Project no.	**212117**	Project acronym:	**FUTUREFARM**
Project title:	**Integration of Farm Management Information Systems to support real-time management decisions and compliance of management standards**		
Instrument:	**Collaborative project**		
Start date of project:	**1st January 2008**	Duration:	**36 months**
Thematic Priority:	**THEME 2 FOOD, AGRICULTURE AND FISHERIES, AND BIOTECHNOLOGY**		

Deliverable 4.1.2 Revision: Final
Machine-readable encoding for definitions of data required to assess compliance to agricultural management and crop production standards

Due date of deliverable: **31/12/2009** Actual submission date: **15/1/2010**

Work package 4: **Knowledge Management in the FMIS of Tomorrow**

Organisation name of lead beneficiary for this deliverable: **Rostock University**

Authors: **Edward Nash (UR), Raimo Nikkilä (TKK), Sascha Kluger (Agrocom), Kai Oetzel (Agrocom), Liisa Pesonen (MTT), Ilkka Seilonen (TKK), Jens Wiebensohn (UR)**

Accepted by Claus Sorensen, 7/2/2010

Accepted by Simon Blackmore, 23/2/2010

Project co-funded by the European Commission within the Seven Framework Programme (2007-2013)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Executive Summary

In order to reliably assess compliance to agricultural management and crop production standards and regulations, a large body of data is required. In order that farmers may gather the correct data to self-check compliance, and that they can produce the necessary data in order to proove compliance to controlling bodies, it is desirable, or even essential, that the exact data required be explicitly and unambiguously defined.

Although this aim is easily defined, no simple solution is available. Two central problems may be identified:

1. Each standard, or rule within that standard, does not in general explicitly state how compliance should be assessed: often there is no clear single data item, or set of data, which is required, or there may be multiple possible alternatives.
2. There is no universal vocabulary or data dictionary for agriculture. Each standard uses its own vocabulary, which may be defined as part of that standard, whilst each FMIS also has an internal data model which may, or more likely may not, correspond to the vocabulary used by any particular standard. Similarly, current data exchange formats for agriculture have a certain vocabulary which does not necessarily correspond to any other. In technical terms, multiple ontologies are in use, which are not always directly comparable.

As a proposed partial solution to this complicated problem, a two-stage system is described in this document:

1. The machine-readable definition of the rules (FutureFarm Deliverable 4.1.1) reference one or more formal ontologies to specify the concepts which are required to define the rule.
2. The FMIS component responsible for evaluating the rules (specified in FutureFarm Deliverable 4.3) provides a mechanism for translating the concepts from the ontologies to a specific data format which is implemented by the FMIS (e.g. agroXML or a software-internal data structure).

1. Introduction and scope

A large and varied collection of data is required in order to assess compliance to agricultural management and crop production standards. In order to enable self-assessment by farmers and to assist them in preparation for external evaluations by controlling bodies., it is necessary that this data is explicitly and unambiguously defined so that it can be collected and stored. However, this is currently not generally the case: the rules which are defined by standards describe what the farmer may or may not do, and only in some cases how compliance should be measured. Even where specific criteria are given, this is frequently couched in somewhat general terms which cannot be easily mapped to concrete data values which may be held in the FMIS database or represented in a standard transfer format such as agroXML.

Since it is not realistic to expect that standards prescribe particular data formats or structures to be used for assessing compliance, a format-neutral, formal machine-readable definition of the terms used within the standards is required. Individual terms may be explicitly defined within a standard (e.g. within a preamble or clauses of a regulation), or general terms may be defined and used industry-wide. A single, centrally-managed list of terms is however unreasonable for a number of reasons:

- The large number of term which must be defined.
- New standards will frequently require the definition of new terms.
- Terms may be used in conflicting ways in different standards, making a harmonisation of terms hard.
- Not all standards are published in the same natural language (e.g. English), and so at a minimum, a multilingual list (or one list per language) would be required.

As a solution, the use of multiple formal ontologies is proposed. How this solution may be used is discussed in section 2.

As an additional aspect, based on the component-based FMIS architecture proposed within the FutureFarm project (Figure 1), for software which should be able to automatically assess compliance to any arbitrary agricultural rules, it is necessary that the component performing the compliance assessment (the Rules App, specified in FutureFarm Deliverable 4.3 (Nash et al, 2009)) be able to identify the concrete data elements which should be retrieved from the database or passed from the Rules Manager. The requirements for identifying data elements are however different depending on the implementation form and so no single solution is possible. Common is however a need to map from terms defined in the ontologies to local data elements. Both of these themes are discussed in this report.

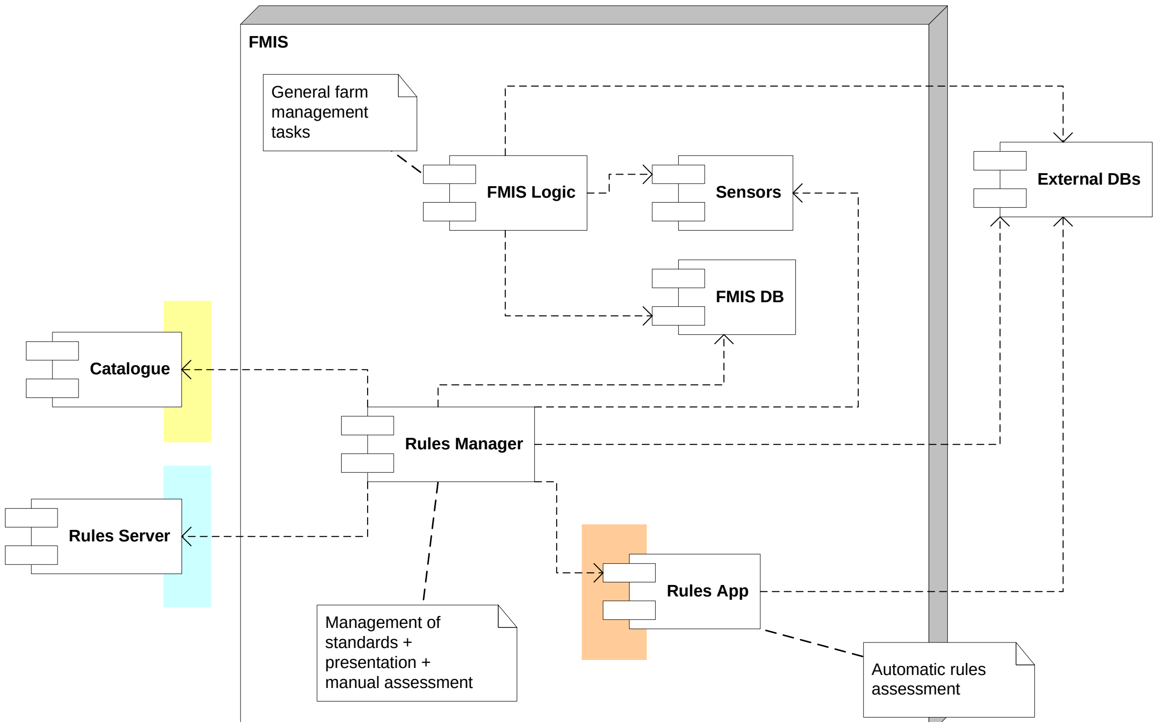

Figure 1. Architecture of FMIS rules subsystem proposed in FutureFarm

2. Ontologies

An ontology may, in computing science, be used to model a particular knowledge domain (e.g. agriculture in general, or one specific management standard). Essentially, an ontology

- "…defines (specifies) the concepts, relationships, and other distinctions that are relevant for modelling a domain.
- The specification takes the form of the definitions of representational vocabulary (classes, relations and so forth), which provide meanings for the vocabulary and formal constraints on its coherent use." (Gruber, 2009)

From this definition it is clear that an ontology can be used to represent both the individual concepts used in defining agricultural standards and the relationships between these concepts. The latter is particularly important when it is established that multiple ontologies must be handled by the system, as it allows for some degree of contextual reasoning when dealing with new ontologies. Given that effectively multiple ontologies are always in use in the proposed system, even where only one standard with a single ontology is in use, as unless the FMIS data model maps directly to the ontology of that standard, which is unlikely, a minimum of two ontologies are present; that of the standard and the (probably implied) internal ontology of the FMIS. The ability to deal intelligently with new ontologies is a critical assumption for the success of the proposed system, as any standard accessed from the Rules Server should be capable of being assessed by the Rules App, and it can not be assumed that all ontologies in use are known in advance.

In order to represent ontologies, the Web Ontology Language (OWL, W3C 2004), a recommendation of the W3C has established itself as the standard methodology.

Although OWL 2.0 has recently been published (November 2009), only OWL 1 is considered here.

Three varieties of OWL are defined:

1. OWL Full, which supports a very broad range of features, but which is not supported by logical reasoning tools available in software. It is therefore not suitable for application-oriented use as is required here.
2. OWL DL (Description Logic), which is a sublanguage of OWL with a number of constraints on usage compared to OWL Full. OWL DL is intended to support only constructs that may be used with software reasoning engines for description logics, and so may be considered a good choice for practical applications.
3. OWL Lite is a sublanguage of OWL DL which is designed to provide a minimal useful subset of langauge features. It places a number of further restrictions on what may be achieved. It may be a good choice for relatively simple applications.

Although most of what is required for the FutureFarm system can probably be accomplished with OWL Lite, it is recommended that OWL DL be considered due to its greater expressive power. The lack of computability of OWL Full means that it must not be considered in this case.

OWL defines both human- and machine-readable (XML) formats for presenting the model. Software for creating OWL models, such as the free software Protégé[1] also allows a graphical visualisation of an ontology. Once the terms are defined, the OWL model may be referenced in the definition of the rule using RIF (Rules Interchange Format) as part of the machine-readable encoding for definitions of agricultural standards (FutureFarm Deliverable 4.1.1, Nash et al 2009). The actual OWL model may be directly embedded in the definition of the standard or referenced by a URL. The individual OWL concepts which provide the data required to evaluate compliance to a rule may be listed as metadata for the rule using the `<requiresData>`-element defined in the schema. Alternatively, it should be possible for the RulesApp component to extract the OWL terms from the RIF-encoded rule.

2.1 Example of an ontology for an agricultural standard

As an example of how OWL may be applied to model the terms defined by an agricultural standard, we here present an ontology derived from the German ”Düngeverordnung” (DüV, Fertilisation Regulation), the national implementation of the EU Nitrate Directive, and forming part of the Cross-Compliance criteria.

In the DüV, a fertiliser with significant nutrient content is defined as a subcategory of fertiliser having either a phosphate content greater than 0.5% or a nitrogen content greater than 1.5%. From this four terms may be identified:

- “Fertiliser” as a more general category.
- “Fertiliser with significant nutrient content” as a more specific category.
- The properties “nitrogen content” and “phosphate content” which may be used to distinguish fertilisers between those having a significant nutrient content and those which do not.

[1] http://protege.stanford.edu/

The following OWL fragment defines this formally. Note that a workaround is required for comparative operators (e.g. greater than) in OWL 1, here accomplished using an ontology defining percentage ranges. With OWL 2 this will no longer be necessary and a straightforward mathematical comparison on data property values may be used.

```
Declaration(Class(duevo:FertiliserWithSignificantNutrientContent))
EquivalentClasses(
    duevo:FertiliserWithSignificantNutrientContent
    ObjectUnionOf(
        ObjectAllValuesFrom(duevo:hasPhosphateContent
            percentages:PercentGreater0.5)
        ObjectAllValuesFrom(duevo:hasNitrogenContent
            percentages:PercentGreater1.5)))
SubClassOf(agrovoc:Fertiliser)
DisjointClasses(duevo:FertiliserWithNoSignificantNutrientContent)
```

(Figure 2) visualises this as part of a hierarchy of fertiliser categories and related terms defined by the DüV.

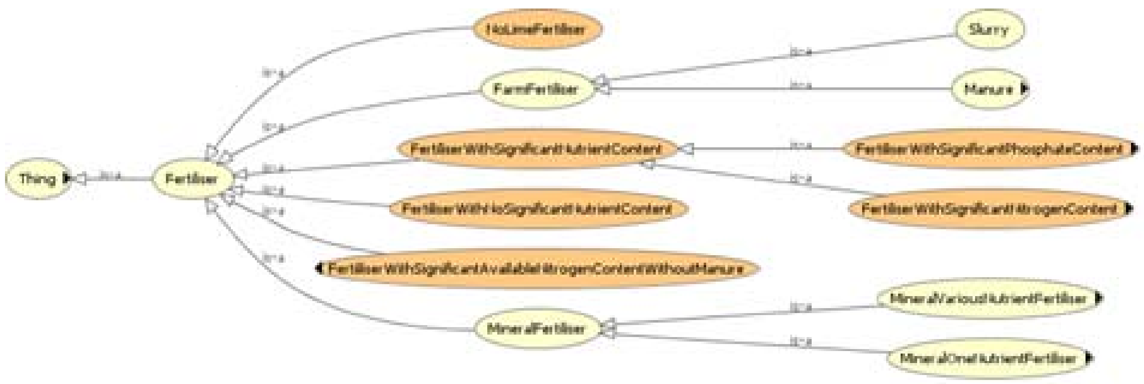

Figure 2. Visualisation of OWL model of fertiliser subcategories

3. Data formats and specifying concrete data elements

As previously discussed, the specification of concrete data elements to be used for evaluation of compliance does not form part of the specification of the standard, and the ontologies used for defining the concepts are not directly related to any data model used in the FMIS. This section therefore can only present some basic examples as to how concrete data elements and structures may be specified in general terms, based on examples from other systems.

As an example set of data we will consider that which is required in order to assess compliance to the rule in the German Fertiliser Regulations (DüV) whichs state that fertilisation with fertilisers with a significant level of available nitrogen is forbidden in the period 1st November – 31st January on cropland and 15th November – 31st January on grassland. To evaluate compliance, the following information regarding the fertilisation is required:

1. Chemical composition of fertiliser applied.
2. Date of operation.

3. Spatial region of operation (allowing a GIS lookup of field type), or field type applied to.

For a strongly-typed language such as Java, the requirements to have a class which can deliver these three pieces of information may be defined by declaring an Interface class specifying the required methods and their data types, where these data types may be further interfaces, e.g.

```
package org.futurefarm
public interface FertiliserApplication {
    public Fertiliser getAppliedFertiliser();
    public Date getDateOfOperation();
    public Geometry getRegionOfOperation();
}
```

The FMIS may then implement this interface in any class, which may subsequently be used for transferring the data to the component performing the compliance evaluation. In order to specify the data to be transferred, the scoped interface name (e.g. `org.futurefarm.FertiliserApplication`) may be used together with Java's reflection features (see e.g. McCluskey, 1998).

For an XML-based transfer format, such as agroXML, one approach would be to use a schema fragment to produce a profile of the XML grammer representing only the data required. However, this has the disadvantage of removing the link to the original schema definition and may in some cases require a significant amount of duplication. As an alternative, XPath (W3C, 1999) may be used. XPath is a language for addressing parts of an XML document. Based on a root element, both subsequent elements and attributes may be specified using XPath expressions. The agroXML encoding for the information required for our example could therefore be specified as:

Root element:

```
agroxml:WorkProcess
```

Child elements:

```
agroxml:SpatialData/gml:Polygon
agroxml:StartDateTime
agroxml:EndDateTime
agroxml:Fertilization/agroxml:Fertilizer/
    agroxml:SubstanceOfContentsDetails/agroxml:SubstanceOfContents
agroxml:Fertilization/agroxml:Fertilizer/
    agroxml:SubstanceOfContentsDetails/agroxml:Concentration
```

(Note that there is currently no mechanism in agroXML for encoding whether a region is cropland or grassland, or for the solubility of a certain substance, both of which would be required for the robust assessment of the rule. It is assumed that the land use of the region could be retrieved based on the `SpatialData` element of the `WorkProcess`, although this implies that a suitable dataset is available to the RulesApp.)

This method of XPath-based element specification is similar to that used in the Schematron XML grammar definition language, and is used for the RESTful Rules App interface specification given as an appendix in FutureFarm Deliverable 4.3 (Nash et al., 2010).

4. Mapping ontologies to data formats

Grubin (2009) states that:

> ”ontologies are typically specified in languages that allow abstraction away from data structures and implementation strategies; in practice, the languages of ontologies are closer in expressive power to first-order logic than languages used to model databases... The key role of ontologies with respect to database systems is to specify a data modeling representation at a level of abstraction above specific database designs (logical or physical), so that data can be exported, translated, queried, and unified across independently developed systems and services.”

This quote neatly encapsulates the advantage of ontologies when used for identifying required data in describing rules, whilst also describing the main disadvantage: the ontology does not directly specify the data, rather it describes it. Where the situation in the FMIS diverges from that described in the quote is that the ontology (or ontologies) used in specifying the standard are not necessarily the same as that (or those) which describe the data in the FMIS or in an external standard data transfer format such as agroXML. Furthermore, it can probably not be assumed that a formal ontology exists for the FMIS internal data model, or even standard data transfer formats.

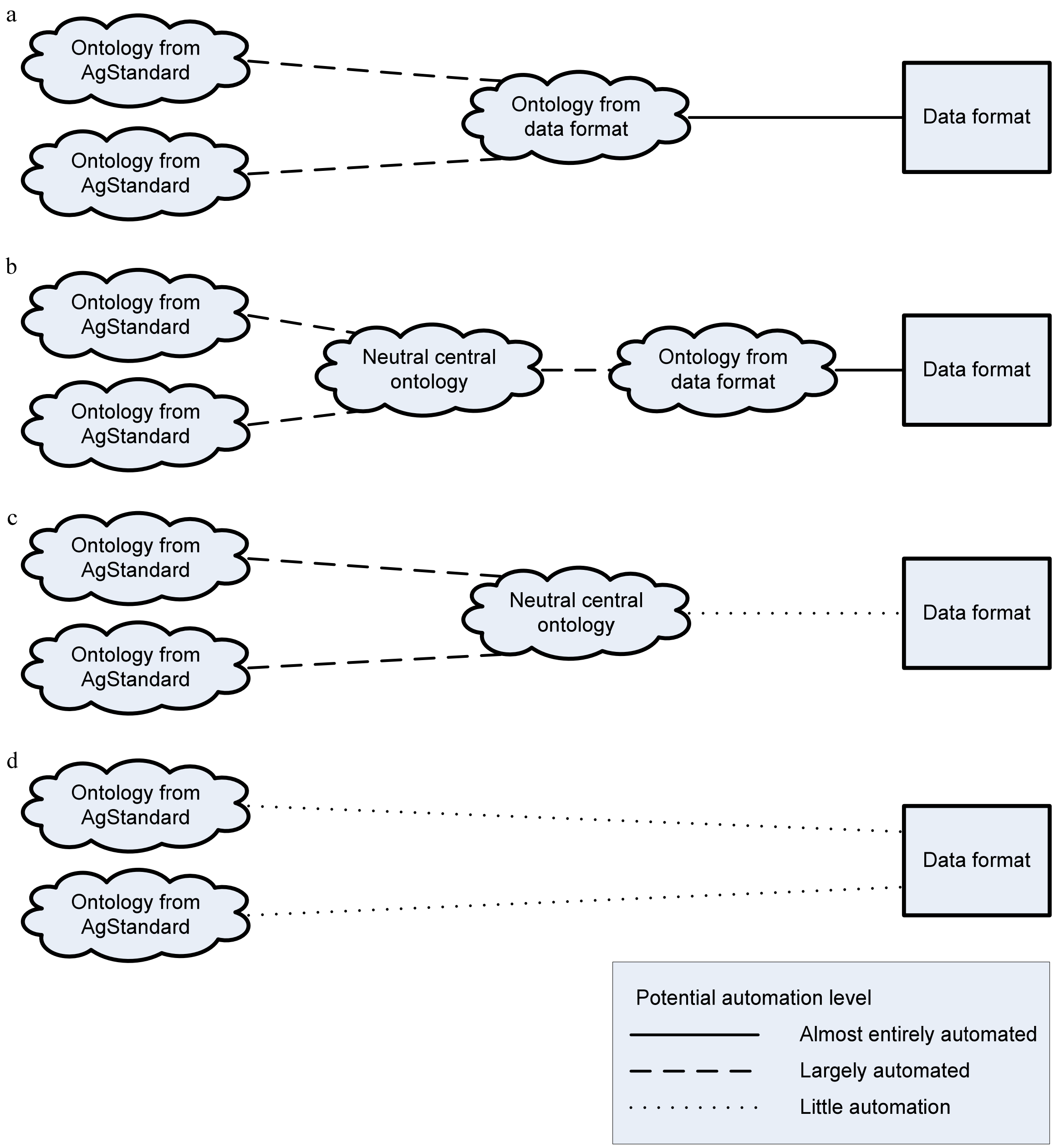

Figure 3. Mapping strategies from ontologies in agricultural standards to data formats; a. via an ontology for the data fomat, b. via a neutral central ontology and a data format ontology, c. via a neutral central ontology and d. directly.

For mapping between two formal ontologies, techniques using syntactic or semantic matching techniques (Giunchiglia & Shvaiko, 2004) may be used to automate the mapping to some extent: in the best case completely, in the worst case with significant manual assistance. For mapping between an ontology and an abritrary data format a low degree of automation can be assumed. If a specific ontology is however available for the data format, the mapping between this ontology and the data format ought to be an entirely automated process.

Four strategies may therefore be identified for mapping between the arbitrary ontologies of an agricultural standard and a given data format (Figure 3). In this it can be seen that the direct mapping between arbitrary ontologies and a data format (Figure 3d) is likely to be an entirely manual process; compared to this, the mapping via an ontology for the data format (Figure 3a) is amenable to a large degree of automation.

One potential technique for simplifying some of the mapping is the use of a neutral central ontology to which all ontologies from agricultural standards and data formats may be mapped (Figure 3b and c). Even if no ontology is prepared for the data format, the single manual mapping to this central ontology (Figure 3c) would only have to be performed once. Similarly, the mapping for each data format would only be required as a 1:1 mapping to the central ontology, rather than a 1:m mapping for each ontology from a standard (giving a total m:n mapping for standards ontologies to data formats), reducing the total manual effort required to produce all possible mappings.

However, there exists no such neutral central ontology (although Agrovoc (FAO, 2010) shows potential for being developed in this direction. Even if such an ontology is developed, the changing nature of standards, and therefore the ontologies defined by them, together with the plethora of agricultural software and data formats, means that to keep the central ontology capable of representing all concepts defined in all standards and data formats would be a mammoth ongoing task. If a concept is defined in one of the standards that cannot be represented in the central ontology, then no mapping would be possible and the data for assessing compliance could not be specified and retrieved.

5. Discussion

This report has presented work on the specification of the data needed to assess compliance to agricultural management and crop production standards. Although this is on the surface a simple problem, closer inspection requires a myriad of difficulties that mean that no single specification mechanism for concrete data elements is possible. Instead, an ontology-based approach is suggested whereby individual concepts may be defined in a formal ontology and then referred to in the machine-readable definition of the standard. These same concepts may later be mapped, potentially via intermediate layers, to concrete data elements in some data format (FMIS software-internal data structures, transfer format such as agroXML, etc.) as required.

The crux of the proposed system is the need to map between ontologies and from ontologies to concrete data formats. Depending on the structures used, the automation of this process is possible to varying degrees. In the proposed system where arbitrary definitions of agricultural standards, and therefore arbitrary ontologies, may be used, a maximal amount of automation is preferred as a manual mapping would in most cases imply that the farmers must identify the data elements themselves.

Ultimately, the ideal solution would be for all standards, and all data formats, to be defined with reference to a single neutral ontology, e.g a further-developed Agrovoc. In the near-term, this is however unrealistic. A more likely interim solution is that each standard will continue to be defined with reference to its own vocabulary, and that each FMIS software will continue to use its own internal data model. Each FMIS may then fully support a limited number of ontologies through a predefined mapping, with some further ontologies partially supported through use of automated inference and mapping techniques. Further ontologies would only be supported through a manual mapping process.

This document has dealt only with the actual data used in assessing compliance to the standard. Also required is a definition of the concepts used in the definition of the standard, e.g. ”compliant”, ”violation”, ”greater than”, ”between”, etc. It is assumed that it will be possible to produce a single central ”meta-ontology” for describing all these terms, since their number is limited when compared to those used in describing the agricultural and management processes themselves, such that FMIS software processing definitions of standards will would be able to implement the interpretation of these concepts in a single static form which will then be applicable for use with all standards.

References

FAO, 2010. Agrovoc Thesaurus. Available at http://aims.fao.org/website/AGROVOC-Thesaurus/sub (11/01/2010).

Giunchiglia, F., Shvaiko, P., 2004. Semantic Matching. The Knowledge Engineering Review 18 (3) 265-280.

Gruber, T., 2009. Ontology. In: Liu, L., Özsu, M.T. (Eds.), Encyclopedia of Database Systems. Springer-Verlag.

McCluskey, G., 1998. Using Java Reflection. Available at http://java.sun.com/developer/technicalArticles/ALT/Reflection/ (07/01/2010).

Nash, E., Nikkilä, R., Pesonen, L., Oetzel, K., Mayer, W., Seilonen, I., Kaivosoja, J., Bill, R., Fountas, S., Sørensen, C., 2009. Machine Readable Encoding for Definitions of Agricultural Crop Production and Farm Management Standards. FutureFarm Deliverable 4.1.1. Available at http://www.futurefarm.eu/system/files/FFD4.1.1_Encoding_of_standards.pdf (05/01/2010).

Nash, E., Nikkilä, R., Kluger, S., Oetzel, K., Pesonen, L., Seilonen, I., Wiebensohn, J., 2010. Specification of a Rules App to handle compliance assessment based on knowledge from repositories. FutureFarm Deliverable 4.3.

Nikkilä, R., Nash, E., 2009. Definition of an interface for a knowledge repository. FutureFarm Deliverable 4.2. Available at http://www.futurefarm.eu/node/196 (06/01/2010).

W3C, 1999. XML Path Language (XPath). Available at http://www.w3.org/TR/xpath (07/01/2010).

W3C, 2004. OWL Web Ontology Language. Available at http://www.w3.org/TR/owl-ref/ (06/01/2010).