

# Implementierung einer agroXML-Schnittstelle für landwirtschaftliche Prozessdaten

Georg Steinberger<sup>1</sup>, Christine Spietz<sup>2</sup>, Matthias Rothmund<sup>1</sup>,

<sup>1</sup> Fachgebiet Technik im Pflanzenbau  
Technische Universität München  
Am Staudengarten 2  
85354 Freising

<sup>2</sup> Kompetenzzentrum für Innovative Informationssysteme  
Fachhochschule Bingen  
Berlinstr. 109  
55411 Bingen  
georg.steinberger@wzw.tum.de  
spietz@fh-bingen.de  
matthias.rothmund@wzw.tum.de

**Abstract:** The aim of agroXML as a data exchange format is to network process and business information along the food and value chain. In this paper the integration of an agroXML interface in the Agricultural Process Data Service (APDS) is presented. After evaluating different implementation methods the implementation itself resulted in recommendations for further development of agroXML.

## 1 Einleitung und Problemstellung

Die Vernetzung von Prozess- und Geschäftsinformation entlang der Lebens- und Futtermittelkette von der Urproduktion bis zum verarbeiteten Produkt führt zu einer Reihe unterschiedlicher Anforderungen an Inhalt und Struktur der transportierten Daten. Weitere Anforderungen ergeben sich durch notwendige Informationsflüsse von und zu dieser Produktionskette, wie bei der Zulieferung von Betriebsmitteln, Einbindung von Beratern und Lohnunternehmern und durch die Informationspflicht gegenüber staatlichen oder beauftragten Kontrollinstanzen. Schließlich sollen innerhalb des landwirtschaftlichen Produktionsprozesses erfasste Daten vor allem dem Betriebsmanagement selbst für unterschiedlichste Anwendungen wie Buchführung, Abrechnung, Schlagkartei oder die teilflächenspezifische Bewirtschaftung zur Verfügung stehen [RA04]. Während innerhalb des landwirtschaftlichen Betriebes bei Nutzung elektronischer Steuerungs- und Dokumentationssysteme vor allem große Mengen von georeferenzierten Maschinen-Prozessdaten transportiert und verarbeitet werden müssen, spielen für den Austausch über die Betriebsgrenzen hinweg eher Informationen über geschäftliche Prozesse, jedoch zunehmend unter Einbeziehung von Geodaten und aus Prozessdaten ermittelten Kenngrößen, eine Rolle.

Für den Austausch von Prozess- und Auftragsdaten zwischen dem Farm Management Information System (FMIS) und mobilen landwirtschaftlichen Arbeitsmaschinen wird derzeit ein XML-basiertes Datenaustauschformat als Bestandteil des Standards für elektronische Kommunikationssysteme ISO 11783 (ISOBUS) eingeführt. Dieses Format ist in Struktur und Inhalt für die Übertragung von Auftragsinformationen und georeferenzierten Steuerungsanweisungen vom FMIS zur Arbeitsmaschine als auch für die Rückführung der auf der Maschine erfassten georeferenzierten Prozessdaten zum FMIS optimiert. Daher wurde bei der Erstellung des Datenaustauschformats insbesondere auf einen minimalen beschreibenden Teil und eine möglichst flache XML-Struktur geachtet. Im Gegensatz dazu wird bei der derzeitigen Entwicklung von agroXML als Austauschformat für landwirtschaftliche Geschäftsprozesse auf eine hohe Flexibilität bezüglich unterschiedlicher Anwendungsbereiche von der Betriebs- und Produktspezifizierung bis hin zur Einbindung von Geo- und Prozessdaten geachtet [MSK06]. Dies führt zu einem deutlichen Anstieg des Anteils an Strukturinformation und zu einer tiefen hierarchischen Gliederung der XML-Struktur. Erste Anwendungen beider Datenschnittstellen sind bei landwirtschaftlichen Softwareherstellern zu finden. Jedoch wird die Erfassung und Verwendung zeitlich und räumlich hochaufgelöster Prozessdaten mittels automatisierter Systeme während des Arbeitsprozesses in bisherigen Ansätzen kaum berücksichtigt. Das Management georeferenzierter Prozessdaten als Bestandteil einer automatisierten Dokumentation erfordert neue Konzepte und Techniken.

Auf der 26. GIL-Jahrestagung 2006 wurde mit dem Agricultural Process Data Service (APDS) ein Konzept für die Speicherung, Verarbeitung und Nutzung von georeferenzierter Prozessinformation vorgestellt [SRA06]. Die Basis hierfür bilden neben einem automatisierten Datenerfassungssystem auf mobilen Arbeitsmaschinen vor allem ein serverbasiertes Datenmanagement und die Verwendung moderner standardisierter Webtechnologien. Ziel dieses Service ist es, die Welt der mobilen Arbeitsmaschinen (ISOBUS) mit der landwirtschaftlichen Geschäftswelt zu verbinden und die in den Prozessdaten enthaltene Information für diese zu erschließen. Zu diesem Zweck wurde nun der APDS mit einer agroXML-Schnittstelle ausgestattet, was ein „Mapping“ der relevanten Inhalte aus ISOBUS- in das agroXML-Format erfordert. Aufgrund der völlig unterschiedlichen Datenstrukturen und der unterschiedlichen Anforderungen an die Präsentation des Inhalts stellt dies eine komplexe Aufgabe dar, die innerhalb des APDS durch die zusätzliche Erstellung einer agroXML-ähnlichen Objektstruktur gelöst wurde. Im Folgenden werden diese Implementierung beschrieben und Alternativen aufgezeigt.

## **2 Methodische Varianten**

Der APDS dient insbesondere der Datenhaltung von Prozessinformationen und der Extraktion der darin enthaltenen Managementinformationen. Um den APDS mit einer agroXML-Exportschnittstelle auszustatten, werden geeignete Datenstrukturen sowohl zum Zugriff auf die Datenbank als auch zur Realisierung der Transformation nach agroXML benötigt. Das Mapping der originären Prozessdaten auf agroXML vollzieht sich demnach auf mehreren Ebenen.

Zur Realisierung der agroXML-Schnittstelle stehen grundsätzlich unterschiedliche Methoden zur Verfügung. Die wohl verbreitetste Vorgehensweise ist in diesem Zusammenhang die Verwendung des Model-View-Controller-Architekturmusters zur Aufteilung des Softwaresystems in die drei Einheiten Datenmodell, Präsentation und Programmsteuerung. Dabei könnte im vorhandenen Datenmodell für die Abbildung von Datenbankinhalten Funktionalität ergänzt werden, die ein direktes Schreiben der Inhalte in ein agroXML-Dokument ermöglicht. Das Mapping der Daten ist dann integraler Bestandteil des Serialisierens<sup>1</sup> in das agroXML-Format.

In der Regel sollte jedoch eine Trennung des Klassen-Modells für den Zugriff auf die Datenbank und die Datenobjekte zur Präsentation der agroXML-Objekte erfolgen. Hierbei werden auf der Basis des agroXML-Schemas neue Klassen erzeugt, deren Objekte in den meisten Programmiersprachen mit Hilfe von Standardbibliotheken serialisiert werden können. Die Klassen selbst lassen sich zum Teil mit Hilfe von Code-Generatoren aus dem agroXML-Schema automatisch erzeugen. Das Mapping erfolgt somit nicht mehr gleichzeitig mit dem Schreiben des XML-Dokuments, sondern davon getrennt, und zwar ausgehend von den bereits vorhandenen Objekten für den Datenbankzugriff zu den neu erzeugten der agroXML-Schnittstelle. Damit ist der Quellcode der Schnittstelle gekapselt und kann in der weiteren Entwicklung getrennt von der restlichen Applikation gehandhabt werden. Denkbar wäre auch eine dritte Variante, in der XML-Dokumente in einem proprietären Format, angelehnt an die Datenstruktur der Applikation, geschrieben und diese dann mit Hilfe von XSLT in agroXML übersetzt werden.

### 3 Implementierung

Bei der Implementierung der Schnittstelle für den APDS wurde die zweite oben beschriebene Variante gewählt. Die in der verwendeten Programmiersprache Python vorhandenen, einfachen Bibliotheken zur XML-Serialisierung waren auf Grund der komplexen Struktur des agroXML-Schemas nicht zu nutzen. Deshalb wurden die notwendigen neuen agroXML-basierten Klassen manuell erstellt. Um ein agroXML-Dokument zu erzeugen, werden dabei im Programmablauf zunächst die vorhandenen Klassen der Applikation bzw. deren Objekte genutzt, um die benötigten Daten aus der Datenbank abzurufen und vorzubereiten. Anschließend werden deren Inhalte in die neuen agroXML-Objekte übertragen und das agroXML-Dokument geschrieben. Der zugrunde liegende Aufwand steigt hierbei sowohl mit zunehmender Komplexität der Struktur von agroXML als auch in Abhängigkeit der Modellunterschiede zwischen der bereits bestehenden Applikation und der Schnittstelle. Um die Möglichkeiten der automatischen Code-Generierung abschätzen zu können, wurden auch Klassen mit der Java Architecture for XML Binding (JAXB2) generiert. Mit der vorläufigen Version 1.2\_draft (Stand 06/06/2006) von agroXML konnten hier die Klassen, anders als bei Verwendung der Programmiersprache Python, automatisch generiert werden.

---

<sup>1</sup> Unter Serialisierung wird eine sequenzielle Abbildung von Objekten in eine externe Darstellungsform verstanden (Wikipedia, 05.12.2006).

<sup>2</sup> <http://java.sun.com/webservices/jaxb/index.jsp>

## 4 Schlussfolgerungen und Ausblick

Um eine hohe Akzeptanz bei Softwareanbietern und damit eine rasche Verbreitung von agroXML zu fördern, sollten nicht nur die Abbildung von Inhalten in agroXML sondern auch die Konsequenzen für die Implementierung der Schnittstelle bei der Entwicklung berücksichtigt werden. Der Aufwand bei der Implementierung einer agroXML-Schnittstelle, aufbauend auf einer bestehenden Daten- und Programmstruktur kann zwar durch die Nutzung von zusätzlichen Bibliotheken zur XML-Serialisierung verringert werden. Meist ist jedoch auch ein großer Teil geschriebenen Codes notwendig, um die Inhalte in das agroXML-Format zu überführen. Der Aufwand dabei ist stark abhängig von der Komplexität des agroXML-Schemas und der applikationsinternen Datenstruktur. Der Einsatz von Klassengeneratoren ist möglich, wird aber durch eine zunehmende Komplexität des XML-Schemas erschwert.

Das Generieren von Klassen mit JAXB für Java wurde erfolgreich getestet. Diese Klassen könnten zusätzlich zum agroXML-Schema veröffentlicht werden, um Softwareanbietern die Umsetzung eigener Schnittstellenimplementierungen zu erleichtern. Bei der weiteren Entwicklung wird es notwendig sein, die Auswirkungen von Versionsupdates zu beachten. Weitere Design-Patterns (Strukturmuster) in agroXML müssen dabei die Wiederverwendbarkeit vorhandenen Codes sicherstellen.

## Literaturverzeichnis

- [MSK06] Martini, D; Spietz, C; Klöpfer, F: Darstellung teilflächenspezifischer Maßnahmen in agroXML als Voraussetzung für die Verwendung im Bereich Precision Farming. In (Wenkel, K-O et al. Hrsg.): Lecture Notes in Informatics, Referate der 26. GIL Jahrestagung, Potsdam, 2006; S. 161-164.
- [RA04] Rothmund, M; Auernhammer, H: Neue Informationsmanagementsysteme für die Pflanzenproduktion und die Arbeitswirtschaft. 14. Arbeitswissenschaftliches Seminar des VDI-MEG Arbeitskreis Arbeitswissenschaften im Landbau, FAT-Schriftenreihe Nr. 62, Tänikon, Schweiz, 2004, S.7-18.
- [SRA06] Steinberger, G; Rothmund, M; Auernhammer, H: Serverbasierte Verarbeitung von Prozessdaten gemäss ISO11783 (ISOBUS). In (Wenkel, K-O et al. Hrsg.): Lecture Notes in Informatics, Referate der 26. GIL Jahrestagung, Potsdam, 2006; S. 289-293.

Danksagung: Die Modellierung und Implementierung von APDS und agroXML-Schnittstellen erfolgt im Rahmen des Projekts *pre agro*, das vom Bundesministerium für Bildung und Forschung (BMBF) gefördert wird.

Thanks to Christine for all the agroXML work!