

Open Geospatial Consortium Inc.

Date: 2005-02-08

Reference number of this OGC™ document: OGC 05-017

Version: 0.3.3

Category: OGC™ Discussion Paper

Editors: Wenli Yang, Arliss Whiteside

Web Image Classification Service (WICS) Implementation Specification

Copyright © Open Geospatial Consortium (2005)

Warning

This document is not an OGC Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type:	OGC™ Discussion Paper
Document subtype:	Interoperability Program Report
Document stage:	Approved Discussion Paper
Document language:	English

Contents	Page
i. Preface.....	v
ii. Submitting organizations	v
iii. Document contributor contact points.....	v
iv. Revision history	vi
v. Changes to the OGC Abstract Specification.....	vi
vi. Future work.....	vii
Foreword.....	vii
Introduction.....	viii
1 Scope.....	9
2 Conformance.....	9
3 Normative references	9
4 Terms and definitions	10
5 Conventions	10
5.1 Abbreviated terms.....	10
5.2 UML notation.....	11
5.3 Document terms and definitions.....	11
6 WICS Overview.....	11
7 Shared aspects.....	14
7.1 Introduction.....	14
7.2 Shared operation request parameters	14
7.3 Operation request encoding	15
8 GetCapabilities operation (required).....	15
8.1 Introduction.....	15
8.2 Operation request.....	15
8.3 Operation response.....	17
8.3.1 Normal response	17
8.3.2 OperationsMetadata section standard contents.....	17
8.3.3 Contents section contents.....	18
8.3.4 Capabilities document XML encoding.....	20
8.3.5 Exceptions.....	23
8.4 Examples.....	23
8.4.1 GetCapabilities request	23
8.4.2 GetCapabilities response.....	23
9 GetClassification (required).....	25
9.1 Introduction.....	25

9.2	Operation request	25
9.2.1	GetClassification request parameters.....	25
9.2.2	GetClassification request KVP encoding (optional).....	27
9.2.3	GetClassification request XML encoding (required).....	28
9.3	Operation response.....	30
9.3.1	GetClassification normal response	30
9.3.2	Exceptions.....	31
9.4	Examples.....	31
9.4.1	GetClassification request	31
9.4.2	GetClassification response.....	32
10	TrainClassifier.....	32
10.1	Introduction.....	32
10.2	Operation request.....	32
10.2.1	TrainClassifier request parameters	32
10.2.2	TrainClassifier request KVP encoding (optional).....	33
10.2.3	TrainClassifier request XML encoding (required)	34
10.3	Operation response.....	36
10.3.1	TrainClassifier normal response	36
10.3.2	Exceptions.....	37
10.4	Examples.....	38
10.4.1	TrainClassifier request.....	38
10.4.2	TrainClassifier response.....	39
11	DescribeClassifier Operation (required).....	39
11.1	Introduction.....	39
11.2	Operation request.....	39
11.2.1	DescribeClassifier request parameters.....	39
11.2.2	DescribeClassifier request KVP encoding (optional).....	40
11.2.3	DescribeClassifier request XML encoding (required).....	40
11.3	Operation response.....	41
11.3.1	DescribeClassifier normal response XML.....	41
11.3.2	Exceptions.....	42
11.4	Examples.....	43
11.4.1	DescribeClassifier request	43
11.4.2	DescribeClassifier response	43
Annex A (informative) Example of the Maximum likelihood supervised classifier schema.....		43
Bibliography		45

Figures	Page
Figure 1 — WICS interface UML diagram.....	14

Tables	Page
Table 1 — Definitions of some operation request and response parameters	15
Table 2 — Operation Request Encoding.....	15
Table 3 — Section name values and contents	17
Table 4 — Required values of OperationMetadata section attributes	17
Table 5 — Optional values of OperationMetadata section attributes.....	18
Table 6 — Parts of Contents section	18
Table 7 —Parameters in ClassifierBrief data type	19
Table 8 — Parameters in KVPPParameterType data type	19
Table 9 — Parameters in GetClassification operation request.....	26
Table 10 — GetClassification operation request URL parameters	28
Table 11 — Exception codes for GetClassification operation	31
Table 12 — Parameters in TrainClassifier operation request.....	33
Table 13 — TrainClassifier operation request URL parameters	34
Table 14 — Exception codes for TrainClassifier operation	38
Table 15 — DescribeClassifier operation request URL parameters	40

i. Preface

NOTE The previous version of this document was numbered 04-082, now renumbered 05-017 after approval and final editing in 2005.

The OGC Web Image Classification Service (WICS) results from design and testing in OGC's OWS-2 Initiative. Suggested additions, changes, and comments on this draft report are welcome and encouraged. Such suggestions may be submitted by OGC portal message, email message, or by making suggested changes in an edited copy of this document.

ii. Submitting organizations

The following organizations submitted this Implementation Specification to the Open Geospatial Consortium Inc. as a part of the WICS Request For Comment (RFC):

- a) George Mason University
- b) BAE SYSTEMS

iii. Document contributor contact points

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

Person	Company	Address	Phone	Email
Wenli Yang	George Mason University	9801 Greenbelt Road, Suite 316-317, Lanham, MD21044	+1 301 552 9360	yang@rattler.gsfc.nasa.gov
Yang Liu	George Mason University	9801 Greenbelt Road, Suite 316-317, Lanham, MD21044	+1 301 552 9467	Yliu5@gmu.edu
George Percivall	NASA / GST, Inc.	NASA Goddard Space Flight Center, Code 101.4	+1 301-286-4073	percivall@gsfc.nasa.gov
Arliss Whiteside	BEA Systems		+1 858 592 1608	Arliss.Whiteside@baesystems.com
Sean Forde	LizardTech		+1 206 902 2874	sforde@lizardtech.com

iv. Revision history

Date	Release	Editor	Primary clauses modified	Description
2004-04-05	0.0.1	Wenli Yang		Discussion outline
2004-04-14	0.1.0	Wenli Yang	All	Initial draft, clauses 8, 9, 10, and 11 not finished.
2004-05-04	0.2.0	Wenli Yang	Added operation request and response schema examples to clauses 8, 9, 10, 11.	Second draft. Blank sub-clauses in clauses 8, 9, 10, and 11.
2003-07-12	0.2.1	Wenli Yang	6-11, A-C	Document reorganized, edited most parts
2003-07-26	0.3.0	Wenli Yang	7, 8, 9, 10, 11	Each WICS server is limited to provide only one classifier. The WICS schema is significantly simplified with this limitation. As a result, significant changes have been made to all major clauses.
2004-08-11	0.3.1	Arliss Whiteside	All	Mainly various editorial revisions, rearrangement of paragraphs, Modifications of XML examples in the getCapabilities clause based on the latest OWScommon.xsd and WICS.xsd, Changes in tables of the getCapabilities' content section based on latest WICS.xsd.
2004-09-10	0.3.2	Wenli Yang	All	Various editorial revisions. Added top level UML model, Added terms and definitions, Modified the WICS.xsd and MiniDistance.xsd schemas. Changed XML examples in the document to reflect all XML schema changes made since 2004-08-14.
2005-01-31	0.3.3	Sean Forde	iii,iv, vi	Put front page items in red font; replaced copyright statement; added indentation to contents page; added generalization statement to clause vi, fixed typos
2005-02-08	0.3.3	Arliss Whiteside	cover, i	Final formatting as approved Discussion Paper

v. Changes to the OGC Abstract Specification

The OGC™ Abstract Specification does not require changes to accommodate the technical contents of this document.

vi. Future work

NOTE Generalization of the service interface now specified in this Discussion Paper has not yet been considered by the OGC. Such generalization of this interface will include considering combining the functions supported here with functions currently supported by other OGC (approved and draft) web services.

Foreword

This document, a draft Interoperability Specification, is the first WICS Interoperability Program Report (IPR) and does not supplement or supersede any existing OGC documents.

This specification uses parts of the OGC Web Services Common Specification [OGC 05-008].

This specification includes one normative annex.

Attention is drawn to the possibility that some of the elements of this standard may be the subject of patent rights. Open Geospatial Consortium Inc. shall not be held responsible for identifying any or all such patent rights. To date, no such patent rights have been claimed or identified.

Introduction

The Web Image Classification Service (WICS) supports classification of digital images. A digital image is composed of pixel values organized into one or more two-dimensional arrays. The two dimensions of an image represent two axes in space based on a spatial coordinate reference system. The dimensions of the different 2-D arrays comprising an image must be the same and represent exactly the same spatial locations.

While the 2D array(s) in an image can be of different physical properties (e.g., electromagnetic radiance, temperature, vegetation index, surface wetness), they are usually scaled digital numbers (DN values) representing measurements or observations of certain physical/chemical/biological properties. The service interface specified in this document supports the general classification of any image data.

Image classification uses the information represented by the DN values in one or more Image components, and classifies each individual pixel based on this information. The objective is to assign all pixels in the image to particular classes/clusters or themes (e.g. water, forest, shrubland, grassland, etc.). The resultant image, a single two-dimensional array, is comprised of one grid of pixels, with each pixel belonging to a class or a theme.

There are two types of classifications. Unsupervised classification classifies an image into groups of pixels that are uniform or similar with respect to their DN values. The classification is solely based on the DN values of the image and no prescribed physical meanings of the classes are required. Supervised classification classifies an image into groups of pixels according to a classification scheme and each resultant group corresponds to a class defined in the scheme, which is usually defined based on specific application needs (e.g., water, forest, shrubland, grassland, etc. in land cover classification).

Web Image Classification Service (WICS) Implementation Specification

1 Scope

This document specifies the interface to a Web Image Classification Service (WICS), which performs classification of image data. The WICS interface supports these server abilities:

- a) Perform unsupervised classification.
- b) Perform supervised classification with archived trained classifiers supplied by a WICS server.
- c) Perform training for supervised classification.
- d) Provide parameters of server archived trained classifiers.
- e) Provide parameters of trained classifiers based on client supplied data.
- f) Provide classification schema supported.

2 Conformance

Conformance with this draft OGC Implementation Specification is not specified in this Interoperability Program Report.

3 Normative references

The following normative documents contain provisions that, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

OGC 5-008, *OGC Web Services Common Specification*

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

In addition, this specification includes several normative XML Schema files. These XML Schema files are bundled in the "OWS-2 IH4DS Schema and WSDL Baseline IPR" package (OGC 04-080).

4 Terms and definitions

For the purposes of this specification, the definitions specified in Clause 4 of the OWS Common Specification [OGC 05-008]. In addition, the following terms and definitions apply.

4.1

image classification

process of grouping pixels in an image into a finite number of categories based on the pixel values

4.2

classifier

algorithm or implemented algorithm of image classification

4.3

train

process of defining the criteria by which different categories are recognized

4.4

supervised classification

type of classification which is based on prior knowledge of classes

NOTE In this type of classification, either the parameters about known classes or sample pixels of known classes need to be provided to the classifier.

4.5

unsupervised classification

type of classification which is not based on prior knowledge of classes. In this type of classification, no parameters about the classes are needed

4.6

class label

identifier, usually an integer number, used to indicate different categories

4.7

cluster label

synonym of class label

5 Conventions

5.1 Abbreviated terms

Most of the abbreviated terms listed in Subclause 5.1 of the OWS Common Implementation Specification [OGC 05-008] apply to this document, plus the following abbreviated terms.

2D Two Dimensional

3D	Three Dimensional
DN	Digital Number

5.2 UML notation

Most diagrams that appear in this specification are presented using the Unified Modelling Language (UML) static structure diagram, as described in Subclause 5.2 of [OGC 05-008].

5.3 Document terms and definitions

This document uses the specification terms defined in Subclause 5.3 of [OGC 05-008].

6 WICS Overview

The specified WICS performs classification of digital images. A digital image is composed of pixel values organized into one or more two-dimensional arrays. The two dimensions of an image represent two axes in space, based on a spatial coordinate reference system. The dimensions of the different 2D arrays comprising an image must be the same and represent exactly the same spatial locations.

While the 2D arrays in an image can be of different physical properties (e.g., electromagnetic radiance, temperature, vegetation index, surface wetness), they are usually scaled digital numbers (DN values) representing measurements or observations of certain physical/chemical/biological properties. The service interface specified in this document supports the classification of any image data. The image data values are referred to as DN values hereafter in this document regardless of their physical/thematic meanings.

NOTE In ISO19124, imagery and thematic data are distinguished by defining the former as being directly measured by a sensor, e.g., radiance, and the latter as being derived and/or interpreted from the former, e.g., surface temperature).

Image classification uses the information represented by the DN values in one or more image components and attempts to classify each individual pixel based on this information. The objective is to assign all pixels in the image to particular classes/clusters or themes (e.g. water, forest, shrubland, grassland, etc.). The resultant image, a single two-dimensional array, is comprised of one grid of pixels, with each pixel belonging to a class or a theme.

There are two types of classifications. Unsupervised classification classifies an image into groups of pixels that are uniform or similar with respect to their DN values. The classification is solely based on the DN values of the image, and no prescribed physical meanings of the classes are required. Supervised classification classifies an image into groups of pixels according to a classification scheme, and each resultant group corresponds to a class defined in the scheme, which is usually defined based on specific

application needs (e.g., water, forest, shrubland, grassland, etc. in land cover classification).

Different client/server interactions exist for the two types of classifications. Unsupervised classification can be conducted through one round of client/user interaction, with the client sending the unclassified image together with a set of classification parameters (e.g., number of classes, the minimum between-class distance, etc.). The server classifies the image based on the client-sent parameters, and returns the classified image.

Supervised classification can be performed through either one or two rounds of client/user interaction. Frequently, a client needs to train a classifier supported by a server with a set of training data and a classification schema in the first round of interaction. The client then sends the image to be classified together with the trained parameters. A client may also select a server's pre-trained classifier to conduct supervised classification. In this case, only one round of client/server interaction is needed.

The WICS interface (currently) specifies four operations that can be requested by a client and performed by a WICS server. Those operations are:

- c) GetCapabilities (required implementation by servers) – This operation allows a client to request and receive back a service metadata (or Capabilities) document that describes the abilities of the specific server implementation. This service metadata includes brief descriptions of the classifiers available from the server, including classifier identifier, type of classifier, and a human readable label or a description element briefly explaining the classifier. This operation also supports negotiation of the specification version being used for client-server interactions.
- d) GetClassification (required implementation by servers) – This operation allows a client to specify a classifier and send a coverage (using a server supported data type/format for that classifier), possibly together with classification parameters. The server responds to the request by classifying the coverage using the client identified classifier (and client-sent classification parameters), and returns the classified coverage or a URL of the classified coverage back to the client.
- e) TrainClassifier (optional implementation by servers) – This operation allows a client to specify a classifier and send training data, likely together with parameters defining the classifier, in compliant type/format declared by the server. The client also needs to identify a classification scheme. The server trains the client-specified classifier and returns training parameters in a declared encoding format or the URL of the parameters.
- f) DescribeClassifier (required implementation by servers) – This operation allows a client to request detailed information about a classifier available from the server. The server responds to this operation with an XML document that describes the details of the classifier(s) identified by the client. This information may include brief theoretical base of the classification algorithm(s), supported data types, applicable spatial/temporal extents, and classifier specific parameters and default values for these parameters in case the client does not send values.

The WICS provides three types of classifications using these four operations:

- a) Unsupervised classification: `getCapabilities`, `describeClassifier`, and `getClassification`.
- b) Supervised classification using server archived trained classifiers: `getCapabilities`, `describeClassifier`, and `getClassification`.
- c) Supervised classification using classifiers trained based on client supplied training data: `getCapabilities`, `describeClassifier`, `TrainClassifier`, and `getClassification`.

The WICS may be compared to other OGC Web Services. It is similar to the WCTS in that both services need to take from the client the raw data to be classified or transformed. Unlike the WCTS, the input data a WICS needs is always a grid coverage, and is very large volume in most cases, while a WCTS can transform only a single feature. A WICS is also similar to the WCS in that it returns grid coverages to clients. Unlike the WCS, a WICS needs an unclassified image from the client and possibly training data, both of which are usually very large.

A unique characteristic of WICS is that it must provide detailed descriptions of the classifiers, such as classification parameters and schemes, which are very different from content metadata for geospatial data. Therefore, an operation designed to provide detailed server side information, i.e., the `DescribeClassifier` operation, is mandatory for a WICS. Due to the wide varieties in classification algorithms, it is difficult, if not impossible, to design a classification schema that can address different structures and classification parameters used in different algorithms. In this document, each classifier is considered as an independent WICS server and each server has its own schema to describe the specific structure and parameters. When a specific server is invoked, the classifier-specific schema shall be inserted into the general WICS schema by replacing the abstract element defined in the general schema.

The WICS operations have many similarities to other OGC Web Services, including the WMS, WFS, and WCS. Many of these interface aspects that are common with other OWSs are thus specified in the OGC Web Services Common Implementation Specification [OGC 05-008]. These common aspects are normatively referenced herein, instead of being repeated in this specification.

Figure 1 is a simple UML diagram summarizing the WICS interface. This class diagram shows that the WICS interface class inherits the `getCapabilities` operation from the `OGCWebService` interface class, and adds the `describeClassifier`, `trainClassifier`, and `getClassification` operations. (This capitalization of names uses the OGC/ISO profile of UML.).

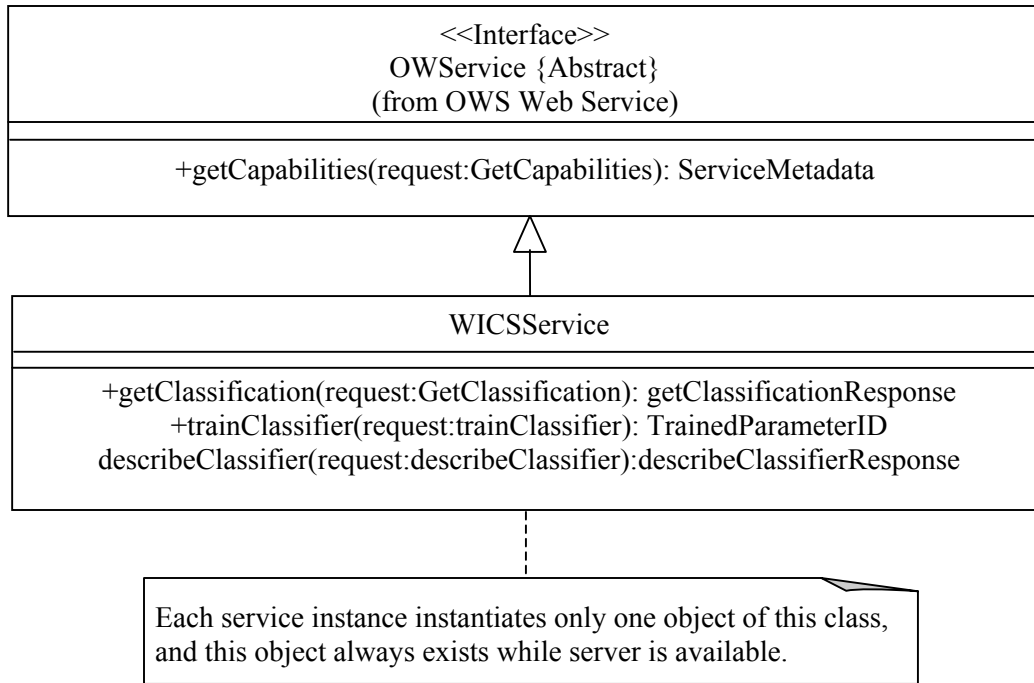


Figure 1 — WICS interface UML diagram

Each of the four WICS operations is described in more detail in subsequent clauses.

7 Shared aspects

7.1 Introduction

This clause describes aspects of Web Image Classification Server behaviour that are independent of particular operations, or that are common to several operations or interfaces.

7.2 Shared operation request parameters

This clause specifies some of the parameters used by multiple operation requests specified in the following clauses. The parameter names, meanings, and formats shall be as specified in Table 1.

Table 1 — Definitions of some operation request and response parameters

Name	Definition	Data type and value	Multiplicity
SourceCoverage	Client input image data file (a grid coverage containing a 3-dimensional image data array).	URL referenced to the source coverage or operation request	One (mandatory for getClassification operation and TrainClassifier operation)
TrainedParameters	Data file containing parameters trained from specific data sets against a supervised classifier.	URL referenced to the TrainedParameters data file	One (mandatory for getClassification operation on a supervised classifier and for TrainClassifier operation)

7.3 Operation request encoding

The encoding of operation requests shall use HTTP GET with KVP encoding and HTTP POST with XML encoding as specified in Clause 11 of [OGC 05-008]. Table 2 summarizes the WICS operations and their encoding methods defined in this specification.

Table 2 — Operation Request Encoding

Operation	Request encoding
GetCapabilities (required)	KVP and optional XML
TrainClassifier (optional)	XML and optional KVP
GetClassification (required)	XML and optional KVP
DescribeClassifier (required)	XML and optional KVP

8 GetCapabilities operation (required)

8.1 Introduction

The mandatory GetCapabilities operation allows WICS clients to retrieve metadata about the services available from a WICS server. The response to a GetCapabilities request shall be an XML document containing service metadata about the server, including brief descriptions of the classifiers available from the server, such as classifier identifier, type, and a human readable label or a description. This clause specifies the XML document that a WICS server must return to describe its capabilities.

8.2 Operation request

The GetCapabilities operation request shall be as specified in Subclauses 7.2.2 through 7.2.4 of [OGC 05-008]. The “service”, “request”, “Sections”, and “AcceptVersions” parameters shall be implemented by all WICS servers. The “updateSequence” parameter is optional implementation by WICS servers. All WICS servers shall implement HTTP GET transfer of the GetCapabilities operation request, using KVP encoding. Servers may

also implement HTTP POST transfer of the GetCapabilities operation request, using XML encoding only.

The value of the “service” parameter shall be “WICS”. The allowed set of service metadata (or Capabilities) XML document section names and meanings shall be as specified in Tables 3 and 7 of [OGC 05-008].

The XML Schema fragment for encoding a WICS GetCapabilities operation request extends ows:GetCapabilitiesType in owsCommon.xsd from Annex A of [OGC 05-008], and is:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.opengis.net/wics"
xmlns:wics="http://www.opengis.net/wics"
xmlns:ows="http://www.opengis.net/ows"
xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
version="0.0.20" xml:lang="en">
  <annotation>
    <appinfo>wicsGetCapabilities.xsd 2004-08-09</appinfo>
    <documentation>This XML Schema encodes the WICS Get Capabilities
operation request message. </documentation>
  </annotation>
  <!-- =====
includes and imports
===== -->
  <import namespace="http://www.opengis.net/ows"
schemaLocation="../../../ows/0.3.20/owsGetCapabilities.xsd"/>
  <!-- =====
GetCapabilities Request
===== -->
  <element name="GetCapabilities">
    <annotation>
      <documentation>Request to a WICS to perform the
GetCapabilities operation. This operation allows a client to retrieve a
Capabilities XML document providing metadata for the specific WICS
server. In this XML encoding, no "request" parameter is included, since
the element name specifies the specific operation. </documentation>
    </annotation>
    <complexType>
      <complexContent>
        <extension base="ows:GetCapabilitiesType">
          <attribute name="service" type="ows:ServiceType"
use="required" fixed="WICS"/>
        </extension>
      </complexContent>
    </complexType>
  </element>
</schema>
```


8.3 Operation response

8.3.1 Normal response

The complete service metadata document shall contain the four sections specified in Table 3. Depending on the values in the Sections parameter of the GetCapabilities operation request, any combination of these sections can be requested and returned.

Table 3 — Section name values and contents

Section name	Contents
ServiceIdentification	Metadata about this specific server. The schema of this section shall be the same as for all OWSes, as specified in Subclause 7.4.3 and owsServiceIdentification.xsd of [OGC 05-008].
ServiceProvider	Metadata about the organization operating this server. The schema of this section shall be the same for all OWSes, as specified in Subclause 7.4.4 and owsServiceProvider.xsd of [OGC 05-008].
OperationsMetadata	Metadata about the operations specified by this service and implemented by this server, including the URLs for operation requests. The schema of this section shall be the same as for all OWSes, as specified in Subclauses 7.4.5 and owsOperationsMetadata.xsd of [OGC 05-008].
Contents	Metadata about the data served by this server. For the WICS, this section shall contain summary data about the classifiers and classification algorithms and schemas known to this server, as specified in Subclauses 8.3.3 and 8.3.4 below.

In addition to these sections, each service metadata document shall include the mandatory “version” and optional updateSequence parameters specified in Table 6 in Subclause 7.4.1 of [OGC 05-008].

8.3.2 OperationsMetadata section standard contents

For the WICS, the XML Schema of the OperationsMetadata section shall be the same as for all OGC Web Services, as specified in Subclause 7.5 and owsOperationsMetadata.xsd of [OGC 05-008]. The mandatory values of various (XML) attributes shall be as specified in Table 4. Similarly, the item listed in Table 5 shall be included when this operation is implemented. In Table 4 and Table 5, the “Attribute name” column uses dot-separator notation to identify parts of a parent item. The “Attribute value” column references an operation parameter, in this case an operation name, and the meaning of including that value is listed in the right column.

Table 4 — Required values of OperationMetadata section attributes

Attribute name	Attribute value	Meaning of value
Operation.name	GetCapabilities	The GetCapabilities operation is implemented by this server.
	GetClassification	The GetClassification operation is implemented by this server.
	DescribeClassifier	The DescribeClassifier operation is implemented by server.

Table 5 — Optional values of OperationMetadata section attributes

Attribute name	Attribute value	Meaning of value
Operation.name	TrainClassifier	The TrainClassifier operation is implemented by this server.

In addition to the optional values listed in Table 5, there are many optional values of the “name” attributes and “value” elements in the OperationsMetadata section, which may be included when considered useful. Most of these attributes and elements are for recording the domains of various parameters and quantities. For example, the domain of the exceptionCode parameter could record all the codes implemented for each operation by that specific server. Similarly, each of the GetCapabilities operation optional request parameters might have its domain recorded. For example, the domain of the Sections parameter could record all the sections implemented by that specific server.

8.3.3 Contents section contents

The Contents section of a service metadata document contains metadata about the data served by this server. For the WICS, this Contents section shall contain brief information about the classifier known to this server, including identifier, type of classifier, brief and detailed description of the classifier, plus supported source and return image coverage formats. The Contents section shall include the parameters specified in Table 6.

Table 6 — Parts of Contents section

Name	Definition	Data type	Multiplicity and use
ClassifierBrief	Brief metadata about one classifier available in this WICS server (TBR)	ClassifierBrief data type, See Table 7	One (mandatory)

Table 7 —Parameters in ClassifierBrief data type

Name	Definition	Data type and value	Multiplicity and use
Name	Identifier of classifier which the server can perform (TBR)	Character String, not empty	One (mandatory)
Title	Human readable title of classifier which the server can perform (TBR)	Character String, not empty	Zero or one (optional)
Abstract	Human readable brief description of classifier which the server can perform (TBR)	Character String, not empty	Zero or one (optional)
Class	Identifier of the class of classifier which the server can perform	Code List, with allowed values: “Supervised” “Unsupervised”	One (mandatory)
Parameter SchemaURL	Reference to a document that provides actual schema for the classifier.	URL	One (Mandatory)
SourceImageFormat	Identifier of input image format supported by classifier	Character String type, not empty	One or more (mandatory) One for each format supported, unordered
ClassifiedCoverageFormat	Identifier of returned classified grid coverage supported by classifier	Character String type, not empty	One or more (mandatory) One for each format supported, unordered
KVPPParameter	KVP that shall provide to the classifier when using http GET method.	KVPPParameter, see Table 8	One (mandatory)

Table 8 — Parameters in KVPPParameterType data type

Name	Definition	Data type	Multiplicity
Keyword	The name of keyword in KVP	Character String, not empty	One (mandatory)
ValueType	Value type for the corresponding keyword	Character String, not empty	One (mandatory)
required	Indicating if the specific KVP is mandatory	Boolean	One (mandatory)

Any WICS server must support one classification, either supervised or unsupervised. If supervised classification is available, the WICS must also support a) performing training based on a 2D training image on which pixel values are class identification numbers and a source image coverage; and b) perform supervised classification on a source image coverage and a trained parameter data file. The trained data parameter file shall be specified using a URL, and it may either be stored in the client side, the server side, or any other remote locations.

The format of the trained parameters shall be defined by the server originally performing the training. When specifying the URL to the trained parameter file, the client must make sure that the format and content of the parameters stored in the file is consistent with the classifier provided by the intended server, and with the source image coverage the client is sending. Generally, the trained parameter data file is only suitable to be sent back to the server originally producing the file, to perform classification.

8.3.4 Capabilities document XML encoding

A XML Schema fragment for a WICS service metadata document extends `ows:CapabilitiesBaseType` in `owsCommon.xsd` of [OGC 05-008], and is:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.opengis.net/wics"
xmlns:wics="http://www.opengis.net/wics"
xmlns:ows="http://www.opengis.net/ows"
xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
version="0.0.20" xml:lang="en">
  <annotation>
    <appinfo>wicsCapabilities.xsd 2004-08-10</appinfo>
    <documentation>This XML Schema encodes the WICS GetCapabilities
operation response message. </documentation>
  </annotation>
  <!-- =====
includes and imports
===== -->
  <import namespace="http://www.opengis.net/ows"
schemaLocation="../../../ows/0.3.20/owsGetCapabilities.xsd"/>
  <!-- =====
GetCapabilities Response
===== -->
  <element name="Capabilities">
    <annotation>
      <documentation>XML encoded WICS GetCapabilities operation
response. This document provides clients with service metadata about a
specific service instance, including metadata about the tightly-coupled
data served. If the server does not implement the updateSequence
parameter, the server shall always return the complete Capabilities
document, without the updateSequence parameter. When the server
implements the updateSequence parameter and the GetCapabilities
operation request included the updateSequence parameter with the current
value, the server shall return this element with only the "version" and
"updateSequence" attributes. Otherwise, all optional elements shall be
included or not depending on the actual value of the Contents parameter
in the GetCapabilities operation request. </documentation>
    </annotation>
    <complexType>
      <complexContent>
        <extension base="ows:CapabilitiesBaseType">
          <sequence>
            <element name="Contents">
              <annotation>
                <documentation>Contents section of WICS
service metadata (or Capabilities) XML document. For the WICS, these
```

```

contents are data and functions (TBR) that the WICS server supports (not
data that can be accessed from this server). </documentation>
    </annotation>
    <complexType>
      <sequence>
        <element name="ClassifierDescriptionBrief"
type="wics:ClassifierBriefingType"/>
      </sequence>
    </complexType>
  </element>
</sequence>
</extension>
</complexContent>
</complexType>
</element>
<!-- ===== -->
<complexType name="ClassifierBriefingType">
  <annotation>
    <documentation>Brief metadata about one classifier available
in this WICS server (TBR). </documentation>
  </annotation>
  <sequence>
    <element name="Name" type="string">
      <annotation>
        <documentation>Identifier of classifier which the
server can perform (TBR). </documentation>
      </annotation>
    </element>
    <element ref="ows:Title" minOccurs="0">
      <annotation>
        <documentation>Human readable title of classifier which
the server can perform (TBR). </documentation>
      </annotation>
    </element>
    <element ref="ows:Abstract" minOccurs="0">
      <annotation>
        <documentation>Human readable brief description of
classifier, such as maximum likelihood supervised classification,
ISODATA unsupervised classification etc. </documentation>
      </annotation>
    </element>
    <element name="Class" type="wics:ClassifierClassType"/>
    <element name="ParameterSchemaURL" type="anyURI"
minOccurs="0">
      <annotation>
        <documentation>Reference to a document that provides
more detailed in-depth description of the classifier, such as algorithm
theoretical document base. </documentation>
      </annotation>
    </element>
    <element name="SourceImageFormat" type="ows:MimeType"
maxOccurs="unbounded">
      <annotation>
        <documentation>Unordered list of identifiers of source
coverage formats supported by classifier. </documentation>
      </annotation>
    </element>

```

```

        <element name="ClassifiedCoverageFormat" type="ows:MimeType"
maxOccurs="unbounded">
            <annotation>
                <documentation>Unordered list of identifiers of return
coverage formats supported by classifier. </documentation>
            </annotation>
        </element>
        <element name="KVPPParameter" type="wics:KVPPParameterType"
maxOccurs="unbounded">
            <annotation>
                <documentation> Unordered list of KVP parameters for the specific classifier. To
use this classifier in an operation invoked using GET, the client must provide the KVP parameters defined here,
which are only for this classifier. </documentation>
            </annotation>
        </element>
    </sequence>
</complexType>
<!-- ===== -->
<simpleType name="ClassifierClassType">
    <annotation>
        <documentation>Identifier of the class of classifier which the
server can perform (TBR). </documentation>
    </annotation>
    <restriction base="string">
        <enumeration value="Unsupervised"/>
        <enumeration value="Supervised"/>
    </restriction>
</simpleType>
<!-- ===== -->
<complexType name="KVPPParameterType">
    <annotation>
        <documentation> KVP parameters for a classifier </documentation>
    </annotation>
    <sequence>
        <element name="Keyword" type="string">
            <annotation>
                <documentation> Name of the keyword in KVP
</documentation>
            </annotation>
        </element>
        <element name="ValueType" type="string">
            <annotation>
                <documentation> The type of value for the corresponding
keyword </documentation>
            </annotation>
        </element>
    </sequence>
    <attribute name="required" type="boolean" default="true">
        <annotation>
            <documentation> Attribute indicating if the KVP is
mandatory </documentation>
        </annotation>
    </attribute>
</complexType>
</schema>

```

As indicated, this XML Schema uses the `owsServiceIdentification.xsd`, `owsServiceProvider.xsd`, and `owsOperationsMetadata.xsd` schemas specified in [OGC 05-008].

8.3.5 Exceptions

When a web image classification server encounters an error while performing a `GetCapabilities` operation, it shall return an exception report message as specified in Clause 8 of [OGC 05-008]. The allowed exception codes shall include those listed in Table 5 of [OGC 05-008], if the `updateSequence` parameter is implemented by the server.

8.4 Examples

8.4.1 GetCapabilities request

To request a capabilities document, a client could issue the following `GetCapabilities` operation request with minimum contents encoded using KVP:

```
http://laits.gmu.edu/WICS_ISODATA?service=WICS&request=getCapabilities
```

The corresponding minimum request encoded in XML is:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetCapabilities xmlns="http://www.opengis.net/wics"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics WICS.xsd"
service="WICS"/>
```

8.4.2 GetCapabilities response

In response to such a request, the web image classification server might generate a document that looks like:

```
<?xml version="1.0" encoding="UTF-8"?>
<Capabilities xmlns="http://www.opengis.net/wics"
xmlns:ows="http://www.opengis.net/ows"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics WICS.xsd"
version="0.0.20">
  <ows:ServiceIdentification>
    <ows:ServiceType>OGC WICS</ows:ServiceType>
    <ows:ServiceTypeVersion>1.0.0</ows:ServiceTypeVersion>
    <ows:Title>Web Image Classification Service</ows:Title>
    <ows:Abstract>Network service for classifying an image grid
coverage </ows:Abstract>
    <ows:Keywords>
      <ows:Keyword>Image Classification</ows:Keyword>
      <ows:Keyword>Supervised classification</ows:Keyword>
      <ows:Keyword>Training</ows:Keyword>
    </ows:Keywords>
    <ows:Fees>NONE</ows:Fees>
```

```

    <ows:AccessConstraints>NONE</ows:AccessConstraints>
  </ows:ServiceIdentification>
  <ows:ServiceProvider>
    <ows:ProviderName>George Mason University</ows:ProviderName>
    <ows:ProviderSite xlink:href="laits.gmu.edu/WICS"/>
    <ows:ServiceContact>
      <ows:IndividualName>Liping Di</ows:IndividualName>
      <ows:PositionName>Director, LAITS</ows:PositionName>
      <ows:ContactInfo>
        <ows:Phone>
          <ows:Voice>+01 301 552 9496</ows:Voice>
          <ows:Facsimile>+01 301 552 9671</ows:Facsimile>
        </ows:Phone>
        <ows:Address>
          <ows:DeliveryPoint>9801 Greenbelt
Road</ows:DeliveryPoint>
          <ows:City>Lanham</ows:City>

          <ows:AdministrativeArea>Maryland</ows:AdministrativeArea>
          <ows:PostalCode>20706</ows:PostalCode>
          <ows:Country>USA</ows:Country>

        <ows:ElectronicMailAddress>lpd@rattler.gsfc.nasa.gov</ows:Electronic
MailAddress>
      </ows:Address>
    </ows:ContactInfo>
  </ows:ServiceContact>
</ows:ServiceProvider>
<ows:OperationsMetadata>
  <ows:Operation name="GetCapabilities">
    <ows:DCP>
      <ows:HTTP>
        <ows:Get xlink:href="laits.gmu.edu/WICS"/>
      </ows:HTTP>
    </ows:DCP>
  </ows:Operation>
  <ows:Operation name="GetClassification">
    <ows:DCP>
      <ows:HTTP>
        <ows:Get xlink:href="laits.gmu.edu/axis/WICS"/>
      </ows:HTTP>
    </ows:DCP>
  </ows:Operation>
  <ows:Operation name="TrainClassifier">
    <ows:DCP>
      <ows:HTTP>
        <ows:Get xlink:href="laits.gmu.edu/WICS"/>
      </ows:HTTP>
    </ows:DCP>
  </ows:Operation>
  <ows:Operation name="DescribeClassifier">
    <ows:DCP>
      <ows:HTTP>
        <ows:Get xlink:href="laits.gmu.edu/WICS"/>
      </ows:HTTP>
    </ows:DCP>
  </ows:Operation>
</ows:OperationsMetadata>

```



```

<Contents>
  <ClassifierDescriptionBrief>
    <Name>TBD</Name>
    <Class>Unsupervised</Class>
    <SourceImageFormat>application/HDF-EOS</SourceImageFormat>
    <ClassifiedCoverageFormat>application/HDF-
EOS</ClassifiedCoverageFormat>
    <KVPPParameter required="false">
      <Name>TBD</Name>
      <ValueType>TBD</ValueType>
    </KVPPParameter>
  </ClassifierDescriptionBrief>
</Contents>
</Capabilities>

```

9 GetClassification (required)

9.1 Introduction

The GetClassification operation allows WICS clients to request classification of an input image. The image grid coverage is usually three-dimensional data array, although two-dimensional might be possible in very limited circumstances. Input image data must be encoded in a format supported by the intended classifier and shall be provided as a URL referencing an image grid coverage file or as a request to a server providing image data, such as to a WCS.

9.2 Operation request

9.2.1 GetClassification request parameters

A request to perform the GetClassification operation shall include the parameters listed and defined in Table 9. This table also specifies the UML model data type, source of values, and multiplicity of each listed parameter, plus the meaning to servers when each optional parameter is not included in the operation request. In addition to the input image data, either classifier parameters or an UUID of trained parameter data file shall be provided.

NOTE 1 To reduce the need for readers to refer to other documents, the first three parameters listed below are largely copied from Table 21 in Subclause 9.2.1 of [OGC 05-008].

Table 9 — Parameters in GetClassification operation request

Name	Definition	Data type and value	Multiplicity and use
Service	Service type identifier	Character String type, not empty Value is OWS type abbreviation, namely “WICS”	One (mandatory)
Request	Operation name	Character String type, not empty Value is operation name, namely “GetClassification”	One (mandatory)
Version	Specification version for operation	Character String type, not empty Value is specified by each Implementation Specification and Schemas version	One (mandatory)
SourceImage	Image coverage data to be classified	One URL or one operation request that returns an image coverage	One (Mandatory)
Classifier Parameters	Parameters that define the structure of a classifier, such as the number of class and maximum iterations in an ISODATA classifier and the numbers of layers and nodes in a multiple layer perceptron (MLP) neural network classifier. It should be noted that this element shall define only the static configuration parameters values of a classifier and shall not include such dynamic values as the connection weights in a MLP network.	This is an abstract element. Each server (Note: one server serves only one classifier) shall define its own element to substitute this abstract element. The parameters to be defined in this element shall include those specifying the configuration of a classifier, such as the number of class and maximum iterations in an ISODATA classifier and the numbers of layers and nodes in a multiple layer perceptron (MLP) neural network classifier. It should be noted that this element shall define only the static configuration parameters values of a classifier and shall not include such dynamic values as the connection weights in a MLP network. Those latter parameters are obtained through the TrainClassifier operation and shall be included, together with the static configuration parameters, in a data file referenced by the trainedParameterID	Zero or one (optional) Include only for unsupervised classification
Trained ParameterID	Identifier of data file to be used as input to a supervised classification	URI Must be result of valid TrainClassifier operation.	Zero or one (optional) Include only for supervised classification

Name	Definition	Data type and value	Multiplicity and use
Format	Identifier of input image format	Character String type, not empty Must be one listed as sourceImageFormat in Capabilities document	One (mandatory)
Classified Coverage Format	Identifier of returned classified grid coverage	Character String type, not empty Must be one listed as classifiedCoverageFormat in Capabilities document	One (optional) If not used, the input image coverage format, as identified by the "Format" parameter, shall be used as classified coverage format

There are different ways to invoke a classification operation. In an unsupervised classification, a WICS client shall specify an input image grid coverage URL (or an operation request to another service such as WCS which returns a URL), and classifier parameters. The WICS shall respond with a classified image referenced to by a URL.

In a supervised classification, a WICS client may either use a pre-trained classifier for a given classification scheme or train a classifier using the TrainClassifier operation (to be described later) before requesting the GetClassification operation. In the first case, the client shall specify a universal unique identifier, TrainedParameterID, which references a data file obtained from a previous TrainClassifier operation. In the second case, the client shall request the TrainClassifier first. In both cases, the client shall specify a classifier and an input image grid data.

It should be noted that in supervised classification, the file referenced by the TrainedParameterID already contains a classifier's static structure parameters such number of layers and nodes in a MLP), in addition to dynamic parameters such as MLP's connection weights. Therefore, no ClassifierParameters are needed when requesting supervised classification. In unsupervised classification, clients must either provide the ClassifierParameters or use classifier's default ClassifierParameter values.

Sometimes a single application may need hybrid classifications. For example, an unsupervised classification for a large image may be done by first applying an unsupervised clustering algorithm to a subset of image data resampled from the original image and then using the cluster centres as training classes to train a supervised classifier. Finally, the original image is classified by the supervised classifier. This type of applications can be performed by chaining unsupervised GetClassification operation, TrainClassifier operation and supervised GetClassification operation.

9.2.2 GetClassification request KVP encoding (optional)

WICS servers can implement HTTP GET transfer of the GetClassification operation request, using KVP encoding. The KVP encoding of the GetClassification operation request shall use the parameters specified in Table 8. The parameters listed in Table 10 shall be as specified in Table 9 above.

Table 10 — GetClassification operation request URL parameters

Name and example ^a	Optionality and use	Definition and format
service=WICS	Mandatory	Service type identifier
request= GetClassification	Mandatory	Operation name
Version	Mandatory	Specification and schema version for this operation
SourceImage	Mandatory	Unique identifier URL of input image coverage.
Format	Mandatory	A character string identifying one of the coverage data formats supported by the server.
ClassifierParameters	Mandatory for unsupervised classifiers. Not used for supervised classifiers Included for supplying parameters for the classifier.	Multiple KVP pairs. The specific contents of these KVP pairs shall be declared by specific WICS servers in the Contents section of the GetCapabilities response.
TrainedParameterID	Mandatory for supervised classification. Not used for unsupervised classification	Unique identifier of a parameter data file.
ClassifiedCoverageFormat	Optional Included for specifying the format of the resultant classified data. If not specified, the input image coverage format shall be used.	A character string identifying one of the coverage data formats supported by the server.
<p>A All parameter names are here listed using mostly lower case letters. However, any parameter name capitalization shall be allowed in KVP encoding, see Subclause 11.5.2 of [OGC 05-008].</p>		

9.2.3 GetClassification request XML encoding (required)

All WICS servers shall implement HTTP POST transfer of the GetClassification operation request, using XML encoding only. The following schema fragment specifies the contents and structure of a GetClassification operation request encoded in XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Originally edited by Yang Liu (yliu5@gmu.edu), Wenli Yang, George
Mason University on 06/11/2004, 07/21/2004, 07/26/2004 -->
<schema targetNamespace="http://www.opengis.net/wics"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://www.opengis.net/xlink"
xmlns:ows="http://www.opengis.net/ows"
xmlns:wics="http://www.opengis.net/wics" elementFormDefault="qualified"
version="0.0.20" xml:lang="en">
  <annotation>
    <appinfo>wics.xsd 2004-08-28</appinfo>
    <documentation>This XML Schema encodes the WICS GetCapabilities
operation request message. </documentation>
  </annotation>
<!-- =====
```

```

    includes and imports
    ===== -->
    <include schemaLocation="./wics.xsd"/>
    <import namespace="http://www.opengis.net/ows"
schemaLocation="http://schemas.opengis.net/ows/owsCommon.xsd"/>
    <!-- =====
        GetCapabilities request
    ===== -->
    <element name="GetClassification">
      <annotation>
        <documentation>Request to a WICS to perform the
GetClassification operation. This operation classifies an input
SourceImage into a two-dimensional thematic data layer whose content are
usually integer numerical values representing certain themes such as
forest, grassland, and cropland. The SourceImage shall be a 3D image
grid coverage. In this XML encoding, no "request" parameter is included,
since the element name specifies the specific operation.
</documentation>
      </annotation>
      <complexType>
        <complexContent>
          <extension base="wics:RequestBaseType">
            <sequence>
              <element name="SourceImage" type="ows:ReferenceType">
                <annotation>
                  <documentation>Reference to a remote resource or a
local payload. This reference can have a human readable description and
a link to metadata. Specifically in this schema the reference shall
include an URI referenced to a 3D image to be classified and a format
identify the format of this image. </documentation>
                </annotation>
              </element>
              <choice>
                <annotation>
                  <documentation>Two different types of input data are
required by the GetClassification operation, in addition to the
SourceImage. For an unsupervised classifier, WICS clients shall provide
values for or use default values of the wics:ClassifierParameters
element. For a supervised classifier, WICS clients shall provide a
TrainedParametersID, as returned by TrainClassifier operation.
</documentation>
                </annotation>
                <element ref="wics:ClassifierParameters"/>
                <element name="TrainedParametersID" type="string"/>
              </choice>
              <element name="ClassifiedCoverageFormat"
type="ows:MimeType" minOccurs="0">
                <annotation>
                  <documentation>Identification of format to be used
for the classification result coverage. If not specified, the same
format as the SourceImage shall be used. </documentation>
                </annotation>
              </element>
            </sequence>
          </extension>
        </complexContent>
      </complexType>
    </element>

```

```
</schema>
```

9.3 Operation response

9.3.1 GetClassification normal response

The normal response to a GetClassification operation request is a URL pointing to the classified coverage data. The following XML Schema fragment specifies the contents and structure of a GetClassification operation response:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Originally edited by Yang Liu (yliu5@gmu.edu), Wenli Yang, George
Mason University on 06/11/2004, 07/21/2004, 07/26/2004 -->
<schema targetNamespace="http://www.opengis.net/wics"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://www.opengis.net/xlink"
xmlns:ows="http://www.opengis.net/ows"
xmlns:wics="http://www.opengis.net/wics" elementFormDefault="qualified"
version="0.0.0" xml:lang="en">
  <annotation>
    <appinfo>wics.xsd 2004-08-26</appinfo>
    <documentation>This XML Schema encodes the WICS GetCapabilities
operation request message. </documentation>
  </annotation>
  <!-- =====
includes and imports
===== -->
  <include schemaLocation="./wics.xsd"/>
  <import namespace="http://www.opengis.net/ows"
schemaLocation="http://schemas.opengis.net/ows/owsCommon.xsd"/>
  <!-- =====
GetCapabilities response
===== -->
  <element name="Classification">
    <annotation>
      <documentation>Response to a valid GetClassification operation
request sent to a WICS. </documentation>
    </annotation>
    <complexType>
      <sequence>
        <element name="ClassifiedCoverage" type="ows:ReferenceType">
          <annotation>
            <documentation> Reference to a remote resource or a local
payload. This reference can have a human readable description and a link
to metadata. Specifically in this schema the reference shall include an
URI referenced to a 2D image which is the result of classification.
</documentation>
          </annotation>
        </element>
      </sequence>
    </complexType>
  </element>
</schema>
```

9.3.2 Exceptions

When a web image classification server encounters an error while performing a GetClassification operation, it shall return an exception report message as specified in Subclause 7.4 of the OGC Web Services Common Implementation Specification. The allowed exception codes shall include those listed in Table 9. For each listed exceptionCode, the contents of the “locator” parameter value shall be as specified in the right column of Table 11.

NOTE To reduce the need for readers to refer to other documents, the first three values listed below are copied from Table 20 in Subclause 8.3 of [OGC 05-008].

Table 11 — Exception codes for GetClassification operation

exceptionCode value	Meaning of code	“locator” value
MissingSourceImage	No input image coverage data was available from the specified source for input source image ^a	Value of SourceImage parameter
InvalidSourceImageURI	Request contains invalid URI for input source image	Name of parameter
InvalidFormat	The format name for the input source image is not supported by the server	Name of the invalid format
InvalidClassifiedCoverageFormat	The format name for the output classified coverage is not supported by the server	Name of the invalid format
InvalidClassifierParameters	Specified value for the ClassifiedParameters inconsistent with that required by the classifier.	Name of the invalid format
InvalidTrainedParameterID	Request contains invalid UUID for the TrainedParameterID parameter	Name of parameter
TBD	TBD	TBD
<p>A The specified source for input image could be the URL of the file containing an image data or an OWS operation request to get an image coverage. If remote access of input image was requested, and an exception report or other error response was returned by that resource, then the first ExceptionText element shall contain the received exception report or other error response.</p>		

9.4 Examples

9.4.1 GetClassification request

If a client wants to use a supervised minimum distance classifier to classify an HDF-EOS encoded image coverage data name residing at <http://laits.gmu.edu/WICSInputCovaerage/MODIS.hdf>, it can issue the following XML encoded GetClassification operation request:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetClassification xmlns="http://www.opengis.net/wics"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics ../wics.xsd"
service="WICS" version="1.0.0">
```

```

    <sourceImage xlink:href="http://laits.gmu.edu/WCSInputCovaerage/MODIS.hdf">
      <Format> HDF-EOS </Format>
    </sourceImage>
    <trainedParametersID>
http://laits.gmu.edu/WCSInput/parametersForMODIS.hdf
</trainedParametersID>
    <classifiedCoverageFormat> HDF-EOS </classifiedCoverageFormat>
</GetClassification>

```

9.4.2 GetClassification response

The response to a request for classifying an image coverage to a thematic coverage the server could be, if no exception occurs:

```

<?xml version="1.0" encoding="UTF-8"?>
<Classification xmlns="http://www.opengis.net/wics"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics ../wics.xsd"
service="WICS" version="1.0.0">
  <classifiedCoverage
xlink:href="http://laits.gmu.edu/WCSInputCovaerage/ClassifiedMODIS.hdf">
    <Format> HDF-EOS </Format>
  </classifiedCoverage>
</Classification>

```

10 TrainClassifier

10.1 Introduction

The TrainClassifier operation allows WICS clients to train a supervised classifier supported by WICS with client supplied training data. Training data include two separate data sets which shall be stored in two separate data files. One data set contains the image data on which training is to be performed. The image data is usually three-dimensional data array, although in very limited occasions two-dimensional might also be allowed. The other data set contains a two-dimensional data array whose values are class labels. The two data sets must have the same numbers of rows and columns and must be geographically co-registered. WICS clients must encode the training data based on the encoding format supported by the server. The training data sets shall be provided as two URLs referenced to the two data set files.

10.2 Operation request

10.2.1 TrainClassifier request parameters

A request to perform the TrainClassifier operation shall include the parameters listed and defined in Table 12. This table also specifies the UML model data type, source of values, and multiplicity of each listed parameter, plus the meaning to servers when each optional parameter is not included in the operation request. Parameters common to the other

operations which are listed in Table 2, i.e., Service, Request, and Version, are not re-listed in the current table.

Table 12 — Parameters in TrainClassifier operation request

Name	Definition	Data type and value	Multiplicity and use
TrainingImage	Image coverage data on which training shall be performed	One URL or one operation request which returns an image coverage	One (Mandatory)
TrainingLabelImage	Two-dimensional grid coverage whose values are known class labels.	One URL or one operation request which returns an image coverage	One (Mandatory)
TrainingImageFormat	Format of the input training grid image coverage	Character String type, not empty Must be one of those listed under the sourceImageFormats of the classifier's supportedFormats	One (Mandatory)
TrainingLabelImageFormat	Format of the training label image	Character String type, not empty Must be one of those listed under the sourceImageFormats of the classifier's supportedFormats	One or zero (optional) Assumed to the same as the TrainingImageFormat if not present
ClassifierParameters	The same as defined in Table 7, subclause 9.2.1 of this document	The same as defined in Table 7, subclause 9.2.1 of this document	One (Mandatory)

The response of the TrainClassifier, trainedParameterID, shall be a UUID of the trained parameters data file generated from the training process. This data file shall be kept at the server side and the encode format is internal to the classifier. Thus, the requesting client does not need to know this internal encoding format. The server may announce an expiration date and time of this UUID if the server will not keep this file permanently. The client can use this trained parameter data in subsequent GetClassification request through referencing to the UUID before the expiration date and time. Alternatively, the client might also store this data file in client side or any other locations and generate a UUID for the stored file. In this way, the client might be able to use this file after the expiration date and time. However, it should be noted it is very likely that a client may not be able to correctly keep track of the trained parameter data files after store them in client side or remote locations, especially after multiple TrainClassifier operations are sent to multiple servers. Thus, in order to make sure that the trained data parameters submitted to it is correct, a WICS server may not allow clients to actually get this file.

10.2.2 TrainClassifier request KVP encoding (optional)

WICS servers can implement HTTP GET transfer of the TrainClassifier operation request, using KVP encoding. The KVP encoding of the TrainClassifier operation request

shall use the parameters specified in Table 13. The parameters listed in Table 11 shall be as specified in Table 12 above.

Table 13 — TrainClassifier operation request URL parameters

Name and example ^a	Optionality and use	Definition and format
Service=WICS	Mandatory	Service type identifier
Request= TrainClassifier	Mandatory	Operation name
Version	Mandatory	Specification and schema version for this operation
TrainingImage	Mandatory	A URL or operation request to an image grid coverage
TrainingLabelImage	Mandatory	A URL or operation request to a two dimensional grid coverage containing classification labels
ClassifierParameterst	Mandatory The same as that defined in Table 8, subclause 9.2.2 of this document	The same as that defined in Table 8, subclause 9.2.2 of this document
TrainingImageFormat	Mandatory	A character string identifying the format of trainingImage data file. The format must be supported by the server
TrainingLableImageForm at	optional	A character string identifying the format of TrainingLabelImage data. Assumed to be the same as the TrainingImageFormat if not present. The format must be supported by the server
<p>^a All parameter names are here listed using mostly lower case letters. However, any parameter name capitalization shall be allowed in KVP encoding, see Subclause 11.5.2 of [OGC 05-008].</p>		

10.2.3 TrainClassifier request XML encoding (required)

All WICS servers shall implement HTTP POST transfer of the TrainClassifier operation request, using XML encoding only. The following schema fragment specifies the contents and structure of a TrainClassifier operation request encoded in XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Originally edited by Yang Liu (yliu5@gmu.edu), Wenli Yang, George
Mason University on 06/11/2004, 07/21/2004, 07/26/2004 -->
<schema targetNamespace="http://www.opengis.net/wics"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://www.opengis.net/xlink"
xmlns:ows="http://www.opengis.net/ows"
xmlns:wics="http://www.opengis.net/wics" elementFormDefault="qualified"
version="0.0.0" xml:lang="en">
  <annotation>
    <appinfo>wics.xsd 2004-07-26</appinfo>
    <documentation>This XML Schema encodes the WICS GetCapabilities
operation request message. </documentation>
```

```

</annotation>
<!-- =====
      includes and imports
===== -->
<include schemaLocation="./wics.xsd"/>
<import namespace="http://www.opengis.net/ows"
schemaLocation="http://schemas.opengis.net/ows/owsCommon.xsd"/>
<!-- =====
      TrainClassifier request
===== -->
<element name="TrainClassifier">
  <annotation>
    <documentation>Request to a WICS to perform the TrainClassifier
operation. This operation trains a supervised classifier with an input
training data set whose target thematic meaning is already known. Each
pixel in the input training data is assigned to one thematic class. The
current classifier is trained using the training data to determine the
parameters of the classifier. This operation only exists for a
supervised classifier. </documentation>
  </annotation>
  <complexType>
    <complexContent>
      <extension base="wics:RequestBaseType">
        <sequence>
          <element name="TrainingImage" type="ows:ReferenceType">
            <annotation>
              <documentation>The input training image used
together with training label image to train the specified
classifier.</documentation>
            </annotation>
          </element>
          <element name="TrainingLabelImage"
type="ows:ReferenceType">
            <annotation>
              <documentation>The image containing class labels
used together with training image to train the specified
classifier.</documentation>
            </annotation>
          </element>
          <element ref="wics:ClassifierParameters" minOccurs="0">
            <annotation>
              <documentation>The input parameters for the selected
classifier. These parameters shall determine the structure of a
classifier but shall not determine the actual coefficients/weights of a
classifier. For example, in a MLP network, the parameters shall
determine the number of hidden layers and number of nodes in each hidden
layer but they shall not determine the connection weights among layers.
The latter shall be determined after the training is finished. This
element can be omitted if the element to substitute this abstract
element is announced optional by a server, which indicates that the
server has default or fixed values for all classifier-specific
parameters. This parameter shall not be used in supervised
classification where trained classifier parameters are supplied through
a TrainedParametersID. </documentation>
            </annotation>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</element>

```

```

        </complexContent>
    </complexType>
</element>
</schema>

```

10.3 Operation response

10.3.1 TrainClassifier normal response

The normal response to a TrainClassifier operation request is a UUID pointing to the trained classifier parameter data file. The following XML Schema fragment specifies the contents and structure of a TrainClassifier operation response:

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Originally edited by Yang Liu (yliu5@gmu.edu), Wenli Yang, George
Mason University on 06/11/2004, 07/21/2004, 07/26/2004 -->
<schema targetNamespace="http://www.opengis.net/wics"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://www.opengis.net/xlink"
xmlns:ows="http://www.opengis.net/ows"
xmlns:wics="http://www.opengis.net/wics" elementFormDefault="qualified"
version="0.0.0" xml:lang="en">
    <annotation>
        <appinfo>wics.xsd 2004-07-26</appinfo>
        <documentation>This XML Schema encodes the WICS GetCapabilities
operation request message. </documentation>
    </annotation>
    <!-- =====
        includes and imports
        ===== -->
    <include schemaLocation="./wics.xsd"/>
    <import namespace="http://www.opengis.net/ows"
schemaLocation="http://schemas.opengis.net/ows/owsCommon.xsd"/>
    <!-- =====
        TrainClassifier response
        ===== -->
    <element name="TrainedParameters">
        <annotation>
            <documentation>Response to a valid TrainClassifier operation
request.</documentation>
        </annotation>
        <complexType>
            <sequence>
                <element name="ID" type="string">
                    <annotation>
                        <documentation>Unique identifier that references a data
file containing the trained parameters, to be used as an input in a
GetClassification operation request sent to the same WICS server
providing supervised classification. The trained parameters shall
include the parameters necessary to perform the supervised
classification using the same classifier as used in the training.
Examples of the trained parameters include the mean and covariance
matrices derived from a maximum likelihood classifier, and node
connection weights of a multiple layer perceptron neural network
classifier. The format of this parameter file is internal to a WICS
server, and the client does not need to know this format. In a

```

```

subsequent GetClassification request, WICS clients can send this
parameter data file ID, together with the input image grid coverage to
be classified, to the WICS server and the server shall get the correct
parameters from the file referenced by the ID. </documentation>
  </annotation>
</element>
<element name="ExpireDateTime" type="dateTime" minOccurs="0">
  <annotation>
    <documentation>Specifies the expiration date and time of
the trained parameters kept by the server. After the expiration time,
the trained parameters will be deleted, and the TrainedParameters ID
that a client got from the TrainClassifier operation will become invalid.
This element is included only for supervised classifiers.
</documentation>
  </annotation>
</element>
</sequence>
</complexType>
</element>
</schema>

```

10.3.2 Exceptions

When a web image classification server encounters an error while performing a TrainClassifier operation, it shall return an exception report message as specified in Subclause 7.4 of the OGC Web Services Common Implementation Specification. The allowed exception codes shall include those listed in Table 14. For each listed exceptionCode, the contents of the “locator” parameter value shall be as specified in the right column of Table 14.

Table 14 — Exception codes for TrainClassifier operation

exceptionCode value	Meaning of code	“locator” value
MissingTrainingImage	No input training image coverage data was available from the specified source for input source image ^a	Value of TrainingImage parameter
InvalidTrainingImageURI	Request contains invalid URI for input training image	Name of parameter
MissingTrainingLabelImage	No input training label image coverage data was available from the specified source for input source image ^a	Value of TrainingLabelImage parameter
InvalidTrainingLabelImageURI	Request contains invalid URI for input training image label	Name of parameter
InvalidFormat	The format name for the input image is not supported by the server	Name of the invalid format
InvalidClassifierParameters	Specified value for the ClassifiedParameters inconsistent with that required by the classifier.	Name of the invalid format
TBD	TBD	TBD
<p>A The specified source for input image could be the URL of the file containing an image data or an OWS operation request to get an image coverage. If remote access of input image was requested, and an exception report or other error response was returned by that resource, then the first ExceptionText element shall contain the received exception report or other error response.</p>		

10.4 Examples

10.4.1 TrainClassifier request

If a client wants to train the LAITS’s minimum distance classifier, LAITS_MINIDIS, using training data residing at <http://laits.gmu.edu/WICSInputCovaerage/TraingDataMODIS.dat> and <http://laits.gmu.edu/WICSInputCovaerage/TrainingLabelForMODIS.dat>, which are encoded in the classifier’s internal LAITS_MINIDIS_BINARY format, it can issue the following XML encoded TrainClassifier operation request:

```
<?xml version="1.0" encoding="UTF-8"?>
<TrainClassifier xmlns="http://www.opengis.net/wics"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics ../wics.xsd"
service="WICS" version="0.0.20">
  <TrainingImage
xlink:href="http://laits.gmu.edu/WICSInputCovaerage/TrainingDataMODIS.hdf" >
    <Format> LAITS_MINIDIS_BINARY </Format>
  </TrainingImage>
  <TrainingLabelImage
xlink:href="http://laits.gmu.edu/WICSInputCovaerage/TrainingLabelForMODIS.dat" >
    <Format> LAITS_MINIDIS_BINARY </Format>
  </TrainingLabelImage>
<!-- =====
```

The abstract ClassifierParameters element is substituted by the following actual classifier parameters for the LAITS's minimum distance supervised classifier.

```

===== -->
<Parameters_MiniDistance>
  <Distance> EuclideanDistance </Distance>
</Parameters_MiniDistance>
</TrainClassifier>

```

10.4.2 TrainClassifier response

The response to a request for train a classifier could be, if no exception occurs:

```

<?xml version="1.0" encoding="UTF-8"?>
<TrainedParameters xmlns="http://www.opengis.net/wics"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics ../wics.xsd"
service="WICS" version="0.0.20">
  <!-- ===== -->
    note that in the following, if a server does not allow clients to
    download trained parameters,
    the ID shall not be a URL but just an ID such as 048821f0-bff8-
    1026-9887-dbadcfbcc22c.dat
    ===== -->
  <ID> http://laits.gmu.edu/WICSOutput/048821f0-bff8-1026-9887-
  dbadcfbcc22c.dat </ID>
  <expireDateTime> 2005-08-30T00:00:00.0 </expireDateTime>
</TrainedParameters>

```

11 DescribeClassifier Operation (required)

11.1 Introduction

The DescribeClassifier operation allows WICS clients to request detailed information a WICS server supported. This information is important to most applications because there are wide spectra of classification algorithms, schema, and implementations. Even for well-known classification algorithms such as ISODATA, the parameters used to configure the classifier also vary from one implementation to another. Therefore, in most cases a WICS client may request DescribeClassifier operation before requesting classification. The WICS server shall respond to the DescribeClassifier request with an XML document with URL or URLs to parameter data file(s) for specified classifier(s).

11.2 Operation request

11.2.1 DescribeClassifier request parameters

One WICS server serves only one classifier and thus no parameters specific to the DescribeClassifier request operation are needed. Parameters common to the other operations which are listed in Table 7, i.e., Service, Request, and Version.

11.2.2 DescribeClassifier request KVP encoding (optional)

WICS servers shall implement HTTP GET transfer of the DescribeClassifier operation request, using KVP encoding. The KVP encoding of the DescribeClassifier operation request shall use the parameters specified in Table 15. There are no classifier specific parameters since each server only serves one classifier.

Table 15 — DescribeClassifier operation request URL parameters

Name and example ^a	Optionality and use	Definition and format
Service=WICS	Mandatory	Service type identifier
request=DescribeClassifier	Mandatory	Operation name
Version	Mandatory	Specification and schema version for this operation

11.2.3 DescribeClassifier request XML encoding (required)

WICS servers shall implement HTTP POST transfer of the DescribeClassifier operation request, using XML encoding only. The following schema fragment specifies the contents and structure of a DescribeClassifier operation request encoded in XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Originally edited by Yang Liu (yliu5@gmu.edu), Wenli Yang, George
Mason University on 06/11/2004, 07/21/2004, 07/26/2004 -->
<schema targetNamespace="http://www.opengis.net/wics"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://www.opengis.net/xlink"
xmlns:ows="http://www.opengis.net/ows"
xmlns:wics="http://www.opengis.net/wics" elementFormDefault="qualified"
version="0.0.0" xml:lang="en">
  <annotation>
    <appinfo>wics.xsd 2004-07-26</appinfo>
    <documentation>This XML Schema encodes the WICS GetCapabilities
operation request message. </documentation>
  </annotation>
  <!-- =====
includes and imports
===== -->
  <include schemaLocation="./wics.xsd"/>
  <import namespace="http://www.opengis.net/ows"
schemaLocation="http://schemas.opengis.net/ows/owsCommon.xsd"/>
  <!-- =====
DescribeClassifier request
===== -->
  <element name="DescribeClassifier">
    <annotation>
      <documentation>Request to a WICS to perform the
DescribeClassifier operation. This operation allows a client to retrieve
the description of the classifier. Because different classifiers shall
be implemented as different servers, no classifier name or
identification shall be needed in requesting a DescribeClassifier
operation. </documentation>
```



```

    </annotation>
  <complexType>
    <complexContent>
      <extension base="wics:RequestBaseType">
        <sequence/>
      </extension>
    </complexContent>
  </complexType>
</element>
</schema>

```

11.3 Operation response

11.3.1 DescribeClassifier normal response XML

The normal response to a DescribeClassifier request shall be the description of the classifier provided by the server. The information returned from the DescribeClassification operation shall include one or more of the following pieces of information: classifier name, label, metadata link(s), keywords, algorithm theoretical base document(s), supported data format(s), and default classifier configuration parameters.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Originally edited by Yang Liu (yliu5@gmu.edu), Wenli Yang, George
Mason University on 06/11/2004, 07/21/2004, 07/26/2004 -->
<schema targetNamespace="http://www.opengis.net/wics"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://www.opengis.net/xlink"
xmlns:ows="http://www.opengis.net/ows"
xmlns:wics="http://www.opengis.net/wics" elementFormDefault="qualified"
version="0.0.0" xml:lang="en">
  <annotation>
    <appinfo>wics.xsd 2004-07-26</appinfo>
    <documentation>This XML Schema encodes the WICS GetCapabilities
operation request message. </documentation>
  </annotation>
  <!-- =====
includes and imports
===== -->
  <include schemaLocation="./wics.xsd"/>
  <import namespace="http://www.opengis.net/ows"
schemaLocation="http://schemas.opengis.net/ows/owsCommon.xsd"/>
  <!-- =====
DescribeClassifier response
===== -->
  <element name="ClassifierDescription">
    <annotation>
      <documentation>Response to a valid DescribeClassifier operation
request sent to a WICS. It provides more detailed information about the
classifier than the Capabilities, such as classification algorithm,
keywords, and metadata. </documentation>
    </annotation>
    <complexType>
      <complexContent>
        <extension base="wics:ClassifierBriefingType">
          <sequence>

```

```

                <element name="ParameterSchemaSummary" type="string"
minOccurs="0">
                    <annotation>
                        <documentation>Human readable text description of
the classifier's parameter schema. This is a summary of the information
included in the ParameterSchemaURL element in the ClassifierBrief. It
is provided here for clients judge quickly if the full schema shall be
requested.
                        </documentation>
                    </annotation>
                </element>
                <element name="ClassifierATBDSummary" type="string"
minOccurs="0">
                    <annotation>
                        <documentation>Human readable text description of
the classifier, including such information as algorithm description,
main mathematical equation(s), and suitable applications. This element
is a summary of the classification algorithm theoretical document base.
It is provided here for clients to judge quickly if the service is
applicable to their applications or if further information shall be
sought from the ClassifierDocumentURL element. </documentation>
                    </annotation>
                </element>
                <element name="ClassifierDocumentURL" type="anyURI"
minOccurs="0" maxOccurs="unbounded">
                    <annotation>
                        <documentation>Reference to document that provides
more detailed information about the classifier than the
ClassifierATBDSummary, such as the full ATBD, research articles, and
classification schema in case of supervised classification.
                        </documentation>
                    </annotation>
                </element>
            </sequence>
            <attribute name="version" type="ows:VersionType"
use="optional">
                <annotation>
                    <documentation> Algorithm version implemented in this
server. A complicated classifier usually has several version upgrades
and this element is used to indicate which version of the classifier is
implemented. Note that this is NOT the WICS schema version
                    </documentation>
                </annotation>
            </attribute>
        </extension>
    </complexContent>
</complexType>
</element>
</schema>

```

11.3.2 Exceptions

No operation specific parameter is required for this operation and therefore no operation exception codes are needed (TBD).

11.4 Examples

11.4.1 DescribeClassifier request

The following XML encoded DescribeClassifier operation request can be issued:

```
<?xml version="1.0" encoding="UTF-8"?>
<DescribeClassifier xmlns="http://www.opengis.net/wics"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics ../wics.xsd"
service="WICS" version="0.0.20" />
```

11.4.2 DescribeClassifier response

The response to a request to the DescribeClassifier could be:

```
<?xml version="1.0" encoding="UTF-8"?>
<ClassifierDescription xmlns="http://www.opengis.net/wics"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wics ../wics.xsd"
service="WICS" version="0.0.20">
  <title>ISODATA_Unsupervised_Classifier</title>
  <abstract> This is the ISODATA unsupervised classifier server
implemented by LAITS/GMU</abstract>
  <class> Unsupervised </class>
  <ParameterSchemaURL>
    http://laits.gmu.edu/WICS/Schema_ISODATA.xml
  </ParameterSchemaURL>
  <sourceImageFormats>HDF-EOS GeoTiff</sourceImageFormats>
  <classifiedCoverageFormats>
    HDF-EOS GeoTiff
  </classifiedCoverageFormats>
  <KVPPParameter>
    <Keyword required="true"> NumberOfClusters </Keyword>
    <ValueType> int </ValueType>
    <Keyword required="false"> MaximumIterations </Keyword>
    <ValueType> int </ValueType>
  </KVPPParameter>
  <ParameterSchemaSummary>
    This classifier include a required parameter, NumberOfClusters,
specifying the maximum number of classes to be classified, and an
optional parameter, MaximumIterations, specifying the maximum number
of iterations a pixel shall be reclassified.
  </ParameterSchemaSummary>
  <ClassifierDocumentURL>
    http://laits.gmu.edu/WICSdocs/Algorithms/ISODATA.pdf
  </ClassifierDocumentURL>
</ClassifierDescription>
```

Annex A (informative)

Example of the Maximum likelihood supervised classifier schema

The parameters needed for a maximum likelihood classifier may include:

1. Number of classes (n)
2. Mean vector for each class (M_c)
3. Covariance for each class (V_c)
4. The *a priori* probability for each class (a_c), default is $1.0/n$.

In addition, number of image bands (or layers), K , is also implicitly included in the above parameters. M_c is a $1 \times K$ vector and V_c is $K \times K$ matrix.

For supervised classification, a client shall either 1) use the trainClassifier operation by supplying a training dataset (including the image data and class label data) to the server to obtain the above parameters (Note: number of classes is implied in the client provided class label data); or 2) use the describeClassifier operation to get/identify an appropriate classifier with a set of know parameters; or 3) directly request the getClassification operation, assume the client already knows and is satisfied with the classifier's parameters.

Upon receiving the image data and classifier parameters, the server shall calculate probability of an unknown pixel, X , belongs to a specific class c , p_c , using the following equation:

$$p_c = \ln(a_c) - (0.5 \ln(\det(V_c))) - (0.5(X - M_c)^T (V_c)^{-1} (X - M_c))$$

where $\det(V_c)$ is the determinant of V_c .

The pixel is classified to class k if and only if $p_k \geq p_i$ for all $i=1, \dots, n$.

Bibliography

- [1] OGC 02-012: OGC Classification Service
- [2] OGC 05-013: WCTS Implementation Specification, draft
- [3] OGC 03-065r6: Web Coverage Service, Version 1.0.0