendment
InterestAmendment
SpatialReplacement
SpatialAdjustment
SpatialInterestAmendment
Interest
RegistryAttribute
RegistryObject
SpatialInterestAmendment
Right
Restriction
Ownership
Appellation
AddressGroup
Zone
AddressPart
LandParcel
BuildingParcel
Easement
Land Administration Ontology: General LA Concepts
AddressSuburb
AddressPostcode
AddressStreet
AddressCity

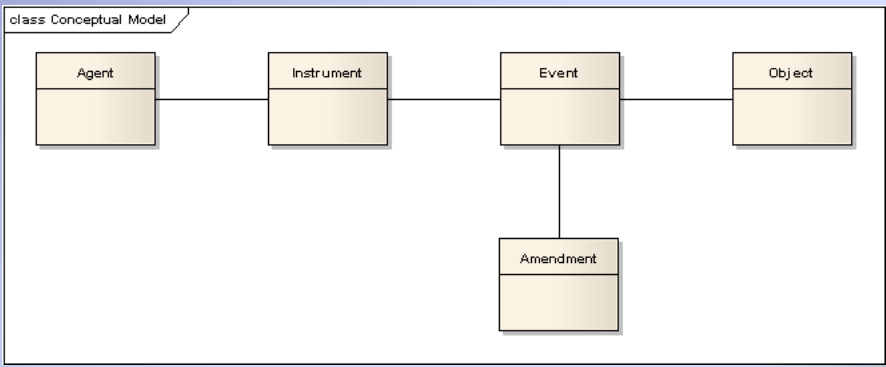

Thing
Interest
Document
OSCARCoreThing
RegistryAttribute
WorkflowLog
RegistryObject
Event
Instrument
Amendment
ThematicAmendment
hasWorkflowLog
hasRegistryObject
hasTargetDocument
hasSourceDocument
hasProcess
hasAmendment
hasDocument
coreObjectProperty
hasAttributeValue
Property inheritance

Simplified Core Ontology (in UML)

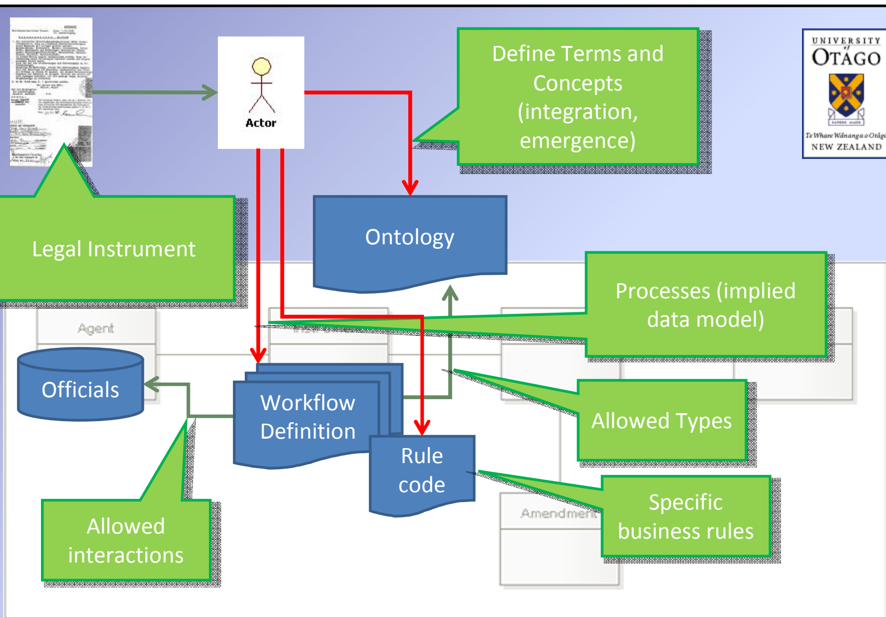


Instrument based application

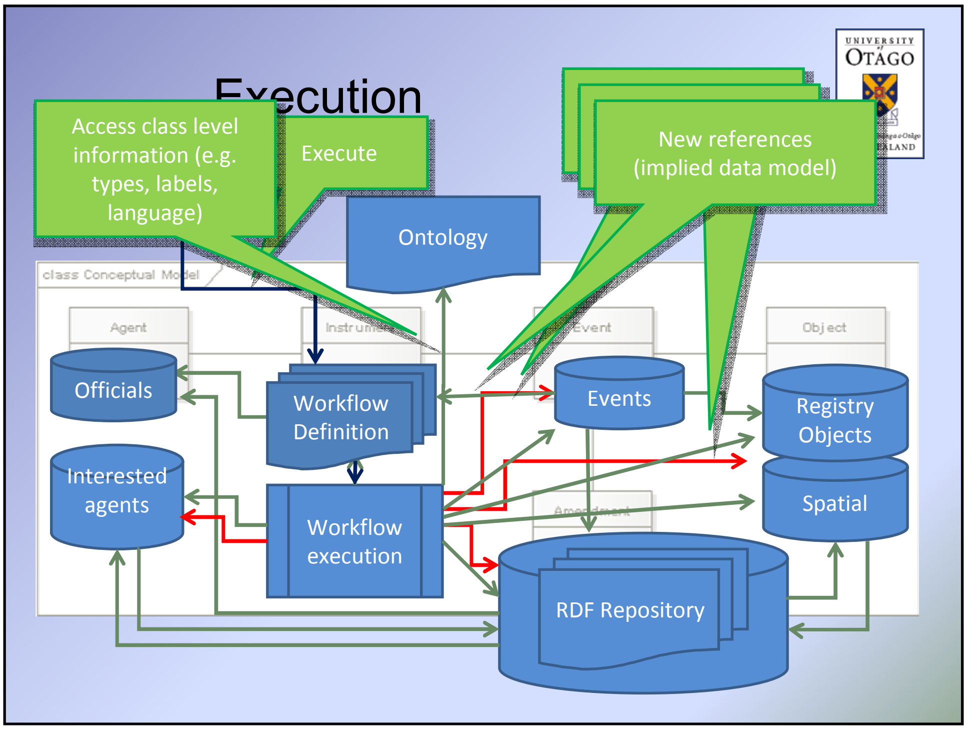
- An *instrument* is a legal definition of a process by which the repository may be modified
- An *instrument* implies a data model
- An *instrument* can be implemented using workflow software that manages the interaction between people and the repository
- The workflow process also manages the temporal and semantic models (analogous to a DBMS). Can also manage the meta-model (emergence and integration) in a configuration process



Computerization of an Instrument



Computerization of an Instrument



Advantages of this approach

- Temporal model stores history
- Semantic model and process orientation (Instrument-based) supports schema evolution and variation
- Improved reusability across jurisdictions, configuration based on translating legal instrument
- FLOSS components reduce costs and improve sharing and cooperation
- SDK environment allows targeted builds i.e. different software for different users
- Process orientation supports intranet/networked application.

Future plans

UNIVERSITY of OTAGO
Te Whare Wānanga o Otāgo
NEW ZEALAND

- Research continuing on cross-jurisdictional schema variation, schema evolution, and ontology definition and reasoning for temporal land registry databases.
- Prototype OS coding is well underway for the core OSCAR components. Initial release is expected in December 2009
- Testing will use land registry database and parcel mapping from Western Samoa
- Initial call of interest for participation in extended project has been issued by FAO

Thanks to Mika Törhönen (FAO).

UNIVERSITY of OTAGO
Te Whare Wānanga o Otāgo
NEW ZEALAND

GeoTools The open source Java GIS toolkit
Jena semantic web framework
W3C OWL
eclipse
JAVA
protégé
W3C Semantic Web
RDF
JBoss The Professional Open Source Company
PostgreSQL
PostGIS
uDig
ADUNA