Community Metadata ISO-19115 Adapter

Qiping Yan

1 Introduction

1.1 About the project

This project has been assigned to Qiping Yan by Dr. Sergiu Dascalu as part of the metadata exchange project which is part of the Track II of NSF EPSoR funded project "Collaborative Research: Cyberinfrastructure Developments for the Western Consortium of Idaho, Nevada, and New Mexico."

This project will be implemented as a web service that transforms two-way between ISO 19115-2 metadata and strongly typed programming objects such as classes.

1.2 Purpose of the this document

This document defines the software requirements and architectural design of the project and specifies the functional operations. This document is intended to be an input to the next phase of development of the code.

1.3 Overview of the document

This document describes the top-level structure of the code and function of the web service software. The software will be implemented in c# on a Microsoft Visual Studio platform.

1.4 Definitions, acronyms and abbreviations

ISO 19115	It defines the schema required for describing geographic information and services
Web service	A method of communication between two electronic devices over the Web
Class	A construct that is used to create instances of itself, class objects, instance objects or simply objects.
OO	Object Oriented (Programming)
URD	User Requirement Document
SRD	Software Requirement Document
oXygen	A XML editor/authorization software package

1.5 References

1 Web Services, http://www.w3.org/TR/ws-arch/

3 ISO 19115, http://www.iso.org/iso/catalogue_detail?csnumber=26020

3 oXygen, www.oxygenxml.com

2 Logical Model Description

In the diagram (Figure 1 on page 3), a user initiates a web service request with proper parameters. The web service then validates the request, record user info (optional), process the request, and return proper data. The request parameters include: Geometry information, sampling interval, and time period, return data format. The return data can be in ISO 19115 metadata format or a specific structure.

2.1 Software Work Flow

The software work flow diagram is shown in Figure 2 on page 4.

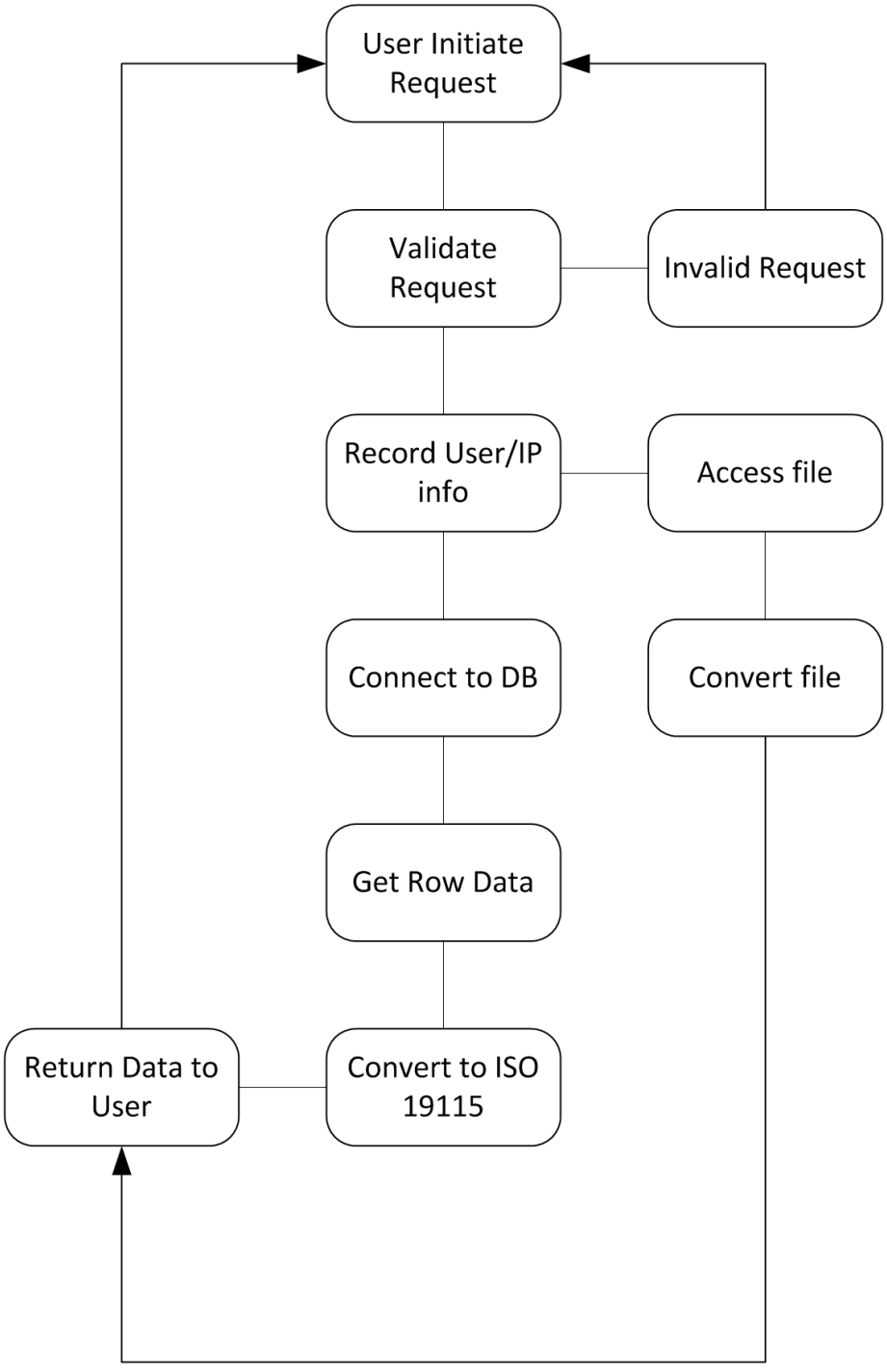

Figure 1: The logical model diagram

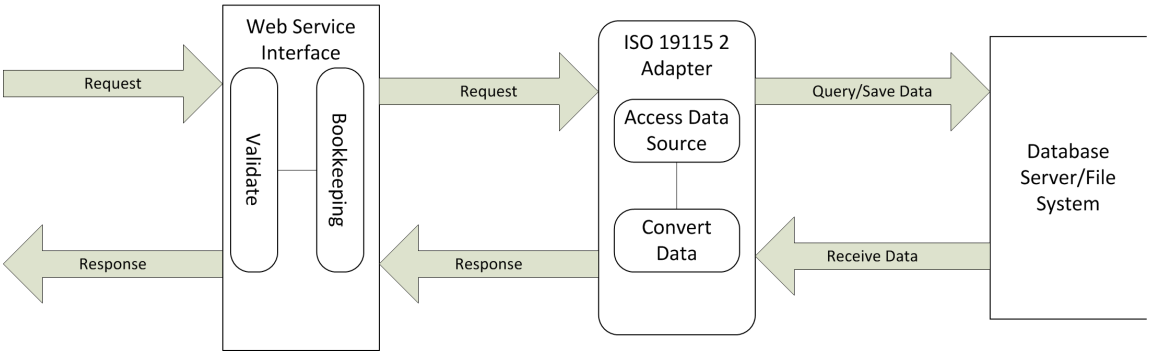

Figure 2: The software work flow diagram

3 Requirements

3.1 Functional requirements

3.1.1 User input interface (SR 1)

SR ID	Description
SR 1.1	A web form for user to enter query parameters
SR 1.2	Geometry information
SR 1.3	Start date/time
SR 1.4	End date/time
SR 1.5	Time interval
SR 1.6	Translation direction
SR 1.7	Upload file
SR 1.8	User/developer reference ID

3.1.2 User Output interface (SR 2)

SR ID	Description
SR 2.1	19115 metadata file saved or output to a mapping interface to visually display data.
SR 2.2	Query status code/error code

3.1.3 Web Service (SR 3)

SR ID	Description
SR 3.1	Validate query parameters
SR 3.2	Response with ISO 19115 metadata
SR 3.3	Response with Error code
SR 3.4	Pass validated query to ISO 19115 Adapter for metadata
SR 3.5	Record user/IP information

3.1.4 ISO 19115 Adapter (SR 4)

SR ID	Description
SR 4.1	Parse input
SR 4.2	Get data from data storage (DB) or file from file system
SR 4.3	Construct ISO 19115 file (translate)
SR 4.4	Serve metadata to web service layer
SR 4.5	Save data to DB or file

3.2 Interface requirements (SR 5)

The user will use a web form to easily construct the request parameters. The web form will provide the necessary data for the web service and 19115 adapter to properly query data and form metadata (translate). For security purpose, a user must provide a valid user id or developer reference number with each service request.

3.3 Operational requirements (SR 6)

The web service and ISO 19115 adapter shall be implemented on a IIS web server with Microsoft .NET framework and MS SQL server. Users can send service request from any platform with an Internet connection.

4 System Design

The diagram (Figure 3 on page 7) shows the class category diagram for the Community ISO 19115 Adapter. The unshaded boxes indicate the new class category required modules to support the adapter/web service functions.

5 Component Description

5.1 main

5.1.1 Type

Main program.

5.1.2 Functions

The main program instantiate the ISO 19115 Adapter, establish database connections.

5.1.3 Interface

Controls the overall execution of the software.

5.1.4 Dependencies

It is the first process executed.

5.1.5 Data

Data are held in Adapter class instance.

5.1.6 Resource

None.

5.1.7 Software requirements met

None.

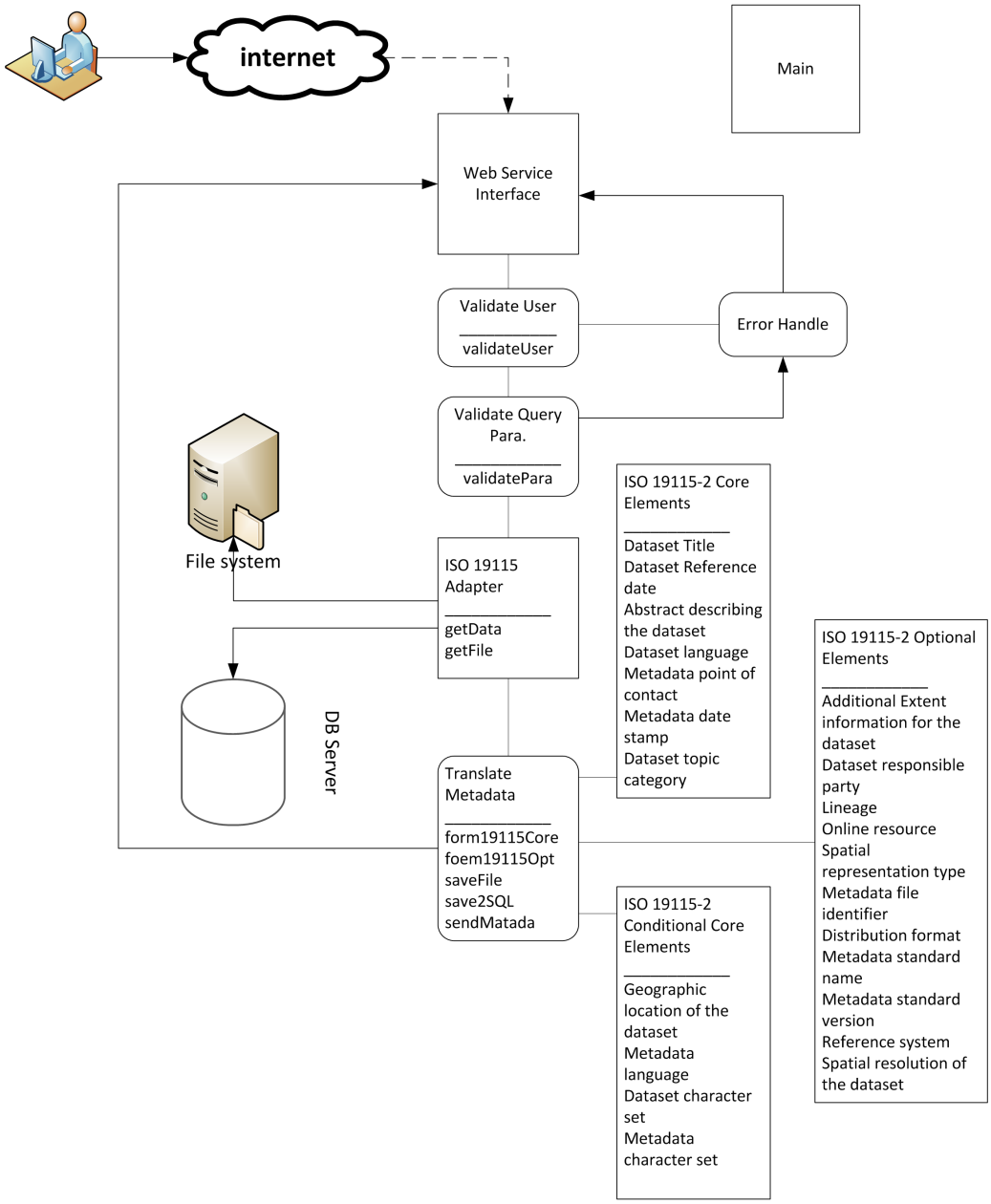

Figure 3: The logical model diagram

5.2 ISO 19115 Adapter

5.2.1 Type

Class

5.2.2 Functions

The ISO 19115 Adapter class is the main class. it stores the source data and translated metadata. Methods include:

- getData
- getFile
- form19115Core
- form19115Opt
- saveFile
- save2SQL
- sendMetadata

5.2.3 Interface

It is initiated by main process.

5.2.4 Dependencies

- ISO 19115 Core Elements
- ISO 19115 Conditional Core Elements
- ISO 19115 Optional Elements

5.2.5 Data

Source file/data from web service requests or from database.

5.2.6 Resource

None.

5.2.7 Software requirements met

[SR 4.1, SR 4.2. SR 4.3. SR 4.4, SR 4.5]

5.3 Translate Metadata

5.3.1 Type

Class method.

5.3.2 Functions

Called by ISO 19115 Adapter class. Functions include:

- form19115Core
- form19115Opt
- saveFile
- save2SQL
- sendMetadata

5.3.3 Interface

Called by Adapter instance.

5.3.4 Dependencies

- ISO 19115 Core Elements
- ISO 19115 Conditional Core Elements
- ISO 19115 Optional Elements

5.3.5 Data

Data are held in Adapter class instance.

5.3.6 Resource

None

5.3.7 Software requirements met

[SR 4.3, SR 4.5]

5.4 ISO 19115 Core Elements

5.4.1 Type

Class

5.4.2 Functions

Called by Adapter class methods.

5.4.3 Interface

Serve as a template for Adapter class methods.

5.4.4 Dependencies

None

5.4.5 Data

None

5.4.6 Resource

Contains the list of core elements for 19115 including data format information.

5.4.7 Software requirements met

[SR 4.3]

5.5 ISO 19115 Conditional Core Elements

5.5.1 Type

Class

5.5.2 Functions

Called by Adapter class methods.

5.5.3 Interface

Serve as a template for Adapter class methods.

5.5.4 Dependencies

None

5.5.5 Data

None

5.5.6 Resource

Contains the list of conditional core elements for 19115 including data format information.

5.5.7 Software requirements met

[SR 4.3]

5.6 ISO 19115 Optional Elements

5.6.1 Type

Class

5.6.2 Functions

Called by Adapter class methods.

5.6.3 Interface

Serve as a template for Adapter class methods.

5.6.4 Dependencies

None

5.6.5 Data

None

5.6.6 Resource

Contains the list of optional elements for 19115 including data format information.

5.6.7 Software requirements met

[SR 4.3]

6 User requirements V.S. Software requirements Traceability Matrix

Table 1 in page 12 show a cross-reference of the user requirements and software requirements.

U\S	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	2.1	2.2	3.1	3.2	3.3	3.4	3.5
1.1															
1.2															
1.3															
1.4															
2.1															
2.2															

Table 1: Traceability matrix for user and software requirements