Publish and Share

LADM Manual

Version 1.0

2019-08-01

Email: contact@its4land.com

Web: https://its4land.com/

Table of Contents

INTRODUCTION **2**
OVERVIEW OF LADM **3**
LADM CONCEPTS IN PUBLISH AND SHARE **6**
QUALITATIVE REPRESENTATION OF SPATIAL DATA **10**
SPECIFICATION OF THE 'QUALITATIVE BASE' SPATIAL PROFILE 12
THE 'QUALITATIVE BASE' SPATIAL UNIT IN PUBLISH AND SHARE 14
CADASTRE DATABASE FOR QUALITATIVE SPATIAL INFORMATION IN PUBLISH AND SHARE 15
SUGGESTED READING **19**

Introduction

This manual describe the utilisation of the Land Administration Domain Model (LADM) in the Publish and Share platform of its4land.

The Land Administration Domain Model (LADM / ISO 19152) [1] provides a conceptual framework for land administration systems. The main goal of LADM is to define a common set of concepts and terms for land administration stakeholders. This common set of concepts and terms is an ontology of the land administration domain.

This manual further more describes the its4land extension of LADM to store qualitative describes data as spatial reference for a tenure registration process.

Overview of LADM

The conceptual framework of LADM is documented in three packages and two sub packages. A (sub)-package is a group of classes (each with its own namespace). The complete model may therefore be implemented through a distributed set of (geo-) information systems, each supporting data maintenance activities and the provision of elements of the model. The model may also be implemented by one or more maintenance organizations operating at the national, regional or local level. This underlines the relevance of the model: different organizations have their own responsibilities in data maintenance and supply, but may communicate based on standardized administrative and technical update processes. The LADM packages are:

- Party
- Administrative
- Spatial Unit
- Surveying (sub-package of the Spatial Unit package)
- Spatial Representation (sub-package of the Spatial Unit package)

An overview of the packages (with their respective classes) is presented in **Figure 1** as an UML class diagram.

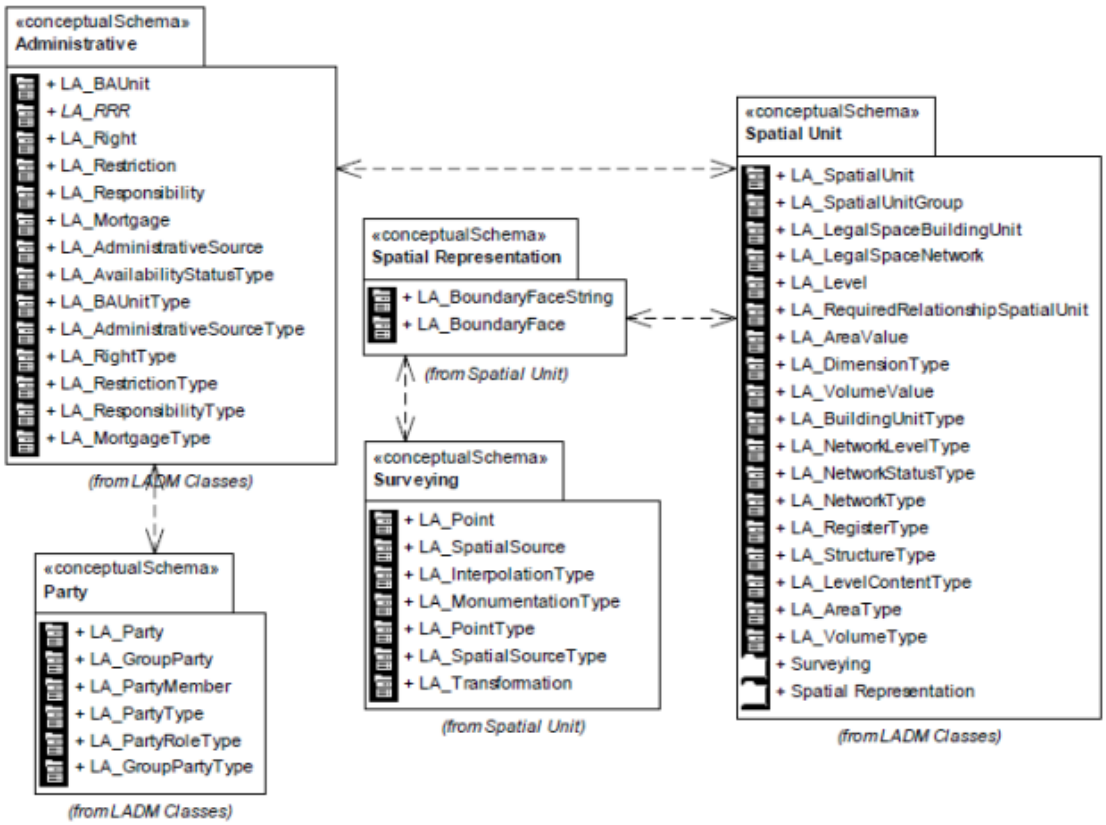


Figure 1: Overview of LADM

In a land administration system based on LADM standards a core set of LADM classes is generally used. These classes are the minimum subset of LADM to build a land administration system [4], [5]:

Class **LA_Party**. Instances of this class are parties. Parties are persons / organizations, or group of persons / organizations, that compose an identifiable entity.

Class **LA_RRR**. Instances of subclasses of **LA_RRR** are rights, restrictions or responsibilities (Formal or informal entitlement to own or to do something, or not to do something).

Class **LA_BAUnit**. Instances of this class are basic administrative units (Legal administrative entity consisting of zero or more spatial units with which (one or more) unique and homogeneous rights are associated).

Class **LA_SpatialUnit**. Instances of this class are spatial units (Single or multiple areas of land / water or single / multiple volumes of space under (one or more) unique and homogeneous rights, restrictions, or responsibilities).

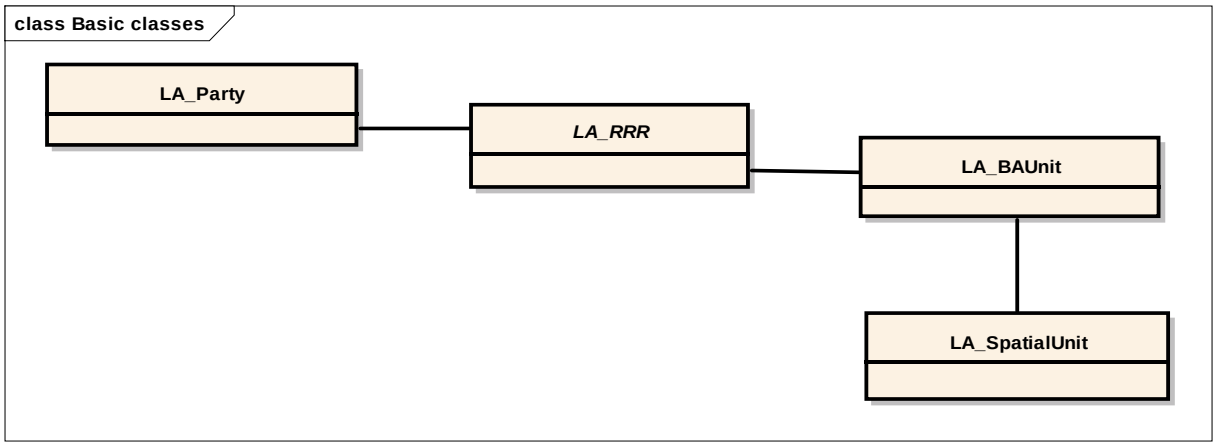


Figure 2: Core classes of LADM[1]

Figure 2 shows the relationship of these basic classes. In a practical implementation, concrete classes will be derived from the LADM classes. This step includes the country specific naming, attribute set, etc.

E.g. the country profile for the Netherlands contains two instantiable classes (**NL_BuildingUnit** and **NL_Parcel**). Both classes are derived from **LA_SpatialUnit** and contain country specific attributes and definition. In an object oriented view **NL_ BuildingUnit** and **NL_Parcel** are specializations of **LA_SpatialUnits** [1].

LADM concepts in Publish and Share

Publish and Share focuses on providing a spatial reference for a Land Administration System. Therefore only a subset of the LADM concepts is necessary. In the current stage of implementation, the following concepts are included:

LADM Package	**LADM Class / Concept**	**Description**
Spatial Unit	LA_SpatialUnit	LA_SpatialUnit is the spatial reference in an LADM based LAS. LA_SpatialUnit is used in Publish and Share, because it is the spatial reference for any kind of tenure registration. The interpretation of LA_SpatialUnit provided by Publish and Share is subject of the context, including the legal framework, in which Publish and Share is utilized. Publish and Share supports several spatial profiles, e.g. the 2D polygon profile from LADM and the qualitative data profile from its4land. An example for a 2D Polygon SpatialUnit is a parcel that was created by a feature extraction tool. An example for a qualitative data SpatialUnit is a ranch created by *Draw and Make.*
Spatial Unit	LA_Level	LA_Level can be used to group LA_SpatialUnits with a geometric or thematic coherence.

		LA_Level is used for grouping SpatialUnits based on their dedicated use in the LAS. Example: Distinguish a parcel that will be registered as an urban parcel from those to be registered as rural parcels. Example: Distinguish qualitative SpatialUnits which represent different types of community land, like seasonal pastures.
Spatial Representation	LA_BoundaryFaceString	LA_BoundaryFaceString forms the outside of LA_SpatialUnit in a 2D geometric representation. LA_ BoundaryFaceString represents a general or fixed boundary. A land administration system can use boundary face strings to create SpatialUnits as needed in the specific implementation. Publish and Share currently supports only non-topological 2D BoundaryFaceStrings. Example: *Automate It* produces general boundaries which are managed as BoundaryFaceStrings in Publish and Share.
Surveying	LA_SpatialSource	LA_SpatialSource documents the evidence of a spatial unit or a boundary face string. LA_SpatialSources are used in Publish and Share as input for tools or

		applications to generate SpatialUnits and/or BoundaryFaceStrings. A LA_SpatialSource could be any kind of document, like orthomosaics, images, surveying sketches, etc. Publish and Share has no restriction. The interpretation of a SpatialSource is part of the utilization of Publish and Share in a specific project or implementation. Example: The SketchMaps used in *Draw and Make* are treated as SpatialSources for the created SpatialUnits. Example: The SpatialSources for the BoundaryFaceStrings created in *Automate It* are the orthomosaics from *Fly and Create*.
Administrative	LA_AdministrativeSource	LA_AdminSource documents the evidence of an interest (see LA_RRR) in land. This includes rights, restrictions, responsibilities and also parties. LA_AdminSources are used in Publish and Share to document rights, restrictions and responsibilities along with parties in a narrative way. These are not legally valid registrations. The legally valid registration is done in the LAS. The interpretation of a LA_AdminSource Document is part of the utilization of Publish and Share in a specific project. This depends highly on the legal framework of the country where the project is conducted. LA_AdminSource Documents can be any kind of document. Starting from a scanned contract, a word document to a structured recording of land

		narratives based on alternative concepts of land rights. Example: *Draw and Make* captures, besides the qualitative description of land, information about land rights und land usage based on community related ontologies as well. This information is stored by *Draw and Make* in the structured form of JSON files and stored as LA_AdminSources in Publish and Share.

With the implemented LADM concepts, Publish and Share can provide the necessary information to a land administration system that is needed for legal registration. A developer who is familiar with LADM will recognize these known concepts and can map them to the structures and localized meanings needed by a specific land administration system.

Qualitative representation of spatial data

Work package 3 “Draw and Make” provides a tool for capturing land tenure registration data in cadastral databases.

In our context, we define a cadastral database as the sum of all spatial objects, which form the spatial reference for the registration of interest in land. Our definition does not determine the spatial representation.

Only the Spatial Unit and its two sub-packages are relevant for our definition of a cadastral database. The cadastral database in our context is the sum of all LA_SpatialUnit class instances.

The LA_SpatialUnit is defined as “single area (or multiple areas) of land […] and/or water, or a single volume (or multiple volumes) of space” [8]. The spatial unit itself does not specify the spatial representation. The spatial representation is defined by the two classes LA_BoundaryFaceString and LA_BoundaryFace in the spatial representation sub package:

- LA_BoundaryFaceString: defining the boundary between a spatial unit and the outside in 2D by a 1-dimensional topological primitive. A boundary face string can have multiple appearance (see spatial profiles below).

- LA_BoundaryFace: defining the boundary between a spatial unit and the outside in 3D by combining 2-dimensional topological primitives. A boundary face can have multiple appearances (see spatial profiles below).

LADM identifies six different types of spatial units.

Spatial profile	**Description**
sketch based	A sketch (a quick drawing of a group of spatial units) is available, e.g. sketch maps (survey sketches), and photographs, in the absence of any better identification.
point based	A point based spatial unit identifies an area by a point within the area. This profile is typically used when only the location and the size is known.
text based	A text based spatial unit is used when an area is entirely described by text. The spatial

	unit is in general associated with one or more boundary face strings, with their location described by free text. No geometry is used in this kind of spatial unit.
unstructured (line) based	An unstructured (line) based spatial unit references boundary face strings. The boundary face string is represented by a polyline. This profile is used, when inconsistencies, such as hanging lines and incomplete boundaries are allowed.
polygon based	A polygon based spatial unit has exactly one link to a polygon. Every spatial unit is treated as a separate entity without topological connection between neighbouring spatial units. Standard GIS supports this profile by default.
topological based	A topological based spatial unit is encoded by referencing boundary face strings. In opposite to the unstructured line or polygon profiles, the boundaries are topologically connected and consistent. Adjacent spatial units share a boundary face string as a joint boundary. The boundary face strings of a spatial unit form a closed loop. The topological profile can be best compared with the node-edge-face model.

None of the existing spatial profiles is able to handle qualitative descriptions of spatial objects in such a way that they can express the uncertainty of boundaries and also allow spatial operations on them. The text based spatial profile would allow a qualitative description, but not spatial operations.

We suggest extending LADM by an additional spatial profile to handle qualitative data. The spatial profile is named “qualitative base”. The spatial profile should satisfy the following requirements:

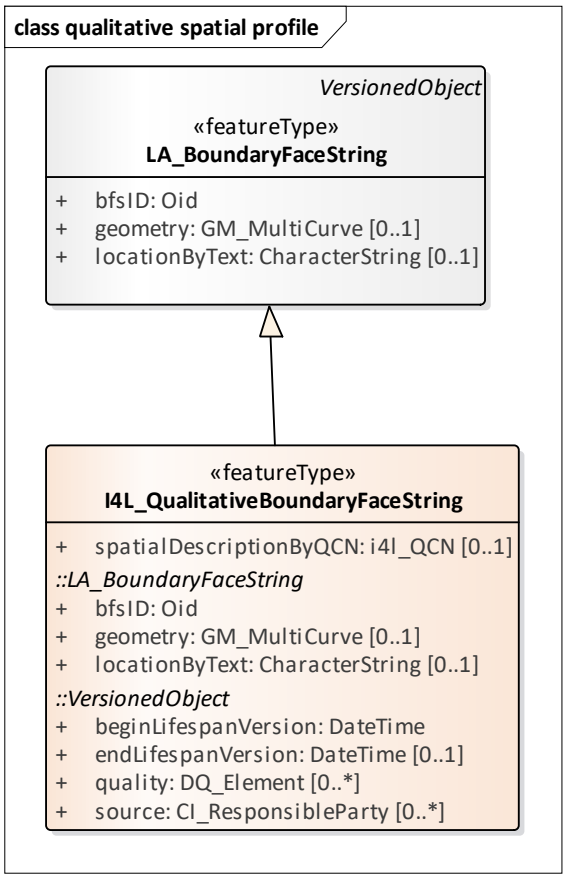
- Store the qualitative description of a spatial unit in a computational way, so that it can be used for spatial queries

- An additional quantitative representation as 2D polygon for the manually created approximation in the SmartSkeMa should be available

Specification of the ‘qualitative base’ spatial profile

The “qualitative base” spatial unit is used when its definition is the outcome of a qualitative mapping process in SmartSkeMa. A spatial unit itself represents one spatial feature in the sketch map. The ‘qualitative base’ spatial profile supports 2D.

The boundary face string for “qualitative base” spatial unit is inherited from the original LADM class LA_BoundaryFaceString and is called “I4L_QualitativeBoundaryFaceString”.



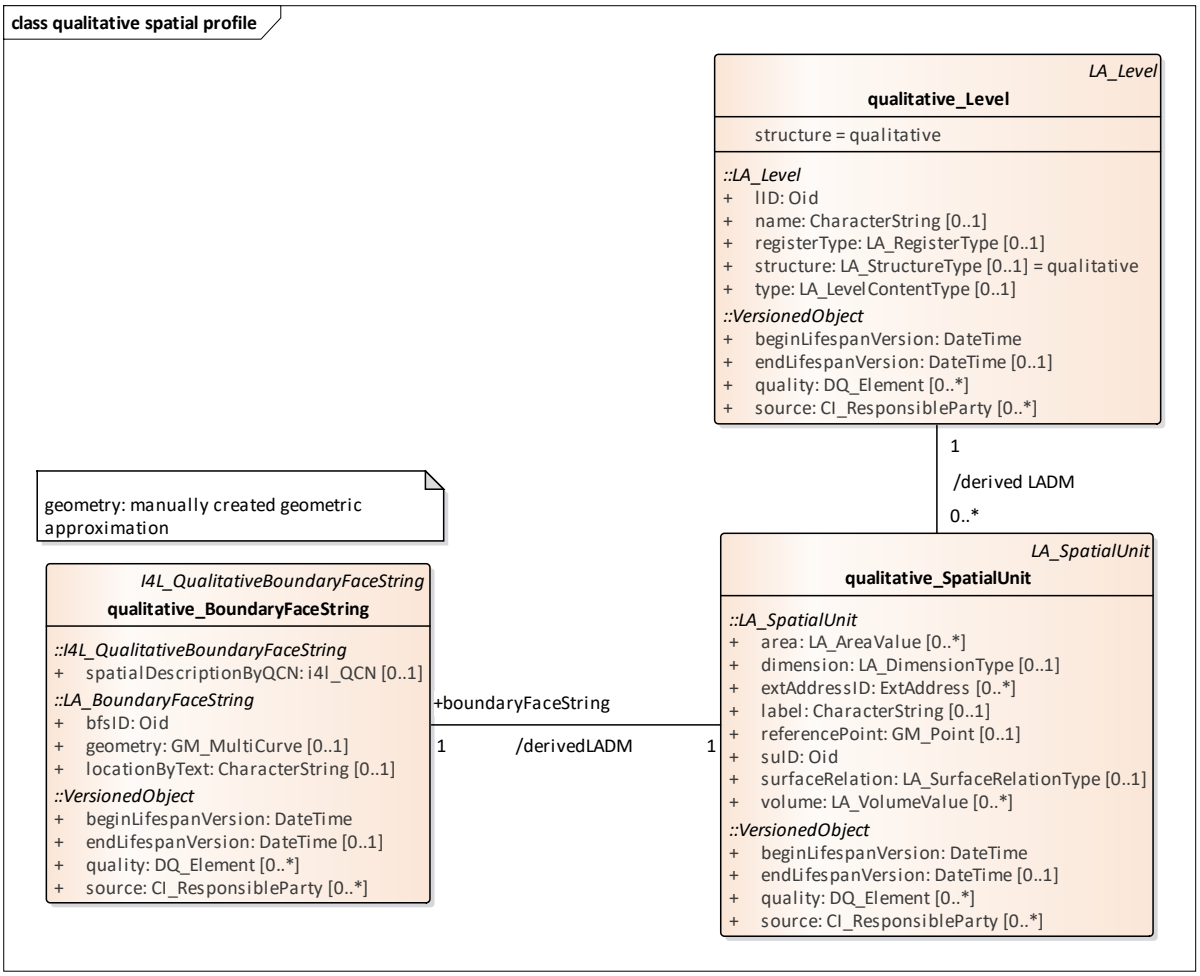
An instance of class I4L_QualitativeBoundaryFaceString is a qualitative boundary face string. The class I4L_QualitativeBoundaryFaceString is a sub class of LA_BoundaryFaceString.

A qualitative boundary face string represents the boundary of a spatial unit. The class I4L_QualitativeBoundaryFaceString has the same properties and can be used in the same situations as the class LA_BoundaryFaceString.

I4L_QualitativeBoundaryFaceString extends LA_BoundaryFaceString by one additional attribute:

- spatialDescriptionByQCN: the boundary represented by the qualitative constrained network (QCN) extracted from geometric representation of input maps in work package 3.

The qualitative representation using the QCN stores the spatial configuration as a set of qualitative relations between spatial features in sketch map. These are determined in the sketch map qualification process in SmartSkeMa.



As a specialisation of LA_BoundaryFaceString, I4L_QualitativeBoundaryFaceString, also supports the geometry attribute for a quantitative representation of the boundary face string as a closed polygon in parallel to qualitative representation. The geometry attribute is used to store the manually approximated geometry of the spatial unit. This approximated geometry is created in SmartSkeMa.

A ‘qualitative base’ spatial unit allows up to one qualitative and up to one quantitative spatial representation of the boundary.

The ‘qualitative base’ spatial unit in Publish and Share

The implementation of the ‘qualitative base’ spatial unit can be fully adopted on the structure of the Publish and Share platform [12][13].

The QCNs representations are stored in JSON files. SmartSkeMa defines the internal structure of the file. This JSON structure is stored as a ContentItem via the corresponding endpoint (POST /ContentItem).

The Public API provides two endpoints to handle spatial units:

- /SpatialUnit - used for standard 2D 'polygon based' spatial unit. However, it also allows to access the quantitative representation of a 'qualitative base' spatial unit. When accessing a 'qualitative base' spatial unit via this endpoint it behaves like a 2D 'polygon base' spatial unit.
- /SpatialUnit/qualitative - provides full access to the 'qualitative base' spatial unit, which includes the qualitative representation. Depending on the size of the QCN JSON structure, it will be embedded into the response of the /SpatialUnit/Qualitative endpoint or retrieved separately via the /ContentItem endpoint.

For integration with standard GIS/LAS workflows, the quantitative representation of a boundary (of a 'qualitative base' spatial unit) can be disseminated via the data dissemination interface of Publish and Share [14].

Cadastre Database for qualitative spatial information in Publish and Share

In this chapter, we describe how the purported benefits of including qualitative spatial information in a LAS can be achieved with the use of the Publish and Share platform. The Public API offered by Publish and Share offers users a means to implement these concepts in their usage scenarios.

The API provides several 'endpoints', which are accessed by a path (URL) on the web server hosting the platform implementation. A *request* to an endpoint consists of sending a data payload and requisite parameters to the correct path. Upon successful processing of a request, the server sends a *response* with a status code indicating success/failure along with an optional payload consisting of processed or requested data. The table below briefly describes the endpoints groups useful for storage of cadastral data including qualitative spatial information. An endpoint group consists of a set of related endpoints.

Table 1 Publish and Share Public API endpoint groups to implement Cadastral Database for Qualitative Spatial Information.

Endpoint Group	**Description**
Projects	Stores and retrieves the project context for Publish and Share. A project's context information contains basic metadata, the spatial area of interest, input spatial sources, tags and any output spatial units.

ContentItems	Refers to raw content that is stored in the backend storage. The content is usually a file. The file and basic metadata about the file are stored without paying attention to the purpose of the file itself.
SpatialSources	A spatial source is an LADM concept, which documents the evidence for a spatial unit, such as sketch, map, orthoimages etc. This endpoint refers to a unique content item and provides meaning to raw data.
AdminSources	Similar to spatial sources, but deals with legal documents pertaining to evidence of interest in land.
SpatialUnits	Stores and retrieves information about the actual piece of land to register. Geometries of the land parcel are described as per a given LADM spatial profile.

The following subsections discuss how API endpoints are useful in implementing the cadastre database.

Storing Spatial Sources

As described previously, the GUI of SmartSkeMa consists of a sketch map panel and its corresponding base map. Sketch maps are stored in the SVG format and are not georeferenced, whereas base maps use features stored in GeoJSON format. Since both of these depict the same area of interest, in LADM terms, both are instances of spatial sources. Publish and Share uses the project context to structure related items, with each project having its own unique identifier. Within the context of a project, the spatial sources in Publish and Share are stored as follows:

1. Use the POST */contentitems* endpoint to upload the sketch map and obtain a UUID. Let us call it *sketch_uid*.
2. Use the POST */projects/{project_uid}/SpatialSources* endpoint to register the content with *sketch_uid,* as a Spatial source.
3. Repeat steps 1 and 2, but this time with the base map to register the GeoJSON as a spatial source in the same project.

When stored in this manner, the actual content containing the spatial data is separated from the notion of a spatial source, which is a higher-level abstract concept. Given a spatial source, the API provides means to obtain its actual contents. A project can have multiple spatial sources; however, each spatial source is related to only one content item.

Storing Qualitative Representations

The qualitative data used to aid a user in the demarcation of land parcels captures spatial constraints between features in the project area. These are described as a qualitative constraint network (QCN). A QCN is serialized in a file and describes the relationship, according to a chosen calculus, between pairs of features. The file describing the QCN is stored via the *contentitems* endpoint.

Storing Non-spatial information

In addition to spatial data, LADM also has provisions to capture the legal rights of parties for a piece of land. Publish and Share stores this non-spatial information via *AdminSources*. An AdminSource documents the evidence for interest in land such as rights like ownership or restrictions such as the right-of-way. The document merely provides evidence for later registration in an LAS and does not constitute the registration itself.

For storing such non-spatial information, the *adminsource* endpoint of API requires a reference to a *contentitem* pointing to the actual document and optionally URLs to external sources along with standard metadata fields such as name and description. The AdminSource document can also be added to the context of a project. Additionally, the AdminSource document can point to one or more spatial units, which are the actual parcels of land being registered.

Support for interactive tools

Tools in Publish and Share can run either in *batch mode* or in *interactive mode*. Batch mode is ideal for computationally demanding tasks with long running times, which do not require user prompts to proceed. For a tool with a GUI such as SmartSkeMa, interactive mode is required. With interactive mode, tools can accept user input when running in order to perform a task.

While SmartSkeMa is capable of running as a standalone application, there are several advantages offered by integrating it with the Publish and Share platform. These benefits accrue depending on to what extent the tool is integrated. Integration consists of using the provided API and workflow guidelines. The integration can be basic or tightly coupled to Publish and Share.

Basic Integration – For a basic integration with Publish and Share, it is sufficient to use the API to use the *projects* endpoint to load and save the project context. A basic integration provides an organized structure for storing related information such as spatial sources and admin sources relevant to the task. Storage of required files however becomes the responsibility of the SmartSkeMa tool. Anytime the project is loaded, the required sketch map, non-spatial information and base map will be retrieved from the tool's storage area or uploaded by a client. This level of integration is sufficient to perform one-off tasks, where one does not expect to reuse stored data frequently.

Complete Integration – In a complete integration, in addition to the Projects endpoint, other endpoints such as SpatialSources, AdminsSources, ContentItems and others are used. The resources are saved in the backend cloud storage of Publish and Share and available for reuse anytime. Resources are available anytime and shared between different running instances of the tool. Another advantage is that dissemination is easy since the platform hosts all resources. Benefits at this level of integration however, come at the cost of greater level of design and development complexity for the tool writer.

Suggested reading

[1] ISO/FDIS 19152, “Geographic information - Land Administration Domain Model (LADM),” vol. 2012, pp. 1–118, 2012.

[2] S. Enemark, K. C. Bell, C. Lemmen, and R. McLaren, “Fit-For-Purpose Land Administration,” 2014.

[3] R. T. Fielding, “Architectural Styles and the Design of Network-based Software Architectures,” University of California, Irvine, 2000.

[4] C. Timm, M. I. Humayun, S. Walter, R. Borchert, S. Crommelinck, and C. Stöcker, “Technical report and software prototype of the mobile image processing system. Its4land Deliverable 6.1,” Enschede, 2018.

[5] C. Timm, M. I. Humayun, S. Walter, M. Chipofya, J. Sahib, and K. Kundert, “Technical report and software prototype of the mobile qualitative data processing system. Its4land Delivery 6.2,” Enschede, 2018.

[6] Network Working Group, “RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1.” 1999.

[7] Internet Engineering Task Force (IETF), “RFC 7946: The GeoJSON Format.” 2016.

[8] SmartBear Software, “OpenAPI Specification Version 2.0,” 2014. [Online]. Available: https://swagger.io/specification/v2/.

[9] Internet Engineering Task Force, “RFC 8259: The JavaScript Object Notation (JSON) Data Interchange Format.” 2017.

[10] World Wide Web Consortium (W3C), “Extensible Markup Language (XML) 1.1 (Second Edition).” 2006.

[11] Node.js Foundation, “node.js,” 2018. [Online]. Available: https://nodejs.org/en/.

[12] “Amazon Web Services S3 Homepage,” 2018. [Online]. Available: https://aws.amazon.com/s3/?nc1=h_ls.

[13] “OpenDroneMap.” [Online]. Available: https://www.opendronemap.org/.