Generating AGRIS AP XML from local databases

July 2005

Language: English
Version 1.0

by GILW, Library and Documentation Systems Division

DOCUMENT METADATA		
element	***scheme***	***value***
title		Generating AGRIS AP XML from local databases
creator. creatorCorporate		FAO of the UN, Rome (Italy). GILW, Library and Documentation Systems Division
subject		Metadata Standards; Guidelines; Dublin Core; Document-like Information Objects; Application Profile; Information Exchange
subject. subjectClassification	ASC	C30
subject. subjectClassification	LCC	Z699.35.M28
subject. subjectThesaurus	AGROVOC	AGRIS; DATA PROCESSING; METHODS; STANDARDS; INFORMATION STORAGE
subject. subjectThesaurus	CABIT	information processing; information systems; information storage; AGRIS
subject. subjectThesaurus	LCSH	Agricultural Metadata Element Set; Handbooks, manuals, etc.; Information storage and retrieval systems; Agriculture; Metadata; Exchange of bibliographic information; Machine-readable bibliographic data formats; AGRIS (Information retrieval system)
publisher. publisherName		FAO of the UN
publisher. publisherPlace		Rome, Italy
description. descriptionEdition		Version 1.0
description.abstract		This document is intended as a guide to exporting local data into valid AGRIS AP DTD compliant XML. Its purpose is to facilitate the process thorough step-by-step instructions.
date.dateIssued	W3CDTF	2005-07
type	DCMIType	Text
format.extent		19 p.
language	ISO639-2	eng
rights		http://www.fao.org/copyrigh.htm

Table of Contents

1. Introduction

This document provides detailed technical information for converting bibliographic data from local databases into valid AGRIS DTD compliant XML records. The process that is to be carried out is, to some extent, complex. However, the result will benefit the Institutions and Centres that produce bibliographic data of agricultural content, in the conversion to XML data that can be disseminated to and harvested by the AGRIS XML repository. The XML repository is aggregated, maintained, organized and made available via the Internet[1].
The main purpose of this guide is to facilitate the conversion process thorough step-by-step instructions and to provide a list of common pitfalls.

This document supplements the AGRIS AP guidelines[2], which provide details on describing the content of the XML elements and schemes. Section 4 also provides a brief content description of the 16 core elements, its sub-elements (qualifiers) and relevant schemes and attributes, together with the correlated examples of encoding of the XML tags.

2. Some definitions

2.1 XML

XML, the eXtensible Markup Language, is the universal format for structured documents and data on the Web. It is designed to improve the functionality of the Web by providing more flexible and adaptable information identification. It is called extensible because it is not a fixed format like HTML (a single, predefined markup language). Instead, XML is actually a 'metalanguage' -- a language for describing other languages -- which lets you design your own customized markup languages for limitless different types of documents. All these features make it an attractive standard for exchanging data.

An XML document is a collection of data. In many ways, this makes it no different from any other file. As a "database" format, XML has some advantages. For example, it is self-describing (the mark-up describes the structure and type names of the data, although not the semantics), it is portable (Unicode), and it can describe data as tree or graph structures.

Except for unparsed entities, all data in an AGRIS AP XML document is PC DATA (for elements) or CDATA (for attributes) text, even if it represents another data type, such as a date or an integer. Generally, the data transfer software will convert data from text (in the XML document) to other types (in the database) and vice versa.

XML is a content mark-up meta-language designed to store and display documents on the World Wide Web. By separating content from presentation, XML enables us to create information that can be more easily integrated with other Web resources.

2.2 The Document Type Definition (DTD)

The purpose of a DTD, or document type definition, is to define the legal building blocks of an XML document. It defines the document structure with a list of legal elements. The advantages of the DTD are many, viz. each of your XML files can carry a description of its own format with it; independent groups of people can agree to use a common DTD for interchanging data; your application can use a standard DTD to verify that the data you receive from the outside world is valid; and you can also use a DTD to verify your own data.

It is essential that the structure of the XML output documents exactly match the structure expected by the DTD. Mapping the local database schema to an XML DTD schema is the most important exercise that is undertaken in this context.

[1] AGRIS Website http://www.fao.org/agris/
[2] AGRIS AP Guidelines http://www.fao.org/docrep/008/ae909e/ae909e00.htm

2.3 Namespaces

The W3C XML community defines a mechanism called *XML namespaces*, which can be used as single XML document containing elements and attributes that are defined for and used by multiple software components. This use by multiple software promotes reuse and restricts reinvention. Their definition:

*An **XML namespace** is a collection of names, identified by a URI reference which are used in XML documents as element types and attribute names. XML namespaces differ from the "namespaces" conventionally used in computing disciplines in that the XML version has an internal structure and is not, mathematically speaking, a set.*

In this context, all the newly defined elements in Agricultural Metadata Element Set (AgMES)[3] constitute a namespace. The AgMES defines elements needed to accurately describe various types of information resources in the domain of agriculture. This element set is maintained at a stable location and identifies a reference point where elements are defined and are maintained to be used by different applications.

2.4 XML and Databases

Today most bibliographic data is stored both in relational databases, such as Oracle and SQL Server 2000 and other database systems that support XML using different approaches. These products allow easy publishing, managing, and sharing of content on corporate intranets and Web. An important characteristic in this respect is that they are bidirectional. That is, they can be used to transfer data both from XML documents to the database and from the database to XML documents.

In this document we focus our attention on XML Enabled Databases, that is on systems that allow exporting data to the XML format. Most of the modern ILMS vendors often offer some sort of XML functionality in their products. However, it is important to note that a more multipart process will allow to extract, convert and generate XML from nearly any type of DBMS, provided that an additional layer is developed after the extraction of the relevant subset of records.

3. General Issues

3.1 Before you start

It is assumed that the readers of this document are already XML savvy. We are not going to discuss in detail XML. The readers are, however, introduced to the very essential issues of converting to XML. This section describes the process of converting your local data to AGRIS AP compliant XML documents.

The AGRIS AP, or in database terms the AGRIS data model, prescribes the vocabulary, content and structure rules that can be used to share information between heterogeneous datasets without requiring any change to the local system. With the possibility of using tools such as XSLT, information extraction and conversion becomes a simple yet extremely important task towards facilitating interoperability. The fact that the resource itself is not required to be attached to the metadata makes it easy to control access rights on it.

The AGRIS AP is accompanied by a DTD[4] which is used to validate inputs from different resource centres. The first steps should be:

- Make sure you have the most recent version of the AGRIS AP Specifications as well as the DTD. All this information is available in the AGRIS Website[5].
- Read the specifications.
- Understand the requirements of the AGRIS AP XML DTD. Particular requirements and constraints are dictated by the local database and conversion systems that are used in the AGRIS data structure. Since the scope of this project is to provide maximum flexibility to both structure and content of the XML documents, the current DTD points to five mandatory core elements.
- Map the elements of your local system structure to the AGRIS DTD elements and schemes.

[3] http://www.fao.org/agris/agmes
[4] The AGRIS AP XML DTD is available for validation at http://purl.org/agmes/agrisap/dtd/ and, for display, in APPENDIX A
[5] http://www.fao.org/agris/

3.2 Exporting from XML-enabled databases

The production and export of XML resources from local databases to the AGRIS AP model (see **Figure 1** below) is facilitated when the source DB is XML-enabled, that is if it supports extensions for transferring data between XML documents and their own data structures.

The following four steps briefly describe the process of generating valid AGRIS XML records from proprietary XML-enabled databases:

1. Identification of the fields in the catalogue of the local database that will match the AGRIS AP XML DTD elements and schemes. The resulting mapping document links the fields of the local database to the elements and qualifiers of the DTD.
2. An XSLT stylesheet encodes the mapping document produced by the cataloguers. The template will link and match the nodes from each field of the local database to the appropriate elements and schemes of the AGRIS AP XML DTD.
3. The well-formed XML documents are converted to AGRIS AP XML resources by means of the XSL processor.
4. The XML documents are validated against the AGRIS AP XML DTD by means of XML parsers (3.2).

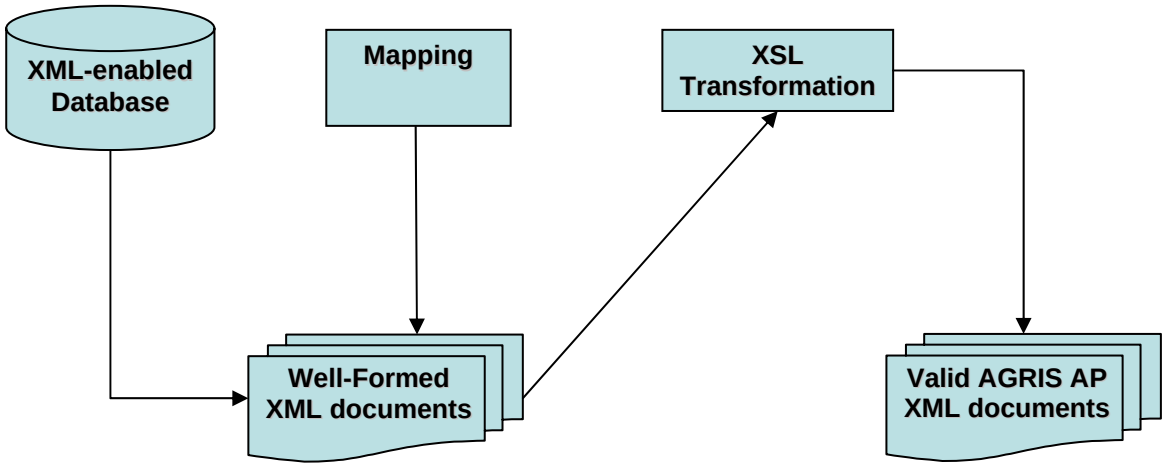


Figure 1: The AGRIS AP XML production process

3.3 The OAI-PMH[6] example

The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH in short), "provides an application-independent interoperability framework based on *metadata harvesting*". Implementing the OAI protocol, data providers are able to generate well formed and valid XML data, by mapping their local repositories to a common DC metadata format. The The Norwegian Univ. Library of Life Sciences[7] as data provider has implemented the OAI protocol, and exposed their metadata to the AGRIS harvester by means of a unique identifier (URI). For detailed information on the OAI-PMH implementation refer to its guidelines[8].

3.4 XSLT transformation to the AGRIS AP metadata

The final production of valid AGRIS AP XML documents is achieved with XSLT Stylesheets. The Extensible Style Language (XSL) provides elements that define rules for how one XML document is transformed into another XML document. In this context, the development of an XSLT can be an easy exercise if the structure of the local repository is DC compliant. For other more complex metadata formats, such as MARC, this can require more work. There are several options that can be considered, each one having particular requirements, such as different fields, different conditions and rules to apply, and accordingly different stylesheets to encode.

---

[6] http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm
[7] http://www.umb.no/
[8] http://www.openarchives.org/OAI/2.0/guidelines.htm

The example provided below is being used by one of the AGRIS Resource Centres that use an XML-enabled ILMS called *InMagic*. It shows how an XML tag of a record extracted by InMagic is transformed (from a well-formed "InMagic" XML to valid AGRIS AP XML) using an XPath expression, addressing the *Title* node of the local DB in the output result tree. In human language the XSLT instructions say: if title element exists, select it and write it with the right AGRIS AP XML tag, in this case the core Dublin Core element dc:title.

Input

```
<inm:Title---Eng-M>Conservation and use of native tropical fruit species biodiversity in Asia</inm:Title---Eng-M>
```

XSLT instructions

```
<xsl:if test="string-length(inm:Title---Eng-M)>0">
<dc:title xml:lang="eng">
        <xsl:value-of select="inm:Title---Eng-M"/>
        <xsl:text/>
        </dc:title>
</xsl:if>
```

Output

```
<dc:title xml:lang="eng">Conservation and use of native tropical fruit species biodiversity in Asia</dc:title>
```

3.5 Tools to validate XML documents

Validating parsers check the well-formedness of the XML documents and verify that the same documents conform to the specific rules of the AGRIS AP XML DTD. The process of validating can be easily achieved with the Microsoft XML Parser (MSXML) which is included in Microsoft Internet Explorer. In the next section we will see that AGRIS AP XML validation is facilitated in that the AGRIS DTD is located in a fixed (PURL) location.

Other XML parsers, many of them freeware, are available on the Internet[9]. The tool that is most widely used is XML Spy[10], a comprehensive package used to create, edit and validate XML, XSL and DTD/XML Schemas documents.

4. Writing the XML[11]

4.1 Header of XML files

The AGRIS DTD provides a set of elements, refinements and schemes for describing and enforcing the structure that makes up the XML format of a bibliographic record. It is essential, when creating or exporting AGRIS AP compliant XML documents, to append the header shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ags:resources SYSTEM "http://purl.org/agmes/agrisap/dtd/">
<ags:resources      xmlns:ags="http://purl.org/agmes/1.1/"
                    xmlns:dc="http://purl.org/dc/elements/1.1/"
                    xmlns:agls="http://www.naa.gov.au/recordkeeping/gov_online/agls/1.2"
                    xmlns:dcterms="http://purl.org/dc/terms/">
```

4.1.1 Declaring XML

All XML documents must declare that they are XML documents by writing the following XML declaration:

```
<?xml version="1.0" encoding="UTF-8"?>
```

This line tells a software that receives the XML data file that you are writing XML and that it should match the file to the XML specification for version 1.0. We shall tackle the encoding issue later in this document. As this is not actually an XML tag containing data, it does not require a closing tag and must be at the beginning of the document.

[9] http://www.xml.com/pub/rg/XML_Parsers
[10] http://www.altova.com
[11] In APPENDIX B an instance of an AGRIS AP XML document

4.1.2 Declaring the document type

When marking up documents using a DTD, it is a standard practice to include a DOCTYPE declaration so that the processing tools 'know' which DTD the document being processed conforms to. When an XML document is validated against the DTD by a validating XML parser, the XML document will be checked to ensure that all required elements are present and that no undeclared elements have been added. The hierarchical structure of elements defined in the DTD must be maintained. The values of all attributes will be checked to ensure that they fall within defined guidelines. In short, every detail of the XML document from top to bottom will be defined and validated by the DTD. This facilitates the process of ensuring uniformity among groups of XML documents, such as those harvested by the AGRIS repository from distributed centres from around the world.

```
<!DOCTYPE ags:resources SYSTEM "http://purl.org/agmes/agrisap/dtd/">
```

The above DOCTYPE declaration for an AGRIS resource document, marked up using the AGRIS DTD, indicates that the document type is ags:resources and that it conforms to the DTD. Requiring that an XML document be validated against the AGRIS DTD ensures the integrity of the data structure. XML documents may be parsed and validated before they are ever loaded by an application.

This declaration points to a PURL (Persistent Uniform Resource Locator), which facilitates the validation, provided that the computer is connected to the Internet. If not, the DTD included in the appendix should be used to validate the XML document.

4.1.3 Declaring the namespaces

The namespace declarations should be the next line following the XML DTD reference. Because documents may contain multiple namespaces, and because the possibility of collisions between prefixes exists, namespaces allow developers to map prefixes to URIs for elements and their contents, not just document-wide.

```
<ags:resources    xmlns:ags="http://purl.org/agmes/1.1/"
                  xmlns:dc="http://purl.org/dc/elements/1.1/"
                  xmlns:dcterms="http://purl.org/dc/terms/"
                  xmlns:agls="http://www.naa.gov.au/recordkeeping/gov_online/agls/1.2">
```

In the above example, there are four namespace declarations: ags, dc, dcterms and agls. In general, a namespace uniquely identifies a set of names or tags so that there is no ambiguity when tags having different origins but the same names are mixed together. Thus, dcterms:citation is different from ags:citation.

4.2 Mandatory elements and schemes

The structures used to describe the “AGRIS” class of documents are *Text Only* (dc:type, dc:source, etc.), *Element only* (dc:citation, agls:availability, etc.) and *Mixed content* (dc:title or dc:relation). All attributes are data character strings (CDATA) with the exception of ags:ARN, which is a unique identifier for the root element ags:resource (ID) and the reserved attribute xml:lang which, where applied, should be constrained to the three-letter ISO639-2 language code[12].

Within the DTD, cardinality of the elements is indicated with the following *cardinality operators*.

(no indicator)	*Required*	One and only one
+	*Required, repeatable*	One or more
?	*Optional*	None or one
*	*Optional, repeatable*	None, one, or more

4.3 XML body of the document

This section will explain how to encode each element, refinement and scheme to create well-formed XML elements. Each table describes the content model of the element, a template explaining how the content should be tagged, the attributes and if the attribute is required.

[12] Codes for the Representation of Names of Languages at http://www.loc.gov/standards/iso639-2/langcodes.html

4.3.1 Attribute ags:ARN

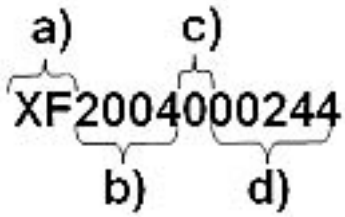
This attribute replaces the previous AGRIS field for Temporary Record Number (TRN). It has an ID validity constraint that provides uniqueness to an AGRIS resource. It is therefore essential that a unique numbering system be used to differentiate between two records. ARN is **mandatory for all records** submitted to AGRIS. The format used for this required attribute is made of 12 characters, divided into three groups. A typical ARN will contain:

a) the two-letter ISO country code of the country where the AGRIS Resource centre is located for the code of the multinational or international institution submitting input. This list can be found in the ISO3166-1 for geographic codes. AGRIS codes that are currently being used by the centres are provided in the Appendix C of AGRIS AP User Guide or in the AGRIS web site.

b) the year in which the input record is created. This must be in four digits and is **not** the year of publication of the resource.

c) the sub-centre code assigned by the Resource Centre, one character only, to be used in countries with more than one resource centre. It may be a letter or a digit. In countries where there are more than nine sub-centres the sub-centre code may be a letter. For countries with one resource centre a zero (0) should be entered in this field.

d) a number made of five digits. This can be assigned on a yearly basis or it can be just a local classification number, such as an internal library number. For example:



4.3.2 Root element ags:resource

This is the root element and it contains all the other core elements and qualifiers. Five of the core elements are mandatory, namely title, date, subject, language and availability information. It is the most important element, as it contains the rest of the document and becomes synonymous with the document type.

XML content model	(dc:title+, dc:creator*, dc:publisher*, dc:date+, dc:subject+, dc:description*, dc:identifier*, dc:type*, dc:format*, dc:language+, dc:relation*, agls:availability+, dc:source?, dc:coverage*, dc:rights*, ags:citation*)	
XML tag	<ags:resource ags:ARN="XF2004000244"> </ags:resource>	
XML attributes/ schemes	ags:ARN (See 4.3.1).	required

4.3.3 Element dc:title

Enter in this element the title of the document. Enter also, if available, the translated title of the resource (dcterms:alternative).

XML content model	(#PCDATA | dcterms:alternative)*	
XML tag	<dc:title xml:lang="eng">title of resource <dcterms:alternative xml:lang="eng">other title, normally translated </dcterms:alternative> </dc:title>	
XML attributes/ schemes	xml:lang	required

4.3.4 Element dc:creator

This element describes all entities (Agents) that handle the resource, i.e. creating or contributing. It may include a person (ags:creatorPersonal); an organization, a service or an agency (ags:creatorCorporate); or a conference (ags:creatorConference).

XML content model	(ags:creatorPersonal | ags:creatorCorporate | ags:creatorConference)*
XML tag	<dc:creator> <ags:creatorPersonal>personal creator</ags:creatorPersonal> <ags:creatorCorporate>corporate creator</ags:creatorCorporate>

	<ags:creatorConference>conference creator</ags:creatorConference> </dc:creator>	
XML attributes/ schemes	-	

4.3.5 Element dc:publisher

Enter in the two refinement elements the information about the publisher. These elements provide the name of the individual, group, or organization which controls or publishes the item (ags:publisherName) and its location (ags:publisherPlace).

XML content model	(ags:publisherName | ags:publisherPlace)*	
XML tag	<dc:publisher> <ags:publisherName>name of publisher</ags:publisherName> <ags:publisherPlace>location of publisher</ags:publisherPlace> </dc:publisher>	
XML attributes/ schemes	-	

4.3.6 Element dc:date

Enter in this element the date when the resource was made available. dc:date must be used together with its qualifier (dcterms:dateIssued).

XML content model	dc:date (dcterms:dateIssued)	
XML tag	<dc:date> <dcterms:dateIssued>date of publ.</dcterms:dateIssued > </dc:date>	
XML attributes/ schemes	scheme (dcterms:W3CDTF)	

4.3.7 Element dc:subject

Enter in this element the subject information about the resource. It can be free-text (dc:subject), come from a classification scheme (ags:subjectClassification) or from a controlled vocabulary (ags:subjectThesaurus).

XML content model	(#PCDATA | ags:subjectClassification | ags:subjectThesaurus)*	
XML tag	<dc:subject> <ags:subjectClassification scheme="ags:ASC">ASC scheme</ags:subjectClassification> <ags:subjectClassification scheme="ags:CABC">CABC scheme</ags:subjectClassification> <ags:subjectClassification scheme="dcterms:DDC">DDC scheme</ags:subjectClassification> <ags:subjectClassification scheme="dcterms:LCC">LCC scheme</ags:subjectClassification> <ags:subjectClassification scheme="dcterms:UDC">UDC scheme</ags:subjectClassification> </dc:subject> <dc:subject> <ags:subjectThesaurus scheme="ags:AGROVOC" xml:lang="eng">AGROVOC term</ags:subjectThesaurus> <ags:subjectThesaurus scheme="ags:ASFAT" xml:lang="eng">ASFAT term</ags:subjectThesaurus> <ags:subjectThesaurus scheme="ags:CABT" xml:lang="eng">CABT term</ags:subjectThesaurus> <ags:subjectThesaurus scheme="ags:NALT" xml:lang="eng">NALT term</ags:subjectThesaurus> <ags:subjectThesaurus scheme="dcterms:LCSH" xml:lang="eng">LCSH term</ags:subjectThesaurus> <ags:subjectThesaurus scheme="dcterms:MeSH" xml:lang="eng">MeSH term</ags:subjectThesaurus> </dc:subject>	
XML attributes/ schemes	ags:subjectClassification • scheme (ags:ASC | ags:CABC | dcterms:DDC | dcterms:LCC | dcterms:UDC) ags:subjectThesaurus • scheme (ags:CABT | ags:AGROVOC | ags:NALT | ags:ASFAT | dcterms:LCSH | dcterms:MeSH) • xml:lang	required required required

4.3.8 Element dc:description

This element indicates different descriptive aspects of the resource. These may include a brief statement, annotation, comment, or elucidation concerning any aspect of the resource (ags:descriptionNotes); formally designated version of the data set or information resource being described (ags:descriptionEdition); or an abstract as a summary of a document designed to give the user a clearer idea about the document's contents (dcterms:abstract).

XML content model	(ags:descriptionNotes | ags:descriptionEdition | dcterms:abstract)*

XML tag	<dc:description> <ags:descriptionEdition>description of edition</ags:descriptionEdition> <ags:descriptionNotes>notes</ags:descriptionNotes> <dcterms:abstract xml:lang="eng">abstract</dcterms:abstract> </dc:description>	
XML attributes/ schemes	dcterms:abstract • xml:lang	optional

4.3.9 Element dc:identifier

The identifiers help locate or/and identify a resource. There can be many numbers assigned to a document. This element is reserved for standard numbers taken from the item. Some of the numbers may be input in authorized form. For web resources, the URI (electronic address starting with: for ex. http:// or ftp://) is also placed in this element. Numbers assigned by cataloguing institutions for internal purposes are **not** entered here, but placed into the agls:availability field.

XML content model	dc:identifier (#PCDATA)	
XML tag	<dc:identifier scheme="ags:DOI">DOI id</dc:identifier> <dc:identifier scheme="ags:IPC">International Patent Classification no.</dc:identifier> <dc:identifier scheme="ags:ISBN">Book ISBN</dc:identifier> <dc:identifier scheme="ags:JN">Job Number</dc:identifier> <dc:identifier scheme="ags:PN">Patent Number</dc:identifier> <dc:identifier scheme="ags:RN">Report Number</dc:identifier> <dc:identifier scheme="dcterms:URI">URI of resource</dc:identifier>	
XML attributes/ schemes	scheme (ags:IPC | ags:RN | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:DOI)	optional

4.3.10 Element dc:type

Although it is not mandatory, the value of this element should be provided when possible. It explains the nature or genre of the content of the resource and also helps to describe the general categories, functions, genres, or aggregation levels for the content of the resource.

If possible, select the dc:type values from the *DCMI Type list*[13]. If using a local type controlled vocabulary, make sure there is no code but instead whole words that describe the genre of the resource.

XML content model	dc:type (#PCDATA)	
XML tag	<dc:type>DC Types controlled vocabularies</dc:type>	
XML attributes/ schemes	scheme (dcterms:DCMIType)	optional

4.3.11 Element dc:format

The extent element (dcterms:extent) is used to indicate the size or duration of the resource. The medium element (dcterms:medium) is used to indicate the material or physical carrier of the resource.

XML content model	dc:format (dcterms:extent | dcterms:medium)*	
XML tag	<dc:format> <dcterms:extent>collation, size, duration of the resource</dcterms:extent> <dcterms:medium> the material or physical carrier of the resource. </dcterms:medium> </dc:format>	
XML attributes/ schemes	-	

4.3.12 Element dc:language

For this element, it is recommended to enter the three letter code from ISO639-2[14]. If your local system does not allow you to provide the 3 letter code, enter the two letter code, indicating the scheme as ISO639-1[15]. If a language does not have a code in the selected scheme, enter the full form of the language without indicating the scheme.

XML content model	dc:language (#PCDATA)
XML tag	<dc:language scheme="ISO639-1">language of resource</dc:language>

[13] http://dublincore.org/documents/dcmi-type-vocabulary
[14] ISO639-2 the three letter language codes http://www.loc.gov/standards/iso639-2/langcodes.html
[15] ISO639-1 and ISO639-2 codes http://www.loc.gov/standards/iso639-2/codechanges.html

XML attributes/ schemes	scheme (ISO639-1 | ISO639-2)	optional

4.3.13 Element dc:relation

This element is used to link one resource to another. It allows the establishment of various relationships between resources and for users to locate related resources. When using relation element, it is important to establish the type of relationship by choosing a value from one side of any of the following pairs of relation refinement types.

XML content model	(#PCDATA | dcterms:isPartOf | dcterms:hasPart | dcterms:isVersionOf | dcterms:hasVersion | dcterms:isFormatOf | dcterms:hasFormat | dcterms:references | dcterms:isReferencedBy | dcterms:isRequiredBy | dcterms:requires | dcterms:isReplacedBy | dcterms:replaces | ags:relationHasTranslation | ags:relationIsTranslationOf)*
XML tag (only dcterms:URI scheme included for each qualifier)	<!-- physical or logical part of the referenced resource --> <dc:relation> <dcterms:isPartOf scheme="dcterms:URI">related URI</dcterms:isPartOf> </dc:relation> <!-- the referenced resource either physically or logically --> <dc:relation> <dcterms:hasPart scheme="dcterms:URI">related URI</dcterms:hasPart> </dc:relation> <!-- a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in content rather than differences in format --> <dc:relation> <dcterms:isVersionOf scheme="dcterms:URI">related URI</dcterms:isVersionOf> </dc:relation> <!-- a version, edition, or adaptation, namely, the referenced resource --> <dc:relation> <dcterms:hasVersion scheme="dcterms:URI">related URI</dcterms:hasVersion> </dc:relation> <!-- same intellectual content of the referenced resource, but presented in another format--> <dc:relation> <dcterms:isFormatOf scheme="dcterms:URI">related URI</dcterms:isFormatOf> </dc:relation> <!-- pre-existed the referenced resource, which is essentially the same intellectual content presented in another format --> <dc:relation> <dcterms:hasFormat scheme="dcterms:URI">related URI</dcterms:hasFormat> </dc:relation> <!-- references, cites, or otherwise points to the referenced resource --> <dc:relation> <dcterms:references scheme="dcterms:URI">related URI</dcterms:references> </dc:relation> <!-- is referenced, cited, or otherwise pointed to by the referenced resource --> <dc:relation> <dcterms:isReferencedBy scheme="dcterms:URI">related URI</dcterms:isReferencedBy> </dc:relation> <!-- is required by the referenced resource, either physically or logically --> <dc:relation> <dcterms:isRequiredBy scheme="dcterms:URI">related URI</dcterms:isRequiredBy> </dc:relation> <!-- required by the referenced resource, either physically or logically --> <dc:relation> <dcterms:requires scheme="dcterms:URI">related URI</dcterms:requires> </dc:relation> <!-- is supplanted, displaced, or superseded by the referenced resource --> <dc:relation> <dcterms:isReplacedBy scheme="dcterms:URI">related URI</dcterms:isReplacedBy> </dc:relation> <!-- supplants, displaces, or supersedes the referenced resource --> <dc:relation> <dcterms:replaces scheme="dcterms:URI">related URI</dcterms:replaces> </dc:relation> <!-- has a translation, namely, the referenced resource --> <dc:relation> <ags:relationHasTranslation scheme="dcterms:URI">related URI</ags:relationHasTranslation> </dc:relation> <!-- a translation of the referenced resource -->

	<dc:relation> <ags:relationIsTranslationOf scheme="dcterms:URI">related URI</ags:relationIsTranslationOf> </dc:relation>	
XML attributes/ schemes	scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:ARN)	required

4.3.14 Element agls:availability

Availability provides users with a number or code that is uniquely associated with an item, and serves to identify that item within an organization. This number is normally assigned by the organization that holds the item. Since this is local information, availability must include the name or code identifying the institution or repository (ags:availabilityLocation) in which the item is housed and the local number (ags:availabilityNumber) with which the resource is locally accessed.

XML content model	agls:availability (ags:availabilityLocation, ags:availabilityNumber)*	
XML tag	<agls:availability> <ags:availabilityLocation>availability location</ags:availabilityLocation> <ags:availabilityNumber>availability number</ags:availabilityNumber> </agls:availability>	
XML attributes/ schemes	-	

4.3.15 Element dc:source

This element (dc:source) provides the reference to a resource of which the current resource is a part. When cataloguing the analytic, this element is used to provide information for identification of the Monograph. Source information that can go into this element includes title of the whole, creators of the whole, etc.

XML content model	dc:source (#PCDATA)	
XML tag	<dc:source>additional information of resource</dc:source>	
XML attributes/ schemes	-	

4.3.16 Element dc:coverage

This element (dc:coverage) provides information about the geographical (dc:spatial) and temporal (dc:temporal) coverage of the resource.

XML content model	dc:coverage (#PCDATA, dc:spatial, dc:temporal)	
XML tag	<dc:coverage>additional information of resource <dcterms:spatial scheme ="dcterms:ISO3166">coverage (ISO3166)</dcterms:spatial> <dcterms:temporal scheme ="dcterms:TGN">coverage (TGN)</dcterms: temporal > </dc:coverage>	
XML attributes/ schemes	dcterms:spatial scheme (dcterms:POINT | dcterms:ISO3166 | dcterms:TGN | dcterms:Box) dcterms:temporal scheme (dcterms:Period | dcterms: W3CDTF)	optional optional

4.3.17 Element dc:rights

XML content model	dc:rights (#PCDATA, ags:rightsStatement, ags:rightsTermsOfUse)	
XML tag	<dc:rights> <ags:rightsStatement>statements of rights</ags:rightsStatement> <ags:rightsTermsOfUse>terms of use<ags:rightsTermsOfUse> </dc:rights>	
XML attributes/ schemes		

4.3.18 Element ags:citation

This is a mandatory entry when the resource is part of a serial. A serial is defined as a publication, usually having numerical or chronological label, and intended to be continued indefinitely. It may be made available on any medium and is issued in successive parts.

XML content model	ags:citation (ags:citationTitle | ags:citationIdentifier | ags:citationNumber | ags:citationChronology)*	
XML tag	<ags:citation> <ags:citationTitle>Title of the serial</ags:citationTitle> <ags:citationIdentifier scheme="ags:ISSN"> Identifier of the Serial </ags:citationIdentifier> <ags:citationNumber>Number of the issue</ags:citationNumber> <ags:citationChronology>Chronological designation of the issue</ags:citationChronology> </ags:citation>	
XML attributes/ schemes	ags:citationTitle xml:lang ags:citationIdentifier scheme (ags:ISSN | ags:CODEN)	 optional required

5. Troubleshooting issues

5.1 Character Sets

One of the most important characteristics of the XML mark-up is its portability. In fact, XML documents can contain any Unicode character except some of the control characters. Unfortunately, many databases offer limited or no support for Unicode and require special configuration to handle non-ASCII characters. If your data contains non-ASCII characters, be sure to check how and if both your database and data transfer software handle these characters. Thus, a proper check on how the database (or the system from where data is extracted) handles and manipulates the encoding should be done.
Given that XML documents should be encoded in UTF-8 (preferable), other encoding schemes, such as ISO-8859-1 are also allowed, provided that it is clearly declared in the starting XML header declaration in order for it to be recognized and displayed correctly in the centralized AGRIS XML repository.

5.2 Predefined Entity References

XML predefines the five entity references as shown below. These predefined entity references are used in XML documents in place of specific characters that would otherwise be interpreted as part of mark-up language.

Character	Entity References
&	&amp;
<	&lt;
>	&gt;
“	&quot;
‘	&apos;

5.3 Whitespace

Four characters are treated as white space in XML data: Horizontal Tab, Line-feed, Carriage-return and ASCII space character. None of these should be represented within the same XML tag. The input should be typed as it should be rendered in the AGRIS web database. Therefore, the following two situations should be avoided:

```
<dc:title xml:lang="eng">Understanding and capturing the multiples values of tropical forest.¶
Proceedings of the International Seminar on Valuation and Innovative Financing Mechanisms in support of Conservation and sustainable management of tropical forest</dc:title>
```

No carriage return within the XML tags

```
<ags:subjectClassification scheme="ags:ASC">E20 </ags:subjectClassification>
```

No blank space before and/or after the value

5.4 Repeatable elements

Repeatable elements or qualifiers should all have multiple tags.
For example, the following is not valid:

```
<ags:subjectClassification scheme="ags:ASC">E20 ; J12</ags:subjectClassification>
```

whereas the following is the correct encoding:

```
<ags:subjectClassification scheme="ags:ASC">E20</ags:subjectClassification>
<ags:subjectClassification scheme="ags:ASC">J12</ags:subjectClassification>
```

5.5 Size of XML documents

Avoid creating and sending large XML documents. If files are bigger than 500Kb, split them into two or more files. When splitting the file, make sure to place correct header in them (See 4.1). However, a preferred approach for handling XML documents in large repositories is splitting each XML resource in each document.

5.6 Scheme for xml:lang

The xml:lang attribute gives XML authors a consistent way to identify the language contained within a particular element. The AGRIS AP uses this attribute for elements for which it was considered necessary to know the language of the content. This extensibility also enables for multiple values of the specified element in different languages. Use **ISO639-2 scheme** (three letter code) for the xml:lang attribute.

5.7 Empty-Elements Tags

No elements, sub-elements and attributes tags should be present without content. When mapping the structure of the database to an XML document, you should be careful that optional element types and attributes are mapped to nullable columns (that is fields allowing null values) and vice-versa. The result of not doing so is likely to produce invalid documents (when transferring data from the database) and to create unwanted empty elements or qualifiers.

5.8 Mandatory tags

Despite the flexibility of the DTD schema, there are a few rules that should be followed, in order to have a valid XML document. Some mandatory elements, qualifiers and schemes must be entered in the relevant tags. This means that valid XML documents should contain at least the following five core elements for each **ags:resource**:

- <dc:title>
- <dc:date>
- <dc:subject>
- <dc:language>
- <agls:availability>

5.9 Inconsistency of local metadata with AGRIS AP elements

Local systems often contain fields that are specific to meet local requirements and have no corresponding element in the metadata structure defined by the AGRIS AP. During the mapping and transformation process this metadata should not be considered. This version of the AGRIS AP metadata is quite flexible and its purpose is also to accommodate varied types of information, and, in future versions it might take into consideration possible additions of key qualifiers or schemes.

5.10 Nesting, dumbing-down and DC compliance

The Dublin Core metadata set is not completely adequate to describe resources and in certain cases, the DC element has been qualified for the purposes of AGRIS resources.
The qualification of the Dublin Core elements is guided by a rule known colloquially as the Dumb-Down Principle. According to this rule, a client should be able to ignore any qualifier and use the value as if it were unqualified. While this may result in some loss of specificity, the remaining term value (minus the qualifier) must continue to be generally correct and useful for discovery. Qualification is therefore supposed only to refine, not extend the semantic scope of an element.
Element refinements (qualifiers) share the meaning of the unqualified element, but with a more restricted scope. When necessary, these qualifiers are “dumbed down” to Dublin Core elements by means of nesting to ensure DC compliancy. The following example dumbs down ags:publisherPlace and ags:publisherName into the core element dc:publisher. This practice results in the element refinement being ignored and the value used as content of the unqualified element core element dc:publisher

With qualifiers	**dumb-down**
<dc:publisher> <ags:publisherPlace>Rome (Italy)</ags:publisherPlace> <ags:publisherName>FAO</ags:publisherName> </dc:publisher>	<dc:publisher>FAO Rome (Italy)</dc:publisher>

APPENDIX A: The AGRIS Document Type Definition (DTD)

```
<!-- Namespaces URIs declarations -->
<!ENTITY agsns "http://purl.org/agmes/1.1/">
<!ENTITY dcns "http://purl.org/dc/elements/1.1/">
<!ENTITY aglsns "http://www.naa.gov.au/recordkeeping/gov_online/agls/1.2">
<!ENTITY dctermsns "http://purl.org/dc/terms/">
<!-- Convenience entities for XML namespace declarations -->
<!ENTITY % agsnsdecl "xmlns:ags CDATA #FIXED &quot;&agsns;&quot;">
<!ENTITY % dcnsdecl "xmlns:dc CDATA #FIXED &quot;&dcns;&quot;">
<!ENTITY % aglsnsdecl "xmlns:agls CDATA #FIXED &quot;&aglsns;&quot;">
<!ENTITY % dctermsnsdecl "xmlns:dcterms CDATA #FIXED &quot;&dctermsns;&quot;">
<!-- The root element -->
<!ELEMENT ags:resources (ags:resource+)>
<!ATTLIST ags:resource
    ags:ARN ID #REQUIRED
>
<!ELEMENT ags:resource (dc:title+, dc:creator*, dc:publisher*, dc:date+, dc:subject+, dc:description*, dc:identifier*,
dc:type*, dc:format*, dc:language+, dc:relation*, agls:availability+, dc:source?, dc:coverage*, dc:rights*, ags:citation*)>
<!-- ELEMENT title -->
<!ELEMENT dc:title (#PCDATA | dcterms:alternative)*>
<!ATTLIST dc:title
    xml:lang CDATA #REQUIRED
>
<!ELEMENT dcterms:alternative (#PCDATA)>
<!ATTLIST dcterms:alternative
    xml:lang CDATA #IMPLIED
>
<!-- ELEMENT creator -->
<!ELEMENT dc:creator (ags:creatorPersonal | ags:creatorCorporate | ags:creatorConference)*>
<!ELEMENT ags:creatorPersonal (#PCDATA)>
<!ELEMENT ags:creatorCorporate (#PCDATA)>
<!ELEMENT ags:creatorConference (#PCDATA)>
<!-- ELEMENT publisher -->
<!ELEMENT dc:publisher (ags:publisherName | ags:publisherPlace)*>
<!ELEMENT ags:publisherName (#PCDATA)>
<!ELEMENT ags:publisherPlace (#PCDATA)>
<!-- ELEMENT date -->
<!ELEMENT dc:date (dcterms:dateIssued)>
<!ELEMENT dcterms:dateIssued (#PCDATA)>
<!ATTLIST dcterms:dateIssued
    scheme (dcterms:W3CDTF) #IMPLIED
>
<!-- ELEMENT subject -->
<!ELEMENT dc:subject (#PCDATA | ags:subjectClassification | ags:subjectThesaurus)*>
<!ATTLIST dc:subject
    xml:lang CDATA #IMPLIED
>
<!ELEMENT ags:subjectClassification (#PCDATA)>
<!ATTLIST ags:subjectClassification
    scheme (ags:ASC | ags:CABC | dcterms:DDC | dcterms:LCC | dcterms:UDC | ags:ASFAC) #REQUIRED
>
<!ELEMENT ags:subjectThesaurus (#PCDATA)>
<!ATTLIST ags:subjectThesaurus
    xml:lang CDATA #IMPLIED
    scheme (ags:CABT | ags:AGROVOC | ags:NALT | ags:ASFAT | dcterms:LCSH | dcterms:MeSH) #REQUIRED
>
<!-- ELEMENT description -->
<!ELEMENT dc:description (ags:descriptionNotes | ags:descriptionEdition | dcterms:abstract)*>
<!ELEMENT ags:descriptionNotes (#PCDATA)>
<!ELEMENT ags:descriptionEdition (#PCDATA)>
<!ELEMENT dcterms:abstract (#PCDATA)>
<!ATTLIST dcterms:abstract
    xml:lang CDATA #IMPLIED
>
<!-- ELEMENT identifier -->
<!ELEMENT dc:identifier (#PCDATA)>
```

```
<!ATTLIST dc:identifier
    scheme (ags:IPC | ags:RN | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:DOI) #IMPLIED
>
<!-- ELEMENT type -->
<!ELEMENT dc:type (#PCDATA)>
<!ATTLIST dc:type
    scheme (dcterms:DCMIType) #IMPLIED
>
<!-- ELEMENT format -->
<!ELEMENT dc:format (dcterms:extent | dcterms:medium)*>
<!ELEMENT dcterms:extent (#PCDATA)>
<!ELEMENT dcterms:medium (#PCDATA)>
<!ATTLIST dcterms:medium
    scheme (dcterms:IMT) #IMPLIED
>
<!-- ELEMENT language -->
<!ELEMENT dc:language (#PCDATA)>
<!ATTLIST dc:language
    scheme (ags:ISO639-1 | dcterms:ISO639-2) #IMPLIED
>
<!-- ELEMENT relation -->
<!ELEMENT dc:relation (dcterms:isPartOf | dcterms:hasPart | dcterms:isVersionOf | dcterms:hasVersion |
dcterms:isFormatOf | dcterms:hasFormat | dcterms:references | dcterms:isReferencedBy | dcterms:isRequiredBy |
dcterms:requires | dcterms:isReplacedBy | dcterms:replaces | ags:relationHasTranslation | ags:relationIsTranslationOf)*>
<!ELEMENT dcterms:isPartOf (#PCDATA)>
<!ATTLIST dcterms:isPartOf
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:hasPart (#PCDATA)>
<!ATTLIST dcterms:hasPart
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:isVersionOf (#PCDATA)>
<!ATTLIST dcterms:isVersionOf
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:hasVersion (#PCDATA)>
<!ATTLIST dcterms:hasVersion
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:isFormatOf (#PCDATA)>
<!ATTLIST dcterms:isFormatOf
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:hasFormat (#PCDATA)>
<!ATTLIST dcterms:hasFormat
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:references (#PCDATA)>
<!ATTLIST dcterms:references
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:isReferencedBy (#PCDATA)>
<!ATTLIST dcterms:isReferencedBy
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:isRequiredBy (#PCDATA)>
<!ATTLIST dcterms:isRequiredBy
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:requires (#PCDATA)>
<!ATTLIST dcterms:requires
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:isReplacedBy (#PCDATA)>
<!ATTLIST dcterms:isReplacedBy
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT dcterms:replaces (#PCDATA)>
<!ATTLIST dcterms:replaces
```

```
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT ags:relationHasTranslation (#PCDATA)>
<!ATTLIST ags:relationHasTranslation
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!ELEMENT ags:relationIsTranslationOf (#PCDATA)>
<!ATTLIST ags:relationIsTranslationOf
    scheme (ags:IPC | ags:PN | ags:ISBN | ags:JN | dcterms:URI | ags:RN | ags:DOI) #REQUIRED
>
<!-- ELEMENT availability -->
<!ELEMENT agls:availability (ags:availabilityLocation, ags:availabilityNumber)*>
<!ELEMENT ags:availabilityLocation (#PCDATA)>
<!ELEMENT ags:availabilityNumber (#PCDATA)>
<!-- ELEMENT source -->
<!ELEMENT dc:source (#PCDATA)>
<!-- ELEMENT coverage -->
<!ELEMENT dc:coverage (#PCDATA | dcterms:spatial | dcterms:temporal)*>
<!ELEMENT dcterms:spatial (#PCDATA)>
<!ATTLIST dcterms:spatial
    scheme (dcterms:Point | dcterms:ISO3166 | dcterms:TGN | dcterms:Box) #IMPLIED
>
<!ELEMENT dcterms:temporal (#PCDATA)>
<!ATTLIST dcterms:temporal
    scheme (dcterms:Period | dcterms:W3CDTF) #IMPLIED
>
<!-- ELEMENT rights -->
<!ELEMENT dc:rights (#PCDATA | ags:rightsStatement | ags:rightsTermsOfUse)*>
<!ELEMENT ags:rightsStatement (#PCDATA)>
<!ELEMENT ags:rightsTermsOfUse (#PCDATA)>
<!-- ELEMENT citation -->
<!ELEMENT ags:citation (ags:citationTitle | ags:citationIdentifier | ags:citationNumber | ags:citationChronology)*>
<!ELEMENT ags:citationTitle (#PCDATA)>
<!ATTLIST ags:citationTitle
    xml:lang CDATA #IMPLIED
>
<!ELEMENT ags:citationIdentifier (#PCDATA)>
<!ATTLIST ags:citationIdentifier
    scheme (ags:ISSN | ags:CODEN) #REQUIRED
>
<!ELEMENT ags:citationNumber (#PCDATA)>
<!ELEMENT ags:citationChronology (#PCDATA)>
```

APPENDIX B: An instance of an XML AGRIS record

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ags:resources SYSTEM "http://purl.org/agmes/agrisap/dtd/">
<ags:resources xmlns:ags="http://purl.org/agmes/1.1/" xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:agls="http://www.naa.gov.au/recordkeeping/gov_online/agls/1.2" xmlns:dcterms="http://purl.org/dc/terms/">
    <ags:resource ags:ARN="NL2004700134">
        <dc:title xml:lang="eng">Effect of oxidation ditch horizontal velocity on the nitrogen removal process</dc:title>
        <dc:creator>
            <ags:creatorPersonal>Abusam, A.</ags:creatorPersonal>
            <ags:creatorPersonal>Keesman, K.J.</ags:creatorPersonal>
            <ags:creatorPersonal>Spanjers, H.</ags:creatorPersonal>
        </dc:creator>
        <dc:date>
            <dcterms:dateIssued>2002</dcterms:dateIssued>
        </dc:date>
        <dc:subject>
            <ags:subjectClassification scheme="ags:ASC">P10</ags:subjectClassification>
            <ags:subjectThesaurus xml:lang="eng" scheme="ags:CABT">WASTE WATER</ags:subjectThesaurus>
            <ags:subjectThesaurus xml:lang="eng" scheme="ags:CABT">NITRATES</ags:subjectThesaurus>
            <ags:subjectThesaurus xml:lang="eng" scheme="ags:CABT">REMOVAL</ags:subjectThesaurus>
            <ags:subjectThesaurus xml:lang="eng" scheme="ags:CABT">PERFORMANCE</ags:subjectThesaurus>
        </dc:subject>
        <dc:description>
            <ags:descriptionNotes>12 refs</ags:descriptionNotes>
        </dc:description>
        <dc:identifier scheme="dcterms:URI">http://www.ewaonline.de/journal/2002_06.pdf</dc:identifier>
        <dc:format>
            <dcterms:extent>p. 213</dcterms:extent>
            <dcterms:medium>internet</dcterms:medium>
        </dc:format>
        <dc:language scheme="ags:ISO639-1">en</dc:language>
        <agls:availability>
            <ags:availabilityLocation>Library Wageningen University and Research Centre, Postbus 9100, 6703 BK
Wageningen ub.library@wur.nl http://library.wur.nl/desktop/</ags:availabilityLocation>
            <ags:availabilityNumber>1700134</ags:availabilityNumber>
        </agls:availability>
        <ags:citation>
            <ags:citationTitle xml:lang="eng">European water management online</ags:citationTitle>
            <ags:citationChronology>2002</ags:citationChronology>
        </ags:citation>
    </ags:resource>
</ags:resources>
```