DECEMBER 2018
FOOD AND AGRICULTURE ORGANISATION OF THE UNITED NATIONS (FAO)

REVIEW SOLA SUITE OF APPLICATIONS

FINAL REPORT

COWI

ADDRESS COWI A/S
Parallelvej 2
2800 Kongens Lyngby
Denmark

TEL +45 56 40 00 00
FAX +45 56 40 99 99
WWW cowi.com

DECEMBER 2018
FOOD AND AGRICULTURE ORGANISATION OF THE UNITED NATIONS (FAO)

REVIEW SOLA SUITE OF APPLICATIONS

FINAL REPORT

PROJECT NO.	DOCUMENT NO.
GCP/INT/328/UK A116394	1

VERSION	DATE OF ISSUE	DESCRIPTION	PREPARED	CHECKED	APPROVED
4	21-12-2018	Report of reviewing SOLA applications and the future options	Jan van Bennekom-Minnema (COWI) jvb@cowi.dk	Lars Bagge-Hommel Nielsen (COWI) lbni@cowi.dk	Lars Bagge-Hommel Nielsen (COWI)

CONTENTS

Summary

SOLA Applications:
Land information system
Registry
Systematic Registration
State Land
Field tenure data collection
Open Tenure
Community Server

The SOLA applications have been reviewed in terms of architecture, data model, functionality, and implementations. The SOLA applications are land information systems being used in two situations:

- The development and deployment of a land information system on behalf of a beneficiary in developing countries (i.e. SOLA Registry for land registry offices).
- Assist beneficiaries in developing countries with the field collection of tenure data (e.g. SOLA Open Tenure on mobile devices and SOLA Community Server for communities, CSO's, and customary groups).

Architecture

The SOLA applications are based on java open source components which make up a comprehensive, robust architecture, which requires specialist expertise to customise, deploy and maintain these. Various recommendations on improving the architecture, source code, and documentation have been made, which, dependent on the chosen future option for SOLA applications, may be addressed partly or whole. A separate review on architecture, source code, and documentation is incomplete and could not be used as the basis of additional recommendations on the quality of SOLA architecture and source code.

Level 1 Complaint with LADM

The data maintained by SOLA applications is compliant (Level 1) with the Land Administration Domain Model (LADM), and the SOLA data model has significant extensions to LADM (e.g. with regard to workflow and processes).

Functionality
SOLA Registry

SOLA Registry has the functionality to support typical registration and cadastral offices, offering support for managing property rights and right holders; geospatial information; searching, retrieve, view and generate reports on alphanumerical and geospatial data; workflow and progress; digital/scanned evidence documents. Non-functional system qualities are implemented for authorisation and authentication; system administration; support multiple languages; flexibility and parametrisation.

Functionality
SOLA Open Tenure

SOLA Open Tenure facilitates the collection of land tenure data with (offline) mobile devices, where the SOLA Community Server facilitates the central storage and moderation of the field collected land tenure records.

Applications comparable to SOLA Open Tenure

A number of applications (i.e. ODK, Cadasta, STDM, and MAST), with functionality comparable to SOLA Open Tenure / Community Server (SOLA OT/CS), have been reviewed in the context of field data collection:

- Flexibility, i.e. the level of parametrisation and suitability for different purposes in ODK, Cadasta, and STDM is higher than SOLA OT/CS (with dynamic forms).
- STDM, MAST, and SOLA OT/CS require specialist expertise to deploy the server on behalf of a community.
- Cadasta is moving away from open source tools and will be based on proprietary software (ESRI); Cadasta takes care of hosting/managing one (1) dedicated server for clients/partners.
- STDM operates on desktop computers via a QGIS plugin, while ODK, Cadasta, MAST and SOLA OT operate on mobile (android) devices.
- MAST does not have installable software and requires software developer expertise to compile and deploy the applications and server software.
- ODK is very flexible, but offers limited support for moderation of the records on a central server, custom development is required to address that.

Customisation vs Configuration

Deployments of SOLA Registry applications involved considerable customisation, i.e. custom development or modification of source code, requiring international developers (i.e. FAO/SOLA staff). Local software development expertise to assist and maintain the deployed system is required and at the same time has proven to be difficult/impossible to obtain in developing countries (with consequences for system sustainability). SOLA OT deployments could suffice with configuration, i.e. fine-tuning of system parameters (i.e. by SOLA staff), but have comprised of customisation exercises as well.

Limit customisation for SOLA Registry

A different approach to consider in deploying SOLA Registry is to minimise custom development instead of stimulating this before initial land information system deployment; for example through increasing the flexibility of SOLA: e.g. configurable workflow, business rules, reports, reference tables.

Host one SOLA Community Server

The flexibility of SOLA Open Tenure could be increased as well (e.g. with regard to workflow, and data to-be-captured), which, together with hosting one (1) community server on behalf of all clients (e.g. various communities with different information needs, similar to the Cadasta platform), could offer a good option for community projects; deployable with little (customisation or configuration) effort and training.

SOLA Open Source (OS) Community

The majority of deployments have been done by international FAO/SOLA project staff, most of which are no longer employed. Development has stopped for SOLA Registry, only SOLA Open Tenure/Community Server is being maintained at a low ebb. The SOLA open source (OS) community is not active. Revitalising it would require funding and governance, to improve functionality and architecture, documentation, and training, as well as for managing and communication/outreach to the SOLA OS community.

Options for future SOLA

The options for FAO with regard to a sustainable future of the SOLA project are to:

- continue governance of the SOLA project by FAO (requiring hiring a software development team);
- to transfer the project over to external entities, for example to a professional body or a private company (requiring improvement of development and deployment documentation and tools before handover);
- or handover to an open source community (requiring the above mentioned improvement of documentation and tools as well as funding for the open source community activities); or
- to discontinue the SOLA project (requiring careful communication to SOLA users).

The most feasible and cost effective option seems to be to transfer the project to an open source community, external to FAO, such as OSGeo, but this needs further discussion and deliberation.

Governance & funding

This option, like most others, requires funding and governance to revitalise the OS community, and also to satisfy requirements that the OS community may have (e.g. accepting contributions, direction by the community instead of FAO, visible activity in the community).

Response to SOLA queries

A checklist for has been defined to assist FAO in responding to the queries with regard to the deployment of SOLA applications. This checklist will create awareness and assist in assessing key factors, to facilitate a successful implementation and long-term sustainability of the deployments of SOLA applications.

In the following sections, SOLA applications will be reflected upon in terms of the functional scope of SOLA and comparable applications; SOLA implementations, customisations and configurations; current SOLA open source community, options for the future of SOLA.

The draft report of this review has been presented at FAO's premises in Rome on the 19th of December, 2018 (see Appendix J for the presentation slides). The comments made before and during this presentation have been incorporated in this final report on the review of the SOLA Suite of Applications.

Jan van Bennekom-Minnema
COWI A/S Denmark
jvb@cowi.dk

1 Functional Scope of the SOLA Suite of Applications

In this chapter the SOLA applications are reviewed in terms of architecture, functionality, data model, competitive solutions and implementations:

- SOLA Registry is a land information system "for land administration agencies that require a secure, robust and transparent tenure registration solution"[1]; also referred to as **SOLA R**.

- SOLA State Land is a land information system that "*assists governments to manage land and property that is owned, occupied or controlled by the state*", also referred to as **SOLA SL**.

- SOLA Systematic Registration is a land information system that "*supports systematic registration activities where tenure information is collected for the first time*", also referred to as **SOLA SR**.

- SOLA Open Tenure is a mobile device application that supports "*in-the-field capture of tenure rights by communities and individuals using mobile devices*", which will be synchronised with SOLA Community Server which provides "*web access to community collected tenure data and manages the processes leading to the community recognition of tenure rights*" also referred to as **SOLA OT** and **SOLA CS**.

Note that the SOLA Web Admin application provides system administration and configuration functions for all SOLA software applications [except SOLA OT]).

[1] http://www.flossola.org/

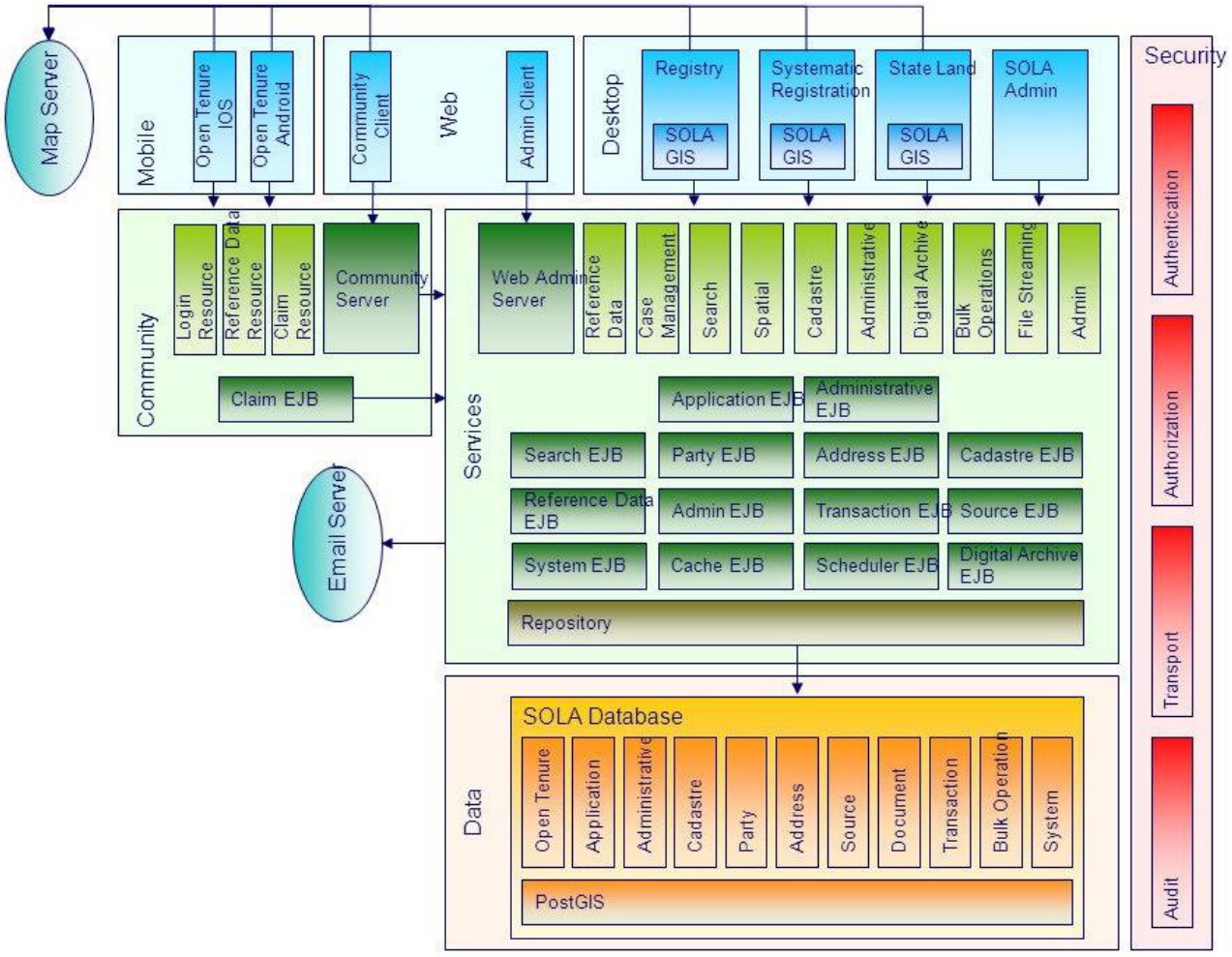
1.1 Architecture

The architecture of SOLA Registry, State Land, Systematic Registration and Community Server are all based on similar components, except for the SOLA OT application on mobile devices (see Figure 1).

Figure 1: SOLA Architectural Elements

	SOLA **R, SR, SL, CS**	SOLA **OT** (Mobile Devices)
Application Server	Glassfish (JAVA EE specification)	Apache HTTP client for API interaction and maps download
Database	PostgreSQL with PostGIS spatial extension	H2 database with JTS spatial extension
GIS server	Geoserver (WMS services)	GoogleMaps SDK for Android
Reports	Jasper reports	Claim report
Security Model	JEE 7 Security Model, Glassfish and Metro.	Android platform + data encryption
Version Control	GitHub	GitHub

Figure 2: SOLA Architecture



The following observations were made with regard to the architecture:

- Initially, system administration was performed through "SOLA Admin Desktop", but this has been replaced by SOLA Web Admin (system administration for all SOLA software applications [i.e. SOLA R, SR, SL, and CS]).

- SOLA OT for mobile devices supported both Android and iOS operating systems; it has now been decided to discontinue development on iOS so only mobile devices with Android can run the latest SOLA OT. Considering the use of SOLA OT in developing countries, this may have less effect, since usually more expensive iOS devices are not as widely used.

- SOLA OT (and SOLA CS) have been developed in a separate setup, instead of re-using the existing SOLA Registry setup, database and components, and use only the necessary business rules (i.e. a stripped down version of SOLA R/SR). For example, instead of (re-)using components Case Management, Application, Administrative for handling claims, parties and rights, a decision was made for creating separate components to handle these. Migration of data from SOLA OT to SOLA R is affected. This is partly explained because SOLA CS in its concept is a web application (and not desktop based) used by communities, receiving collected data from mobile devices running SOLA OT.

Merge SOLA applications into a modular integrated web-based system

- Various reports were made on the (disputed) need for maintaining different SOLA applications and architectures, and recommendations were made with regard to changing and merging the SOLA applications into a modular integrated web-based system, that can be deployed fully or partially as the need arises. This would faciltiate easier deployment, configuration and use of the SOLA Suite.

- During the requirements and design phases of SOLA OT, Open Data Kit (ODK) was considered as an option to provide the functionality of the SOLA OT dynamic forms. The conclusion was that ODK has limitations: "*form based surveys have little flexibility when the number and type of fields and attachments is not known at form design time. Overcoming these limitations would require a development effort reducing the advantage of using an existing framework*". Another limitation of ODK is related to a global view of collected data, allowing the mobile device user to see the parcel being collected in the context of other, neighbouring, collected parcels. The spatial data editor in ODK (at the time of conceptualising SOLA OT) had limited functionality (which meanwhile has been improved in ODK and GeoODK Collect apps). As a result, the functionality around dynamic forms was developed from scratch into SOLA OT/CS.

- The architecture of SOLA R, SR, SL and CS is comprehensive, based on many proven components which make up a robust architecture (for example components for hosting applications, database, GIS/Geo server, report server, etc.). This strength of the architecture also means that deployment, configuration, maintenance and especially customisation

(custom development) will require specialists, experienced and skilled in all these components. It has proven difficult to find and/or train local specialists to perform these functions with regard to SOLA software.

Review lifecycle of SOLA architectural components

- The support (by Oracle) for one of the architectural components for application server (i.e. Glassfish) has become an issue, so a Payara Application Server is being considered to replace that architectural element. There may be other architectural elements that will no longer supported in the near future.

- A SOLA Community Server can only handle one survey purpose; e.g. multiple CSOs for example cannot use the same server (if the survey form/questionnaire needs are different). Only one dynamic form template can be used at a time. To support different CSOs and communities with different data collection needs, multiple physical servers (or multiple virtual servers on one physical server) need to be deployed.

- Development has stopped for SOLA R, SR, and SL (last update 2016); the development of SOLA OT/CS is ongoing, last update in August 2018.

Support developers with development environment setup instructions

- Consider creating, updating or improving documentation on how to setup development environment, as relevant, to assist the open source community.

Support users and developers with SOLA installers

- Installers for the SOLA apps maybe considered for setting up operational and even development environment; see for example the STDM installer taking care of the installation of apache applications server, the geoserver, and the PostgreSQL database, etc.).

1.1.1 Separate Review of SOLA Architecture

A separate technical quality review has been conducted on the systems architecture, the source code, and the existing documentation on SOLA; a final report (i.e. a PowerPoint presentation in pdf format) was delivered beginning December[2]. The main summary was a slide "Assessment Output" (Figure 3) in which the following source code characteristics were evaluated:

- Source code complexity; relevant to ease of maintenance of the source code.
- Hard coding; changes to "hard coded" elements would require customisation of software instead of configuration[3].
- Error handling; handle and communicate run time errors.
- Security
- Source code comments[4]
- Unit test coverage; assisting in (automatically) testing all elements of the source code.

[2] www.zensar.com
[3] https://stackoverflow.com/questions/1895789/what-does-hard-coded-mean
[4] https://medium.freecodecamp.org/code-comments-the-good-the-bad-and-the-ugly-be9cc65fbf83

- Issues reported by Sonar[5]; the result of automatic code analysis.
- Duplication of source code[6]; copy-and-paste development, also relevant to ease of maintenance.
- Critical issues; possibly the outcome of Sonar automatic analysis.
- Overall source code quality.
- Technical debts; relating to the number of resources / person days to correct the all issues.

Figure 3: Assessment Output

	SOLA R	SOLA SR	SOLA SL	SOLA CS	SOLA OT	SOLA WA (web admin)
Source code complexity	High	Medium	Medium	Low	High	Medium
Hard coding	High	High	High	High	High	High
Error handling	Medium	Medium	Medium	Medium	Medium	Medium
Security	Medium	Medium	Medium	Medium	Medium	Medium
Source code comments	Low	Low	Low	Low	Low	Low
Unit test coverage	Low	Low	Low	Low	Low	Low
Issues - sonar	2000+	2000+	2000+	1699	2000+	1019
Duplication of source code	High	Medium	High	Low	High	Low
Critical issues	25	25	26	15	50+	14
Overall source code quality	Low	Medium	Medium	Medium	Low	Medium
Technical debts	High	Medium	High	High	Medium	Medium

Source: Adapted from Sola Suite Final Assessment Report.pdf (5-12-2018, ZenSar)

The technical quality review outlines an alarming condition of the SOLA architecture; the overall source code quality of SOLA is low/medium, and that all SOLA applications require architectural changes. Two recommended approaches are provided and detailed with the technical quality reviewer's resource planning and resources:

1 Technical quality improvement (622 Person Days).
2 Convert to new architecture based on *micro services* (1100 Person Days).

Improve SOLA architecture and source code (extent unconfirmed)

The conclusions of the technical quality review are not or insufficiently justified, for example the review mentions 150 critical issues, but does not provide details about these critical issues, nor are they classified to allow for some prioritisation. The review also generated many critical comments with regard to compliance to the relevant ToR, completeness, correctness & evidence, usefulness, understanding the SOLA application context, original and impartial advice. *Therefore, this "Review of the SOLA Suite of Applications" cannot base or adjust its recommendations on the outcome of the technical quality review (in its current form).*

[5] http://www.sonarsource.org
[6] https://solidsourceit.wordpress.com/2012/08/03/does-source-code-duplication-matter/

Remarks on documentation

- The technical quality review was expected to assess and provide recommendations for SOLA technical documentation, which seemed to have been limited to architecture documents and installation guides.

Improve documentation: requirements, architecture, design, development, installation, system administration, user manual

- Documentation, especially for SOLA R, SR, and SL is found to be a bit sketchy, inconsistent, or too extensive with no clear summary at times: consider the enhancement of the documentation.
- Documentation on SOLA OT / CS is more developed / comprehensive than SOLA R, SR, SL, the focus of SOLA team has been on SOLA OT.

1.2 Functional Scope

The functional scope of the SOLA applications has been assessed in detail, see Appendix G; an overview is provided in the following sections. Figure 4 shows the functionalities and qualities of the different SOLA applications:

Figure 4: Overview SOLA Functional Scope

Green	Available
Yellow	Partly available
Red	Not available

	Registry (SOLA R)	Systematic Registration (SOLA SR)	State Land (SOLA SL)	Open Tenure (SOLA OT / CS)
Functionality				
Workflow management				
› Dispute handling				
Property management				
› Systematic Registration				
› Property valuation				
Spatial unit management				
Map spatial information				
Manage rights and right holders				
Digital document management				
Search, retrieve and display				
Reporting				
System Qualities				
Security				
Maintainability/Flexibility				
› System administration				
› Business rules				
› Survey form templates				
› Multiple languages				
Other				
› Multiple operating systems				
› E-mail notifications				
› Data import/export				
› Offline use of data				

1.2.1 SOLA Functional Scope

The SOLA applications provide functionality with regard to:

- Workflow support; management of applications and services (dashboard).
  - Applications can flexibly be composed of services (Not in SOLA OT/CS).
  - The workflow also comprises of handling disputes/challenges (not in SOLA R).

- Property management; handling properties (basic administration units).
  - SOLA SR offers functionality for systematic registration (i.e. public display maps and lists).
  - SOLA SL offers functionality for property valuation.

- Spatial unit management; managing parcels and cadastral objects (offline when using SOLA OT).

- Use and show integrated spatial (geometric) information (map viewer).

- Manage rights, restrictions, responsibilities and right holders.
- Digital document management; scanned documents, images, photos, categorised and attached to workflow.
- Search, retrieve and display land administration records (on server and mobile device for SOLA OT).
- Reporting; various pre-fined report formats to assist workflow.

1.2.2 SOLA System Qualities

SOLA provides functionality with regard to system qualities:

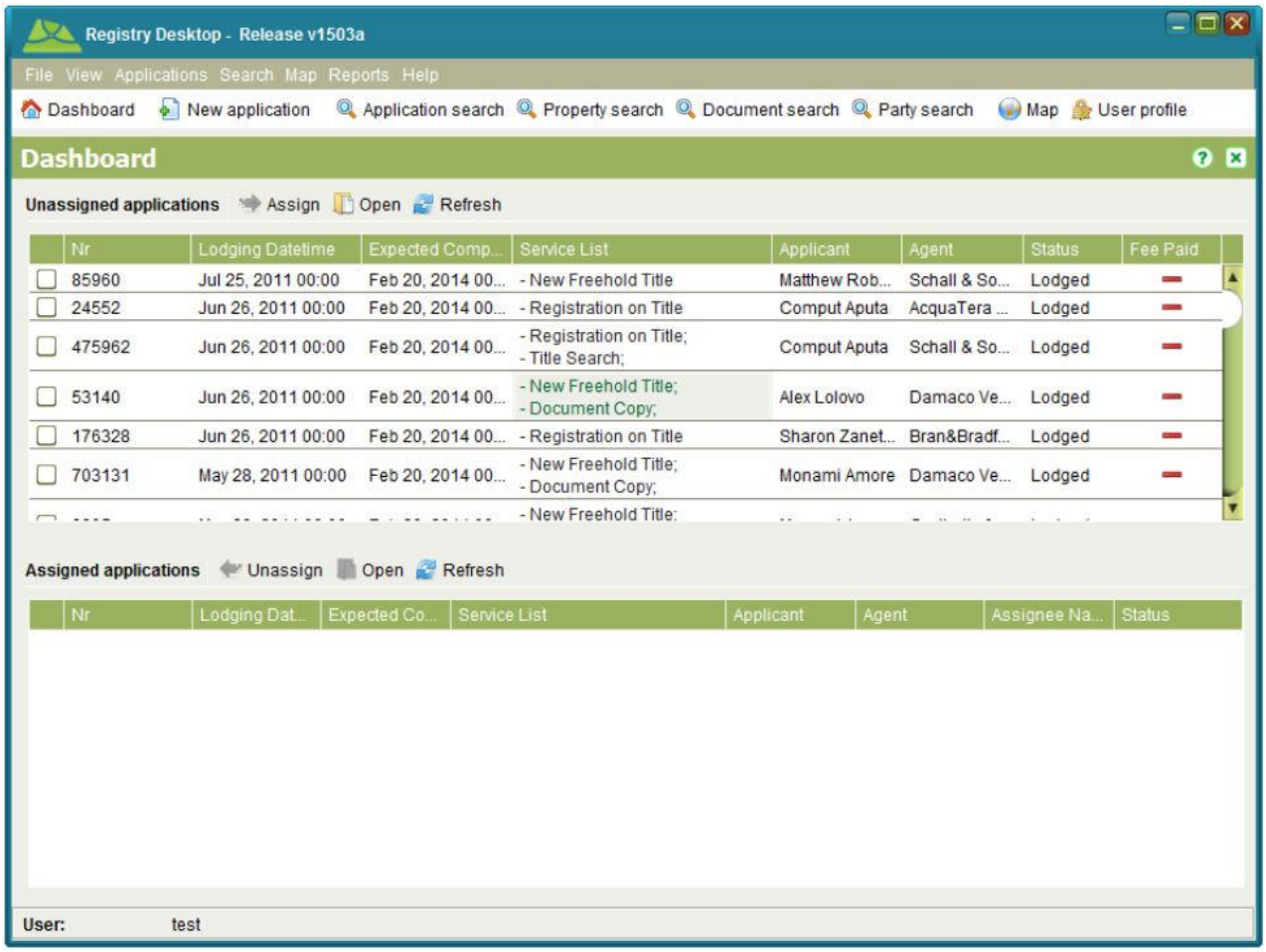
- Security; Authorisation, user authentication & management, record security, auditing and logging, backup.
- Maintainability/Flexibility; system administration, maintain reference data (lists of standards values), localisation support (multiple languages), *parametrisable* business rules (Not in SOLA OT/CS), flexibility survey form templates for capturing (Only in SOLA OT/CS).
- Other system qualities:
  - Server capable of sending e-mail notifications.
  - Support for multiple operating systems (Linux, MS Windows Server, Android [support for SOLA OT on Apple's iOS has been discontinued]).
  - (Bulk) data import/export in various formats.
  - Help function.
  - SOLA OT is capable of downloading off-line data on mobile device (existing claims, attachments, raster and vector data). Syncing and using imagery and vector data is an important capability for developing countries where access to Internet and/or affordable Internet services are an issue.

Observations

- The workflow (for example application status) is not configurable; the workflow setup is not flexible in the sense that workflow (steps) cannot be altered without source code changes. Flexibility in this section (of all SOLA applications) could address variations in local customs and regulations via configuration (limiting the need for customisation, see section 1.4.3).
- As indicated in section 1.1, limited current development is carried out on SOLA R, SR and SL, only SOLA OT is being currently used and developed. The majority of SOLA software architects and developers are no longer employed by FAO. Some updates funded by local initiatives have been done, e.g. SOLA R and SR in Nigeria and Sierra Leone. Futher development with regard to SOLA R and SOLA OT has been requested in Kenya, and with regard to SOLA R in Tonga.

› Flexible / configurable Business Rules ("*separate business rules from application code via a Rules interface*") could be a way to increase flexibility/configurability and adjustment to local situations. Configurable (as opposed to customisable) business rules are and could be a great (improved) feature for SOLA (from SOLA R to CS/OT). Note that in the current setup (SOLA Web Admin), the system administrator has to know the SOLA database structure and must have Standard Query Language (SQL[7]) skills to be able to configure business rules.

Figure 5: Impression SOLA Registry



1.2.3 SOLA Registry (SOLA R)

SOLA Registry is one of the most robust SOLA applications, providing the rigour and completeness required for formal property registration and cadastral functions (and has been tested in a considerable number of projects). The SOLA R and SR functionality is considerable/extensive, estimated at covering 70-80% of the needs of a land registry/cadastre office. SOLA Registry is still operational in a number of deployments, and received positive reviews.

[7] https://en.wikipedia.org/wiki/SQL

Consider e-governance: online delivery of government services

Initially, a SOLA Web Application was envisioned to support deployment as a public counter service available from land administration offices. During the last years, e-governance (electronic governance) has been promoted and supported in some developing countries (e.g. Rwanda). Electronic governance is understood as the online delivery of government services (e.g. application for land registry transactions); exchange of information; integration of various stand-alone systems and services, in a convenient, efficient and transparent manner.

The option of having a link to eGovernment services, citizen One-Stop-Shop services centres (e.g. citizens requesting information or starting applications/transactions via internet) is becoming increasingly popular in developing countries, this could be considered as a future development for SOLA Registry (and perhaps SOLA CS). Note that the architecture of SOLA R with Service Oriented Architecture would support eGovernment Portals and services, which, although mentioned in the original approach of SOLA, has not been developed. Future scaling-up needs with regard to eGovernment services added to SOLA functions could be detailed when considering the merging the SOLA applications into a modular integrated web-based system (section 1.1).

1.2.4 SOLA Systematic Registration (SOLA SR)

SOLA Systematic Registration has only been *deployed once* where considerable customisations have been realised. The overlap in data and functionality between SOLA R and SR is considerable, for example the SOLA R and SR applications are reported to work on identical databases. SOLA SR could be used by companies or institutions engaged in systematic adjudication and registration of field collected land tenure data

SOLA SR, initially did not operate with *mobile devices* used in the field for data collection, nor with offline data. All functionality is offered in a central system capable of (bulk) loading of external survey data and scanned documents to support the adjudication records. It was found that elements of SOLA OT (section 1.2.6) would improve the value of SOLA SR, for example the use of mobile devices to capture field data. Nowadays, it is unlikely, that systematic registration projects are designed around paper based tenure data collection, or that the use of mobile devices is not being considered. In Nigeria, a "bridge" was created between SOLA SR and SOLA OT, so that mobile devices can be used in the context of SOLA SR. A real integration would be facilitated by transforming SOLA SR into a web based system (see section 1.1).

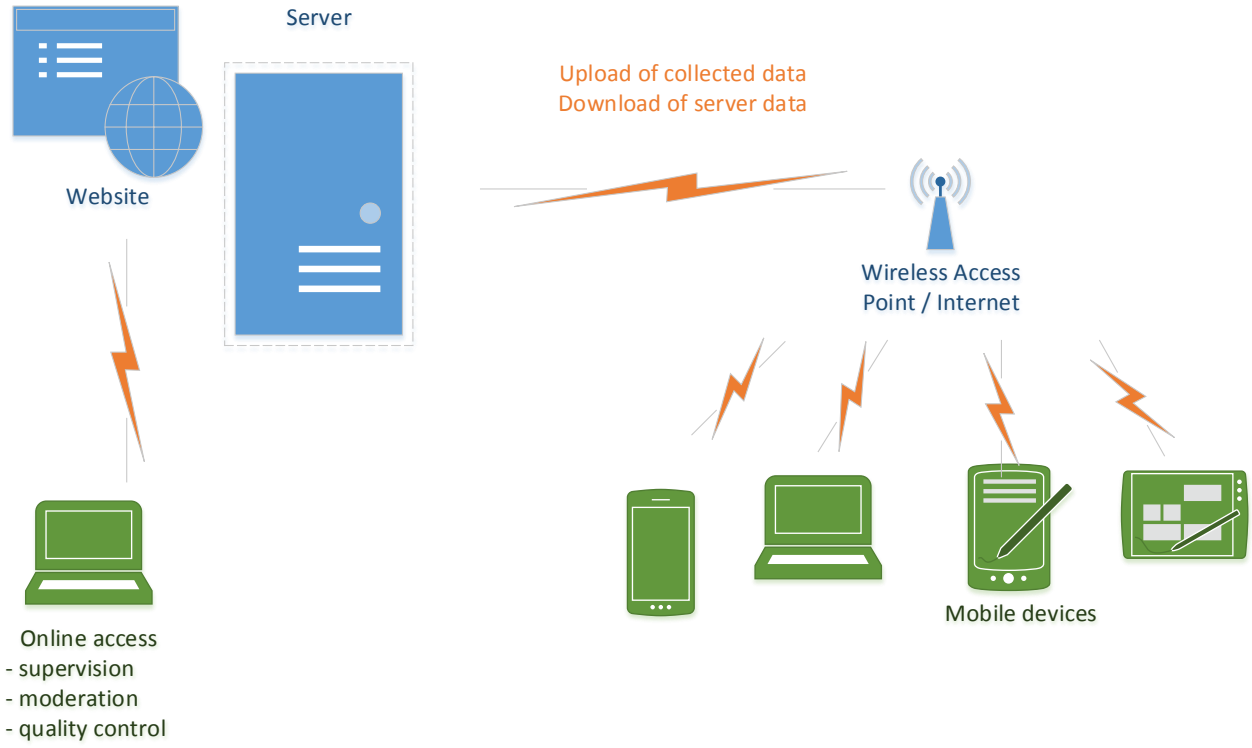
1.2.5 SOLA State Land (SOLA SL)

SOLA State Land has not once been implemented. SOLA State Land was developed because of an anticipated (funding of a) project in Ethiopia, considering rural state land. Considering very tight deadlines, the development was started before an official agreement between the relevant parties was reached. When it was clear the agreement would not be made, SOLA SL was fully developed.

1.2.6 SOLA Open Tenure (SOLA OT)

SOLA Open Tenure[8] is a system for field data collection, focused at parcel based tenure right claims. SOLA OT is designed for communities to do their own community/customary tenure recording and mapping. The architecture consists of a (Cloud) server (SOLA CS, Community Server) which receives information collected (offline) in the field with mobile devices, based on a questionnaire/form (see Figure 6).

Figure 6: overview of architecture for field data collection



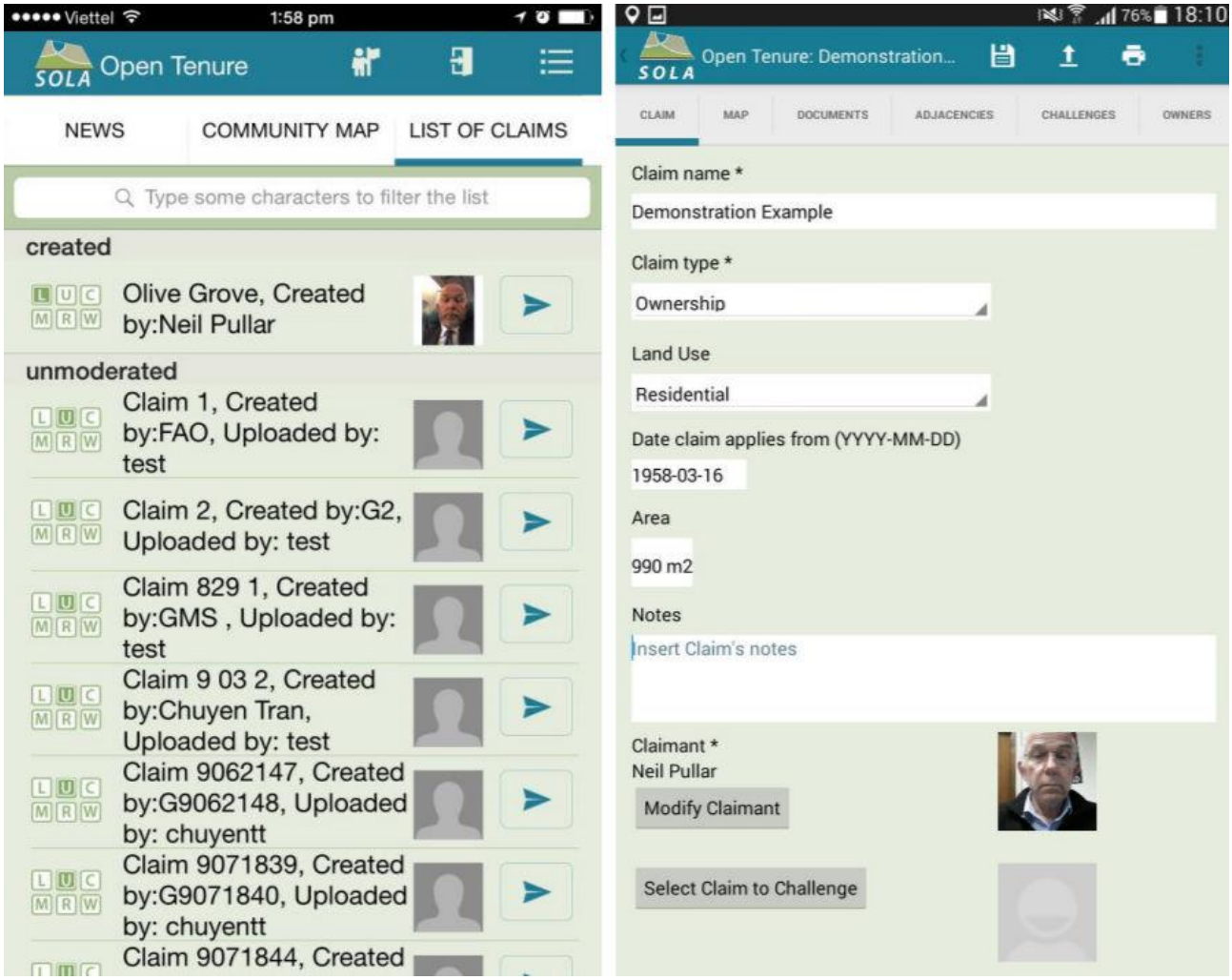
The main components are a data collection app and a central server.

- A data collection app (SOLA OT), is running on a mobile device (mobile phone or tablet), to be used in data collection (the first collection and entry of claims). The data collection app is able to work offline under conditions where internet connectivity and speed are limited. The mobile devices will have supporting offline spatial data available, such as google earth imagery, other ortho(aerial) photos, supporting vector information, such as downloaded open street map data.

- A server, which hosts a central repository (i.e. a PostgreSQL database) with all collected data. The central repository will contain the definition of the information elements and attributes to be used in the field data collection. The server also hosts a web application (SOLA CS): a web site, allowing further handling of the claim, supervision and monitoring, quality control, updates, and management of template forms.

[8] http://www.fao.org/fileadmin/user_upload/nr/land_tenure/OPEN_TENURE.pdf

Figure 7: Impression SOLA Open Tenure

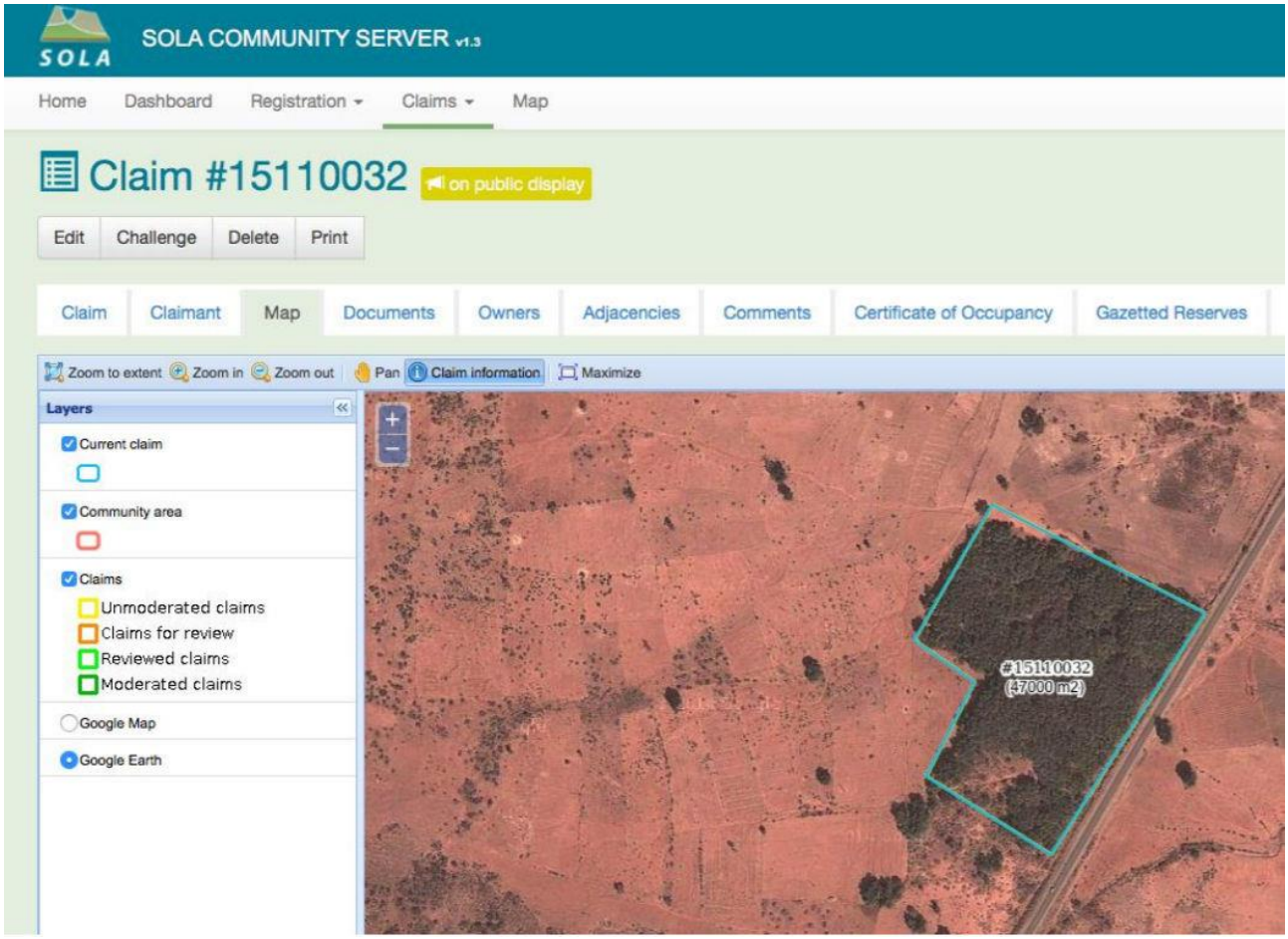


- Open Tenure is designed for *parcel based* tenure right data collection process. The claim form of the Open Tenure app is split into a static part, with "hard-coded"/standard information controlling the logic of the application, and a dynamic part (i.e. dynamic forms), which offers flexibility with regard to specifying other information that should be captured, for a specific purpose or project. With these dynamic forms, the structure of the survey form for the claim can be configured, without the need for changing the source code of either the mobile application or the community server (i.e. customisation). This ability to configure additional information to be captured is common to several data collection applications/toolkits, offered in varying degrees of flexibility.

Improve guidance and validation for SOLA OT forms

- The flow of a questionnaire (to guide SOLA OT users in using the form) cannot be enforced in SOLA OT, especially around the dynamic forms. Validation of forms and tabs/sections in the app (field validation on *out-of-field* navigation, and record validation at *save* actions) could be improved.

Figure 8: Impression SOLA Community Server



Consider SOLA OT functionality with regard to claims for multiple parcels by one claimant

- SOLA OT is parcel based. The process starts with a parcel for which a claim with details of the claimant must be entered. If a claimant claims multiple parcels, claimant data will have to be entered, redundantly, multiple times. Alternatively, the claimant could be taken as starting point and then entering one or more parcels, all claimed by claimant.

Consider dynamic information elements inside the static sections

- The above mentioned static part makes assumptions about the information that always needs to be captured. For example, in SOLA OT, the name of a person consists of FirstName and LastName. When MiddleName (or OtherNames) need to be captured, this requires the definition and entry of this element in the dynamic part (in another part of the app). The option to place dynamic information elements inside the static sections could be considered, instead of only in dynamic sections.

Consider tuple rules

- SOLA OT cannot specify tuple rules / constraints. The content of one field cannot be made dependant on what was entered in other fields (of the same record) previously. For example, when the user in one field selects one of the regions, the next dropdown list cannot be limited to only show the districts within the selected region.

Consider default field values

- Default values for fields are not possible in Open Tenure, for example setting that the gender of a homestead head is male by default.

The workflow for handling claims is "hard coded". Making the workflow configurable to a certain degree would limit the need for customisation of workflow each and every implementation (see section 1.4.3).

Improve source code exception handling

- SOLA OT running under Android 8.1.0 (downloaded 26-10-2018) crashed several times (e.g. at navigating main menu items), which suggest that additional testing could be done, as well as improved exception handling in the source code.

Improve SOLA OT flexibility and capability of "dynamic forms"

- The dynamic forms of SOLA OT is a good start towards configurability/flexibility of the application, but could be improved to bring it to the same level as offered by Cadasta & ODK (see next section 1.2.7).

- Marketing in an open source context is relevant; marketing materials and efforts have been developed and executed for SOLA OT/CS, aimed at land administration and LIS professionals and communities, but less for SOLA R/SR.

1.2.7 Comparative Solutions

The following applications, comparative to SOLA applications (i.e. SOLA OT/CS), have been reviewed:

- Mobile Applications to Secure Tenure (MAST).
- Cadasta.
- Social Tenure Domain Model (STDM).
- Open Data Kit (ODK).

Figure 9 shows the functionalities and qualities of the different applications for tenure data collection:

Figure 9: Overview Comparative Solutions

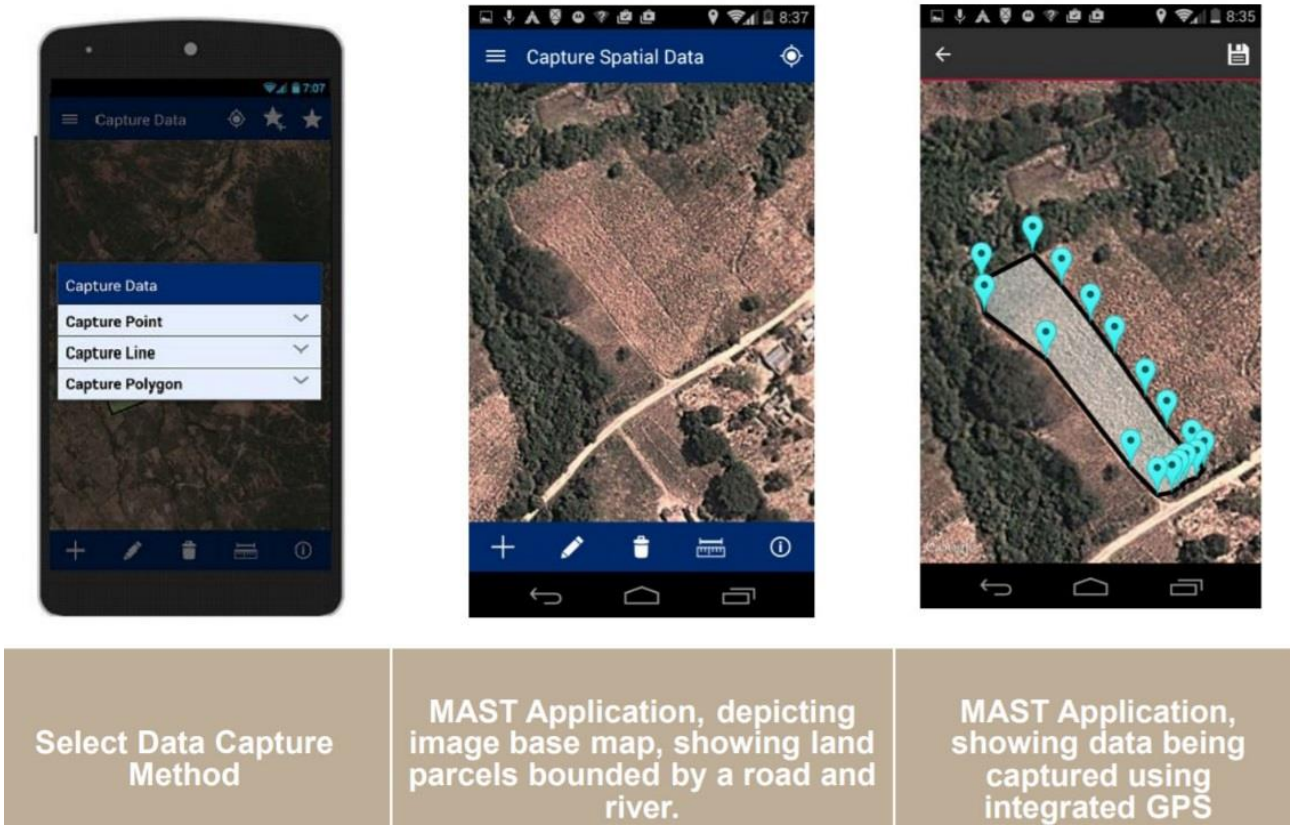
Green	Available	SOLA OT	Cadasta	MAST	STDM	ODK
Yellow	Partly available					
Red	Not available					
Functionality						
Workflow management						
› Dispute handling						
Property management						
› Systematic Registration						
Spatial unit management						
Map spatial information						
Manage rights and right holders						
Digital document management						
Search, retrieve and display						
Reporting						
System Qualities						
Security						
Maintainability/Flexibility						
› System administration						
› Survey form templates						
› Multiple languages						
Other						
› Multiple operating systems						
› E-mail notifications						
› Data import/export						
› Offline use of data						
› Downloadable Installers						
› Host Web data server						
› Support						

- MAST, CADASTA, ODK and SOLA OT perform similar functions based on a similar architecture (server – mobile device synchronisation). STDM offers similar functions but on different platforms and devices (Desktop based).
- SOLA OT has a more sophisticated architecture than MAST and STDM, with much better security and more advanced mapping tools available in the mobile application.
- Cadasta, Mast and STDM describe the use of survey grade GPS receivers connected to a mobile device to improve accuracy of measured coordinates, which are less affordable by communities.
- STDM, MAST and to a certain degree SOLA OT require/rely on a project office and on GIS expertise to advance the data collection work; there is a need for specialists and trainers to set the data collection project and team up.
- Cadasta is hosting one (1) server for all client/partners.
- SOLA, Cadasta and STDM are capable of providing support and training.

1.2.8 Mobile Applications to Secure Tenure (MAST)

Mobile Applications to Secure Tenure (MAST[9]) is an USAID development for tools that use mobile devices and a participatory approach to map and document land and resource rights (see Appendix H.1).

Figure 10 Impression MAST[10]



- MAST software is not available via downloadable installation programs, no demo environment is available. Software developers will always be required to deploy MAST; the sources[11] need to be compiled (for server and mobile app), which requires (senior) software developer knowledge of several technologies (e.g. java IDE, Maven, Android SDK etc.). "*Without knowledge of these it would be difficult for the user to setup the source code and configure to work in debug environment*". The information and instructions on GitHub are limited.

- MAST also provides a simple registration module to administer land records after they have been formalized through a community mapping process: e.g. registration transactions including sales, leases, mortgages, gift, and parcel splits. This registration module is called TRUST, developed for district land offices to manage collected records produced by MAST. Trust is not available in MAST source code on GitHub, it is unclear whether it is available as open source.

[9] https://www.land-links.org/tool-resource/mast-technology/
[10] source http://ansaf.or.tz/wp-content/uploads/2016/05/MAST-PROJECT-FOR-AG-POLICY-2-23-2016.pdf
[11] https://github.com/Mastusaid

- MAST was developed originally for specific project in Tanzania. It was not developed as a generic solution as for example SOLA OT was. Deployments from this version to other countries are based on this Tanzania version. The software architecture has been reported to have issues.

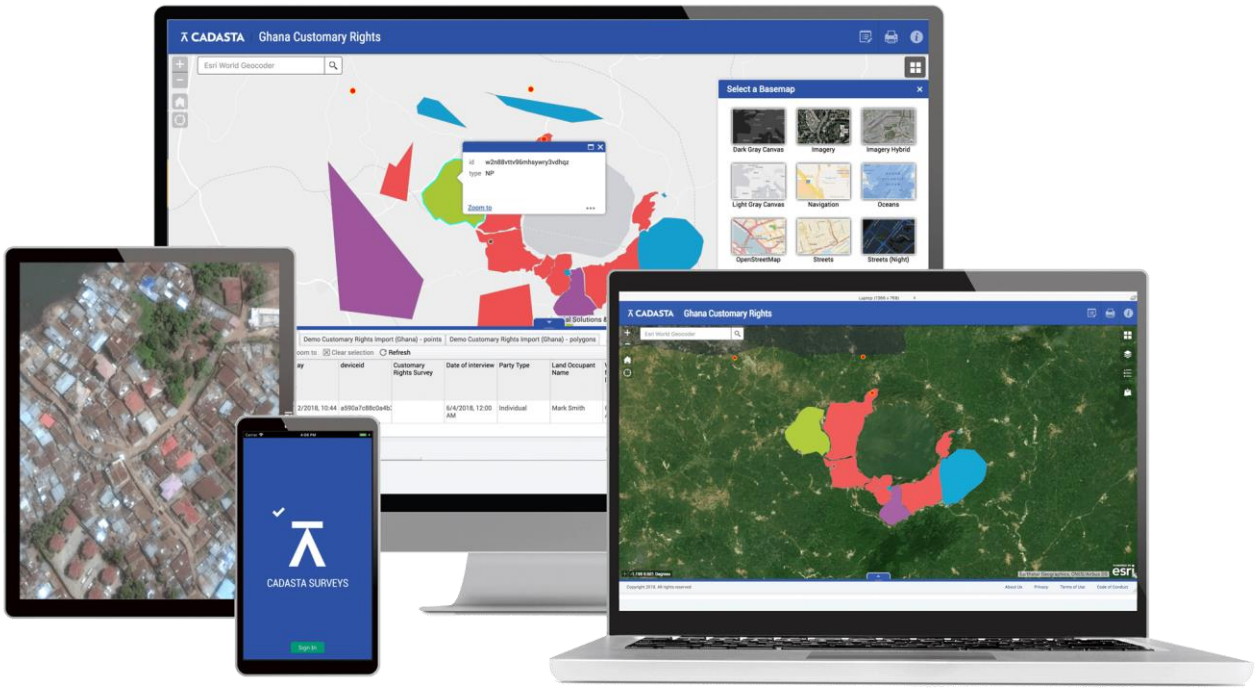
- If a project requires higher GPS accuracy than is available on board a mobile device, MAST can use external GPS devices to capture coordinate locations with more precision; note that this comes with a cost for these external GPS devices.

- MAST is published as open source, but seems to have been just "*dumped on the internet*".

1.2.9 Cadasta

Since 2015, the Cadasta Foundation develops and promotes the use of simple digital tools and technology to capture, analyse and share land and resource rights information, using a full open source stack with tools such as ODK / GeoODK Collect Mobile Application (data collection), QGIS (Desktop GIS) and plugins, PostgreSQL (database). This is labelled the Cadasta Platform 1.0, see Appendix H.2 for a copied description of the Cadasta platform 1.0 elements.

Figure 11: Cadasta Platform[12]



- Since mid-2018, Cadasta 2.0 is introduced, based on the ESRI[13] platform and tools: "*Key learnings from the past two years of working with partners have highlighted a number of needed improvements. To most efficiently address these limitations, we are partnering with Esri…*", and: "*The Cadasta team will be working to onboard all new and existing partners to the Cadasta Platform 2.0 as we begin to transition over to the Esri environment*". Cadasta reported to be currently (November 2018) halfway in making this transition.

- In the context of an approach based on open source tools and open source communities, Cadasta concluded that large scale solutions (e.g. a data collection exercise with hundreds of data collectors) cannot properly be supported by a platform based on open source tools. The uptake, contribution and support by an open source community didn't really materialise (see section 1.4.6).

[12] source https://cadasta.org/platform
[13] https://www.esri.com

- Cadasta issued an RFP to identify a partner for the elements/software for platform 2.0 and consequently selected ESRI. Concerns with regard to data privacy, security, control in ESRIs cloud-based services, have been handled in the agreement between Cadasta and ESRI; the data remains to be the sole property of the relevant partners.

- A license pricing was negotiated with ESRI for their products and services as part of platform 2.0. Cadasta's commitment is to ensure that communities, local NGOs, and customary groups (usually lacking funds) managing less than 10,000 parcels will retain free access and use of the platform. Grants are available to support these groups with training, configuration and customisation services. Organisations with larger datasets and (commercial) project setups, will be billed for a cost recovery of the data storage, and services will be billed at a daily rate.

- Cadasta is available via Google Play store (last update September 2017); technical information[14] last updated July 2017.

- Cadasta is hosting a dedicated webserver[15], physically at Cadasta's premises, offering cloud services for organizations collecting land and resource rights of individuals and communities, and will continue doing that based on the following tools offered (i.e. software as a service [SaaS]):

  - Data Collection Tools:
    - ESRI Survey 123[16]
      - ODK based, only point geometry, white labelled[17]
    - ESRI Data Collector[18]
      - all geometry types, require more advanced mapping skills
  - Use of base maps:
    - Esri Basemaps.
    - Digital Globe satellite imagery.
    - OpenStreetMap.
    - Esri StreetMaps.
  - Project Manager, a web application providing functionality with regard to workflow, data review, quality control, map viewing, digitising, reporting, and exporting data.
  - Operations Dashboard.
  - Advocacy, storytelling on the data with story and thematic maps.
  - New developments with regard to workflow/workforce management and geo-tagged photos is rolled out.

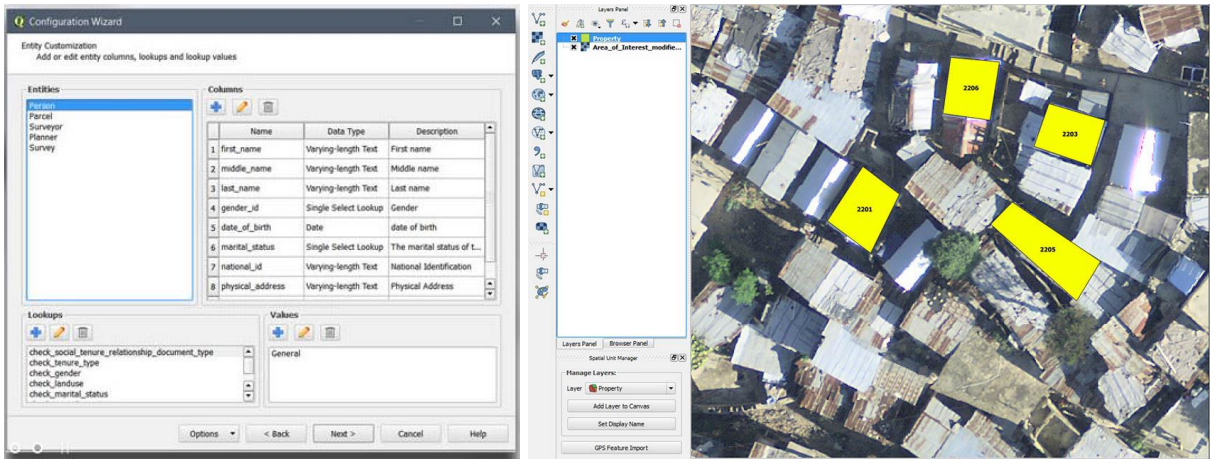
- Cadasta offers support during Cadasta platform project setup and definitions, e.g. training, configuration, special reports, as well as custom development.

[14] https://docs.cadasta.org
[15] https://platform.cadasta.org/dashboard/
[16] https://survey123.arcgis.com/
[17] i.e. branded as Cadasta product
[18] https://doc.arcgis.com/en/collector/

1.2.10 Social Tenure Domain Model (STDM)

The Social Tenure Domain Model (STDM[19]) is an information tool, developed and maintained by Global Land Tool Network (GLTN[20]) as part of their GLTN land tools, "*to bridge the gap between formally registered land and land that is not registered. It is a pro-poor, participatory and affordable land tool for representing a person-to-land relationship along the land rights continuum*" (see Appendix H.3 for more details).

Figure 12: impression STDM[21], a QGIS Desktop GIS plugin



- STDM operates on "client" computers (laptop, desktop) as a plugin to QGIS[22] Desktop GIS (running on operating systems Windows, Mac OS x, Linux); it does *not* operate on mobile devices.
- STDM data is meant to operate on "client" computers with local PostgreSQL database. A central database could be used, but is not part of the standard vision.
- Being a QGIS plugin, STDM is developed in Python program language[23]. Currently, STDM only works on the before last release of QGIS (before version 3); GLTN is working on STDM 2.0 which is a major upgrade that will work QGIS 3.
- A significant degree of flexibility/configurability is offered in STDM.
- Offline data collection is provided through the Open Data Kit (ODK) technology on mobile devices (one of the additions to the latest version).
- Access to external GPS devices is also being considered.
- Activity by open source community seems to be limited.

[19] https://stdm.gltn.net/
[20] https://www.gltn.net/
[21] https://stdm.gltn.net/features/
[22] https://www.qgis.org/en/site/
[23] https://www.python.org/

1.2.11 Open Data Kit (ODK[24])

Open Data Kit (ODK[25]) is a *set* of tools *developed/maintained by University of Washington and a large community of developers, implementers and users. The tools help organizations author, field, and manage mobile data collection solutions*. The architecture consists of a server (ODK aggregate server) which receives information collected (offline) in the field with mobile devices, based on a questionnaire/form:

- ODK is based on (multiple) XLSForms[26] and XForms, an open form standard, to define forms/questionnaires for use in in web and mobile data collection platforms. XForms offer a considerable flexibility in the definition of questionnaire, its questions and their sequence, rules and constraints, default values, and other features that guide and at same time constrain the enumerator.

- ODK is mainly focused at collection of individual questionnaires and offers more flexibility in that than SOLA OT and MAST, Cadasta is partly based on ODK. ODK is a set of tools that can support many distinct use cases, SOLA OT/CS is tailored towards one specific use case: the parcel based field data collection and *moderation*.

- ODK primarily has a focus on initial, first registration (moderation capabilities in ODK are limited). However, the open source community seems to be heading towards a full bi-directional information exchange between server and mobile devices.

- The training of enumerators, operating the mobile devices in the field, is minimal (because of flexibility in designing guiding forms/questionnaires).

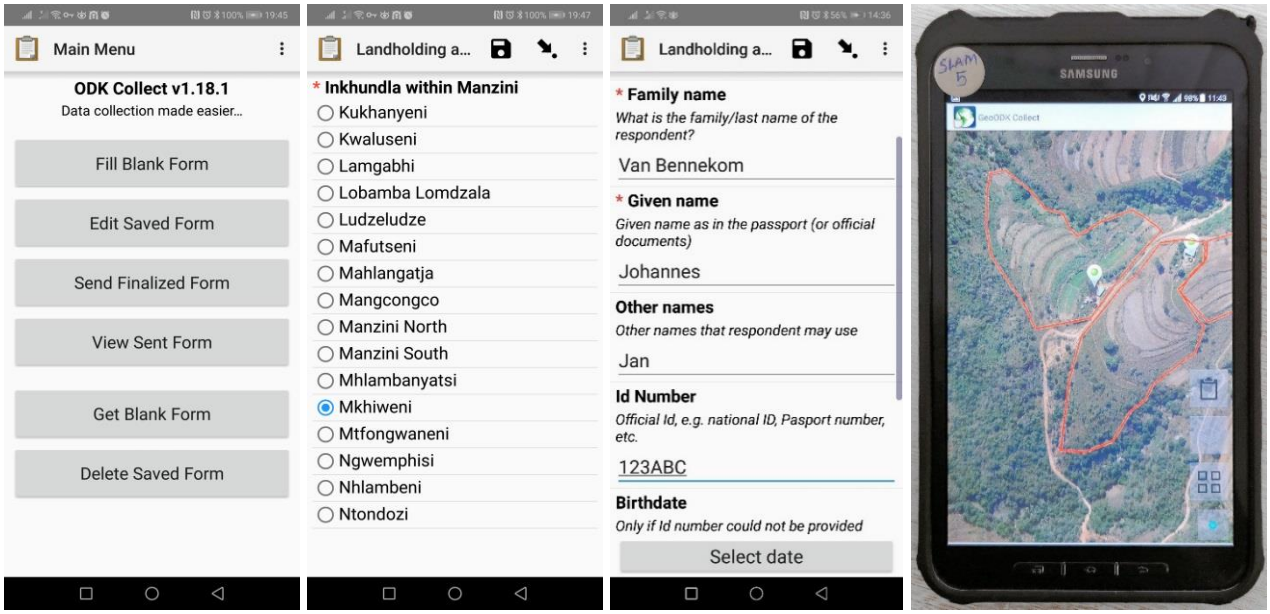
- Specialist expertise in setting up the environment (hosting server, database, edit environment) and defining the questionnaires (in XLS and XForm format) is required.

  ODK tools are only supported on mobile devices with Android operating system.

- ODK has a large, seemingly active, open source community, used in many products and projects (e.g. in STDM, Cadasta, ESRI's Survey 123, FAO's Pan Africa Bean Research Alliance (PABRA) program[27]).

[24] https://docs.opendatakit.org/collect-intro/
[25] https://opendatakit.org/
[26] http://xlsform.org/en
[27] http://www.fao.org/3/I9162EN/i9162en.pdf

Figure 13: Impression ODK and GeoODK Collect (Eswatini/Swaziland)



Example ODK implementation

In Eswatini (Swaziland) a deployment of open source ODK tools was done to collect data on the basis of a questionnaire with 6 sections and 50 information elements on homesteads and households, with several geometries (points and polygons) on homestead fields and community areas. The mobile devices, deployed with GeoODK and ODK Collect (operated by local field teams) are synchronising field collected data with a central server (i.e. an ODK Aggregate web application using a PostgreSQL database). The data clean-up and quality control was performed by local GIS operators, with QGIS Desktop GIS operating on the PostgreSQL database:

- The complete (specialist) setup and deployment of ODK tools was done with 2-3 weeks setup (involving customisation and configuration of questionnaires by senior international IT expert), and 2 weeks guidance/maintenance of the architecture.

- Both ODK and SOLA OT were evaluated by the client, and a choice was made for ODK, mainly because ODK was found to be more flexible to exactly implement the client's requirements.

1.3 Data Model and LADM Compliance

The data models of the SOLA applications have been assessed in the context of the Land Administration Domain Model (LADM, ISO 19152:2012). The SOLA data models are all at least Level 1 Compliant with LADM. Some of the packages are Level 2 or Level 3 compliant (see Figure 14 below). The details of this assessment are provided in Appendix E.

Figure 14: SOLA applications' compliance with LADM packages

	Compliance 'Registry'	Compliance 'Systematic Registration'	Compliance 'State Land'	Compliance 'Open Tenure'
Party Package	Level 1	Level 1	Level 1	Level 1
Administrative Package	**Level 3**	**Level 3**	**Level 3**	Level 1
Spatial Unit Package	**Level 2**	**Level 2**	**Level 2**	Level 1
Surveying and Representation Sub package	Level 1	Level 1	Level 1	Level 1

Consider increasing LADM compliancy for Party Package

- SOLA is Level 1 Compliant with LADM.
- The SOLA implementation of the LADM has added and extended the LADM significantly.

Merge and improve data models (and improve documentation) to support merging of SOLA applications

- Data models for the different SOLA applications seem to have inherited, elements of other SOLA applications, in other words there is large overlap in different data specifications (see Appendix F). For example, the data model for SOLA registry contains sections (schemas) for administrative, application, cadastre, document, party, transaction, but also for *opentenure*. The data model for Community Server (Open Tenure) contains fields that are intended for systematic registrations. This may be related to manner in which SchemaSpy[28] software was used to document the database. Scrutinizing the data model in the context of the individual SOLA apps is not straightforward with the provided data models. The documentation on data could be improved with additional information on how the data is maintained and used by which SOLA applications.

1.3.1 SOLA Registry

- It should be noted that SOLA has added and extended the LADM significantly, especially in the area of workflow, applications, transactions and services, (digital) documents, system & bulk operations. SOLA has defined (alternative) data model implementations in the SOLA database; for example the implementation of party (right holders) members and groups of parties involved in rights.

[28] http://schemaspy.org/

- Modelling processes and transactions was considered outside the scope of the LADM Edition I (2012), because they were considered to be country specific, and generic processes were expected to be too difficult to model. It should be noted that the authors of LADM are now reconsidering this view given developments as Fit-For-Purpose land administration, mobile device apps and block chain[29]. Standardised administrative and technical update processes, are envisioned to be addressed LADM Edition II.

- Some of the entities, are labelled to indicate an implementation of/link with a certain LADM class, and at the same time indicate that this SOLA entity is "Not used by SOLA" (for example group_party, party_member, building_unit_type, legal_space_utility_network, spatial_source_type). Those classes/tables and their relationships were generally created and left initially unpopulated and without associated software in case, in future customisations they would be required. In modelling SOLA, the most generic form of implementation has been used (level 1). *In assessing LADM compliance these tables were not assessed as Conformant*.

1.3.2 SOLA Systematic Registration

- SOLA Systematic Registration is reported to work on exactly the same data model as SOLA Registry.

1.3.3 SOLA State Land

- The data model for SOLA State Land resembles the SOLA Registry datamodel, but has considerable additions in the application area (related to public display, notifications, and disputes). Unlike registry data model, there is no schema for opentenure.

1.3.4 SOLA Open Tenure / Community Server

- For SOLA Open Tenure / Community Server, a separate, different structure has been setup for open tenure (schema opentenure) where elements of SOLA registry could have been used (re-using data, functionality, and business rules), avoiding the need of data migration, for example from Open Tenure to SOLA Registry (see section 1.1 on architecture on the same).

- While at application level a distinction is made between Open Tenure Mobile App and Community Server, this distinction at data model level is not clear.

- The data model Community Server contains the schema "opentenure" but also the schemas known from SOLA Registry and Systematic Registration (administrative, application, cadastre, document, etc.).

[29] Lemmen CHJ, Unger EM, van Oosterom PJM, Kalantari M. Exploring Options for Standardisation of Processes and Transactions in Land Administration. 2018. Paper presented at 19th Annual World Bank Conference on Land and Poverty 2018, Washington, United States.

1.4 Implementations

The implementations of SOLA applications (and customisation vs. configuration) will be discussed in this chapter; focusing at the use cases and types of projects that the SOLA applications facilitate:

- SOLA Registry. To perform core, sporadic cadastre and property registration *processes*.
- SOLA Systematic Registration. To perform systematic first registration *processes* to collect, publish and confirm land rights and right holders.
- SOLA State Land. To perform mass valuation and State Land administration *processes*.
- SOLA Open Tenure. To perform tenure claim data collection on (online and offline operated) mobile devices, which will be synchronised with the SOLA Community Server; facilitating community access, review, editing of field collected claim data.

The SOLA applications are land information systems (LIS) being used in two types of projects:

- Information System Development. The development and deployment of a land information system on behalf of a beneficiary in developing countries, e.g. SOLA Registry for land registry offices.
- Field collection tenure data. Assist beneficiaries in developing countries with the field collection of parcel based land tenure data in target areas, with help of a land information system, e.g. SOLA OT & CS for communities, local CSOs, and customary groups.

1.4.1 Project Phases

There are many ways (based on Information System Development Methods, ISDMs) in which information system development and deployment projects can be structured and described; for the purposes of this review, the following phases will be considered in assessing the deployment of the SOLA applications for both of the above mentioned types of projects: information system development and field tenure data collection.

As input to assessing cost effectiveness and feasibility, an indication of the resources and tasks required for implementing SOLA applications and the selected comparative solutions (see section 1.2.7) has been provided in Appendix I for each of the considered project phases: project orientation, project definition, design and development (customisation), deployment (configuration), and sustained operation.

Orientation

A phase in which the feasibility of investing in a land information system development project or performing an field data collection effort is being assessed; establishing needs and availability of budget, funding, involvement of institutions, organisations and individuals, local laws and regulations, other resources. This phase would typically result in a feasibility report and initial terms of references.

Definition

Following or part of a feasibility study, the (functional and non-functional) requirements for the land information system or for the field data collection effort are specified as well as an estimation (and allocation) of the required resources to implement those requirements. This will describe equipment, responsibilities of local institutions, but also involvement and training of local specialists (e.g. software developers, data collectors/enumerators).

Development

Based on the requirements and available resources, the design and development of the land information system is performed. For example, a well-designed data structure of the data to be captured and/or maintained. The development may consist of customisation (i.e. custom development of source code) and/or of configuration / parametrisation of parts of the system without the need for changing and compiling source code. When deploying an LIS, customisation is likely to consume a considerable part of budget; when deploying a system for field data collection, the focus (and budget) will be more on configuration than on customisation.

Deployment

The deployment phase will comprise of activities to prepare for operation: testing and fine-tuning, conversion of existing data, prepare base data (e.g. imagery), possibly parallel / shadow run of new and legacy LIS, initial and scaled-up installation, training LIS users, data enumerators, and system administrators as needed, outreach / communication to users, citizens, beneficiaries.

Sustained operation

After successfully performing the deployment activities, with LIS projects, a phase of maintenance and support will start during which activities with regard to quality control, maintenance, defect/"bug" fixing and change requests may be performed. In field data collection, this will be the most significant phase, in terms of resources and duration, in which the deployed data collection system (server and mobile devices) will be used to collect tenure records, and make these subject to the defined data collection workflow including quality control.

1.4.2 Project Roles

During implementation of land information systems and field tenure data collection, the following roles may be relevant:

- Private sector. International and /or local companies (and individuals).
  - Responding to (international) tenders to implements comprehensive IT/LIS (land information system) solutions: The SOLA applications Registry and State Land are typically implemented within government, for example in Land Offices under a responsible ministry. The implementation of such systems often occurs with help of private sector: private sector companies take on the responsibility for the complete system integration.
  - Private sector involved in consultancy in specific information system development phases: conducting (smaller) assignments to provide expertise, advice and services with regard to LIS development, in one or more of the phases: orientation, definition (e.g. feasibility study, ToR), development (e.g. QA/QC for IT architecture, software testing), deployment (e.g. training. hosting servers for web applications).
  - Private sector managing (commercial) projects and project offices dedicated to field collection of land tenure data, possibly with SOLA applications (e.g. SOLA OT, CS, SR).
- Civil Society Organisations. Usually local, often voluntary organisations without significant government controlled participation, pursuing interests in the public domain, for example community based organisations with an interest in environmental issues, gender rights, and land tenure rights.
  - Assisting local communities in documenting customary rights, i.e. field collection of land tenure data, with "fit-for-purpose" information systems, possibly with SOLA applications (e.g. SOLA OT, CS). Typically with limited budget for resources and high participation of local community members (relatively lower computer literacy) to perform data collection tasks.
- Governmental institutions. For example land offices/agencies responsible for management of land and property registration, cadastre information.
  - Subject to the implementation of a comprehensive LIS solutions to establish formal registries.
  - Recipient of field collected and moderated tenure data deliver by dedicated project offices, either the result of law based systematic first registration or of a special data collection exercise.
  - Recipient of community collected and approved customary tenure data, to be considered in formal systems.

- Universities. Providing education and training with regard to technologies and activities with relevance to SOLA applications.
  - For example: open source technologies (system development, java programming, application/report/geo server), project management, etc.

1.4.3 Customisation vs Configuration

During the project phases development and deployment (section 1.4.1), two types of adjustment of the land information system to the requirements occur: customisation and configuration:

- Customisation. The creation (programming), modification and compilation of source code, requiring software developers' expertise.
- Configuration. The configuration of system parameters, without the need for changing and compiling source code, however, requiring knowledge of the (administrative parts of the) system.

With SOLA Registry implementations, customisation of software; i.e. the creation, modification and compilation of SOLA source code has been encouraged, supported and executed by the SOLA development team. SOLA Registry software customisation was deemed inevitable because each country has different legislation for registration and cadastre etc. and a different legacy of land records to be dealt with.

Customisation requires international software developers to lead the source code modification, as well as to train local software developers; these developers would execute part of the source code development and would maintain the system when in operation.

For SOLA Registry & SOLA Systematic Registration, the (rough) estimated average custom development ranged from 5 – 10 working months for local software developers and an average involvement of 1 working months of an international SOLA developer, aimed at custom development of:

- Digital capture of certificates, integration with Open Tenure, digitalization of existing paper based maps, customized certificates (Nigeria).
- Integrated Cadastre & Registration system with Digital Archive (Samoa).
- Include Lease management (Lesotho).
- Lease applications, land grants (Tonga).
- Customized Survey Plan, support for coordinate systems (colonial, UTM 28, UTM 29), manual data entry, map viewer switching between different coordinate systems (Sierra Leone).
- Flexible workflow, additional checklists to verify the completion of application and services (Ghana).
- Nepali language, Nepali calendar (Nepal).
- Improvement of generic SOLA source code.

SOLA Open Tenure implementations, *usually* comprise of system configuration, during which configuration / parameterisation of the OT and CS options are being fine-tuned, for example with regard to geoserver layers of available information, standard reference lists and the dynamic forms for specifying extra attributes.

SOLA Open Tenure typically operates in situations where software development/deployment expertise is unlikely/unavailable (for example in communities); this system configuration is usually conducted by SOLA staff during project definition/deployment. However, there have been SOLA OT implementations which also involved *customisations*.

For SOLA OT/CS, the (rough) estimated average for custom development ranged from 1 working month for local software developers and an average involvement of 5 working months of an international SOLA developer, aimed at for example:

- Migration of data from SOLA OT to SOLA R and SR (Nigeria).
- Administrative boundaries demarcation, print maps at different administrative levels (Sierra Leone).
- Customized certificates and reports (Uganda).
- Alternative workflows management (Angola).

Comments SOLA R

In almost all SOLA reviews and implementations it is made clear that availability, commitment and funding of local software developers, capable of customising SOLA and maintain the software over time is key to the success of the SOLA R/SR implementations. Unfortunately, qualified/skilled (java) developers of a certain, required, calibre are hard to find, allocate (afford) and retain, and therefore most implementations depend on international SOLA developers' input. A conclusion is that this local support during development, deployment and operation is critical to sustainability of the land information system, and should be condition *sine qua non* to deployment of SOLA registry.

Reconsider the approach to customise a system 100% of the local requirements before deployment: deploy SOLA as is

With regard to SOLA R implementations, the goal to customise a system 100% to the local requirements, processes and regulations, at first deployment, should be reconsidered in future implementations. Typically a deployment of an LIS occurs in situations where users are not used to digital information systems and software, they have no experience is drawing up requirements for a LIS. Beneficiaries are used to ways of working (their business processes) which are based on analogue methods, which will change or even be obsolete when a digital system is deployed. Attempting to holistically specify requirements to address all (existing) business processes and perform customisations to implement all of these, at the beginning of the project, bears the risk that beneficiaries and users, gaining experience with digital systems, will see the need to change the requirements during operation or even during customisation:

- Consider improving SOLA Registry to cover 70-80% of expected (fit-for-purpose) functionality and focus at deploying SOLA R *as is* (e.g. limit the customisations). Define a certain period of operation and then after this period do a requirements specification (now that users are more knowledgeable), a business process redesign and determine if a customisation is required and feasible. See also the comments on LADM Edition II (section 1.3.1).

Improve with flexible workflow and business rules

- Consider improvement with regard to the workflow engine (applications, services, tasks, claims, statuses, etc.) as well as the business rules in terms of making these flexible and configurable. This could increase the flexibility to adapt/configure to the local processes and regulations, and decrease the need for customisation.

- Some of the customisation by SOLA international staff was related to customised certificates and other reports. These could have been done with local experts with knowledge of SOLA data model and Jasper Reports[30]?

Merge SOLA applications

- As future improvements, the integration of the SOLA applications could be considered: SOLA OT/CS could, in a standard configuration, be made to work together with SOLA R/SR. possibly with a bi-directional message/migration tool to exchange data. SOLA OT, running on mobile apps, for collecting initial data, then linking it to SOLA Registry to continue maintenance of property rights. The maintenance of community records after initial collection and moderation could be offered through SOLA R (light). This would be similar to what MAST is offering (section 1.2.8): to have a simple light registration system to maintain data after initial collection. Consider also comments made in the context of architecture with regard to merging SOLA applications into a modular integrated web-based system.

Comments SOLA OT/CS

Host one (1) community server serving multiple purposes

The SOLA OT app can be used by local communities after some training, however, hosting and maintaining a server would in most cases be not feasible /sustainable; support by SOLA in hosting these community servers would facilitate (and stimulate) the use of SOLA OT: Consider adapting the SOLA community server to allow the hosting of one (1) community server serving multiple communities with different information needs (one server deployed, for example hosted within FAO IT environment or on third party servers). Thus, only one server would need to be *maintained*, facilitating many different communities with different information needs. This setup could also be feasible for private sector / land offices engaging in field data collection exercises of a considerable size. During the presentation of this review to FAO staff, comments were made stating that this would already now be possible with limited or no customisation.

[30] https://www.youtube.com/watch?v=yRLvJgz9Dxk

This approach is used by Cadasta (see section 1.2.9) and by many others. Another similar example, used for field (spatial) data collection in Malawi is Gis4Mobile[31]. Gis4Mobile allows clients to setup their questionnaires and user profiles in a flexible but simple manner, and then, with the app on mobile devices in off-line mode, data can be collected and then uploaded to the Cloud server, physical located in Denmark. Data can be viewed on the webserver and downloaded to other systems and databases.

1.4.4 SOLA Registry in Lesotho

In 2012, the Land Administration Authority of Lesotho, required a land administration system, particularly to handle records that where delivered by several systematic registration projects. SOLA was considered for a "lease management system" and it was found that 60% of the specified local requirements was offered through the standard functionality of SOLA Registry (requiring only configuration). Of the remaining 40% of specified requirements, 20% would require customisation of *existing functions* and 20% would require *new source code development*. According to the Director General involved in this deployment, beginning 2013, SOLA had been installed and customized to *fully* meet LAA's lease management needs, with the help of FAO oversight and a team of 4 local contract software developers, and functioning successfully and satisfactory in an operating environment.

1.4.5 SOLA OT in Myanmar

SOLA OT was also used for data collection in 8 rural communities in Myanmar (2016), managed by a local non-governmental organization called Tharthi Myay Foundation (Peaceful Land, the Myanmar office of Partners Asia[32]). This was done with support by former SOLA team members aimed at workshops, training (e.g. user, system admin), localisation to the Burmese language, and remote e-mail based support. This support was limited in size, compared to other SOLA OT implementations (section 1.4.3), with no large customisations; due to a NGO that was reported to be providing good management and well-prepared guidance during the deployment and operation of SOLA OT.

[31] http://gis4mobile.dk/en
[32] https://partnersasia.org/grantee-highlights/commitment-to-local-leadership/

1.4.6 Open Source Community

FAO has attempted to build an open source community for the SOLA applications, with limited result. The case study report "Community Engagement Strategy (CES) for the Sola Open Source Software" [2018, Andrew McDowell (one of the SOLA developers)] summarises that *"Unfortunately, the open source community supporting SOLA never flourished and with insufficient sponsorship, the viability of the project is now under threat unless the OS community can be inspired to intervene".*

SOLA has never experienced significant contributions by developers to its open source project. This does not necessarily mean it is unsuccessful; it may indicate that the SOLA applications are mature and satisfy the requirements of the institutions that have deployed SOLA. However, if the number of deployments have decreased significantly, this could indicate that interest is fading, and even that the open source project is *dead in the water*. This seems to be the case for SOLA Registry, and may be the case for SOLA Open Tenure.

The CES describes recommendations to energize and grow the SOLA open source community. These recommendations can be summarised as:

Consider CES stage 1 recommendations on governance, sponsorship, funding

1 Simplify the SOLA architecture with light weight components to facilitate development and deployment.

2 Engage and communicate with open source community through social media.

3 Encourage universities to adopt SOLA applications in their land administration curriculum.

4 Offer online accessible SOLA training for development, deployment and operation.

5 Improve SOLA documentation for development, deployment, operation, support and promotion.

6 Increase visibility of SOLA open source community and applications at land administration related events.

7 Establish governance of the SOLA open source community/project.

8 Ensure funding for open source community/project (e.g. FAO, corporate sponsors).

Consider CES stage 2 recommendations on simplifying architecture, improve communication, training, documentation

The CES identifies the transfer of the project to OSGeo (which was the original intention of FAO) where recommendation 7 and 8 on governance, sponsorship and funding are deemed to be key in a first *viability* stage to the likelihood of the SOLA project enduring (and capable of transferring to OSGeo). Stage 2 would focus at other listed recommendations, and could start if stage 1 is successful.

An important consequence of bringing the SOLA project under OSGeo is highlighted in the CES which would be that the community would decide on the direction and enhancements of SOLA (for example put all focus and energy in only one of the SOLA applications), and this should be seen as a sign of a healthy open source community doing its work.

Internet research reveals a lot of information on the topic of successful open source software communities; no academic research has been done into these as part of this review, however, it is clear that most open source projects are not successful and will be abandoned; a few key factors seem to be dominant, for example to consider whether a community should be further supported/funded. A phrase often encountered/copied "*Software development is hard. Running a large open source project is even harder*":

Consider key factors for successful open source software communities

- Funding foundations and companies are significant in the success of open source communities.
- A vision and clear objectives and structure for the open source project and effective communication (marketing) though websites, forums, training, documentation, active bug-tracking system are key for success.
- Modular software architecture to facilitate normalizing and simplifying development tasks.
- Development should be inspired by focusing at user needs.
- The user/installed base does not have to be large, as long there is a clear need for the software by a clear group of users.
- Involvement of a group of developers interested in continuing the use and development, with open source experience.
- Larger communities are more responsive e.g. with regard to fixing bugs, adding new features, etc.

2 SOLA Road Map

In this section the options for the future business/service model for the SOLA applications are discussed (section 2.3 to 2.7):

- Governance by FAO.
- Governance by Professional Body.
- Governance by Consultant.
- Governance by Open Source Community.
- Discontinue SOLA project.

In describing these options, a distinction is made between the two types of projects (i.e. addressing the 2 categories of use cases) that deploy SOLA applications, as mentioned in section 1.4:

- Information System Development (section 2.1).
- Field Collection of Land Tenure Data (section 2.2).

Section 2.1 and 2.2 will describe elements that will have to be considered, regardless of the chosen option: governance by FAO, professional body, consultant, open source community.

2.1 Information System Development

The SOLA application SOLA Registry (and possibly SOLA SR, SL) will be customised and deployed for land registry offices to support management of land tenure records, in information system development projects with considerable customisation and other services (project management, feasibility studies, training, etc.).

Remarks

- Reviews and initial responses with regard to SOLA open source (which in the beginning comprised of SOLA R, SR, SL) were positive, SOLA had a great momentum, but interest from the professional field seems to have died out. The open source community is currently not active; SOLA has not been implemented by others than by (previous) members of the FAO development team, and no ongoing projects or indication of interest in the future deployments.

- Previously, SOLA knowledge was solely available within the SOLA project team. Development resources would have to be allocated and possibly trained as part any of the below options.
  - Software developers (java, etc.).
  - System administrators (i.e. the various server components in use).

- SOLA architecture and sources may also have to be improved to assist other developers to contribute (note that the technical quality review of did not provide a final and accepted conclusion on the state of the architecture, source codes and documentation: section 1.1.1).

- Fixing known bugs.
- Improving documentation.
- Improving source code (and inline documentation).
- Develop functionality (installers).

Improve / simplify SOLA architecture

- SOLA architecture and functionality may also have to be improved (simplified) to promote and stimulate uptake of SOLA by users and developers, for example:
  - No-customisation "as is" configurable deployments.
  - Merge & modularise SOLA applications.
  - Hosting 1 core community server.
  - Make workflow configurable.

- Local development resources are a condition for deployment and maintenance of the LIS during operation.
  - Software developers.
  - System administrators.

2.2 Field Collection of Land Tenure Data

The SOLA applications SOLA OT & CS, configured and deployed for communities and local CSOs, are deployed to support collection of (customary) field data. This includes hosting of a community server on behalf of community servers, and limited or no customisation, but with services with regard to hosting, training, configuration, and support.

Remarks

- A hosting environment is required for the community server.
  - Server running database and website(s).

- Resources need to be available and trained.
  - System administrator required to maintain server and applications.
  - Trainers / team leaders to train SOLA OT users.
  - In case of customisation; trained software developers are required.

- Interest for SOLA OT seems to be higher than for SOLA R, SR, SL, addressing increasingly popular focus on pro-poor, fit-for-purpose, open & transparent land information tools.

2.3 Governance by FAO

Since 2010, the SOLA project has been under governance the Land Tenure and Management Unit (NLRA) of the Food and Agriculture Organisation of the United Nations (FAO). Funding for the initial open source project, and the subsequent implementations was provided by various sources, e.g. the Finnish Government for initial project, and the UK Department for International Development (DfID), Millennium Challenge Corporation (MCC) for country implementations and customisations). FAO could decide on providing continued/renewed support and governance of the SOLA project, addressing the following remarks on governance, funding and resources.

Remarks on Information System Development (section 2.1)

- SOLA developers are currently not employed/available.
- Funding to provide for sustaining the support of SOLA has been indicated as essential. E.g. hiring minimum (part-time) development team to fix bugs, improvement of documentation, enhancement of functionality (routine software maintenance and support), promotional support, web sites (demo server / flossola), etc. Note that various sources reported insufficient funding of the SOLA project in the past years as partly explaining the "unfinished" state that the source code, user interface, manuals and documentation of SOLA are in.
- FAO has demonstrated it is possible to develop open source tools for land administration and field data collection as perhaps a UN agency should be demonstrating; however FAO's core business and organisation are not focused at information system development in the context of open source communities.
- Instead of managing turn-key information system development projects on the basis of SOLA, FAO may opt for providing (smaller) consultancy in specific phases of these projects, for example feasibility studies or requirements phases. Resources would have to be trained to provide these services in the context of SOLA.
- There may be a reluctance / reduced interest by professional community to participate in open source community because of FAO's (tight) control of the opportunities in the past.
- Reports have been made with regard to insufficient promotion and communication on SOLA applications, e.g. in land related conferences, workshops, white papers, publications, actively approaching professionals in the field, universities. etc.

Remarks on Field Tenure Fata Collection (section 2.2)

- SOLA infrastructure (demo server and flossola website) including hosting community server could done by within FAO by Information Technology Division (CIO). Note that earlier attempts seem to have failed,

presumably/reportedly on account of flexibility of hosting environment to SOLA parameters and tools.

- Currently the Demo Server[33] and the SOLA Website[34] (latest news 2016) are being operated and funded on a voluntary basis by former SOLA staff, through third party hosting providers (external to FAO), these websites will expire beginning of 2019.

- Funding for sustaining the support of SOLA (i.e. hosting/development team) is essential; the issue of such funding is a recurring report received during this review.

Note that the CIO, FAOs Information Technology Division, has made a recommendation to take ownership over the SOLA software intellectual property, lifecycle, support and maintenance, which also includes the software quality improvement as suggested by the technical quality review (see section 1.1.1); Appendix J, slide 80-81 of the presentation provides a summary).

2.4 Governance by Professional Body

Another option would be to consider the handover of the SOLA project to an organisation that is more equipped and can rely on staff with the right skills and experiences with regard to technologies used in SOLA. This could be considered for SOLA R, SR, and SL (Information System Development, section 2.1) as well as for SOLA OT/CS applications (Field Collection of Land Tenure Data, section 2.2).

Before a handover of responsibility for governance of SOLA applications to a professional body (or a private company or open source community, see next sections) an investment is required to implement recommendations: fixing bugs, improvement of documentation for developers, installers for SOLA applications, critical issues, etc.

2.4.1 Global Land Tool Network (GLTN)

The Global Land Tool Network (GLTN, see section 1.2.10) is an alliance of international partners[35] with an interest in land and tenure security. GLTN develops land tools contributing to "*land reform, good land governance, inclusive land administration, sustainable land management, and functional land sector coordination*"; FAO, USAID (MAST[36]), OSGeo, as well as Cadasta[37] are partners in the Global Land Tool Network.

[33] https://demo.opentenure.org
[34] http://www.flossola.org
[35] https://gltn.net/gltn-partners-3/
[36] https://gltn.net/2018/11/15/securing-land-tenure-with-smartphones
[37] https://gltn.net/2017/05/04/gltn-welcomes-the-cadasta-foundation-as-its-78th-partner/

The GLTN partners, representing represent global and regional institutions, organizations or networks, commit to join the network for non-commercial purposes and engage in scale-able land tool development. The STDM[38,39] is one of the flagship land tools; GLTN land tools are pro-poor and gender responsive; the following features should be provided by the land tools that GLTN hosts:

- Affordable. To the poor and institution managing the tool.
- Equitable and gender-responsive. Address needs fair and equal.
- Pro-poor. Aim to reduce poverty.
- Sustainable. Implementable without large-scale external inputs.
- Systematic, large scale. Usable at district or even national level.
- Governance. Address local decision-making and conflict resolution on land.
- Subsidiarity. Sensitive to local situations, needs and capability.

Remarks

- SOLA R, SR, SL may not qualify as sustainable since their implementation, currently, is rarely happening without considerable, large-scale external inputs.

- SOLA OT, maybe considered as land tools that satisfies the features above, however STDM is being expanded with ODK based tools which would provide similar functions. There may be no interest within GLTN to host SOLA OT/CS alongside with STDM.

- Finding a suitable professional body may be challenging, since interest in SOLA has diminished, and the development knowledge is in other areas (e.g. Python vs Java programming language).

2.5 Governance by Consultant

A private company may be interested to create a set of services around SOLA, on the one hand around SOLA Registry (section 2.1), on the other hand around field data collection (section 2.2).

Remarks

- Previously, SOLA projects and pilots included the responsibility of identifying, hiring and training personnel, such as software developers, system administrators, trainers. Private (local) companies in the business of providing information system development, with sufficient resources might be better able to manage and oversee these development projects and hired experts.

- As the case study report "Community Engagement Strategy (CES) for the Sola Open Source Software" also mentions, it may be difficult to find a java development consultant who wants to take the risks involved with investing in SOLA, with a very small open source community.

[38] https://gltn.net/2018/09/25/improving-tenure-security-on-customary-lands-in-zambia/
[39] https://gltn.net/job/part-time-consultants-python-developers-for-the-social-tenure-domain-model/

2.6 Governance by Open Source Community

The responsibility for the SOLA project and its applications could also be handed over to an open source community, for example as one of the projects in OSGeo. This could be considered for SOLA R, SR, and SL, or for SOLA OT/CS applications as well. See section 1.4.6 on recommendations for SOLA open source project and key factors for successful open source communities.

2.6.1 Open Source Geospatial Foundation (OSGeo)

Should FAO wish to leave/transfer the SOLA project, it would preferable to hand it over to (other) developers and users who care about and will continue it. FAO could consider SOLA to become part of an open source community such as OSGeo which would increase promotion and visibility of the SOLA applications. The Open Source Geospatial Foundation (OSGeo[40]) is a "*not-for-profit organization whose mission is to foster global adoption of open geospatial technology*" which brings open source users, developers, and other open source web mapping community participants together in providing links to events, documentation, websites, and other information. OSGeo manages so called OSGeo Projects in different technical fields, stable and widely used:

- Content Management Systems (e.g. GeoNode).
- Metadata Catalogs (related Spatial Data Infrastructure).
- Desktop Applications (e.g. QGIS).
- Spatial Database (PostgreSQL's extension PostGIS).
- Geospatial Libraries (e.g. Geotools and GDAL(OGR).
- Web Mapping (e.g. GeoServer and Open Layers).

The OSGeo Projects have been subject to extensive mentorship and preparation by OSGeo during an incubation phase. In addition, OSGeo hosts OSGeo Community projects such as proj4 (transform geospatial data between coordinate reference systems) and pgRouting (network/routing analysis of road datasets).

SOLA may be eligible to join the Open Source Geospatial Foundation as a community project (i.e. in the process of building the user and developer community and establishing a governance structure) if the project has a geospatial nature, an open source license, and is participatory, i.e. accepts contributions (Figure 15).

Open source projects can also enter an incubation process, mentored by a incubations committee[41], to become an OSGeo Project, requiring much more, for example: source code check, transparent communication and decision making process, and active and healthy community with user and developer collaboration, user and developer documentation (Figure 16 and the Project Graduation Checklist[42] [June 2014]).

[40] https://www.osgeo.org
[41] https://www.osgeo.org/about/committees/incubation/incubation-process/
[42] https://www.osgeo.org/wp-content/uploads/graduation_checklist.pdf

Figure 15: OSGeo Community

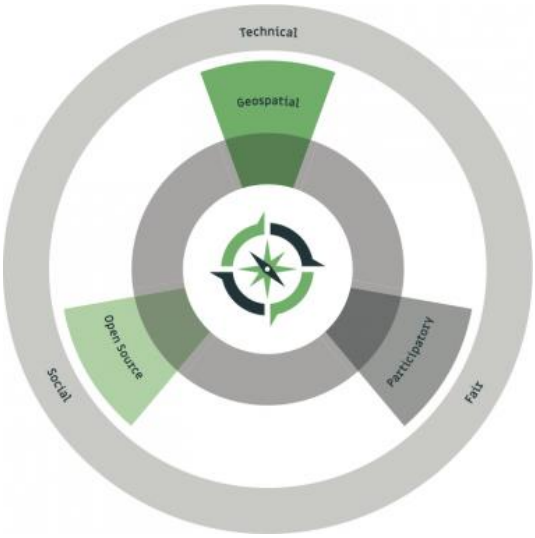


Figure 16: OSGeo Project

Source: OSGeo

OSGeo indicates to be sensitive to projects which are open-source in license but do not have any developers. These may be published as open source, but are in effect "*dumped on the internet*", lacking any form of governance enabling contributions. Lacking an active and healthy community is also addressed in the CES in its recommendation B on considerations for joining the OSGeo foundation:

The difficulty for SOLA will be attracting enough support to complete the restructure necessary to comply with OSGeo requirements. …. As an interim step, it may be necessary for SOLA to join OSGeo as a community project …. This would require significantly less effort than incubating SOLA to become a full OSGeo project but the issue of who would guide SOLA through this process remains unanswered.

To avoid barriers for contributing, OSGeo does not impose membership fees on its participants, members, or charter members[43] , but works with supporters who donate money to the Foundation. OSGeo reserves the right to have levels of support and/or yearly subscriptions in the future.

Remarks

- Transferring the project to OSGeo will also require identifying, selecting and setting up and (partly) subsidising a team that wants to continue and maintain SOLA applications. Funding will be required to establish the SOLA project as an OSGeo community or even OSGeo project: to perform the incubation process, the activities of the project owner to manage

[43] https://www.osgeo.org/about/membership-rules/

community promotion; and improve SOLA applications and documentation, as well as maintenance.

- Handing over the SOLA project to OSGEO would also mean allowing the open source community to set the priorities within the SOLA project, which should be seen and accepted as a consequence of a healthy open source community. The CES states that the community decides on how OT, CS and the other SOLA solutions develop. "*Potentially the community will not see any value in State Land, Registry, or Systematic Registration and will choose not to support those solutions. If so, that should just be viewed as the community in action*".

- Support is provided through community participation, but also by service providers who build a (profitable) service model around open source.

- Strengthening the community side of the SOLA project through establishing more SOLA implementations is not expected to be enough. Many different roles within an open source community need to be supported; the goal should be to engage anyone that has a genuine interest in land and open source land information systems.

- OSGeo has a focus on (and preference for) *technical* aspects of geospatial open source projects, consider the more land tenure domain related objectives of the Global Land Tool Network (GLTN).

2.7 Discontinue SOLA project

The most feasible option seems to be the transfer of the SOLA project to an open source community, external to FAO, possibly as an OSGeo project. This option, as all other options discussed in previous sections, requires funding for governing the project, as well as for improving the documentation and tools for developers and system administrators (before the actual handover).

If this funding cannot be made available, possibly inspired by the reflection on the reasons for the open source project not picking up, discontinuing the SOLA project should be considered. Part of this consideration should be the following:

Remarks

- Discontinuation should be handled with care also in the context of preserving FAO's reputation as a responsible institution (with regard to the existing institutions/users having deployed SOLA).

- In all matters with regard to closing down the SOLA project, clear communication with the existing user community is required on FAO's intentions and each step, to minimise the concerns of existing community and its users. A smooth transition is preferable rather than just stopping it.

- The SOLA sources should be made available (e.g. on GitHub); the source code(s) do not need to be removed but can remain in "read-only" mode. Users would still be able to "fork" it into other (uncontrolled) development. Other elements that may be part of the transfer / archiving may be the documentation, the SOLA repository, wikis, websites, support infrastructure.

2.8 SOLA Quick Guide

Many land administration modernisation projects have been carried out over the last decades in developing countries with varying results. Big contracts seem to have a lower success rate; whereas small contracts, piloting/prototyping, modular/step-by-step approaches, in-house (home-grown) development are relatively more successful, simple is better than initial "big bang" approaches. Failing to align the system and IT with the business goals of the organisation, underestimation of the complexity and size, and the failure to adequately plan and manage the projects (beyond deployment), are often mention as reasons for projects not providing sustainable results.

FAO receives requests with regard to the deployment of SOLA applications; either with regard to the deployment and development of a land information system (LIS), but in the past few years, more often addressing field collection of land tenure data (Data Collection).

This checklist may be further influenced and adapted once a final choice for the road map option for FAO on SOLAs future has been made. For example support questions related to existing deployment could be referred to SOLA open source community (section 2.6), should that be the pursued option.

As a first response to requests regarding future implementations of SOLA, the below checklist (Figure 17) should be shared with the requesting institutions and communities, requiring them to provide information in response to the questions therein. The check are covering topics that should be considered to ensure a successful implementation and long-term sustainability, and are grouped in:

- Strategy & Scope. Approach, business process, requirements.
- Governance and Leadership. Project management, institutional support.
- Funding.
- Resources. Staffing, services, equipment.

Along with the checklist, an initial specially compiled information package could be created from the elements of the existing (improved) documentation, to guide the institutions and communities.

Figure 17: Checklist for considering SOLA deployment

Strategy & Scope		
Project Definition	A clear description of the vision, project objectives, needs and outcome, as well as and the related/responsive *high-level* or initial (business and information system) requirements must be created: › LIS. Determine what the system will do, also to assist in assessing the required customisation of SOLA applications. › OT. Determine what data will be captured, also to assist in assessing the required configuration of dynamic forms.	Spend enough time on designing the project at the beginning: the topics with regard to project strategy and scope should be addressed, a feasibility study may be executed to bring clarity on these.
Project approach	Is a clear approach and work planning defined or all project phases, from orientation, definition, and development to deployment and operation & maintenance (section 1.4.1); short and long term visions: › LIS. Approach to operation & maintenance, often "forgotten", should be described to ensure sustainability. › LIS. Approach to customisation and information system deployment ("big bang" or incremental). › Where is the system going to be implemented, and when?	
Agreed business process	› LIS. Agreement on the system supported business process(es). Which process will support the objectives? Has a business process reengineering (BPR) exercise been executed to capture information on existing land registry and cadastre processes and data, as well as defining and agreeing on an alternative more efficient system processes (as opposed to automation of existing manual and paper-based processes.	Deployment of IT system provide opportunities to transform and improve current processes.
Legacy Data	› LIS. Agreement on data to be converted and made available in the system.	
Detailed requirements	› LIS. Is there an agreement on detailed requirements and customisation approach and scope? › Have key users been involved in defining requirements? › Have Non-functional requirements been defined: performance, compatibility, usability, reliability, security, maintainability, portability. › Is it clear exactly how the system will provide benefits for the receiving institution?	Clarity on the exact requirements that SOLA applications offer will be essential. A scope can be too ambitious.
Governance & Leadership		
Project Management	Is project management defined/established?	The project needs to be planned, managed, monitored, and evaluated by a project manager.
Institutional support and engagement	Involvement, commitment, approval by relevant local stakeholders; institutions, organisations and individuals; their advice with regard to local customs, laws & regulations that may influence the project.	

Project risks	Have project risks and mitigations been documented? Are project risks understood and managed (avoided)?	› Lack of governance › Poor management › Development risk › Deployment Risk › Operational Risk
Funding		
Capital planning	Have cost estimations/calculations of all resources been made for all phases (short / long term)?	Total Cost of Ownership: The acquisition or development cost is a part of the TCO, but even more significant are the scale-up and maintenance costs.
Funding available	Has funding been secured/allocated? Have institutions and donors committed to the specified budgets?	
	› LIS. Funding for services beyond deployment of the system are essential for sustainability. The notion that a LIS can be customised and deployed once and then used without changing it, is a misconception. Developing an LIS is a continuous process rather than a one-time job. After deployment, the need for new requirements and additional functions will arise, especially during successful operation of a system.	
Resources		
Staffing & Services	Has the team been setup and are all required skills for development, deployment and sustained operation identified?	
Beneficiary's commitment	Allocation of beneficiary's staff ("champion[44]"), coordinator, product owner (scrum term) testers. Allocation of office space for development and deployment/operation.	
Local specialists	› LIS. Identification and recruitment of suitable local specialists with the right skills and experience: system development (e.g. Java, PostGIS) and system administration (JasperReports, Geoserver, Glassfish, etc.). Availability of local company / subcontractor with experience in system development. Availability and commitment of local experienced software developers should be a condition for successful implementation and continued operation of SOLA applications. › OT. Local specialist managing field work.	
Equipment	Is equipment available and has maintenance been factored in? Hardware and software will need to be upgraded, requiring time and effort.	
LIS equipment	› LIS. The system will be deployed in local premises for at least a few years: › Database & application server (backup media). › Computers, laptops. › Printers, scanners, plotters & consumables.	

[44] http://www.businessdictionary.com/definition/champion.html

	› Power supply. Sufficient/continuous electricity available, UPSs. › Local Area Network (switches, Wi-Fi, cabling, network points). › Existing Infrastructure sufficient for scaling up? › Development software (and licenses if applicable).	
OT equipment	› OT. The system may be deployed for only a limited period during field data collection: › Mobile devices (phones, tables). › Power supply (solar panels, power banks). › Community server: either a physical server purchased and hosted locally, or a third party virtually hosted server (backup media). › Reliability and availability of tele/Internet communication; with consequence for server setup. › Printer for printing certificates & consumables. › External GPS devices.	
Other	Outreach to future users or affected communities? For introducing IT systems for land records or field work with regard to data collection.	