

Open Geospatial Consortium

Approval Date: 2013-01-18

Publication Date: 2013-02-01

Reference number of this document: OGC 12-119r1

Reference URL for this document: <http://www.opengis.net/def/doc-type/per/OWS9-OGCMobileApps>

Category: Engineering Report

Editor: Ingo Simonis

OWS 9: OGC Mobile Apps: Definition, Requirements, and Information Architecture

Copyright © 2013 Open Geospatial Consortium.

To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

Warning

This document is not an OGC Standard. This document is an OGC Public Engineering Report created as a deliverable in an OGC Interoperability Initiative and is not an official position of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard. Further, any OGC Engineering Report should not be referenced as required or mandatory technology in procurements.

Document type: OGC® Engineering Report
Document subtype: NA
Document stage: Approved for public release
Document language: English

Abstract

This engineering report represents the results of the OWS-9 innovations thread on mobile applications. Initially, the goal was to help understanding the requirements for developing standards-based geospatially-enabled mobile applications. The report describes how OGC Enabled Mobile Apps can be integrated into information architectures based on OGC standards. Particular emphasize has been put on the future work section, as it provides valuable recommendations for further standardization work (and, equally important, highlights aspects that could be excluded from standardization)

Keywords

ogcdoc, ows9, mobile apps, geopackage, ows context, architecture

What is OGC Web Services 9 (OWS-9)?

OWS-9 builds on the outcomes of prior OGC interoperability initiatives and is organized around the following threads:

- **Aviation:** Develop and demonstrate the use of the Aeronautical Information Exchange Model (AIXM) and the Weather Exchange Model (WXXM) in an OGC Web Services environment, focusing on support for several Single European Sky ATM Research (SESAR) project requirements as well as FAA (US Federal Aviation Administration) Aeronautical Information Management (AIM) and Aircraft Access to SWIM (System Wide Information Management) (AAtS) requirements.
- **Cross-Community Interoperability (CCI):** Build on the CCI work accomplished in OWS-8 by increasing interoperability within communities sharing geospatial data, focusing on semantic mediation, query results delivery, data provenance and quality and Single Point of Entry Global Gazetteer.
- **Security and Services Interoperability (SSI):** Investigate 5 main activities: Security Management, OGC Geography Markup Language (GML) Encoding Standard Application Schema UGAS (UML to GML Application Schema) Updates, Web Services Façade, Reference Architecture Profiling, and Bulk Data Transfer.
- **OWS Innovations:** Explore topics that represent either new areas of work for the Consortium (such as GPS and Mobile Applications), a desire for new approaches to existing technologies to solve new challenges (such as the OGC Web Coverage Service (WCS) work), or some combination of the two.

- **Compliance & Interoperability Testing & Evaluation (CITE):** Develop a suite of compliance test scripts for testing and validation of products with interfaces implementing the following OGC standards: Web Map Service (WMS) 1.3 Interface Standard, Web Feature Service (WFS) 2.0 Interface Standard, Geography Markup Language (GML) 3.2.1 Encoding Standard, OWS Context 1.0 (candidate encoding standard), Sensor Web Enablement (SWE) standards, Web Coverage Service for Earth Observation (WCS-EO) 1.0 Interface Standard, and TEAM (Test, Evaluation, And Measurement) Engine Capabilities.

The OWS-9 sponsors are: AGC (Army Geospatial Center, US Army Corps of Engineers), CREAM-GeoViQua-EC, EUROCONTROL, FAA (US Federal Aviation Administration), GeoConnections - Natural Resources Canada, Lockheed Martin Corporation, NASA (US National Aeronautics and Space Administration), NGA (US National Geospatial-Intelligence Agency), USGS (US Geological Survey), UK DSTL (UK MoD Defence Science and Technology Laboratory).

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER'S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR's sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications.

This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

None of the Intellectual Property or underlying information or technology may be downloaded or otherwise exported or reexported in violation of U.S. export laws and regulations. In addition, you are responsible for complying with any local laws in your jurisdiction which may impact your right to import, export or use the Intellectual Property, and you represent that you have complied with any regulations or registration procedures required by applicable law to make this license enforceable

Contents	Page
1 Introduction.....	1
1.1 Scope.....	1
1.2 Document contributor contact points.....	1
1.3 Revision history	2
1.4 Future work.....	2
1.5 Forward.....	4
2 References.....	4
3 Terms and definitions	4
4 Conventions	5
4.1 Abbreviated terms.....	5
5 Management Summary.....	5
6 OGC Enabled Mobile Applications.....	6
6.1 Definition	6
6.2 Role in the OGC Architecture.....	7
6.3 Access to Data.....	7
6.3.1 Data Made Available at OGC Web Service Interfaces.....	7
6.3.2 Local Data.....	8
6.3.3 Combined Local and Remotely Available Data	9
6.4 OGC Enabled Mobile App Deployment & Discovery	9
6.4.1 App Store	10
6.4.2 Manual Download.....	11
6.4.3 Enterprise Device Management.....	12
6.5 Data Access & Data Loading.....	13
6.5.1 Local Data.....	13
6.5.2 Data Provided at OGC Web Service Interfaces	14
6.6 Usage, Editing & Synchronization	14
7 Usage Scenarios.....	15
8 Further Aspects.....	16
8.1 Certification	16
8.2 Security	16
8.3 File Size of Local Files	17

Figures	Page
Figure 1: Role of OGC Enabled Mobile App in the OGC Architecture	7
Figure 2: OGC Enabled Mobile App to OGC Web service communication	8
Figure 3: Simplest architecture. OGC Enabled Mobile App uses data stored in <i>GeoPackage</i>	8
Figure 4: OGC Enabled Mobile App combined data handling	9
Figure 5: Deployment options for OGC Enabled Mobile Apps	10
Figure 6: OGC Enabled Mobile App discovery and deployment, app store scenario	11
Figure 7: Use case direct app download	12
Figure 8: Use Case Enterprise Device Manager	13
Figure 9: Use Case <i>GeoPackage</i> Packaging Server	14

OWS-9 OWS Innovations Mobile Apps Engineering Report

1 Introduction

1.1 Scope

This engineering report represents the results of the OWS-9 innovations thread on mobile applications. Initially, the goal was to help understanding the requirements for developing standards-based geospatially-enabled mobile applications. The following elements had been identified to be of particular interest:

- Certification and accreditation of mobile apps
- Communication technologies, issues, solutions
- Data loading and synchronization
- Security and data privacy
- Mobile apps distribution options
- Quality of results and services
- Metadata aspects for mobile apps and web-based services

During the context of OWS-9, it became clear that a much more important key question has to be answered first: “*What is an OGC Enabled Mobile App?*” Therefore, this engineering report first defines the OGC Enabled Mobile App and focuses on the key characteristics of OGC Enabled Mobile Apps. The report describes how OGC Enabled Mobile Apps can be integrated into information architectures based on OGC standards. Particular emphasize has been put on the future work section, as it provides valuable recommendations for further standardization work (and, equally important, highlights aspects that could be excluded from standardization).

1.2 Document contributor contact points

All questions regarding this document should be directed to the editor or the contributors:

Name	Organization
Ingo Simonis	iGSI GmbH

1.3 Revision history

Date	Release	Editor	Primary clauses modified	Description
02.10.2012	0.1	Ingo Simonis	All	Initial version
26.12.2012	1.0	Ingo Simonis	All	Final version
16.01.2013	1.1	Ingo Simonis	All	Version r1: Changed OGC Mobile App to OGC Enabled Mobile App throughout the document

1.4 Future work

This document was developed in the context of OWS-9. Future initiatives shall define the various aspects of mobile apps more precisely to improve ease of use, market acceptance, and interoperability between system components related to mobile applications. In particular, the following aspects shall be addressed:

Synchronization

Synchronization aspects have been briefly addressed in OWS-9, but require further discussion. In addition, it was not clear if the synchronization mechanism for large data sets, e.g. new tiles for satellite imagery, is a topic for OGC standardization at all. Distributing image tiles might rather be a typical situation of rapid distribution of large data sets than a geo-sync scenario.

Independently of the distribution of large data sets (and given the growing computing power of and the opportunity to connect external sensors to cell phones, a two-way synchronization may become relevant), the two-way synchronization of feature data sets is already a topic that requires further OGC testbed activities. Two general patterns are applicable, i.e. either service oriented messaging to send update requests to service endpoints with or without dedicated synchronization services (which would require explicit knowledge about all service endpoints and the data origin at the endpoint in case of sets of data layers), or subscription to update feeds on both sides.

Once the *GeoPackage* specification is fully endorsed, the two-way synchronization can be elaborated in full detail. *GeoPackage* is a powerful tool for offline data handling, but makes a two-way synchronization challenging: Who submits the changes to the feed/sync service, the geopackaging service or the *GeoPackage* itself? Who really understands which base data corresponds to what type of data in a *GeoPackage*? Wouldn't that be a tough requirement for a geopackaging service? Keeping track of all mappings between original data sources and created *GeoPackages*? What kind of extensions would be necessary for the geosync-service to support the feed approach?

In addition, updates to base data sets need to get communicated through all layers. Let's assume a typical scenario to illustrate this requirement: Routes are stored on the device in a *GeoPackage*. If new road data gets in, the synchronization approach should not only take care of the new underlying road data, but needs to handle the routes data as well, which is usually built on top of the underlying road data. This is not a trivial problem, as – in particular in the enterprise world – lots of aspects are in fact layered on the base data.

In addition to the synchronization of feature data, the synchronization of raster data might become a topic. Here, initial questions based on requirements need to be answered first: What happens if only a subset of the raster get updated? Do we restrict synchronization to full raster data sets? In any case, the current geosync-service is optimized for vector data. It does not handle map tile synchronization or raster layer sub-set synchronization yet.

OWS Context

The role of OWS Context: Is it possible to use OWS Context as it is currently defined to organize situation specific data source organization, i.e. to point to online and locally installed data sources? Does this really play a role for OGC Enabled Mobile App development? Though it has been discussed in OWS-9, there was general hesitance to develop something like a descriptive format for data source definition. Due to their smaller screen, touch gesture control, and mobile device sensor integration, it is more expected to see specialized applications rather than generic applications that could be loaded on demand as defined in a context document. Therefore, a proper requirements analysis is recommended before OWS Context is further explored.

GeoPackage Packaging Service

A GeoPackage service shall be developed that allows packaging data into GeoPackages.

Security

OWS-9 recommends running a requirements-analysis to better understand the targeted level of granularity and access control. For the time being, a page-level encryption service on the data base side (for *GeoPackages*) using a passphrase protection has served all requirements mentioned so far. Secure communication with OGC service endpoints can be realized by using application servers as part of the communication design. In this case, no changes to OGC security standards occur. Applying the full OGC security portfolio directly on mobile phones seems less practical.

Branding and Accreditation

The definition of OGC Enabled Mobile App allows the development of the two-step testing and accreditation process as described further below. The OGC CITE program could be extended.

Other Aspects

Other aspects that came up during OWS-9 worth mentioning include:

- Vector tiling (to improve rendering performance)
- Generalization of vector data
- Styling of vector data
- Metadata description of GeoPackage

1.5 Forward

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

2 References

The following documents are referenced in this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

OGC 06-121r3, *OGC[®] Web Services Common Standard*

OGC 12-128r1, *OpenGIS[®] GeoPackage Implementation Specification*

NOTE This OWS Common Standard contains a list of normative references that are also applicable to this Implementation Standard.

3 Terms and definitions

For the purposes of this report, the definitions specified in Clause 4 of the OWS Common Implementation Standard [OGC 06-121r3] shall apply. In addition, the following terms and definitions apply.

3.1

Mobile app

Mobile applications (mobile apps) are software products developed for mobile devices, such as mobile phones or smart phones. These applications can be pre-installed on phones during manufacture, downloaded by customers from various mobile software distribution platforms, or delivered as web applications using server-side or client-side processing (e.g. JavaScript) to provide an "application-like" experience within a Web browser.

3.2

OGC Enabled Mobile App

The term OGC Enabled Mobile App is used to differentiate from mobile applications in general. A detailed definition is provided and discussed in section 5.

3.3

Mobile Device

A mobile device (also known as a handheld device, handheld computer or simply handheld) is a small, hand-held computing device, typically having a display screen with touch input and/or a miniature keyboard and weighing less than 2 pounds (0.91 kg). A handheld computing device has an operating system (OS), and can run various types of application software, known as apps. Most hand held devices can also be equipped with WI-FI, Bluetooth and GPS capabilities that can allow connections to the Internet and other Bluetooth capable devices. (Wikipedia, 12/2012).

4 Conventions

4.1 Abbreviated terms

API	Application Program Interface
COTS	Commercial Off The Shelf
OS	Operating System
UID	User Identifier
UI	User Interface

5 Management Summary

OWS-9 produced a very vivid discussion on the nature of OGC Enabled Mobile Apps, resulting in a clear definition (see clause 6.1), an integration into the OGC architecture, usage scenarios, and a detailed description on future work items. After several months, agreement has been reached on most aspects. Saying that, it has to be emphasized that the OWS-9 innovation thread/mobile applications group followed a rather pragmatic approach, means that only requirements resulting from group members' other projects have been brought to the table. Requirements à la "I could imagine that" have not been taken into account. In summary, OGC Enabled Mobile Apps are characterized as follows:

- OGC Enabled Mobile Apps do not require any change to existing data serving Web service interfaces, such as WMS, WMTS, WFS, WCS, SOS, etc.
- OGC Enabled Mobile Apps do not require any change to existing processing Web service interfaces, such as WPS or SPS.
- OGC Enabled Mobile Apps may require changes to synchronization services to handle vector, raster, and tile data in all situations of varying and potentially intermittent connectivity

- OGC Enabled Mobile Apps behave in REST-ful architectures the same way as desktop applications
- The smaller screen size leaves less room to render versatile messages retrieved from a server, e.g. GetCapabilities responses, but this remains an implementation detail less relevant to standardization activities
- GeoPackage is a central new piece to the architecture, as it enables working offline (map cache, locally collected/updated data, essential data needed to support working in the field) and serves as a cache for deferred synchronization

The following aspects express more general impressions and anticipations, rather than being hard facts about OGC Enabled Mobile Apps. Nevertheless, they provide valuable insight into the modern approach of designing, developing, and using OGC Enabled Mobile Apps in OGC architectures.

- We see a paradigm shift: Away from ever enriching the server side interfaces and data exchange encodings towards more emphasize on the client side and protocol simplification
- OGC Enabled Mobile App to application server communication becomes an essential part of OGC Enabled Mobile App architectures that facilitates lots of communication, client role handling, billing, security, etc.
- It is anticipated that the nature of OGC Enabled Mobile Apps is less explorative (in terms of *what is available*), but rather a production oriented
- It is more likely to find five OGC Enabled Mobile Apps running on the same device than a one-app-fits-all situations

6 OGC Enabled Mobile Applications

6.1 Definition

An OGC Enabled Mobile App runs on a mobile operating system; has a user interface optimized for a touch screen on a mobile device; utilizes the device's sensors as contextual input; implements one or more OGC services and/or data container specifications; and visualizes or updates those data.

The definition of OGC Enabled Mobile Apps given above emphasizes the key characteristics of OGC Enabled Mobile Apps. Whereas the first part applies to almost all mobile apps that are currently available, the second part addresses OGC specific aspects, as there are the visualization and potential modification of data either served by OGC services or OGC data containers.

OWS-9 has produced long discussions about what aspects shall be relevant for OGC as a standardization organization, and which aspects are implementation specific and shall not be addressed by OGC. Those discussions resulted in the definition provided above, and highlight the fact that an OGC Enabled Mobile App is seen as a client application that

operates within the OGC architecture framework. If those requirements are fulfilled, an app shall be labeled OGC Enabled Mobile App.

One of the key differences between mobile apps and desktop apps is the connectivity between the client application and distributed data sources. Whereas desktop applications are usually developed under the premises of continuous connectivity to the Internet and subsequently their distributed data sources, mobile applications have to deal with intermittent connectivity and potentially high communication costs.

6.2 Role in the OGC Architecture

The following architectural sketches illustrate the role of OGC Enabled Mobile Apps within infrastructures implemented using the OGC standards portfolio. In summary, OGC Enabled Mobile App fit well into existing spatial data infrastructures. They do not require modifications to existing systems, and still provide the advantage that continuous connectivity to other services is not required any longer. This is realized by combining HTTP based Web service communication with in-file data storage, implemented as *OGC GeoPackage*. An OGC Enabled Mobile App may support any number of additional features, such as communication with remote Web services, application servers, or data stores using proprietary protocols, or access and manipulation of locally stored proprietary data files or data bases.

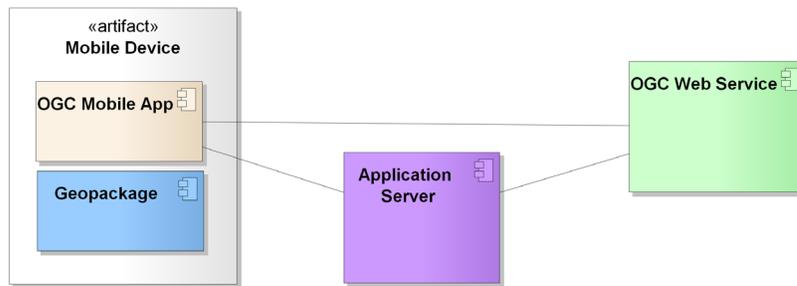


Figure 1: Role of OGC Enabled Mobile App in the OGC Architecture

Any OGC Enabled Mobile App shall at least support the communication with OGC Web services, or with OGC data containers such as *GeoPackage*. The app interacts with those services/formats and makes the data available to the user. OGC Enabled Mobile Apps cover the full range from locally installed data usage up to fully synchronized Web service communication in intermittent connectivity scenarios.

6.3 Access to Data

6.3.1 Data Made Available at OGC Web Service Interfaces

In the simplest scenario, the OGC Enabled Mobile App accesses OGC standards compliant Web services such as WMS, WFS, WMTS, etc. according to the various OGC standards. The app communicates with those Web services either directly or via an application server.

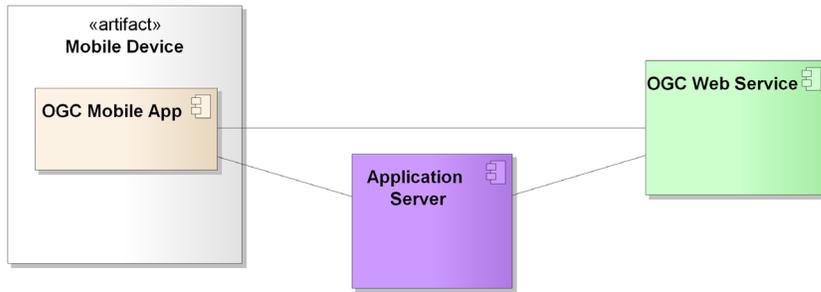


Figure 2: OGC Enabled Mobile App to OGC Web service communication

The communication between the OGC Enabled Mobile App and the application server might use OGC standards compliant or proprietary protocols. At this stage, the OGC Enabled Mobile App definition is slightly stretched, as an app is still called OGC Enabled Mobile App, even if the communication to OGC Web services uses proprietary protocols on the app to application server communication leg.

The OGC Enabled Mobile App may include the links to the Web services or application servers by default; provides a list of potentially linked services, or allows the user to add additional services manually. Analogue to traditional spatial data infrastructures, the OGC Enabled Mobile App may make use of catalogue services. Due to the limited screen size and the resulting exploration experiences, catalogue interfaces shall provide small-screen optimized result representation. Practically, it is currently expected that this approach will only play a minor role in service discovery.

6.3.2 Local Data

OGC Enabled Mobile App can use data stored locally on the mobile device, if the data is compliant with OGC container formats. Theoretically, an OGC Enabled Mobile App may access data served by a locally installed Web service. Practically, performance aspects will most likely negate such a solution.

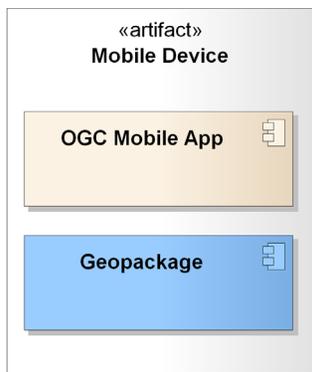


Figure 3: Simplest architecture. OGC Enabled Mobile App uses data stored in *GeoPackage*

If on-device data handling is supported, then OGC Enabled Mobile Apps shall support the data container format *GeoPackage* (see OGC 12-128r1, OpenGIS® *GeoPackage* Implementation Specification).

6.3.3 Combined Local and Remotely Available Data

OGC Enabled Mobile App may integrate both solutions described above. This most complex form supports scenarios with intermittent connectivity.

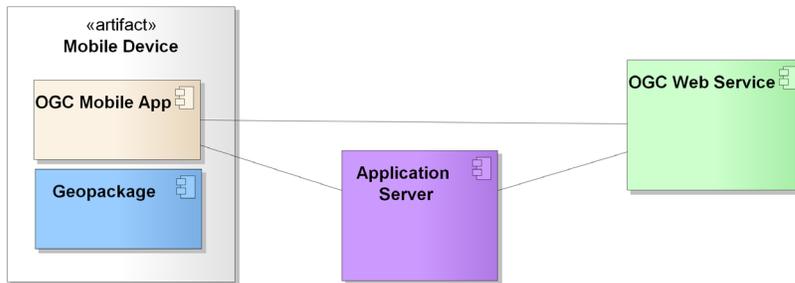


Figure 4: OGC Enabled Mobile App combined data handling

As long as connectivity and bandwidth allow, the OGC Enabled Mobile App uses remotely stored data. If the connection gets interrupted, the app switches to the locally stored data and optionally synchronizes as soon as connectivity is reestablished.

6.4 OGC Enabled Mobile App Deployment & Discovery

Mobile apps as products are already well established and matured outside of the OGC realm. Usually, the different operating system providers operate different flavors of app stores allowing discovery, deployment and maintenance of mobile apps. Among the most important app stores, there is the Google Play Store for Android, the Apple App Store for iOS, the Windows Store for Windows 8 and Windows RT, the BlackBerry App World for BlackBerry OS, and a number of stores operated by the mobile device manufacturers. Essentially, all stores follow the same principle, including catalogues and browsing platforms for app discovery as well as app updating mechanisms. Any OGC Enabled Mobile App will be distributed either using an app store, directly by the software developer, or using proprietary mechanisms in case of enterprise device management.

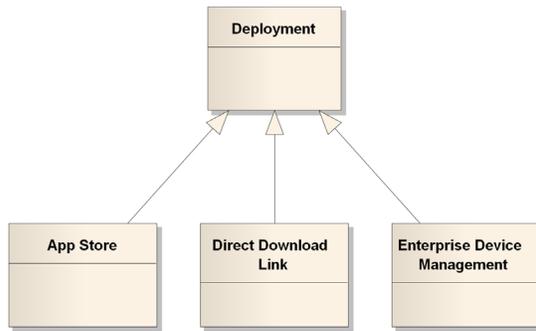


Figure 5: Deployment options for OGC Enabled Mobile Apps

In the first case, the OGC Enabled Mobile App will be registered in the app store by the developer similar to any other non-OGC app. Following standard app store procedures, the user can discover, (purchase), download and install the OGC Enabled Mobile App.

In the second case, the software developer provides a download link to the user. Users can use this link to download and install the OGC Enabled Mobile App locally. In this case, the exact procedure differs slightly from operating system to operating system. Some operating systems do not support manual app installation in general. Alternatively, app (and data) are provided on a memory card. The installation procedure remains the same, just that the download link points to a local resource.

In case of enterprise device management, the device is controlled by a central device management system. Once connected to the private network, the device manager takes over control and deploys software and data. The following use case diagrams illustrate the potential solutions.

6.4.1 App Store

The first use case illustrates the most common scenario for personal use. The user explores the apps provided by the app store using app store exploration tools and selects the identified OGC Enabled Mobile App. This process can be controlled either using the mobile device or a Web-based browsing platform optimized for larger screens.

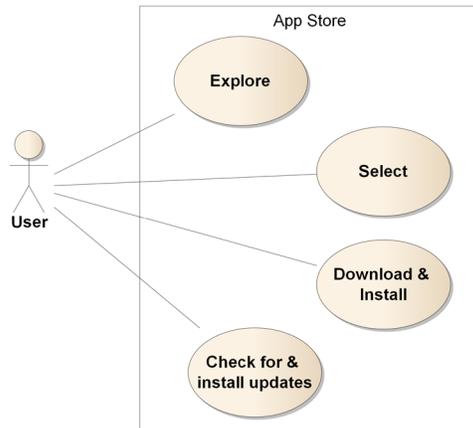


Figure 6: OGC Enabled Mobile App discovery and deployment, app store scenario

Following the app store procedures, the app will be downloaded and automatically installed on the local device. All modern devices support automated checks for updates, which could be downloaded and installed upon user confirmation. OWS-9 participants disliked the idea of a dedicated OGC App Store, as the various stores already provide all required functionality.

The app stores with their in-built evaluation and commenting system helps dealing with trust issues as well. All OS-providers have their specific approach to authenticate app developers. For example, Android requires developers to sign their code. Android code signing usually uses self-signed certificates, which developers can generate without anyone else's assistance or permission. One reason for code signing is to allow developers to update their application without creating complicated interfaces and permissions. Applications signed with the same key (and therefore by the same developer) can ask to run with the same UID. This allows developers to upgrade or patch their software easily, including copying data from existing versions. The signing is different than normal Jar or Authenticode signing however, as the actual identity of the developer isn't necessarily being validated by a third party to the device's user. Developers earn a good reputation by making good products; their certificates prove authorship of their works. [Burns 2008]

6.4.2 Manual Download

The second scenario circumvents the app store. The software developer makes the app available to the user by offering the installation file on the Internet or on a memory card.

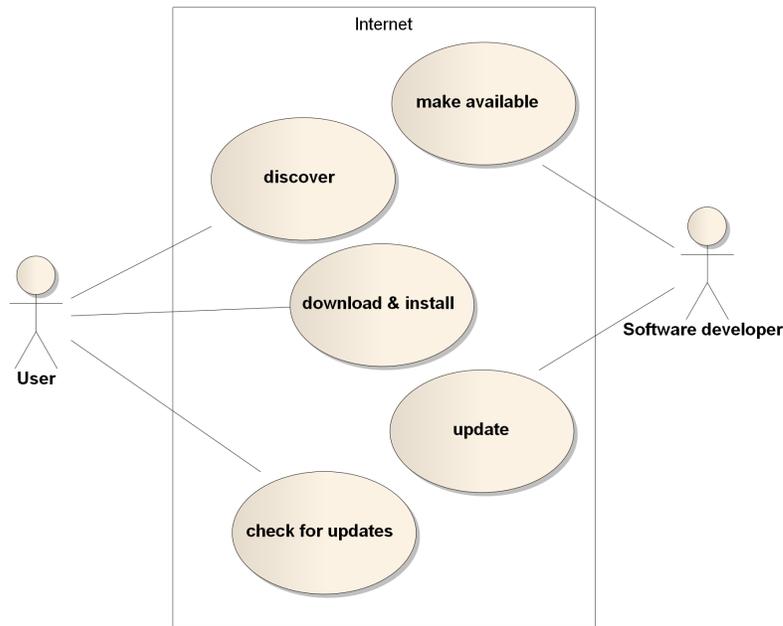


Figure 7: Use case direct app download

The user discovers the app by using a search engine or by retrieval of the download link. The installation file will be downloaded and installed. Currently, the operating systems do not support regular update checks for apps manually installed. The user has to initiate those checks manually if the software does not have an in-built update handler. Updates can be provided by the software developer simply by making new versions of the software available on the Internet.

6.4.3 Enterprise Device Management

In the third scenario, the user does not have direct control over the mobile device. Instead, the device gets managed by a device management system, which is usually operating on the private company network.

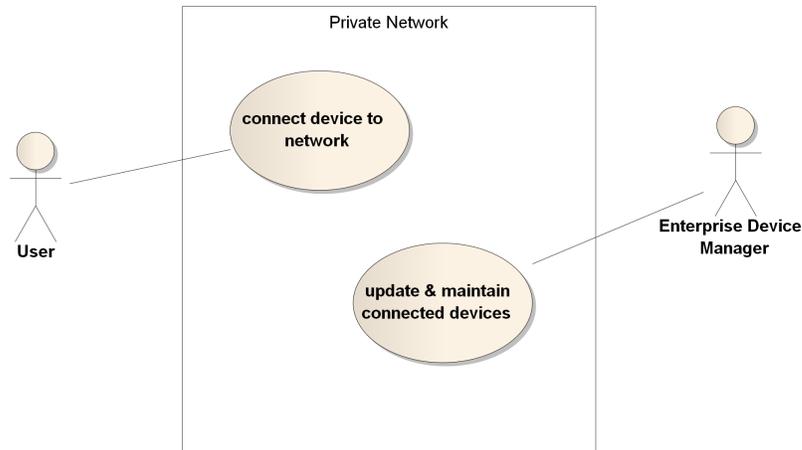


Figure 8: Use Case Enterprise Device Manager

The user simply connects the device to the network. All app and data installation and updating will be handled by the enterprise device manager. The user may have the option to select software and data, or is provided with software and data depending on his specific mission or role in the company.

6.5 Data Access & Data Loading

6.5.1 Local Data

OGC Enabled Mobile Apps can access data stored locally on the mobile devices as well as data made available on the Internet. Locally stored data shall be provided in the form of *GeoPackages*. The OGC *GeoPackage* specification ensures that spatial feature data, tiles, and raster data gets in and out of a data container consistently. An OGC Enabled Mobile App may support access to a single or to many *GeoPackages*. Data sets, packaged as *GeoPackage* can optionally be

- Downloaded directly together with the OGC Enabled Mobile App during initial install,
- Referenced from within the OGC Enabled Mobile App and downloaded on demand,
- Made available on the Internet and downloaded using third party tools and eventually manually linked to the OGC Enabled Mobile App,
- Deployed using an enterprise mobile device management system

As handling multiple *GeoPackages* on a single device causes more management overhead than a single *GeoPackage*, an interesting option is the *GeoPackage* on demand server. This server allows defining the individual data sets that will be packed into the on-demand generated *GeoPackage*.

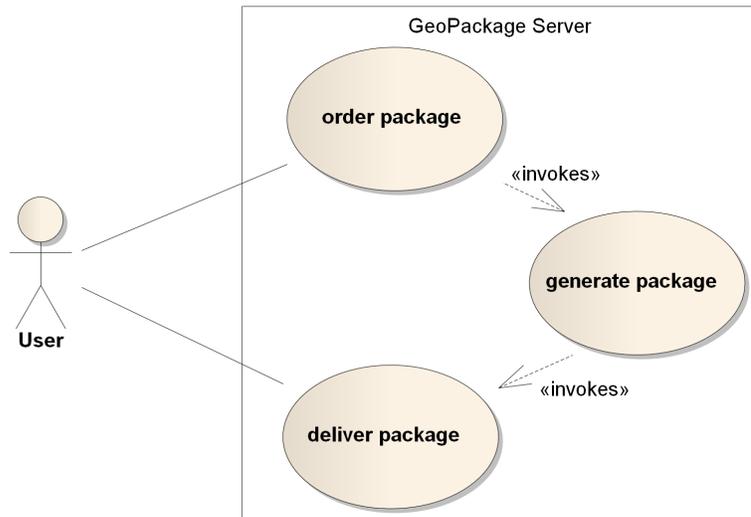


Figure 9: Use Case *GeoPackage* Packaging Server

During the order process, the user defines the required data sets to be contained in the *GeoPackage*. The *GeoPackage* server generates the *GeoPackage*, which will be made available to the user. As the generation of the *GeoPackage* might be a time consuming task, asynchronous communication between the user and the *GeoPackage* server is recommended.

6.5.2 Data Provided at OGC Web Service Interfaces

Any OGC Enabled Mobile App may communicate with other OGC Web services. This communication does not differ from desktop-based applications communicating with OGC Web services. The following Web services have been analyzed: WMS, WMTS, WFS, WCS, SOS, WPS, and SPS. The analysis checked all operations defined in the various standards for specific characteristics of mobile applications that would require changes to the standards.

6.6 Usage, Editing & Synchronization

In the editing phase, the user uses the app to add additional data items to either the local data store or an online source. Full CRUD (create, read, update, delete) support is important for a wide range of scenarios.

On a mobile device, there might be any number of *GeoPackages*. It has been agreed that it will be impossible to integrate *GeoPackages* provided by different providers into a single *GeoPackage*. Though a *GeoPackage* server can integrate different data sets into a single *GeoPackage* (and therefore ensures an easy to manage single package on the device situation), *GeoPackages* will rather stand for themselves once created. It is general consensus among OWS-9 participants that it will be impossible to get sufficient coordination between *GeoPackage* providers to create one super *GeoPackage* that

contains all data of all providers. Experiences made during past attempts to create something like a merge operation between any number of data containers have proven that merging data containers is a very complex task that can barely be handled by a standard covering a wide portfolio of situations. If the *GeoPackages* allow, updates shall be handled in all affected *GeoPackages*.

For OWS-9, it was agreed that creation, updating, and deletion of items only applies to feature data, not to any raster or tile data. Though in future, it is imaginable that a user has additional sensors, models, or other technology at hand that allow creating data that should be added to existing tile or raster data sets. For example, the camera of a phone may create an image or stream of images that get tiled on the fly and should be added to an existing tile data source; or the user makes a number of observations in the field, runs an interpolation model and updates an existing raster data source.

The current synchronization pattern, which is used for example by the geo-synchronization service (and applies to vector data exclusively), foresees changes to be made on the device to get forwarded to the sync service in order to update the original service.

Keeping track of new or changed items compared to the original data in a remote data store is another effort that becomes necessary in case that data needs to be synchronized in chunks due to limited bandwidth or even intermitted connectivity. There are different options to support this capability. Either the *GeoPackage* would contain the necessary aspects to support this functionality, or a separate sync service in combination with a dedicated data format is used. There is a preference of the latter, though this aspect requires further discussion. Keeping all potential synchronization scenarios in line is subject to another testbed, as a range of single and multiple user as well as low and high bandwidth scenarios have to be differentiated.

7 Usage Scenarios

The usage scenarios illustrate the potential usage of OGC Enabled Mobile Apps by three different types of users: The general user, the power user, and the expert. The purpose of this clause is to understand OGC Enabled Mobile Apps from the user's point of view.

Simplicity is key for the general user. The user uses the app store exploration software provided by the app store operator to discover the app, downloads and installs it, and starts using it. Either the data required to operate the app has been downloaded during the initial install, or the app features online access to data. In both cases, the general user is not even aware of the fact that the app is an OGC Enabled Mobile App that corresponds with OGC compliance servers and/or makes use of a local data container such as *GeoPackage*. Updates to feature data sets get communicated directly to the server in online mode, whereas no synchronization is supported without connectivity.

The power user understands the general principles about data containers for map tiles, raster and feature data. This user selects the required data sets either by using OGC Enabled Mobile App's capabilities, or uses a third party data generation tool to package all required data into one or many data containers. Once generated, the user loads the data and links it to the OGC Enabled Mobile App. The OGC Enabled Mobile App may support offline update caching and full two-way synchronization.

The third type of user is the expert. This type is rather the developer than the user of OGC Enabled Mobile Apps. The user can manipulate data containers directly issuing SQL commands and interacts with OGC Web services on the command line. This user is not addressed here any further.

8 Further Aspects

8.1 Certification

Certification of OGC Enabled Mobile Apps is a two-step process, as OGC Enabled Mobile Apps may be optimized for a specific subset of OGC Web services or data containers. Therefore, standard compliancy of the service or container has to be ensured first, then the communication between the OGC Enabled Mobile App and the service/container will be tested. If both work well, certification could be credited.

8.2 Security

This section discusses security and privacy aspects on a platform independent level. It does not address aspects that are specific to any particular operating system. A detailed analysis of the security models of the various platforms needs to be subject of future initiatives if necessary.

There are two main categories of mobile application risks. The category of malicious functionality is a list of unwanted and dangerous behaviors that are stealthily placed in a Trojan app that the user is tricked into installing. Instead of installing the intended software, users get hidden spyware, phishing UI, or other malicious functionality instead, such as:

- Activity monitoring and data retrieval
- Unauthorized dialing, SMS, and payments
- Unauthorized network connectivity (exfiltration or command & control)
- UI Impersonation
- System modification (rootkit, APN proxy configuration)
- Logic or time bomb

The category of vulnerabilities includes errors in design or implementation that expose the mobile device data to interception and retrieval by attackers. Vulnerabilities can also expose the mobile device or remotely provided applications used from the device to unauthorized access.

- Sensitive data leakage (inadvertent or side channel)
- Unsafe sensitive data storage
- Unsafe sensitive data transmission
- Hardcoded password/keys

It can be stated that in general, the same concepts apply as for non-mobile applications. In OWS-9, we tested a SQLite based page-level encryption library called SQL cipher. “SQLCipher is a security extension to the SQLite database platform that facilitates the creation of encrypted databases. It uses the internal SQLite Codec API to insert a callback into the pager system that can operate on database pages immediately before they are written to and read from storage” (<http://sqlcipher.net>). SQL cipher worked well in combination with *GeoPackage* files.

To better understand the security requirements for mobile devices, a full requirements analysis becomes necessary. Though a number of potential situations can be imagined, it was general consensus among OWS-9 participants that a wide range of typical security aspects can be covered by using a page level encryption mechanism with a pass phrase that needs to be provided by the user. Additional role-based authorization settings are often better handled in system designs featuring application servers as part of the communication chain between mobile apps and Web services.

8.3 File Size of Local Files

Currently, the maximum file size is often limited to 4GB on many devices due to constraints set by the file system used on the mobile device, which makes the case for multiple data packages on a single device. In the future, if exFAT or other file systems will be used more widespread, this limitation will go away. exFAT (Extended File Allocation Table) is a proprietary file system designed especially for flash drives. It is used in cases where the NTFS file system is not a feasible solution, due to data structure overhead, or where the file size limit of the standard FAT32 file system (without FAT+ extension, the limit is 4GB) is unacceptable. exFAT has a recommended maximal disk size of 512 TB, though the theoretical max is 64 ZB and a maximal file size of 16 EB. The Android operating system version 4 makes use of exFAT, for example.

Bibliography

- [Burns 2008] Burns, J. (2008) Developing Secure Mobile Applications for Android. An introduction to make secure Android applications. isec partners. Online accessed on Oct. 2, 2012 at:
http://isecpartners.com/files/isec_securing_android_apps.pdf
- [Templeman 2009] Templeman, R., Rahman, Z., Crandall, D., Kapadia, A. (2012) PlaceRaider: Virtual Theft in Physical Spaces with Smartphones. Online accessed on Oct. 3, 2012 at:
<http://arxiv.org/pdf/1209.5982v1.pdf>