agroXML: Enabling Standardized, Platform-Independent Internet Data Exchange in Farm Management Information Systems

Mario Schmitz[1], Daniel Martini[1], Martin Kunisch[1], and Hans-Jürgen Mösinger[2]

[1] Association for Technology and Structures in Agriculture, Bartningstraße 49, 64289 Darmstadt, Germany
m.schmitz@ktbl.de, d.martini@ktbl.de, m.kunisch@ktbl.de
[2] Software AG, Uhlandstraße 9, 64297 Darmstadt
Hans-Juergen.Moesinger@softwareag.com

Abstract. agroXML is a standardized language for data exchange in agriculture. It is based on the eXtensible Markup Language (XML) using XML Schema as its definition language. agroXML is used to submit data from farm management information systems to external partners, like e. g. product processing industries in the food supply chain or agricultural service providers. In addition, data about operating supplies like fertilizers or pesticides can be made available to the farmer by their respective suppliers. In the future, using XML linking technologies might provide a dynamic and flexible mechanism to link documents to such external information sources.

1 Introduction

Documentation of agricultural practices is becoming more and more of an issue for farmers. On the one hand, they are increasingly obliged to it by legislation, on the other hand, integrative planning of agricultural production requires thourough information about measures and events in the past. For data exchange processes in agriculture up to now only individual interfaces between different communication partners were available. Even if the farmer had electronic systems to record production data, the required data had to be transferred by hand from one software to another or from screen into paper forms. A standardized system for electronic data exchange offers new possibilities for information-directed agricultural production increasing sustainability and keeping adversary effects to the environment at a minimum. By allowing for an integrated view of farm production data and other data like e. g. climate or geographic data, measures can be adapted to different conditions, optionally leveraging algorithms or expert systems provided by third parties. Using agroXML as a data exchange language, these procedures are facilitated and the individual interfaces between communication partners are substituted by universally usable data exchange processes.

2 The agroXML Schema, Profiles and Content Lists

The agroXML schema is a model of the real-world objects and their attributes and of the processes in agricultural production. It is based on eXtensible Markup Language [1]. Definition of document structures is done using the XML Schema Language from the W3C [2]. Its architecture is data-centric and currently monolithic. At the moment, agroXML can describe data for plant production. An agroXML document is divided into four parts: A header providing information about the farm in general like e. g. address, name of farm manager etc., a block of data about the fields, like e. g. area and geographic coordinates, a further block of data about the cultivation on different fields, like e. g. the plant species, catch crops etc. and finally, data about the individual measures carried out: fertilization, seeding, pest control, tillage etc.

On the one hand, agroXML can be used to generate consistent stand-alone XML documents. But following the extensibility paradigm of XML, it also offers a collection of data types and elements reuseable and embeddable in other documents.

To facilitate integration with geographic services, spatial vector data are modeled in agroXML reusing constructs from the Geography Markup Language (GML). The reuse is achieved by creating a profile of GML according to [3] and importing this profile together with the GML-namespace into agroXML. GML was chosen as it is the only widely acknowledged XML-based format for vector data. Whereas other more or less open binary formats (like e. g. Shapefiles or TIGER in the U. S.) are available, they are not easy to integrate in a manner allowing tight interconnections and references between the data contained within and agricultural data represented in agroXML.

Practical feasibility in different computing environments is an important factor while developing models to integrate other XML vocabularies. Farm management information systems are written in different programming languages. Components of these management systems providing certain functionality run on a variety of hardware platforms from handhelds to powerful servers for web applications. While in theory combination of different vocabularies seems desirable, in practice it often leads to large, bulky constructs unmanageable by common XML tools. Simplicity, clarity and generality are key properties of well engineered IT systems [4]. For these reasons, vocabularies, which are too extensive and do not allow restriction like e. g. GML does, are currently out of scope for agroXML.

The schema is available at http://www.agroxml.de/schema under the W3C open source licence. Since June 2007, Schema development is carried out in the English language, so that further versions will be accessible and understandible to developers from non-germanspeaking countries.

For a specific data exchange case, application profiles define the obligatory elements. They allow for extraction of a subset of a larger datatype and element collection. To create a profile, elements are selected from the agroXML schema and the necessary restrictions applied to the data types. The basic rule is, that an instance conforming to a profile must also conform to the schema as a whole.

It is possible to turn optional elements into mandatory ones, to set attributes to fixed values and to restrict the cardinality of any particle. The process is described in detail in [3]. Using application profiles, lean and clear instances can be generated.

Another component of agroXML are the content lists. They provide the functionality of XML Schema enumerations, however the mechanism of how they are included in the schema allows to add to their content dynamically without effecting a change in the schema itself. In addition, they not only contain the enumeration values themselves but also a name and a description of the item at hand. The lists conform to a unified schema and can be downloaded at http://www.agroxml.de/content. Several lists exist containing e. g. soil types, machine types, fertilizer types, pesticides and plant variety names. Where possible, content for the lists is obtained from the respective official agencies, like e. g. the plant variety offices. Software systems implementing agroXML can either use a local copy of the content lists for filling instances or use the version on the web. Different caching strategies are possible to ensure a recent data pool even if the internet connection is only intermittent. Technological details concerning integration of list content into instances and provision of recent lists are described in [5].

It is important to note, that due to the dynamic integration into an XML instance (lists are referenced by their Uniform Resource Locator), it is possible to include different lists than the ones provided at http://www.agroxml.de/content for special purposes or containing language- or country-specific content.

3 Applications

An interface which can write or read agroXML is the prerequisite. The transport of an instance on the internet can be conducted using standard protocols like the hypertext transfer protocol (HTTP), the file transfer protocol (ftp) or the simple mail transfer protocol (SMTP). Exchange is currently done in a document-oriented manner: a complete agroXML document is transferred in a single file. For the transfer of only selected items in a dialog-enabled system, query mechanisms are needed. For this purpose, web services based on a service oriented architecture (SOA) can be used. But such systems are complex and resource-hungry during runtime. A lightweight alternative would be to use a standardized XML query language, e. g. XQuery [6] or using services following the paradigm of representational state transfer (REST) [7].

An example of an application using agroXML in association with geo-data is the system developed by the KTBL in the course of a feasibility study supported financially by the Federal Ministry of Agriculture and the Landwirtschaftliche Rentenbank [8]. The prototypical application provides a catalogue with a set of adresses of mapservers which provide useful data like outlines of areas with land use restrictions. It enables farmers to collect geospatial data associated to their fields from official authorities and embed them in agroXML documents readable by farm management information systems.

4 Outlook

In the works is further upgrading of geo-data (also raster data) functionality as well as addition of elements for livestock farming and cultivation of vegetables and fruit. This increasing demand from agricultural sectors other than only plant production leads to technological issues, which are currently dealt with. Especially, a schema architecture and design to allow modularization and extensibility while at the same time keeping internal consistency, is needed. The goal of the work is to provide a schema which can be used only in part to implement the datatypes needed for a specific application, while not breaking application interoperability.

Data about operating supply items, like e. g. fertilizers or pesticides are copied directly into XML instances by the farm management information systems. Suppose a company changes the nitrogen content of a certain fertilizer. If a farmer is not aware of the change, he will transmit incorrect data in further transactions. So, in most cases, a better model would be to leave this information at the place where it is produced, i. e. in the example above on a web server at the fertilizer producer, and use generic link mechanisms like the XLink Standard [9], to only reference the information. URIs offer an excellent system to provide globally unique identifiers. Ressources like fertilizers could be described using agroXML element hierarchies. Not only would this enable real distributed data storage and ensure recent information, but would also enhance the possibilities to harvest and link data to build real knowledge bases. Key factor for the success of such an architecture is a simple and easily adoptable standard.

References

1. World Wide Web Consortium: Extensible Markup Language (XML) 1.1. 2nd edn. (2006) http://www.w3.org/TR/xml11/.
2. World Wide Web Consortium: XML Schema Part 0: Primer. 2. edn. (2004) http://www.w3.org/TR/xmlschema-0/.
3. Open GIS Consortium, Inc.: OpenGIS Geography Markup Language (GML) Implementation Specification. (2004)
4. Kernighan, B.W., Pike, R.: The Practice of Programming. Professional Computing Series. Addison-Wesley (1999)
5. Martini, D., Frisch, J., Kunisch, M.: agroXML-Inhaltslisten – Konzeption und Inhalte. In: Proceedings 27. GIL-Jahrestagung. (2007) 139–142
6. World Wide Web Consortium: XQuery 1.0: An XML Query Language. (2007) http://www.w3.org/TR/xquery/.
7. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. PhD thesis, University of California, Irvine (2000)
8. Geißner, G., Grimm, S., Frisch, J.: Geodaten in der Landwirtschaft – Bereitstellung mit agroXML. In: agroXML – Informationstechnik für die zukunftsorientierte Landwirtschaft. Volume 454 of KTBL-Schrift. Munich (2007) 103–118
9. World Wide Web Consortium: XML Linking Language (XLink) Version 1.0. (2001) http://www.w3.org/TR/xlink/.