Open Geospatial Consortium Inc.

Date: 2005-02-14

Reference number of this OGC Project Document: **OGC 04-040**

Version: **1.0**

Category: OpenGIS® Discussion Paper

Editor: William Lalonde

Style Management Services for Emergency Mapping Symbology

Warning

This document is not an OGC Standard or Specification. This document presents a discussion of technology issues considered in an Interoperability Initiative of the OGC Interoperability Program. The content of this document is presented to create discussion in the geospatial information industry on this topic; the content of this document is not to be considered an adopted specification of any kind. This document does not represent the official position of the OGC nor of the OGC Technical Committee. It is subject to change without notice and may not be referred to as an OGC Standard or Specification. However, the discussions in this document could very well lead to the definition of an OGC Implementation Specification.

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: OpenGIS® Discussion Paper
Document stage: Final
Document language: English

2/17/2005

Special Notice:

Bill Lalonde, a valued participant in OGC activities and one of the recipients of the 2003 Gardels Award, passed away on December 30, 2004, after a long illness. Bill was CubeWerx's director of product development. In this role, he was instrumental in the development and deployment of OGC based CubeWerx technology. He was the editor of the OpenGIS Styled Layer Descriptor Specification and was a driving force in several Interoperability Initiatives. In these initiatives, Bill was the consummate team player working to insure the successful completion of the initiative. His practical, results-oriented approach, hard work, and diplomacy contributed immensely to the success of numerous OGC initiatives.

Contents

i. Preface

This Interoperability Program Report (IPR) document was developed as part of the Emergency Mapping Symbology testbed, Phase 1 (EMS-1). This version of the IPR reflects the status of the specification as of the change history.

ii. Submitting organizations

The following organizations submitted this document to the Open GIS Consortium Inc.:

- Compusult Inc.
- CubeWerx Inc.
- Galdos Systems Inc.

iii. Document Contributor Contact Points

All questions regarding this submission should be directed to the editor or the submitters:

COMPANY	CONTACT	ADDRESS	PHONE/FAX	EMAIL
CubeWerx Inc.	William Lalonde	200 rue Montcalm Suite R-13 Hull, Quebec Canada J8Y 3B5	Phone: 819-771-8303 Ext 206 Fax: 819-771-8388	wlalonde@cubewerx.com
CubeWerx Inc.	Craig Bruce	200 rue Montcalm Suite R-13 Hull, Quebec Canada	Phone: 819-771-8303 Ext 205 Fax: 819-771-8388	csbruce@cubewerx.com

		J8Y 3B5	819-771-8388	
Compusult Inc.	Larry Bouzane	Compusult Limited P.O. Box 1000 40 Banister Street Mount Pearl, Newfoundland, Canada A1N 3C9	Toll Free: 888-307-7707 Tel: 709-745-7914 Fax: 709-745-7927	larry@compusult.nf.ca
Galdos Systems Inc.	Milan Trninic	Suite 200, 1155 West Pender Street Vancouver, BC V6E 2P4 Canada	Tel: +1 (604) 484-2750 Fax: +1 (604) 484-2755	mtrninic@galdosinc.com
Image Matters LLC	John Davidson	214 S. King St Leesburg, VA 20175	Tel: 703-669-5510	johnd@imagemattersllc.com

iv. Revision history

Date	Release	Author	Paragraph modified	Description
1/2/04	0.1	John Davidson	Initial draft.	
3/22/04	0.2	Bill Lalonde	Annexes.	Updated with new Annexes and Annex content
6/18/04	1.0	John Davidson	All.	Overall scrub; updated Annexes

v. Changes to the OpenGIS® Abstract Specification

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

vi. Future Work

The contents of this specification should maintain alignment with the following approved OGC Documents and may result in future changes to these documents:

- 02-070 Styled Layer Descriptor (SLD V1.0.0) Specification (or its successor document). Note: this document is dependent on the successor to SLD V1.0.0 as specified in the pending SLD 1.0 Change Proposal (document 04-009).

- 02-058 Catalog Service (V1.1.0) Specification (or its successor document). Note: this document is dependent on the pending CS V2.0 (document 04-021r2).

Foreword

The Open GIS Consortium Inc. shall not be held responsible for identifying any or all such patent rights.

Introduction

This document describes the proposed system design for the OGC Style Management Service (SMS).

The SMS must manage distinct objects that represent styles and symbols and provide the means to discover, query, insert, update, and delete these objects.

Styles provide the mapping from feature types and feature properties and constraints to parameterized Symbols used in drawing maps. Symbols are bundles of predefined graphical parameters and predefined fixed graphics.

Style Management Services for Emergency Mapping Symbology

1 Scope

The Emergency Mapping Symbology, Phase 1 (EMS-1) initiative attempts to fulfill part of the modernization goals of its sponsors – to increase capabilities to leverage existing market driven, Standards–based Commercial Off-The-Shelf (SCOTS) solutions for fulfilling their missions. Part of the mission is to ensure that commercial industry addresses interoperable technology requirements. Like all OGC Interoperability Program initiatives, EMS-1 exists to facilitate OGC members and industry vendors to develop, test and validate interface specifications leading to commercial products suitable for use by the sponsors, their customers and the broader geospatial community, including international, federal, state and local agencies involved in emergency management and response activities.

The enabling technologies for EMS-1 are Style Management Services (SMS) that were first described and demonstrated in the Open Web Services Phase 1.2 (OWS-1.2) test bed initiative. While SMS can be thought of as a type of service, it is perhaps more practically considered a logical composition of design-patterns, service interfaces and encodings. As such, SMS defines an architecture for enabling flexibile, scalable and interoperable management of symbols and styles in support of cartographic portrayal processes. The key elements of the SMS are:

- **Symbol:** a set of predefined graphical representation parameters and/or fixed graphic icons; the instructions for how vector graphics are to be represented (e.g., geometry/graphic, fill, color, stroke, font, orientation, size, opacity, etc.); the instructions for how raster graphics are to be represented (e.g., opacity, R/G/B channel selection, color map, shaded relief, contrast enhancements, etc.). HSWG[1]

[1] Symbols for Emergency Management and First Responder communities – a map symbology set under development by the FGDC Homeland Security Work Group (HSWG). See: http://www.fgdc.gov/publications/homeland.html and http://www.fgdc.gov/HSWG/

and GeoSym[2] are specifications defining sets of symbols for cartographically portraying features**.**

- **Style:** maps feature types, properties and constraints to one or more parameterized symbols; also the properties and rules describing how features are drawn during a graphical rendering process (e.g., order of layers, associate symbol type X with feature type Y, or how to apply one or more symbols to drawing a road at its centerline, etc.); **Styled Layer Descriptor (SLD)** is the XML language for defining rules for styling features and coverages.

- **Feature**: objects/phenomena on the Earth that are normally represented as graphical entities on a map (e.g., a house, political boundary, lake).

- **Feature Type** – identifies the semantic, structure (properties and property types) and behavior of Feature instances and can be defined with a GML Application Schema.

- **Coverage** – a feature that associates positions within a bounded space (its spatio-temporal domain) to feature attribute values (its range) (e.g., a digital terrain model or image)

- **Registry Information Model (RIM):** the information model that provides the means to package, publish and discover feature, style and symbol metadata by Catalog Service- Web Profile (formerly Web Registry Service).

- **Web Map Service (WMS):** a service that uses SLD to generate cartographic portrayals of features on the Web.

- **Catalog Service for the Web (CSW):** the means to record and store instances of service, feature type, symbol and style metadata for discovery and access on the Web.

Note: The Catalog Services Revision Work Group of the OGC Technical Committee has released a new revision of the Catalog Services Implementation Specification 2.0 (OGC Document 04-021r3) that will incorporate, as a profile, a "stateless" Web interface that is derived from earlier work on Web Registry Service (WRS) including implementations developed and demonstrated in previous Interoperability Program initiatives. Final approval of the Catalog Services 2.0 specification is pending results of an an electronic vote by the OGC membership that is presently underway.

---

[2] Geospatial Symbols (GeoSym) for Digital Displays. The map symbology set defined by NIMA to portray Vector Product Format (VPF) data. See: http://www.nima.mil/ast/fm/acq/mil89045.pdf

Another key specification used and implemented for the EMS-1 project is the "Catalog Service – Web Profile" (OGC Project Document 03-094, v0.8.3). The terms Registry, Catalog, WRS, CSW and RIM are used interchangeably throughout this document to refer to the body of work represented by these two documents (04-021r3 and 03-094).

The architecture presented herein is intended to be a starting point. This document should be considered a draft and extensions and modifications to this architecture will be generated from lessons learned through the EMS-1 and other OGC initiatives.

1.1 Critical Infrastructure Collaborative Environment

The Critical Infrastructure Collaborative Environment (CICE) is a prototype open distributed geoprocessing environment based on open architectures enabling publishing, discovery, and use of geospatial information for Critical Infrastructure Protection and other activities. CICE leverages OGC Web Services (OWS) to enable:

- The publication of the availability of critical infrastructure services and data
- The registration and categorization of published service and data providers
- The discovery and use of needed critical infrastructure services and data

CICE will also establish a leave-behind capability of services, data, applications, partners, relationships, use cases, test cases, and scenarios (see CICE Architecture references listed in Appendix A).

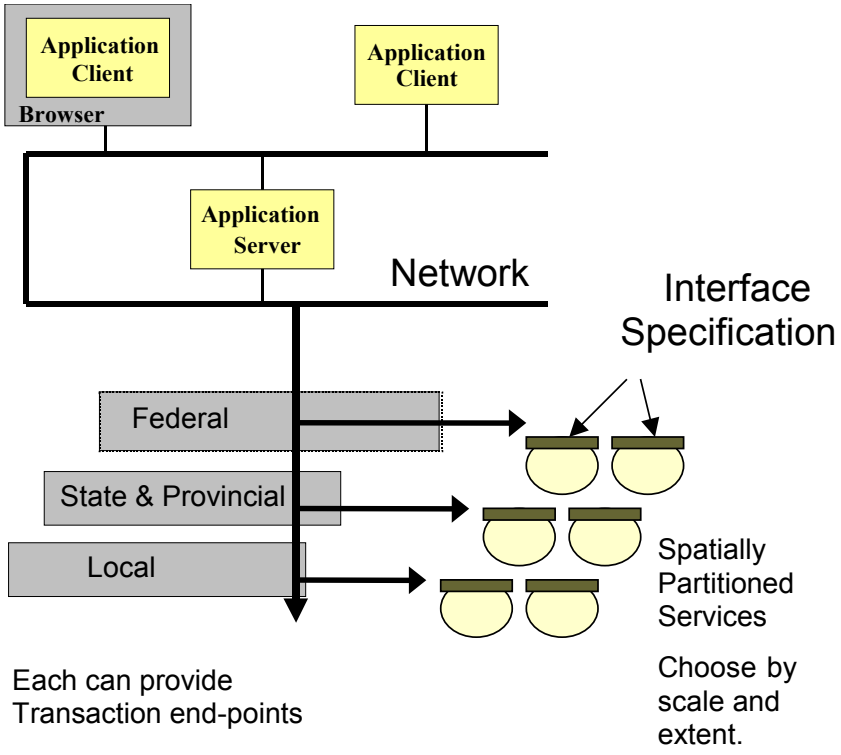


Figure 1 - Critical Infrastructure Collaborative Environment (CICE)

1.2 EMS-1 Reference Model

The structure of this Architecture is loosely based on the Reference Model for Open Distributed Processing (RM-ODP). The five views into the EMS-1 Architecture are described in further detail in Sections 6 through 10 of this annex.

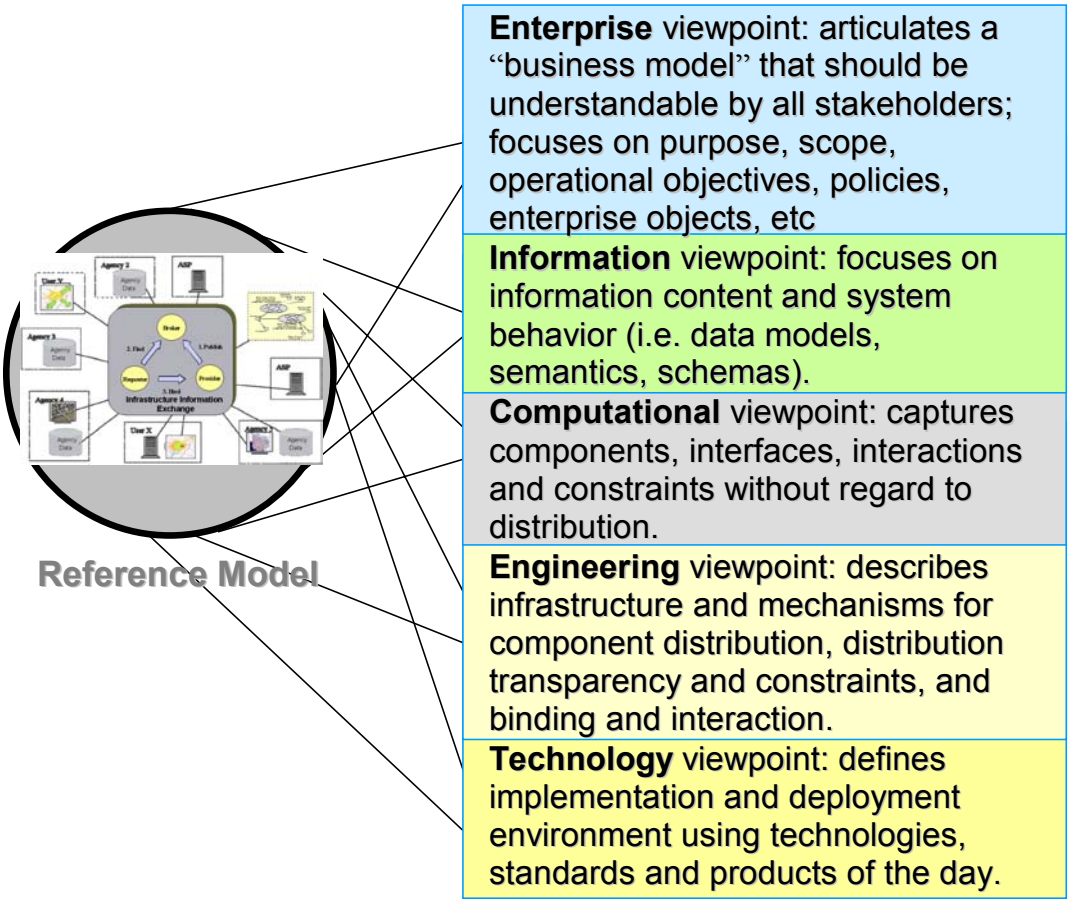


Figure 2. EMS-1 Reference Model

- Section 6 (Enterprise view) describes the Enterprise Architecture for the EMS-1. This architecture describes the high-level system concept and presents representative use cases.

- Section 7 (Information View) describes the Information Architecture for the EMS-1. This architecture describes the basic information building blocks of EMS-1.

- Section 8 (Computational View) describes the Computational Architecture for the EMS-1. This architecture describes the basic service building blocks of EMS-1.

- Section 9 (Engineering View) describes the Engineering architecture for the EMS-1. This architecture describes the core components that are to be deployed and the infrastructure to integrate them into a single environment.

- Section 10 (Technology View) describes the target deployment environment for EMS-1 components in terms of technologies, standards and products.

- Annex A details SMS usage-scenarios.

- Annex B details the revised normative schema for SLD.

- Annex C details Symbol and Style metadata schema and provides worked examples.

- Annex D lists a sample GML Application Schema and worked examples of SLD, Style and Symbol documents.

- Annex E describes the Web-Mapping Module for CSW

- Annex F provides worked-examples of CSW requests and responses for EMS resources.

- Annex G describes SMS Client interactions.

2 Conformance

Not required for an IP IPR, DIPR, or Discussion Paper.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of OGC 02-046. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of OGC 02-046 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

CGI, *The Common Gateway Interface,* National Center for Supercomputing Applications, http://hoohoo.ncsa.uiuc.edu/cgi/

IETF RFC 2045 (November 1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodie*s, Freed, N. and Borenstein N., eds.,

http://www.ietf.org/rfc/rfc2045.txt

IETF RFC 2616 (June 1999), *Hypertext Transfer Protocol – HTTP/1*.1, Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T., eds., http://www.ietf.org/rfc/rfc2616.txt

IETF RFC 2396 (August 1998), *Uniform Resource Identifiers (URI): Generic Syntax*, Berners-Lee, T., Fielding, N., and Masinter, L., eds.,

ISO/DIS 19109, *Geographic information – Rules for Application Schema*, ISO/TC 211, http://www.isotc211.org/protdoc/DIS/ISO_DIS_19109_(E).pdf

ISO/DIS 19110, *Geographic information – Methodology for Feature Cataloguing*, ISO/TC 211, http://www.isotc211.org/protdoc/DIS/ISO_DIS_19110_(E).pdf

OGC 02-112r2 , *The OpenGIS Abstract Specification Topic 12: OpenGIS Service Architecture (Version 4.3*), Percivall, G. (ed.), January 2002 http://www.opengis.org/techno/abstract/02-112.pdf

OGC 01-068r2 Adopted Implementation Specification: Web Map Server version 1.1.1, February 2002.

XML 1.0 (October 2000), *Extensible Markup Language (XML) 1.0 (2nd edition*), World Wide Web Consortium Recommendation, Bray, T., Paoli, J., Sperberg-McQueen, C.M., and Maler, E., eds., <http://www.w3.org/TR/2000/REC-xml>
http://www.ietf.org/rfc/rfc2396.txt

4 Terms and definitions

4.1
application schema
Conceptual schema for data required by one or more applications [ISO/DIS 19109]

4.2
classification scheme
A taxonomy used to classify phenomena… used to characterize catalogue content [ebRIM 9.1]

4.3
client
Software component that can invoke an operation from a server

4.4
feature
Abstraction of real world phenomena; objects or phenomena on the Earth that are normally represented as graphical entities on a map (e.g., a house, political boundary, lake). NOTE: A feature may occur as a type or an instance. Feature type or feature instance should be used when only one is meant. [ISO/DIS 19109]

4.4
feature attribute
feature property
A characteristic of a feature. NOTE 1: A feature attribute may occur as a type or an instance. Feature attribute type or feature attribute instance is used when only one is meant. NOTE 2: A feature attribute type has a name, a data type and a domain associated to it. A feature attribute instance has an attribute value taken from the domain of the feature attribute type. [ISO/DIS 19109]

4.4
feature type
Class of real world phenomena with common properties; identifies the semantic, structure (properties and property types) and behavior of Feature instances and can be defined with an Application Schema. EXAMPLE: The phenomenon 'Eiffel Tower' may be classified with other similar phenomena into a feature type 'tower'. [ISO/DIS 19110]

4.3
interface
Named set of **operations** that characterize the behavior of an entity [18]

4.2
operation
Specification of a transformation or query that an object may be called to execute [18]

4.5
service
Distinct part of the functionality that is provided by an entity through **interfaces** [18]

4.6
service instance
server
Actual implementation of a **service**

4.6
style
Styles provide the mapping from feature types and feature properties and constraints to parameterized *symbols* used in drawing maps; the properties and rules describing how features are drawn during a graphical rendering process (e.g., order of layers, associate symbol type X with feature type Y, or how to apply one or more symbols to drawing a road at its centerline, etc.); **Styled Layer Descriptor (SLD)** is the XML language for defining rules for styling features and coverages.

4.7
symbol
Symbols are bundles of predefined graphical parameters and predefined fixed graphic icons ("images" or strokes); the instructions for how vector graphics are to be represented (e.g., geometry/graphic, fill, color, stroke, font, orientation, size, opacity, etc.); the instructions for how raster graphics are to be represented (e.g., opacity, R/G/B channel selection, color map, shaded relief, contrast enhancements, etc.). **HSWG**[3] and **GeoSym**[4] are specifications defining classification schemes and instances of symbols for cartographically portraying features**.**

5 Conventions

5.1 Symbols (and abbreviated terms)

CGI - Common Gateway Interface
DCP - Distributed Computing Platform
DTD - Document Type Definition
EPSG - European Petroleum Survey Group
GIF - Graphics Interchange Format
GIS - Geographic Information System
GML - Geography Markup Language
HTTP - Hypertext Transfer Protocol
IETF - Internet Engineering Task Force
JPEG - Joint Photographic Experts Group
MIME - Multipurpose Internet Mail Extensions
OGC - Open GIS Consortium
OWS - OGC Web Service
PNG - Portable Network Graphics
RFC - Request for Comments

---

[3] Symbols for Emergency Management and First Responder communities – a map symbology set under development by the FGDC Homeland Security Work Group (HSWG). See: http://www.fgdc.gov/publications/homeland.html and http://www.fgdc.gov/HSWG/

[4] Geospatial Symbols (GeoSym) for Digital Displays. The map symbology set defined by NIMA to portray Vector Product Format (VPF) data. See: http://www.nima.mil/ast/fm/acq/mil89045.pdf

SLD - Styled Layer Descriptor
SVG - Scalable Vector Graphics
URL - Uniform Resource Locator
WebCGM - Web Computer Graphics Metafile
WCS - Web Coverage Service
WFS - Web Feature Service
WMS - Web Map Service
XML - Extensible Markup Language

5.2 UML Notation

The diagrams that appear in this document are presented using the Unified Modeling Language (UML) static structure diagram. The UML notations used in this document are described in the diagram below.

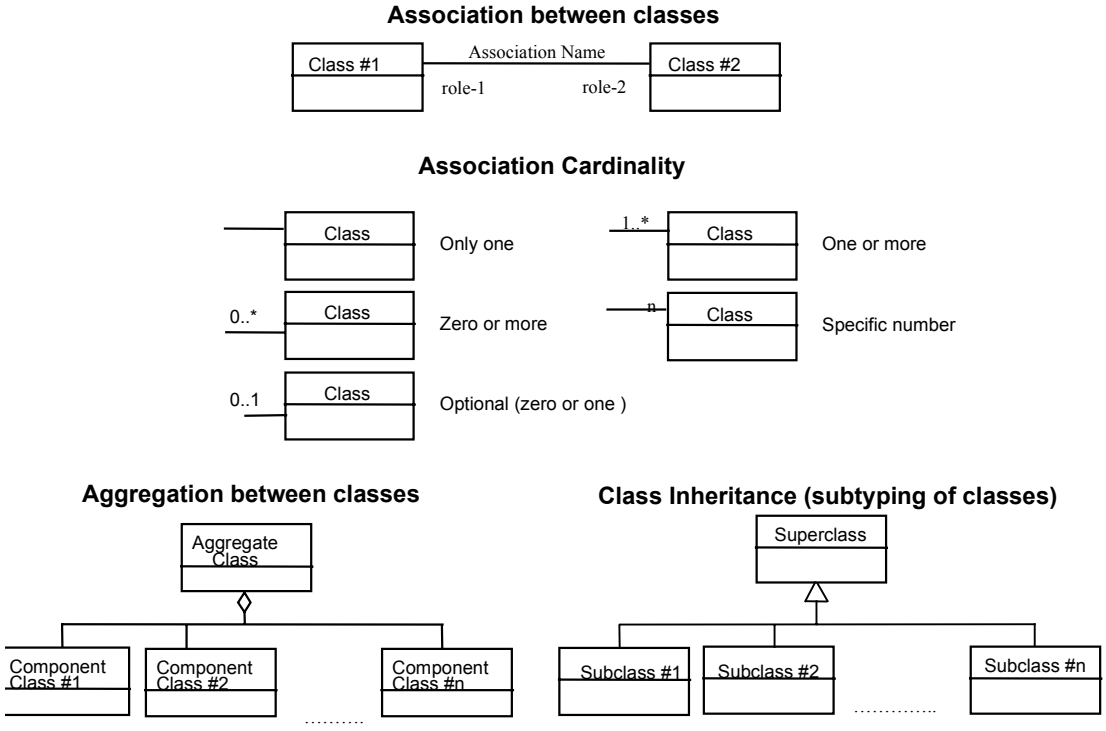


Figure 3 — UML notation

In this diagram, the following three stereotypes of UML classes are used:

<<Interface>> A definition of a set of operations that is supported by objects having this interface. An Interface class cannot contain any attributes.

<<DataType>> A descriptor of a set of values that lack identity (independent existence and the possibility of side effects). A DataType is a class with no operations whose primary purpose is to hold the information.

<<CodeList>> is a flexible enumeration that uses string values for expressing a list of potential values.

6 Enterprise Viewpoint

The EMS-1 Enterprise Architecture captures the capabilities that must be present in support of emergency management and response operations. The capabilities identified in the enterprise view provide the requirements to be met by the EMS-1 services and information architecture. Success of the EMS-1 is measured by the accuracy of the enterprise view and how well the information and systems architectures support that view.

The Enterprise Architecture is defined by a high-level system concept and use-cases. The system concept illustrates the operational setting, major system components, and major interfaces. The use cases provide descriptions of the behavior of the system from the point of view of users.

Figure 4 depicts the basic components of the EMS-1 concept. Data sources for emergency management are provisioned as Web-accessible services by many different, geographically distributed organizations. The organizations providing data for Emergency Management and Response are "vertically" and "horizontally" integrated across public and private sector organizations at international, national, regional, state and local levels. In the EMS-1 architecture, these data sources are published and accessible via OpenGIS® Web Service (OWS) interoperability specifications such as GML, SLD, WMS, WFS, CSW described in this and other supporting documents.

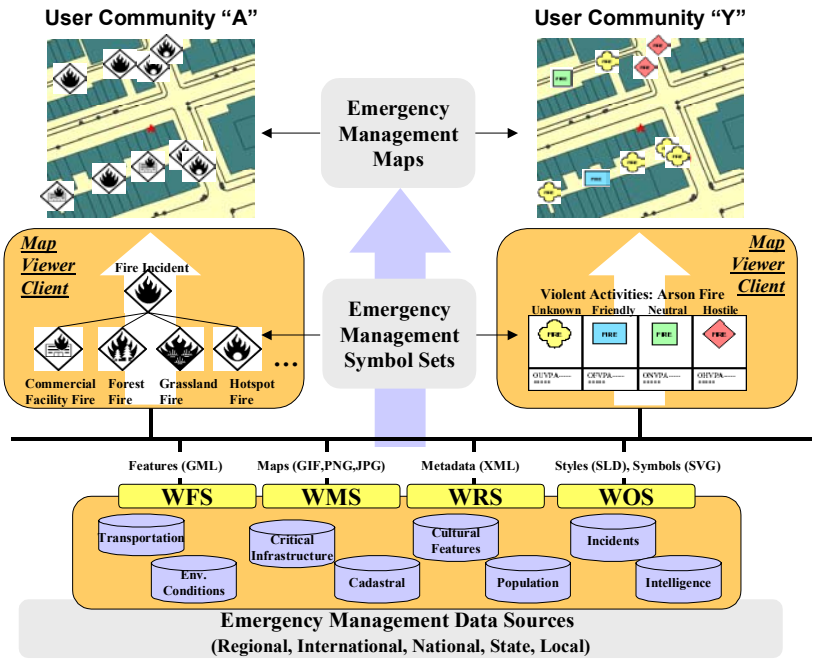


Figure 4 – EMS-1 Operational Context

In Figure 4, people (e.g., first responders and emergency management personnel) use "Map Viewer Clients" to dynamically generate and view maps of emergency incidents, critical infrastructure and other related information. The users of the system may, however, represent a wide range of organizations and "information communities" engaged in different emergency management activities including support for emergency detection, preparedness, prevention, protection, response and recovery. While all communities may use the same sources and standards for accessing geographic data, each user-community may have specialized rules for cartographically representing emergency-related information on maps. In many cases, the disparate user-communities may mandate use of styling rules and symbol sets that are designed specifically for generating map products tuned for their intended use in supporting the specialized mission of their users. Thus, in Figure 4, the users in "User-Community Y" (e.g., first responders) may be accustomed to viewing maps with incident symbols presented one way (presumably meaningful to their mission) and users in "User-Community A" (e.g., incident recovery planners) another way.

The challenge presented above can be mitigated somewhat to the degree that cartographic styling rules and symbol sets can be standardized and adopted across the user-communities. Nevertheless, there will always be information communities with different (possibly contradictory) requirements for map portrayal that result in the definition and use of different styling rules and symbol sets for map production. The EMS-1 Architecture is intended to enable interoperability, flexibility and re-use through a

common framework of service interfaces and encodings for styles, symbols and associated metadata.

6.1 Problems and Objectives

Table 1 – EMS-1 Problem & Solution Summary

Problem Statement	Description of solution and solution inputs, outputs, constraints, variables and alternatives
1. Lack of access to relevant/appropriate/current geospatial information products. A standard solution is sought.	A system that provides transparent information discovery and access of information based on stated information needs/requirements. A system that enables data to be transparently (from user perspective) discovered and then accessed (via standard service interfaces and data encodings) regardless of their physical location on networks (local or wide-area networks, private or open networks). Portable and interoperable across different service and repository implementations (i.e., SCOTS-based) Scalable
2. Lack of interoperability of geodata (formats and semantics), symbols and services for portrayal. A standard solution is sought.	• A system that encodes disparate but commonly-used geospatial "data products" according to community/application/product-specific schemas using a common markup language (e.g., GML, SLD) • A system that defines and uses common mechanisms for defining and describing basic information types, messages, interfaces and services. • A system that defines and uses common types, messages and interfaces and ways for describing services to enable service connectivity, composition and interoperability. • A system that can use metadata (including typing information) to relate or associate symbols and styles to suit stated information needs/requirements of users.
3. Inability to easily/automatically generate customized information products fusing geospatial with other sources of information about an incident. A standard solution is sought.	• A system that enables specialized products (views) of the same geospatial information (models) to be produced, defined and/or chosen and applied. • A system that can present other (non-geospatial) data along with geospatial information in a meaningful way. • A system that supports customized packaging and delivery of information that is "tuned" for users

Table 2- EMS-1 Objectives

Objectives
1. Stabilize, test, package and demonstrate OGC "Technical Baseline". o Build on OWS1.2 initiative (e.g., SMS symbol/style management, registries, RIM, GML3, Web Feature Service, Web Object Service, etc.) o Build on GOS (TP & PI) initiatives (e.g., data modeling, portal infrastructure, etc) o Build on CIPI (1&2) initiatives (e.g., security, information models, alert notification, portal nodes, schema translation, GML3 schema profiles, WFS transactions, etc.)
2. Extend OGC technical baseline with tested implementations supporting automation of cartographically styled portrayals of geospatial data (features and imagery) for the emergency mapping problem domain. Provide capability to configure map services with standard style and symbol types for use with datasets for critical infrastructure and emergency management. Support ability to style geospatial information in a consistent way, regardless of the source of data, according to specified "information product" rules. Support standards-based portrayal of vector, image and terrain data. Provide ability for client applications to choose standard views/presentations of geospatial information.
3. Extend and demonstrate capabilities for enabling "interoperable information communities". Provide service infrastructure for publication and discovery of standardized application schema and feature catalogs. Develop a standard classification of feature types for Home Land Security, Critical Infrastructure and Emergency Management domains and make this available on the Web. Develop a standard classification of style and symbol types that apply to feature types for HLS, CI, and EM domains and make this available on the Web. Implement portrayal clients and services that use these published resources (classifications and services).

6.2 Functional Requirements

The requirements listed are adapted from requirements originally developed during the OWS1.2 testbed initiative and are included here to further refine the scope and basic functionality envisioned of the EMS-1 architecture.

General Requirements

Must allow symbols and styles to be dynamically applied (bound) to geodata independent of its source and type, according to:

1. Feature type (i.e., the semantic and structure of a geographic feature)

2. Metadata for symbols, styles and feature datasets

3. Must allow applications (application clients or application services) to control how geodata will be visually presented.

4. Must allow client applications to create and store styles and symbols for use (possibly by other users and applications) for cartographic portrayal of geodata.

5. Must allow metadata to be created for describing symbol, style and feature types (and collections of these) and associated with instances of these.

6. Must allow symbol, style and feature types and their instances to be discovered and accessed for use in cartographic portrayal of geospatial information.

7. Must allow a style instance definition to be retrieved by name and identifier.

8. Must allow a symbol instance to be retrieved by name and identifier.

9. Must allow users to discover well-formed style names that are available within a style repository.

10. Must allow users to discover well-formed symbol names that are available within a symbol repository.

11. Must allow authorized users to insert new style definitions into a style repository. When a style is inserted in the style repository, the style must be validated and metadata describing the new style definition must be classified and associated with the inserted style.

12. Must allow authorized users to insert new symbols into a symbol repository. When a symbol is inserted in the symbol repository, metadata describing the new symbol must be classified and associated with the inserted symbol.

13. Should allow users to delete a style definition from the style repository.

14. Should allow users to delete a symbol from the symbol repository.

15. Store style definitions as ”feature styles” (SLD <FeatureTypeStyles> elements). This style information must be described with metadata and published and classified in the style registry.

Style and Symbol Discovery Requirements

16. The system must allow users to discover styles referenced by a specified feature type. In this discovery, a user requests style names, giving the feature type name as a search parameter. The system must respond to the request with a list of well-formed style names that apply styles for the given feature type.

17. The system must allow users to discover styles based on one or more properties defined within the style metadata and a style classification.

18. The system must allow users to discover symbols based on one or more properties defined within the symbol metadata and a symbol classification.

Style and Symbol Metadata Requirements

19. Style definitions must be annotated with metadata that describe properties of the style. These properties shall include application domain (e.g. “weather forecasting” or “command and control”) and organization domain (e.g. “NIMA” or “USGS”) identifiers.

20. Symbols definitions must be annotated with metadata that describe properties of the symbol. These properties shall include application domain (e.g. “weather forecasting” or “command and control”) and organization domain (e.g. “NIMA” or “USGS”) identifiers.

Style and Symbol Representation Requirements

21. The system must support different symbol representations (i.e., encodings, data file formats) without requiring knowledge of their representation. The system must have knowledge of the metadata representation. Therefore, the system will have the ability to work with style and symbol representations based on their metadata.

22. The system should have the capability to transform style information between representations, i.e. to provide a style in any of the formats/encodings the system supports. Considering the previous item, supporting a representation means the ability to always produce style and symbol document instances in a supported encoding.

23. The system must be able to check the validity of symbol and style documents to be stored in the repository. The validation will be performed exclusively against the schema of the document that is provided with the request together with the input data. Invalid data must be rejected. The validation will be performed only once – on input.

6.3 Architecture Requirements

1. The Architecture shall provide enough detail to show how it fits into the OGC Critical Infrastructure Collaborative Environment (CICE)
2. The Architecture shall focus on the exchange of information through online services
3. The Information Architecture shall focus on Critical Infrastructure and Emergency Mapping related geographical data
4. The Architecture shall be built on open standards for interoperability
5. There shall be two Roles that organizations will play in support of the EMS-1:
   - 5.1. Requestors of EMS Information
   - 5.2. Providers of EMS Information
6. The EMS-1 shall provide the ability for actors in those Roles to:
   - 6.1. Publish EMS Information
   - 6.2. Access EMS Information
7. Create and Update EMS Information
8. Portray maps.
9. Within the Architecture, Distributed Geoprocessing Resources shall be defined as information from multiple sources available as network addressable instances of typed data or services (OpenGIS® Web Services) including Data Services, Portrayal Services, Processing Services, and associated Encodings
10. Local, State, Federal, and private sector organizations shall be able to publish, find, access, integrate and apply Distributed Geoprocessing Resources across a collaborative network environment
11. Distributed Geoprocessing Resources shall be accessible:
    - 11.1. Vertically - Information sharing among Federal, Local and State departments/agencies, Non Governmental Organizations, and Private Sector Companies
    - 11.2. Horizontally - Information sharing between Federal, Local, or State departments/agencies, Non Governmental Organizations, and Private Sector Companies.
12. Information may also be shared on a transnational basis.
13. The EMS-1 Architecture efforts shall maintain coordination with ongoing related activities.

14. Within the EMS-1 Architecture, Application Clients shall be able to address HLS requirements between and across organizations where the data/service (transaction) end-points are at Federal, State, or Local levels.

6.4 Design Principles

No architecture is truly meaningful or valuable until it is shown to be implemented and useful. A process framework is therefore needed to support and enforce conformant implementation. This clause presents elements of a process framework, including design principles and high-level goals, constraints, assumptions and guidelines for EMS-1 design and specification. Elements include:

- Everything is a network resource including clients, services, data content, appliances, and computers.
- Resources (especially services) have contracts (i.e., resources must have well-defined roles, responsibilities, interfaces, and semantics)
- Interoperability of services over time is maintained by focusing on commitment to contracts not adherence to static protocols.
- Design for availability through dynamic discovery of resources:
    - Assume networks are unreliable and will fail
    - Do not assume given resources are always available in the same location
    - Promote "self-healing" through dynamic discovery and fail-over to other resources
    - A dynamic distributed application relies on the availability of multiple instances of any given resource type; the more effective the typing framework, the more dynamic and therefore reliable the application can become.
- Maximize stateless behavior of resources:
    - Resources are accessed on a transient basis.
    - Persistent state should be maintained solely on the tier that is interested in (responsible for) the specific computational transaction (this may also be termed separation of concerns)

  - Minimize dependencies of resources on the network; maximize loose coupling (i.e., minimize hard-coded dependencies between resources such as client to service instances).

  - Services should maintain state only within the context which requires it for a particular application (e.g. single request, single transaction of multiple requests with one client, single task sequence of transactions with multiple clients, etc.).

  - Service states which are to be maintained indefinitely (e.g. Web Map client contexts, user access profiles) should be managed as (metadata) resources in their own right rather than as stateful service behavior.

  - Clients and services should limit the duration over which they hold resources to minimize chances of losing the resource and maximize reuse.

- To maximize reuse, assume resources will be deployed and used in different application and deployment contexts over time

- Assume different deployment platforms and network communications protocols (e.g., transactional synchronous, point to point asynchronous messaging, broadcast asynchronous messaging, real time streams, low bandwidth formats, etc) will be required

6.5 Use Cases

The use cases in Annex A describe the use of SMS resources in support of EMS-1. These should be viewed as context for the requirements listed above and the Information, Computation and Engineering viewpoints presented in the clauses that follow.

7 Information Viewpoint

The information viewpoint is concerned with the information processing semantics, information system constructs (information model, semantics, data models, schemas), concepts, rules and structures of the architecture independent of distribution and implementation details.

Digital symbolization standards that support multiple portrayals of geospatial data are required to facilitate interoperable map analysis and data sharing. There are many ways to graphically portray geospatial information. Geographic features, for example, may have multiple symbols assigned to them. The choice of which symbol to apply to a feature for portrayal may need to be made dynamically, depending on its type, values of its properties, the application in which it is portrayed and the preferences of the user viewing

it. There is a need to portray the same feature in different ways. The Style Management Services (SMS) architecture is designed to enable more flexible and interoperable map portrayal.

The SMS information architecture (Figure 5) has two major components that can function independently: a Style Manager and a Symbol Manager. Each of these components is comprised of two subcomponents: a Catalog-Registry and a Repository. The purpose of the respective catalog objects is to manage metadata about extrinsic data representing style and symbol objects and to provide an interface for searching and discovery of these objects. The purpose of the repository objects is to store instances of style and symbol objects and to provide an interface for clients (e.g., SMS Clients) to access to them.

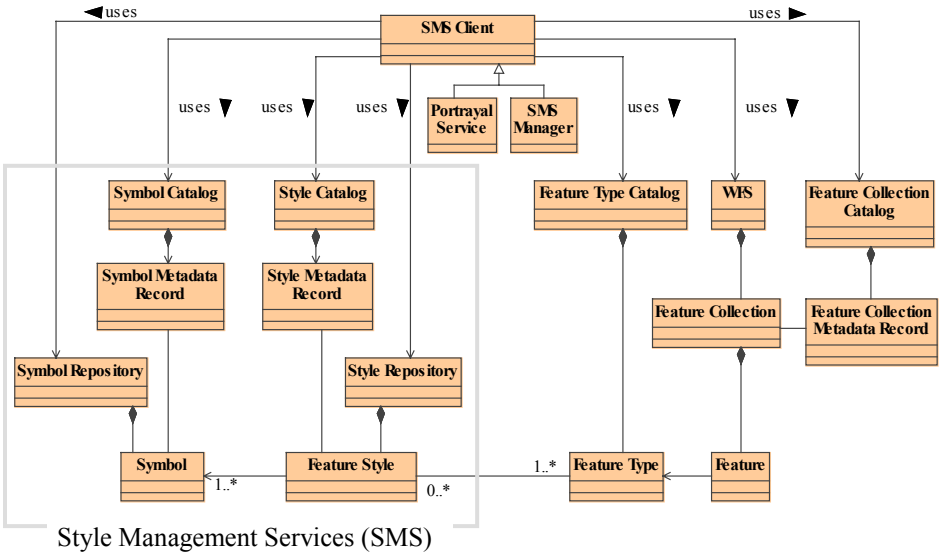


Figure 5. SMS Information Model

The following clauses describe the objects and their relationships depicted in the object model (Figure 5) adapted from the SMS Discussion Paper (OGC Project Document 03-031).

7.1 SMS Client

An SMS Client object is a consumer of SMS (Symbol Catalog, Symbol Repository, Style Catalog, Style Repository) object services. There are two kinds:

Portrayal Service – provide visualization of geospatial information. Portrayal Services are components that, given one or more inputs, produce rendered outputs (e.g., cartographically portrayed maps, perspective views of terrain, annotated images, views of dynamically changing features in space and time, etc.).

SMS Manager – augment geospatial information by allowing style and symbol objects and their metadata to be created, read, updated and deleted. Supports ability to manage associations between Feature Type and Feature Style objects and Feature Style and Symbol objects. Allows users to preview the appearance of symbol and style instance objects bound to feature objects.

7.2 Symbol Catalog

Provides the basic mechanism for SMS Client objects to publish and discover essential operational information about Symbol Objects. Allows the ability to manage (create, read, update and delete) and search for Symbol Metadata Records.

7.3 Symbol Metadata Record

Symbol Metadata Record is an object describing Symbol Objects. The metadata requirements of Symbols are analogous to those of Style Objects.

7.4 Symbol Repository

Allows the ability to create, read, update and delete Symbol Object instances in a repository. Allows SMS Client objects to publish and access Symbol Object instances.

7.5 Symbol Object

Symbols are pieces of graphics used by Portrayal Services to represent geographic features on a map. Symbols are the instructions for how vector and raster graphics are to be portrayed. Vector graphic symbols may have properties such as geometry, graphic, fill, color, stroke, font, orientation, size, opacity etc. Raster graphic symbols may have properties such as opacity, RGB channel selection, color map, shaded relief, contrast enhancements, etc.

Symbols may be represented as a set of graphical instructions using encoding languages such as SLD, SVG or CGM or as a raster graphic (image file) encoded in one of a set of standard formats (e.g., JPG, PNG, GIF, CGM). In principle, any number of Symbol representations may be used with the SMS architecture.

7.6 Style Catalog

Provides the basic mechanism for SMS Client objects to publish and discover essential operational information about Style Objects. Allows the ability to manage (create, read, update and delete) and search for Style Metadata Records.

7.7 Style Metadata Record

A Style Metadata Record is an object describing Style Objects. The metadata includes descriptive information such as titles, keywords and contact information (such as defined in ISO 19115/19117/19119 specifications).

Each Style Object must have a name that can be used as a primary key for identification and access. In addition to referencing this style name, Style Metadata Record objects may also contain references for all of the Feature Types to which a style can be applied. This allows a SMS Client, after locating feature styles, to more easily locate and access Style Objects

7.8 Style Repository

Allows the ability to create, read, update and delete Style Object instances in a repository. Allows SMS Client objects to publish and access Style Object instances.

7.9 Feature Style Object

A Feature Style Object is a description of how to portray a type of feature or coverage in a particular way. Styles are used by Portrayal Services. Styles can specify conditions that allow Features to be portrayed based on the value of a Feature Object's type and its geometric and non-geometric attributes. Styles are composed of one or more Symbols that may be stored with the Style Object or separately managed by SMS. Associations of Symbol Objects with geographic Feature Objects are explicitly given through Styles.

In principle, any number of style representations may be used with the SMS architecture. The primary representation, however, is based on the OGC Styled Layer Descriptor Implementation Specification (SLD).

Figure 6 shows the Feature Style to be the locus of associations between nodes of classification(s) of Feature Types and the Symbol Objects used to represent features in maps. Feature Styles consist of rules defined to form dynamic (run-time) associations of feature instances to Symbol Objects depending on the value of a feature's properties. Notice too that Style Metadata Records capture the association of Feature Type to Style Objects so that these associations can be queried to discover all Feature Styles associated with a given Feature Type.

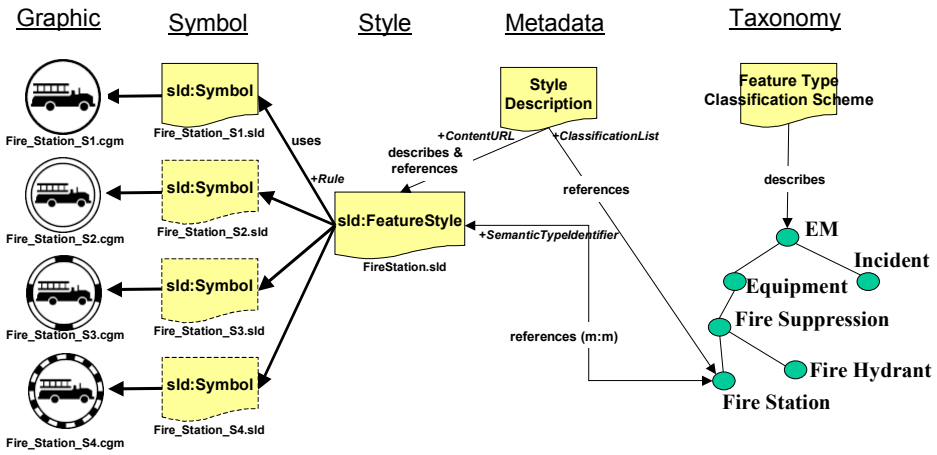


Figure 6. Information Elements of Feature Style

7.10 Feature Type Catalog

Provides the basic mechanism for application objects (including SMS Clients) to discover operational information about Feature Types. Allows the ability to manage (create, read, update and delete) and search for Feature Types.

Figure 7 shows the relationships and roles of features, feature types, catalogs (i.e., registries) and repositories. The central idea of this information viewpoint is that features (instances) of well-known type are measured (i.e., digitized) and recorded (i.e., archived) according to a Collection Process and the types of features to be collected. Feature instances and associated metadata are archived as Feature Collections. Feature Collections have metadata describing their content, use, lineage, sources and feature types.

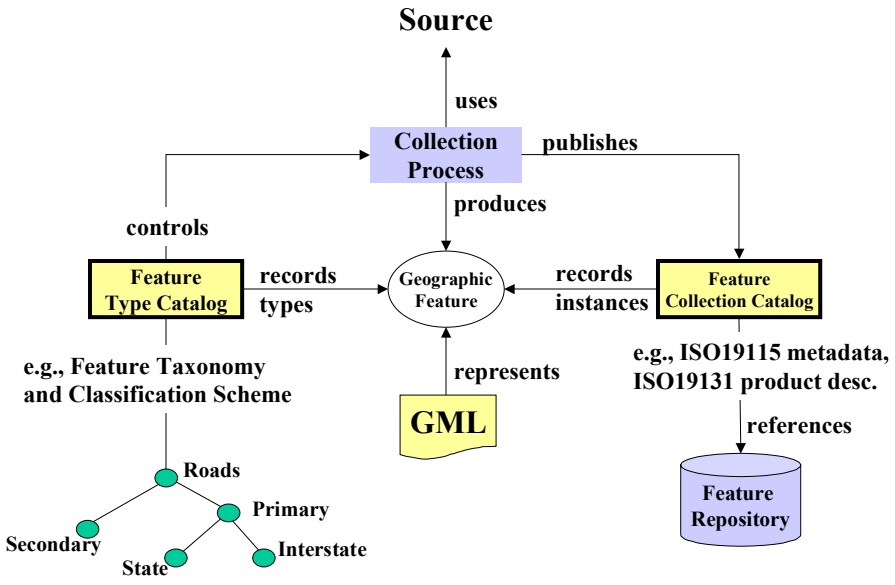


Figure 7. Role of Feature Taxonomies, Catalogs and Repositories

Feature instances have well-known types that are classified according to taxonomies with commonly agreed-upon semantics for a domain. Feature types are instantiated according to a schema defining a standard way of representing domain-specific types with a common set of properties, relationships and semantics (as GML). In a feature production process, an authority typically controls the feature type taxonomy and the schema. Taxonomies, schema and metadata are resources about feature types and feature instances (also resources) that are controlled, referenced and accessed in Catalogs and Repositories.

Taxonomies, schema, metadata, feature instances, feature collections, registries and archives are all resources that must be related and accessed within a "semantic web". Catalogs of resource types and instances play a critical role in enabling semantic interoperability.

7.11 Feature Type Object

Feature Type Objects are nodes in a taxonomy of feature types. They have or reference information about the feature type taxonomy itself, schema describing the structure of feature instances of the type as well as zero or more Feature Style Objects (and their metadata) that are appropriate for use in visual portrayal of feature instances of the type. Schema defining Feature Type objects may be encoded with GML.

7.12 Web Feature Service

A service object to access Feature Object representations from Feature Collections stored in repositories.

7.13 Feature Collection

A Feature Collection is a collection of Feature Object instances. Feature Collections are themselves valid features and can have location and other properties as defined in their schema.

7.14 Feature Object

Feature Objects are instantiations of Feature Types and abstractions of real world phenomena. Representations may be encoded with GML.

7.15 Feature Collection Catalog

Provides the basic mechanism for application objects (including SMS Clients) to discover operational information about Feature Collections. Allows the ability to manage (create, read, update and delete) and search for Feature Collections. Refer to Feature Type Catalog discussion and Figure 7.

7.16 Feature Collection Metadata Record

A Feature Collection Metadata Record is an object describing Feature Collection Objects. The metadata includes descriptive information such as titles, keywords and contact information (such as defined in ISO 19115/19117/19119 specifications). In addition, Feature Collection Metadata Records may directly or indirectly reference the Feature Type Objects of the Feature Objects it contains. Feature Collection Metadata Records are managed by Feature Collection Catalogs.

8 Computational Viewpoint

The Information Technology environment in which the EMS-1 will take place is the Critical Infrastructure Collaborative Environment (CICE). The CICE is based on the OpenGIS® Service Framework (OSF) defined in the OpenGIS® Reference Model, and relevant elements of the US National Spatial Data Infrastructure (NSDI) and Canadian Spatial Data Infrastructure (CSDI).

The OSF describes a computational model for OpenGIS® Services. The objective of the OSF is to detail how geospatial software services plug into broader interoperability infrastructures to use and extend diverse, loosely coupled sources of data and services. The OSF draws on Topic 12 of the OGC Abstract Specification (Service Architecture-

ISO 19119) but focuses more specifically on current technologies, platforms and mechanisms for enabling implementation of interoperable services.

The EMS-1 Computational Architecture provides a platform for geospatial interoperability between applications and critical infrastructure and emergency management information resources. This Framework includes the following:

- The OpenGIS® Service Framework (OSF) of the OpenGIS Reference Model establishes the basis for common service interfaces and data exchange protocols that can be utilized by any application.

- OpenGIS® Implementation Specifications provide guidance to application developers on how to build their applications to comply with this framework.

- OpenGIS® Services are implementations of services that conform to OpenGIS® Implementation Specifications.

Compliant applications, called OpenGIS® Applications, can then "plug into" the framework to join the enterprise operational environment. This loosely coupled approach to enterprise development results in very agile systems.

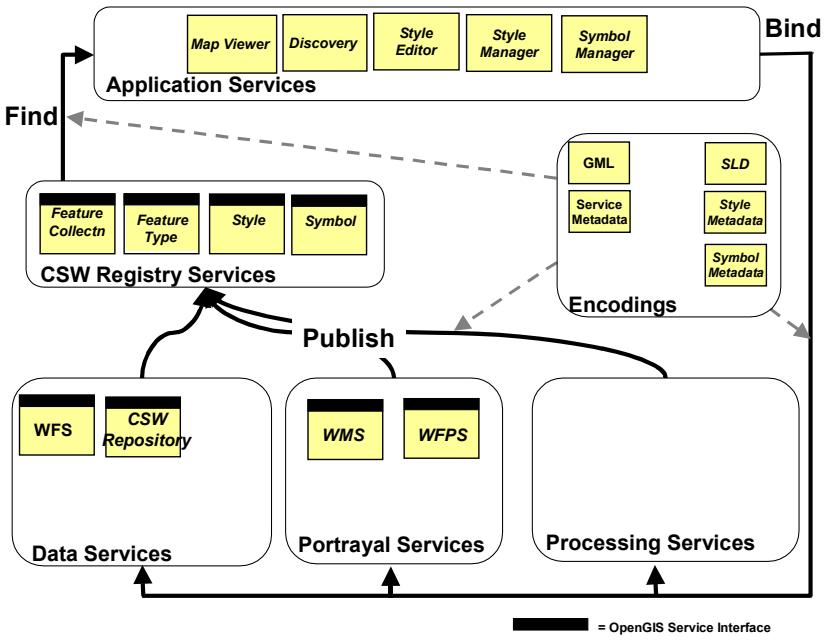


Figure 8. SMS Framework for EMS-1

Figure 8 identifies for Style Management Services Framework, a profile of the OpenGIS Service Framework, for EMS-1. All components shaded (yellow) in the above figure are key to the SMS Framework for EMS-1. Italicized components are anticipated to require enhancement based on EMS-1 requirements and previous work to enhance the specifications on which they are based. Elements of this computational framework are described in greater detail below.

8.1 Application Services

8.1.1 Web Map Viewer Client

A Web Map Viewer Client, including server-based clients, can issue GetMap requests for different maps to several independent Web Map Servers. If each map has the same geographic area and physical dimensions, and if their backgrounds are transparent, then they can be overlaid in a single window to produce a combined map. For example, server A might produce a topography image, server B a map of rivers and lakes, and server C a map of watershed boundaries. Each server maintains the type of data in which it specializes, but the end user can obtain a combined presentation of the three Layers. The Web Map Viewer Client may itself perform the portrayal process, acting as a tightly coupled Portrayal Service, or it may delegate to a loosely coupled Portrayal service, such as a WMS, to produce a map.

Web Map Viewer Clients must support the ability to use the SMS components when requested by a user to generate a cartographically portrayed picture. If the Web Map Viewer Client acts as a Portrayal Service (see description below) to render maps itself, it must specifically support the baseline version of SLD developed for SMS (refer to Section 8.5.2).

8.1.2 Value Added Client

Value-Add Clients are a class of Application Service specializing in supporting the ability for users to collect and submit user input that augments geospatial information originally supplied by a data producer. Value-Add Application Services support augmentation of data by creating new features, and updating or deleting existing features, styles, symbols and metadata. Value-Add Application Services typically support human interaction controls, the ability to add and remove layers, and the ability to create, select, and display cartographic styles to support of the value-adding process. Value-Add Application Services may also support the ability to generate preview graphics, draw on a background map and commit updates to repositories and databases using OpenGIS Data Services such as WFS.

A SMS Manager client is envisioned for supporting the management of styles, symbols and metadata as required to support map production for interoperable Emergency Management and Response applications. The SMS Manager should support the ability to:

- Discover Feature Types, Styles and Symbols published to SMS-enabled facilities (Catalogs and Repositories).
- Create, read, update and delete Style Objects (i.e., SLD elements) to Style Repositories and associated Style Metadata Records to Style Catalogs.
- Create, read, update and delete Symbol Objects (e.g., SVG or CGM elements) to Symbol Repositories and associated Symbol Metadata Records to Symbol Catalogs.
- Link Feature Style Objects to Feature Type Objects and save results to SMS facilities.
- Link Symbol Objects to Feature Style Objects and save results to SMS facilities.
- Generate preview graphics (and/or full-fledge maps) showing the dynamic binding of Styles to Feature Objects.

8.2 Catalog Services

Catalog Services provide a common mechanism to classify, register, describe, search, maintain and access information about network resources. Resources are network addressable instances of typed data or services. Catalog may be differentiated by their role such as for cataloging data types (e.g., types of geographic features, coverages, sensors, symbols), online data instances (e.g., datasets, repositories, symbol libraries), service types (e.g., portrayal, processing, data services) and online service instances.

The metadata content published to the catalog, while conforming to the same information model, describes different kinds of resources using metadata that may be structurally and semantically different than metadata for resources of other types or for other purposes or organizations. The OpenGIS Catalog Service for theWeb profile (CSW) defines a common information model and the service interface to discover and access resource offers, regardless of the type of resource and the content of the metadata.

Type Catalogs contain metadata about resource (data and service) types (e.g., types of features, feature collections, styles, symbols, and services) as taxonomies that are shared and used within information communities. The ability to publish and share this information is an essential requirement for distributed applications to share and exploit, with a common semantic, these resources. Type Catalog Services provide access to these metadata and taxonomies of types. Support for publishing and referencing taxonomies is explicitly supported in the SMS information model specification.

8.2.1 Feature Type and Feature Collection Catalog

It is anticipated that both the CSW will be used and extended (as required) to support the requirements of EMS-1. In particular, the following capabilities are desired of Catalog-Registries for Feature Types and Feature Collections:

1. Support for construction and publication of feature type taxonomies
2. Ability to publish and find schema describing the structure of feature types.
3. Ability to publish and find metadata about features and/or feature collections.
4. Ability to associate feature instances and feature collections to classification nodes in a feature taxonomy
5. Ability to bind schema to feature types and feature instances
6. Ability to bind styles to feature types
7. Ability to bind symbols to styles

8.2.2 Style and Symbol Catalog

As with the Feature Type and Feature Collection Catalog, the role of Style and Symbol Catalog is to support discovery, access and management of style (SLD) and symbol libraries. Style and Symbol Catalog Services provide access to metadata about these resources and to the published symbol and style instances accessible in repositories.

The following capabilities are desired of style and symbol registries:

- Support for construction and publication of symbol type taxonomies
- Ability to publish and find metadata about styles and symbols.
- Ability to associate style and symbol instances to classification nodes in a style and symbol taxonomies
- Ability to bind styles to feature types
- Ability to bind symbols to styles

8.3 Portrayal Services

Portrayal Services provide specialized capabilities supporting visualization of geospatial information. Portrayal Services are components that, given one or more inputs, produce rendered outputs (e.g., cartographically portrayed maps, perspective views of terrain,

annotated images, views of dynamically changing features in space and time, etc.) Portrayal Services can be tightly or loosely coupled with other services such as Data and Processing Services and transform, combine, or create portrayed outputs. Portrayal Services may use styling rules specified during configuration or dynamically at runtime by Application Services.

8.3.1 Styled Layer Descriptor enabled Web Map Service (WMS)

A Web Map Server (WMS) generates "pictures" of georeferenced data. Independent of whether the underlying data are simple features (such as points, lines and polygons) or coverages (such as gridded fields), the WMS produces an image of the data that can be directly viewed in a graphical web browser or other picture-viewing software. An extension of the basic Web Map Server is the Styled Layer Descriptor (SLD) Web Map Server. The SLD enabled WMS inherits all of the attributes from the Web Map Server and adds support for the use of Styled Layer Descriptor documents to specify styling. Instead of generating maps of particular named layers in one or more predefined styles, an SLD Map Server extracts features from a data provider and renders them using a stylistic description encoded in XML.

The WMS instances used for EMS-1 must support the ability to interconnect with one or more WFS instances to access feature data and apply appropriate styles and symbols (as specified by SLD-encoded style elements associated with feature types) to produce a cartographically rendered map.

8.3.2 Styled Layer Descriptor enabled Coverage Portrayal Service (CPS)

The Coverage Portrayal Service (CPS) defines a standard interface for producing visual pictures from coverage data. Typically coverage data are retrieved via a WCS instance. CPS extends the WMS interface and uses the Styled Layer Descriptor (SLD) language to support rendering of WCS coverages. CPS facilitates wider use of coverage data by making views of coverages visible within thin-clients (e.g., Web browsers). To a service requestor, the CPS appears as a WMS instance, but with additional parameters to control the retrieval and/or rendering of coverage data. The CPS may require the client to specify the targeted WCS.

CPS may be used to support:

- assigning multi-spectral bands in an image to color channels in a picture,
- creating chloropleth maps from coverage data using client-specified color-bins
- preset rendering mechanisms such as hill-shaded elevation
- combining multi-spectral pixel values according to client-specified or server-defined formulas (e.g., Normalized Difference Vegetation Index).

The CPS instances used for EMS-1 must support the ability to access coverage data and apply appropriate styles and symbols (as specified by SLD-encoded style elements associated with coverage types) to produce a cartographically rendered map.

8.4 Data Services

Data Services provide access to collections of data in repositories and databases. Resources accessible by Data Services can generally be referenced by a name (identity, address, etc). Given a name, Data Services can then find the resource. Data Services usually maintain indexes to help speed up the process of finding items by name or by other attributes of the item. The OpenGIS® Framework defines common encodings and interfaces in which multiple, distributed Data Services are accessed and their contents "exposed" in a consistent manner to other major components. The sections below describe the current set of Data Services of the OpenGIS® Framework.

8.4.1 Web Feature Service (WFS)

The Web Feature Service supports the query and discovery of geographic features. In a typical Web-base scenario, Web Feature Service delivers GML (XML) representations of simple geospatial features in response to queries from HTTP clients. Clients (service requestors) access geographic feature data through a WFS by submitting a request for just those features that are needed for an application. The client generates a request and posts it to a WFS instance (a WFS server on the Web). The WFS instance executes the request, returning the results to the client (service requester) as GML. A GML-enabled client can manipulate or operate on the returned features.

8.5 Encodings

8.5.1 Geography Markup Language (GML)

The Geography Markup Language (GML) is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features. GML utilizes the OpenGIS® Abstract Specification geometry model which has been harmonized with the ISO geospatial geometry model. Unlike a simple geometry model, the GML Specification also includes the ability to handle complex properties. Work is underway to harmonize this complex property model with the ISO efforts in the same arena.

GML is used to represent geographic features conforming to well-defined application schema for purposes of transport across computational interfaces.

8.5.2 Style Layer Descriptors (SLD)

The Styled Layer Descriptor (SLD) encoding specifies the format of a map-styling language for producing georeferenced maps with user-defined styling. This language is used to create XML documents that control the visual portrayal of the data with which they work. The ability for a human or machine client to define the styling rules requires a styling language that the client and server can both understand. The SLD language can be used to portray the output of Web Map Servers, Web Feature Servers and Web Coverage Servers. The SLD is defined using XML Schema.

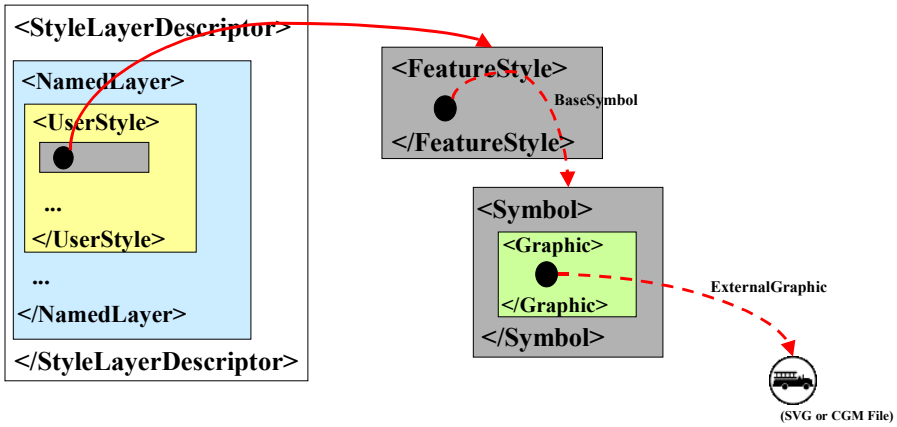


Figure 9. Remote references to Style and Symbol Objects

For EMS-1, the SLD 1.0 schema-set has been refactored to allow Feature Style, Symbol and Graphic elements to be remotely referenced (Figure 9). In this way, SLD documents are now more modular, allowing them to be more easily and dynamically constructed. Also, these elements can now be published, managed and accessed independently (via CSW) so they may be reused in multiple applications and applied to multiple datasets.

8.5.3 Style Metadata

Metadata for describing Style Objects. Refer to Annex C for further description of structure and use of Style and Symbol Metadata.

8.5.4 Symbol Metadata

Metadata for describing Symbol Objects. Refer to Annex C for further description of structure and use of Style and Symbol Metadata.

9 Engineering Viewpoint

The Engineering Viewpoint is concerned with the infrastructure required to support system distribution. It focuses on the mechanisms and functions required to: a) support distributed interaction between objects in the system and b) hide the complexities of those interactions. It exposes the distributed nature of the system, describing the infrastructure, mechanisms and functions for object distribution, distribution transparency and constraints, bindings and interactions.

Recall that Symbol Management Service (SMS) is an architecture for describing service behavior and functionality and may be implemented as a physical service at a single endpoint or a logical service implemented as a composition of interfaces at different endpoints. As with all OWS services, the binding mechanism that enables transparent service distribution is HTTP GET/POST with XML and MIME encodings used for message transport.

Figure 10 shows a possible deployment of SMS services on the Web. Notice that all SMS components (Symbol Catalogs and Repositories, Style Catalogs and Reposities, Feature Catalogs and Repositories) can be physically separated, replicated and even re-purposed for use in different applications at different facilities and servers. These same SMS services and the SMS Client could just as easily and transparently be physically deployed all on the same host server.

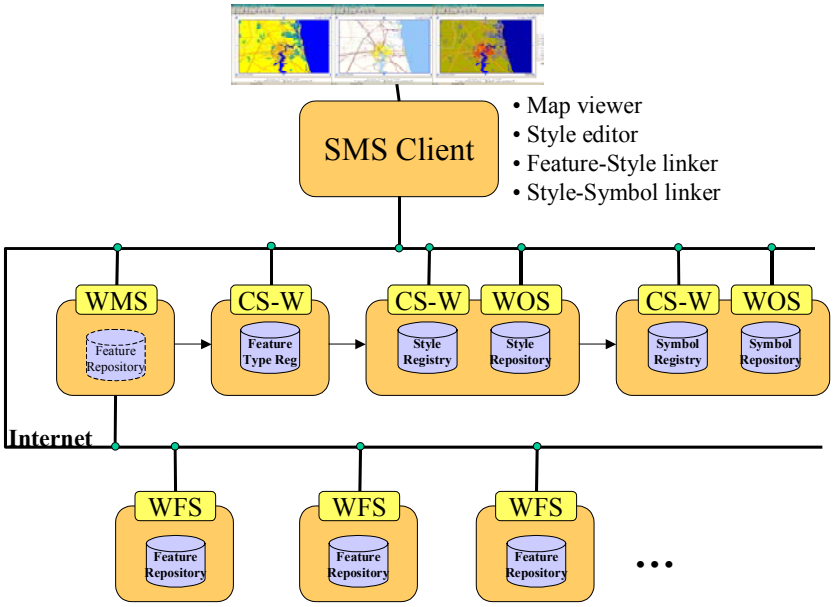


Figure 10. Deployment of SMS Framework for EMS on the Web

10 Technology Viewpoint

The Technology Viewpoint is concerned with the choice of technologies to support system distribution. It defines the implementation and deployment environment using technologies, standards and products of the day, providing "reference points" for conformance testing.

The following specifications are to be supported as the technical baseline for implementation of EMS-1 capabilities. These specifications may however require enhancements to achieve the EMS-1 objectives and successful implementation of requirements. In this case, software implementations must also be modified to support the enhanced specifications.

Data	
Geographic Vector Data	○ GML 3.0 – OGC Geography Markup Language 3.0. OGC Document 02-023r4. ○ Level 0 Profile of GML3 for WFS – CIPI1.2 Interoperability Program Report. OGC Document 03-003r9.
Symbol Data	○ Scalable Vector Graphic. W3C Recommendation http://www.w3.org/TR/SVG11/

	○ Computer Graphic Metafile (CGM). ISO/IEEE 8632-1 CGM Standard. ○ Symbols for Emergency Management and First Responder communities – a map symbology set under development by the FGDC Homeland Security Work Group (HSWG). See: http://www.fgdc.gov/publications/homeland.html and http://www.fgdc.gov/HSWG/ ○ Geospatial Symbols (GeoSym) for Digital Displays. The map symbology set defined by NIMA to portray Vector Product Format (VPF) data. See: http://www.nima.mil/cda/article/0,2311,3104_12137_118865,00.html and http://www.nima.mil/ast/fm/acq/mil89045.pdf ○ MIL-STD-2525B is a Department of Defense Interface Standard that defines Common Warfighting Symbology http://symbology.disa.mil/symbol/mil-std.html.
Query Languages	
OGC Filter	○ OGC Filter Encoding Implementation Specification 1.0.0. OGC Document 02-059. ○ OGC WFS 1.0 and Filter 1.0 Change Requests. OGC Document 02-063.
Styling Description Languages	
SLD	○ SLD – Styled Layer Descriptor Implementation Specification 1.0. OGC Document 02-070. ○ SMS[5] – Style Management Services Discussion Paper. OGC Document 03-031.
SVG	○ Scalable Vector Graphic. W3C Recommendation http://www.w3.org/TR/SVG11/
Data Access Services	.

[5] The OWS1.2 Testbed initiative produced and tested significant enhancements to the SLD 1.0 specification and schema that are important to the technical approach in this initiative. Review the referenced SMS Discussion Paper (OGC Project Document 03-031) for more information about the enhancements to the SLD specification.

WFS	○ OpenGIS® Web Feature Service Implementation Specification 1.0.0. OGC Document 02-058. ○ OGC WFS 1.0 and Filter 1.0 Change Requests. OGC Document 02-063.
Portrayal Services	
WMS	○ *OpenGIS® Web Map Server version 1.2* ○ OpenGIS® Map Context Documents Implementation Specification, version 1.0.
CPS	○ Coverage Portrayal Service (CPS) Interoperability Program Report. OpenGIS Project Document 02-019r1.
Catalog Services	
CSW[6]	○ OpenGIS® Catalog Service Implementation Specification, version 2.0. OpenGIS Project Document 04-021r2. ○ OpenGIS® Catalog Service – Web Application Profile (CSW). OpenGIS® Project Document 03-094 (version 0.8.3).

[6] The Catalog Services Revision Work Group of the OGC Technical Committee has issued a new revision of the Catalog Services Implementation Specification 2.0 (OGC document 04-021r2) that incorporates, as a profile, a "stateless" HTTP binding that is derived from earlier work on WRS including implementations developed and demonstrated in previous Interoperability Program initiatives. The CS 2.0 specification has not been formally approved. The Catalog Service – Web Application Profile (CSW) (OGC document 03-094) extends the CS 2.0 specification to define HTTP bindings and an information model specifically defined for use on the Web.

Annex A – Use Cases

(informative)

The use-cases found in this annex extend or derive from use-cases in *OWS1.2 Registry Service Requirements Interoperability Program Report* (OGC Project Document 03-027) and the *OWS1.2 SMS Requirements Interoperability Program Report* (OGC Project Document 03-030).

A.1 Publishing SMS Resources

These use-cases describe a series of interactions for publishing SMS resources including feature types, styles, symbols and their metadata.

The basic premise of SMS publishing is that there will be enough information in the registry to allow an end user of the system to find data instances and render them without having to perform any manual operations. To this end, there are a series data instances published in the registry that have all the information required to render them available in the registry as well. The use cases in this section describe how the various pieces of information required to make this a reality are structure and stored in the registry.

A.1.1 talks about publishing classification schemes that will be used to associate registry objects with their intended use. For EMS-1, the Digest/FACC and FGDC Emergency Management Classification hierarchies were used to provide the required classifications.

A.1.2 talks about publishing the Application Schemas of the data instances that are available. To encourage reuse of styling information, the Application Schemas or data instances are published to the registry. Styling documents are tied to the Application Schema of a data instance so this level of indirection will allow a single styling document to be used by multiple data instances that share an Application Schema.

A.1.3 gets into publishing Feature Styles that are the actual styling instructions for rendering data instances. The Feature Styles use element names from the Application Schemas in the generation of styling information. For the purposes of EMS-1, we are also recommending that Feature Style segments access Symbol objects that are stored in the registry.

A.1.4 talks about publishing Symbols to the registry. For EMS-1 the symbols were represented as SVG documents that were stored in the registry and accessed from the

Feature Style documents. The classification of the symbols to the classifaction schemes allows for powerful querying possibilities to find and use symbols of interest.

A.1.5 talks about how a WFS should be published into the registry to leverage all the other objects. The recommended linkages provide the end user with powerful tools to render WFS data instances.

A.1.1. Publish Classification Scheme to Registry

Name	Publish Classsification Scheme to a registry
Priority	High
Description	All content in the registry is classified by one or more classification schemes. Classification schemes are the main method for users to relate objects when dealing with the SMS. Data instances, styling information and symbols will all be tagged with appropriate classifications which will simplify the user's job when navigating or searching the registry to find data and styling information. To support this, users need to publish classitication trees to the registry.
Precondition	The client has access to the registry. The client has a classification tree they wish to publish.
Flow of Events – Basic Path	
1.	The client submits a Classification request to the registry to publish a entire classification tree..
2.	The registry populates the Classification Scheme and Classification Node entities in the Registry.
Alternate Flows	
	The client can query Classification information from the registry and use the identifiers returned to submit individual Classification Node entries.
Postcondition	The Classifiction Scheme has been registered.

A.1.2. Publish Application Schema to a registry

Name	Publish Application Schema to a registry
Priority	High
Description	Stying instructions in the registry are tied to the schema of the dataset being styles. Multiple instances of a particular schema can exist so Application Schemes are published separately from data instances. Application Schemas are XML schema documents that are obtained from WFS instances. Application schemas are unique based on a selected Element Name and Target Namespace of the XML Schema document.
Precondition	The client has access to the registry. The client has a WFS Application Schema document they wish to publish to the registry and the list of classifications they wish to use to classify the document. The client has to know any other metadata he/she wishes to register in connection with the taxonomy scheme.
Flow of Events – Basic Path	
1.	The client submits a PutRepositoryItem request to the registry containing the XMLSchema representation of the Application Schema they are publishing.
2.	The registry receives the request and saves the XML Schema document as a repository item.
3.	The client submits a transaction request to insert an Extrinsic Object of type 'Application Schema' with a linkage to the repository item inserted above. This request includes slots for 'TargetNamespace' and 'Element Name' to uniquely identify this schema.
4.	The client submits a series of transaction requests to classify the 'Application Schema' extrinisic object as appropriate Classification Schemes and Classification Nodes.
Alternate Flows	
	The insertion of multiple registry objects can performed with a single request instead of performing a transaction for each object.

Postcondition	The Application Schema document is loaded into the registry and classified as requested to allow for discovery.

A1.3 Publish a Feature Style to a registry

Name	Publish a Feature Style to a registry
Priority	High
Description	The client wishes to publish a Feature Style to registry to define a set of drawing rules. A Feature Style is a subset of a full SLD document thatwill be used by client systems to construct full SLDs for rendering data. The Feature Style will be associated with an Application Schema already published in the registry. This Feature Style will be classified as necessary to ensure that registry users can navigate or search by Classification to find styling documents of interest. As part of creating a Feature Style, clients will generally query the registry to find appropriate symbols for use in defning that style. The primary means of quering symbols will be based on classification although other means of quering supported by the registry (e.g., Name, Keyword) can be used.
Precondition	The client has access to the registry. The client has a Feature Style document that they wish to publish to the registry and the list of classifications they wish to use to classify the document.
Flow of Events – Basic Path	
1.	The client submits a query to the registry to find the Application Schema that matches the Feature Style they are creating. The TargetNamespace and Element name slots are used as search parameters.
2.	If the Application schema is not already published, the client publishes it as described in the Publish Application Schema to a registry use case.
3.	The client submits a PutRepositoryItem request to the registry containing the Feature Style XML document that the user wishes to publish.
4.	The registry receives the request and saves the Feature Style document as a repository item.

5.	The client submits a transaction request to insert an Extrinsic Object of type 'Feature Style' with a linkage to the repository item inserted above.
6.	The client submits a series of transaction requests to classify the 'Feature Style' extrinisic object as appropriate Classification Schemes and Classification Nodes.
7.	The client submits a transactions request to define an association of type 'Portrays' between the 'Application Schema' extrinisic object and the 'Feature Style' extrinisic object.
Alternate Flows	
	The insertion of multiple registry objects can performed with a single request instead of performing a transaction for each object.
Postcondition	The Feature Style document has been stored into the registry and the Feature Style has been registered.

A.1.4 Publish a symbol to a registry

Name	Publish a symbol to a registry
Priority	High
Description	The client wishes to publish a symbol to the registry.
Precondition	The client has access to the registry. The client has a symbol content to be published to the registry and the list of classifications they wish to use to classify the symbol. When a registry user wishes to build a Feature Style document using symbols in the registry, they will query the registry for Map Symbols of the desired Classifications to get a list. The symbol's themselves are available as repository items that can be referenced via URL from a Feature Style document.
Flow of Events – Basic Path	
1.	The client submits a PutRepositoryItem request to the registry containing the Symbol definition (SVG) that the user wishes to publish.

2.	The registry receives the request and saves the Symbol as a repository item.
3.	The client submits a transaction request to insert an Extrinsic Object of type 'Map Symbol' with a linkage to the repository item inserted above.
4.	The client submits a series of transaction requests to classify the 'Map Symbol' extrinisic object as appropriate Classification Schemes and Classification Nodes.
Alternate Flows	
	The insertion of multiple registry objects can performed with a single request instead of performing a transaction for each object.
Postcondition	The symbol object is stored as a registry object and the symbol has been registered.

A.1.5 Publish a WFS to a registry

Name	Publish a WFS to a registry
Priority	High
Description	In order to find data instances that can be rendered with the Feature Style and Symbol documents that are published above, it is necessary to publish Web Feature Services and their advertised FeatureTypes to the registry and to associate them with appropriate application schemas. In general terms, server software might query for a WFS URL and perform a GetCapabilities operation against it to get a list of FeatureTypes provided by that server. For each FeatureType, the server software might prompt for additional metadata about the feature type as well as appropriate classifications for the FeatureTypes. The collected information could then be published into the registry with appropriate classifications and linkages to support discovery and use by client applications. When searching the registry for data instances, the client software will key in to FeatureType Extrinsic Objects and/or Metadata Documents linked to Feature Type Extrinisic objects.

Precondition	The client has access to the registry. The client has a Service definition and a series Extrinisic Objects of type 'Feature Type' (probably extracted from a GetCapabilities request of the WFS) they wish to publish which represents the WFS that they wish to Publish. Optionally, the client has a series of metadata documents to which will be published to the registry as Extrinsic Objects of type 'Dataset Description' which will be associated with the Feature Types they will be publishing.
Flow of Events – Basic Path	
1.	The client submits a transaction requests to insert a Service entry for the WFS being published.
2.	The client submits a series of transaction requests to insert Extrinsic Objects of type 'FeatureType' for each FeatureType from the WFS being published.
3.	The client submits a series of transaction requests to classify each 'FeatureType' extrinisic object with appropriate Classification Schemes and Classification Nodes.
4.	The client submits a series of PutRepositoryItem requests to the registry containing the Metadata documents that describes the WFS Feature Types the user wishes to publish.
5.	The client submits a series of transaction requests to insert Extrinsic Objects of type 'Dataset Description' with a linkage to the repository item inserted above.
6.	The client submits a series of transaction requests to insert 'Describes' associations between the each 'Dataset Description' Extrinisic Object and its associated 'FeatureType' Extrinsic Object.
Alternate Flows	
	The insertion of multiple registry objects can performed with a single request instead of performing a transaction for each object.
Postcondition	The association has been updated.

A.2 Discovery and Access of SMS Resources

These use-cases describe a series of interactions for discovering and accessing SMS resources including feature types, styles, symbols and their metadata.

A.2.1 Create map from single or multiple WFSs

Priority	High.
Description	A user of a Web-based mapping system wishes to find a dataset and view the results. This will typically be accomplished by the user invoking a search screen where they will pick from a list of search criteria. The user will likely be presented with a navigatible version of the classification trees available in the registry to limit search results. Keyword and Geographic Coverage are two other likely candidates for searching. Once the search is performed, the user select a Feature Type of interest from the matching WFS FeatureTypes and ask the mapping system to render it. The rendering system will query the registry to find a list Feature Style extrinisic objects associated with the Application Schema that's associated with the selected Feature Type. The user will be asked to select one of the matching Feature Style entries. The mapping system will then create an SLD by appending the selected FeatureStyle with a remoteOWS component it constructs from the WFS parameters it retrieved from the registry in the initial search. The mapping system then invokes a WFS Portrayal service (either statically chosen by the mapping system or by quering the registry for a list of available ones) with and SLD parameter pointing to the contructed SLD and other required WMS parameters (e.g. SRS, BBOX, WIDTH, HEIGHT, …) to create the view required.
Preconditions	The user must know the names of the features or feature types that he wants portrayed or be able to search the registry to find them. The user must either know the names of desired styles or some other properties that uniquely identify the style or be able to identify the style of interest by reviewing the metadata returned from the registry search.
Typical Flow of Events	
1. User queries SMS Service Registry for list of WFS Feature Types of interest, performs additional SMS Service Registry requests to retrieve additional metadata about objects that are found and selects a Feature Type of interest.	

2. User queries SMS to find Application Schema associated with selected Feature Type. If an Application Schema cannot be found, there is no sytling information stored in the registry and the user will be unable to render the data.

3. User queries SMS to find list of FeatureStyles associated with the Application Schema found above.

4. The user reviews the metadata associated with the returned Feature Styles (possibly running additional queries to retrieve additional metadata about a particular entry) and selects the one of interest.

5. The user retrieves the Registry Object associated with the selected Feature Style. This Registry Object is an actual XML snippet that will be used to help create the SLD for styling the data.

6. The user queries the Registry for Service Binding information for the WFS that contains the selected Feature Type above and uses the returned information to construct the remoteOWS section section of the SLD document.

7. The remoteOWS section and the FeatureStyle section are combined to construct a complete SLD that will be used as a parameter to a WFS Portrayal Service.

8. The user queries the Registry to find a WFS Portrayal Service of interest and retrieves its Service Binding information.

9. A WMS URL is constructed from the binding information, the SLD that was created and other WMS parameters as necessary to construct the desired map view.

10. The above procesure can be repeated for additional data instances and the resulting WMS URLs can be show to the user using overlapping browser images (with appropriate transparency) or the images can be merged together by a middle-ware system to provide a single image to the user.

Alternative Paths

Postconditions

Unless exceptions are raised, a raster (or possibly vector) depiction of a map is returned.

A.2.2 Browsing Available Feature Styles

Priority	Medium.
Description	The user wishes to browse the registry to review the list of Feature styles that are available. The user searches the registry by Classificastion (or other query criteria) to find the set of Feature Styles that are available. For any particular Feature Style of interest, the user can download the associated SLD segment or perform additional queries of the registry to find data instances that can be rendered using the Feature Style they selected.
Preconditions	None.
Typical Flow of Events	
1. User issues a request for a list of available Feature Styles. They might query the registry by Classification or any other query method support by the Service Registry. 2. User browses the Feature Style list and selects a Feature Style of Interest. From this selection, the user can download the XML document that was saved for a Feature Style of interest. 3. Alternately, the user can query the Registry for the Application Schema associated with the Feature Style and from this Application Schema, the user can further query the registry to find data instances that implement this schema. 4. Once a data instance of interest has been identified, the user can further query the registry to retrieve the service and service bindings that offer the data instance and construct a WMS call via a Web Feature Portrayal Service to view the Feature Style in action.	
Alternative Paths	
None.	
Postconditions	
An SLD document or other style definition is returned to a user based on the user's request.	

A.3 Additional detailed use-cases.

Refer to the SMS Requirements Interoperability Program Report (OGC Project Document 03-030) for the remaining detailed use cases whose names are listed here:

1. Requesting Styles from a Style Management Server
2. Retrieve style by name
3. Find Style by Feature Type Name List
4. Find Style by Style Metadata
5. Find Style by Symbols Used
6. Find Style by Combination of Criteria
7. Dynamic Creation of SLDs
8. Pushing Information Back to a Style Management Service
9. Inserting a Style into a Style Management Service
10. Inserting a Symbol into a Style Management Service
11. Deleting a Style into a Style Management Service
12. Deleting a Symbol from a Style Management Service
13. Combining Multiple Styles
14. Symbol Management
15. Find Symbol by Classification
16. Find Symbol by Property
17. Styles as they Relate to Other OGC Services
18. Query a Portrayal Service for Styling Capabilities
19. Find a Feature Server providing data applicable to a given style
20. Using Styles
21. Using parameterised styles

22. Using Feature Type styles

Annex B – Styled Layer Descriptor Schemas for SMS

(normative)

For the EMS-1 project, the SLD 1.0 schema-set was refactored to support the requirements for SMS. These changes have been packaged into an OGC TC Change Proposal document and have been submitted to the SLD Revision Working Group (RWG) for further consideration and adoption.

The SLD change proposal (OGC Project Document 04-009) containing the SMS-compatible schema-set used for the EMS-1 project is available on the OGC Pending Document Archive (http://portal.opengis.org/files/?artifact_id=5263&version=1).

Annex C –Style and Symbol Metadata

This annex contains a discussion of the metadata documents that are used to describe both styles and symbols. Section C.1 is the XML schema for style and symbol metadata. C.2 walks through a sample metadata document. Section C.3 describes issues with the metadata. C.4 has example listings of style and symbol metadata XML. C.5 is an example form describing metadata elements to be completed by subject matter experts for capturing styleand symbol metadata.

C.1 Style and Symbol Metadata Schema (normative)

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="http://www.opengis.org/sms"
xmlns:tns="http://www.opengis.org/sms" xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
    <!-- Top level elements. -->
    <xs:element name="StyleMetadata" type="tns:StyleOrSymbolMetadataType">
        <xs:annotation>
            <xs:documentation>Gives metadata about a style that is stored in an OGC
registry.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="SymbolMetadata" type="tns:StyleOrSymbolMetadataType">
        <xs:annotation>
            <xs:documentation>Gives metadata about a symbol that is stored in an OGC
registry.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <!-- Subordinate elements. -->
    <xs:element name="Name" type="xs:string">
        <xs:annotation>
            <xs:documentation>Gives a short, mnemonic name for the style or symbol to
which this metadata refers.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Description" type="xs:string">
        <xs:annotation>
            <xs:documentation>Gives a brief, free-text description of the style or symbol
to which this metadata refers.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Language" type="xs:language">
        <xs:annotation>
            <xs:documentation>Gives the two letter code for the language that this
metadata's free text is written in.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="Contact" type="tns:ContactType">
        <xs:annotation>
            <xs:documentation>Gives contact information about the author or maintainer of
this metadata.</xs:documentation>
        </xs:annotation>
    </xs:element>
```



```
    <xs:element name="ContactInfo" type="tns:ContactInfoType">
        <xs:annotation>
            <xs:documentation>Information about how to contact the maintainer or
publisher of this metadata.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <xs:element name="ApplicationSchemaInfo" type="tns:ApplicationSchemaInfoType">
        <xs:annotation>
            <xs:documentation>Indicates the schema for the style or symbol that this
metadata describes.</xs:documentation>
        </xs:annotation>
    </xs:element>
    <!-- Type definitions. -->
    <xs:complexType name="StyleOrSymbolMetadataType">
        <xs:sequence>
            <xs:element name="ContentURL" type="xs:anyURI">
                <xs:annotation>
                    <xs:documentation>This tag gives the URL of the style or symbol that
this metadata refers to.  The URL may point to any server anywhere, not just the server
where the style's existence is registered.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element ref="tns:Name" minOccurs="0"/>
            <xs:element ref="tns:Description" minOccurs="0"/>
            <xs:element ref="tns:Language" minOccurs="0"/>
            <xs:element ref="tns:Contact" minOccurs="0"/>
            <xs:element name="DateStamp" type="xs:dateTime" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Date and time when this metadata was last
modified.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element ref="tns:ApplicationSchemaInfo" minOccurs="0"/>
            <xs:element name="FeatureTypeList" type="tns:FeatureTypeListType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation>List of feature types that this style applies
to.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="SemanticTypeIdentifierList"
type="tns:SemanticTypeIdentifierListType" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>List of semantic types styled by this style or
symbol.  (See the SLD specification for a description of the meaning of "semantic
type".)</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="LibraryNameList" type="tns:LibraryNameListType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation>List of names of the symbol or style libraries that
an object belongs to.  This will have children whose text is "NIMA VPF" or "MIL-STD-2525"
or similar.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="ClassificationList" type="tns:ClassificationListType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Lists the classifications of this style or
symbol.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="FeaturePropertyList" type="tns:FeaturePropertyListType"
minOccurs="0">
```

```
                <xs:annotation>
                    <xs:documentation>Lists the names of the feature properties that this
symbol or style (or its constituent parts) use in their "Rule"
elements.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="SymbolTypeList" type="tns:SymbolTypeListType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Lists the types of symbols that this style (or
symbol) contains.  This list will contain strings such as "POINT", "LINE", and
"POLYGON".</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="StyleParameterList" type="tns:StyleParameterListType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Lists the "external" parameters that this style
depends on.  This may be used, for example, if a style has a dependency on the time of
day.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="KeywordList" type="tns:KeywordListType" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>List of arbitrary keywords that the author of the
style may wish to list in order to faciilitate searching.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ContactType">
        <xs:sequence>
            <xs:element name="IndividualName" type="xs:string" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Name(s) of the individual(s) who maintains or
published this metadata.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="OrganisationName" type="xs:string" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Name of the organisation that maintains or
published this metadata.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="PositionName" type="xs:string" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Position of the maintainer(s) of this
metadata.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element ref="tns:ContactInfo" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ContactInfoType">
        <xs:sequence>
            <xs:element name="PhoneNumber" type="xs:string" minOccurs="0"/>
            <xs:element name="Address" type="xs:string" minOccurs="0"/>
            <xs:element name="OnlineResource" type="xs:anyURI" minOccurs="0"/>
            <xs:element name="ContactInstructions" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ApplicationSchemaInfoType">
        <xs:sequence>
            <xs:element name="Name" type="xs:string" minOccurs="0">
                <xs:annotation>
                    <xs:documentation>Gives a human-readable name of the schema to which
this element refers.</xs:documentation>
                </xs:annotation>
```

```
            </xs:element>
            <xs:element name="SchemaLocation" type="xs:anyURI">
                <xs:annotation>
                    <xs:documentation>Gives a URL that can be used to retrieve the schema
for the style or symbol that this metadata refers to.</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="SchemaLanguage" type="xs:string">
                <xs:annotation>
                    <xs:documentation>Gives a string that indicates what language the
schema is written in.  It is expected that this will only ever be "DTD" or
"XMLSchema".</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="FeatureTypeListType">
        <xs:sequence>
            <xs:element name="FeatureType" type="xs:string" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="SemanticTypeIdentifierListType">
        <xs:sequence>
            <xs:element name="SemanticTypeIdentifier" type="xs:string" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="LibraryNameListType">
        <xs:sequence>
            <xs:element name="LibraryName" type="xs:string" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="ClassificationListType">
        <xs:sequence>
            <xs:element name="Classification" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:simpleContent>
                        <xs:extension base="xs:string">
                            <xs:attribute name="scheme" type="xs:string"
use="required"/>
                        </xs:extension>
                    </xs:simpleContent>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="FeaturePropertyListType">
        <xs:sequence>
            <xs:element name="FeatureProperty" type="xs:string" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="SymbolTypeListType">
        <xs:sequence>
            <xs:element name="SymbolType" minOccurs="0" maxOccurs="unbounded">
                <xs:annotation>
                    <xs:documentation>This should probably have an enumerated list of
valid values.</xs:documentation>
                </xs:annotation>
                <xs:simpleType>
                    <xs:restriction base="xs:string"/>
                </xs:simpleType>
            </xs:element>
```

```
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="StyleParameterListType">
        <xs:sequence>
            <xs:element name="StyleParameter" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="KeywordListType">
        <xs:sequence>
            <xs:element name="Keyword" type="xs:string" minOccurs="0"
maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

C.2 Style and Symbol Metadata Description (informative)

This section describes the style and symbol metadata schema by stepping through a fictitious metadata instance. The contents of this section are informative only.

First, there is the standard XML header:

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

Then the opening tag:

```
<StyleMetadata xmlns="http://www.opengis.org/sms">
```

When the metadata is describing a style object, the opening tag will be `StyleMetadata`. If the metadata describes a symbol, the opening tag will be `SymbolMetadata`. Both `Style`- and `SymbolMetadata` have the exact same structure.

All content of the `StyleMetadata` tag is optional except for the `ContentURL` which indicates where the style object can be retrieved. This gives a URL where the style (or symbol) described by this metadata can be retrieved:

```
<ContentURL>http://www.polexis.com/sms/repository.pl?REQUES
T=GetObject&amp;ID=1234567890</ContentURL>
```

Note that in this example, the URL is actually a request to a repository to retrieve an item based on its ID. This might happen if the style object were first inserted into a Web Object Service before being registered with a registry.

In order to identify style or symbol instances through the metadata, a human readable name is associated with a style or symbol instance.

```
<Name>VMAP0 Oceans</Name>
```

The name value is used by clients to construct a list of appropriate style or symbol names when multiple style or symbol metadata documents are returned from a registry query.

The next optional tag is Description. This content allows the creator of style or symbol metadata to provide users an informative description about the style or symbol instance in which the metadata refers.

<Description>NIMA VMAP0 Style for Oceans or Sea Boundaries</Description>

Next comes a tag which gives the two character code indicating the language used in any comments or descriptions contained in the metadata:

```
<Language>EN</Language>
```

The content of this tag must be a two character language code as defined by ISO 639-2. (The valid possibilities are also listed in Annex F of ISO 19115.)

Next is a tag that gives contact information for the author or maintainer of the metadata and style. This tag is a subset of the information present in the ISO 19115 data type "`CI_ResponsibleParty`".

```
<Contact>
    <IndividualName>Chris Dillard</IndividualName>
    <OrganisationName>Polexis, Inc.</OrganisationName>
    <PositionName>Software Engineer</PositionName>
    <ContactInfo>
        <PhoneNumber>(555) 555-5555</PhoneNumber>
        <Address>123 A St., Nowhere, CA 95000</Address>

<OnlineResource>http://www.polexis.com</OnlineResource>
        <ContactInstructions>
            Either call the given number or e-mail
            cdillard@polexis.com
        </ContactInstructions>
    </ContactInfo>
</Contact>
```

This is followed by a tag that gives the time when the style (or symbol) was last modified:

```
<DateStamp>2002-07-22T14:30:00-08:00</DateStamp>
```

Next is a tag whose content can list the schema for the style or symbol document described by this metadata:

```
<ApplicationSchemaInfo>
    <Name>SLD 0.7.3</Name>

<SchemaLocation>http://www.abc.def/ghi.xsd</SchemaLocation>
    <SchemaLanguage>XMLSchema</SchemaLanguage>
</ApplicationSchemaInfo>
```

The `Name` tag is descriptive, and used only to provide a human readable name for the schema that constrains the style. The content of the `SchemaLocation` tag must be an absolute URL that can be used to retrieve a document (in this case an XML schema instance). And the `SchemaLanguage` tag must contain either "DTD" or "XMLSchema". (Other tags may be introduced in the future to support other schema languages.)

Next is a tag containing a list of feature types:

```
<FeatureTypeList>
    <FeatureType>OCEANSEA_1M</FeatureType>
</FeatureTypeList>
```

Each `FeatureType` element names a data type that is used by the style or symbol or its constituent parts. In the above example, `OCEANSEA_1M` is an implicit reference to an application schema that defines the data type `OCEANSEA_1M`. There is currently not a way to explicitly reference the schema where this type is defined. This may be added in the (near) future.

Next is a tag containing a list of semantic type identifiers.

```
<SemanticTypeIdentifierList>
   <SemanticTypeIdentifier>nima:vmap0:bnd:oceansea
   </SemanticTypeIdentifier>
   <SemanticTypeIdentifier>generic:polygon
   </SemanticTypeIdentifier>
</SemanticTypeIdentifierList>
```

The SemanticTypeIdentifier element describes the suitable feature types in which a feature styles can be used and corresponds to the same SemanticTypeIdentifier element in the SLD specification. These values can be generic geometries such as generic:point, generic:line, generic:polygon, generic:text and generic:raster when a style can be used with many different feature types.

Next is a tag that contains a list of "library names". These names are human readable strings that describe a collection of styles or symbols to which a style or symbol belongs.

```
<LibraryNameList>
    <LibraryName>NIMA VPF</LibraryName>
</LibraryNameList>
```

The library name may serve as a way to implicitly group styles or symbols. The user may, for example, request all styles in the "NIMA VPF" library. However, there is no standard (yet) that defines the range of valid library names, so it is not clear if this will be usable in an interoperable manner.

In order to support the idea of a hierarchical classification scheme (as defined in ebRIM), there is a `ClassificationList` element that lists references to classifications in well-known schemes. For example:

```
<ClassificationList>
    <Classification scheme="FACC">
        A001
    </Classification>
    <Classification scheme="ApplicationDomain">
        Agriculture
    </Classification>
</ClassificationList>
```

This example snippet says that the style refers to a feature of type "A001" in the FACC classification scheme and that it should be used for styling agricultural features. It has not yet been decided exactly how the pair of strings for the classification scheme will refer to an ebRIM classification node, but it is presumed that an SMS will be able to use the data in a `Classification` node to uniquely identify an ebRIM classification object.

Next there is a `FeaturePropertyList` element that lists which GML properties of the feature will be used by the Rules which comprise the style or symbol.

```
<FeaturePropertyList>
    <FeatureProperty>NUMBER_OF_LANES</FeatureProperty>
</FeaturePropertyList>
```

Currently, this list is only descriptive since there is not a way to explicitly indicate the XPath and/or namespace that the given feature property refers to. As more research is done to determine the best way to reference a feature's properties, the nature of this element may change.

Next in the metadata is an element that lists the types of symbols that comprise the style or symbol to which the metadata refers.

```
<SymbolTypeList>
    <SymbolType>POINT</SymbolType>
    <SymbolType>TEXT</SymbolType>
</SymbolTypeList>
```

This example could describe a style containing two symbols, one that draws a point symbol and one that draws a text label.

The valid choices for the `SymbolType` element are `POINT`, `TEXT`, `LINE`, `POLYGON`, and `RASTER`. If the metadata is describing a symbol, then it must be that case that there is exactly one `SymbolType` child of `SymbolTypeList`.

It was recognized that some styles may depend on external parameters, such as the time of day or the brightness of the location containing a computer screen. In order to support this, the style and symbol metadata have the following element which lists the names of parameters to the style or symbol:

```
<StyleParameterList>
    <StyleParameter>TIME_OF_DAY</StyleParameter>
</StyleParameterList>
```

No mechanism exists (yet) to specify the data type that the given property is expected to contain. Also, it is not yet defined how the style or symbol elements (in SLD 0.7.3) can reference these properties. More work has to be done to clarify these points.

Next, there is a section in the metadata for arbitrary keywords that the user wishes to list:

```
<KeywordList>
        <Keyword>sample</Keyword>
        <Keyword>example</Keyword>
</KeywordList>
```

This is followed by the closing tag that ends the metadata document:

```
</StyleMetadata>
```

C.3 Unresolved Issues (informative)

The issues mentioned in this section C.1 are collected here for easier reference:

1. The `FeatureTypeList` tag may need a way to specify the schema of the feature types that are listed (although it is unclear how an SMS client would make use of this, other than to provide feedback to the human user).
2. The `LibraryNameList` tag may be too flexible to be useful in an interoperable manner.

3. It is unclear how the `ClassificationList` should reference ebRIM classification nodes. Should the classification scheme be a reference (perhaps a URL) to a scheme registered in an OGC registry? Should it just be a "well-known" name?
4. It has yet to be decided how the properties listed in `FeaturePropertyList` will reference the GML structure, if at all. Should this be an XPath expression? Should it convey any type information about the feature property?
5. The SLD specification does not yet provide a mechanism for styles and symbols to reference the external parameters listed in `StyleParameterList`. (This is not difficult to correct, however, and a couple of mechanisms have been considered.)

More issues, comments and suggestions (submitted by Milan Trninic, Galdos):

1. ***Specifying schema of feature types*** (Issue 1 above)

The way to specify schema can be the namespace: a namespace is enough to find the schema location (using a catalog).

2. ***Language specification***

Why not use xml:lang attribute instead of tns:Language element? The attribute is already there, we just need to import the xml namespace.

3. ***ID***

Is there a need for an ID of the Style/SymbolMetadata or is it enough to let the catalog component assign an ID?

4. ***FeatureList/FeatureProperty***

A style applies to a couple Feature / Property, or in some cases to the triplet Feature/GeometryProperty/Geometry. The feature and property names to which the style applies should probably be always specified together. We can use Xpath for this:

```
<Feature>app:Road/app:curbLine</Feature>
<Feature>app:Railway/app:centerLineOf</Feature>
```

rather than:

```
<FeatureTypeList>
   <FeatureType>app:Road</FeatureType>
   <FeatureType>app:Railway</FeatureType>
<FeatureTypeList>
<FeaturePropertyList>
```

```
    <FeatureProperty>app:curbLine</FeatureProperty>
    <FeatureProperty>gml:centerLineOf</FeatureProperty>
<FeaturePropertyList>
```

since the style obviously doesn't apply to:

```
app:Railway/app:curbLine
app:Road/gml:centerLineOf
```

5. FeatureType / FeatureProperty in SymbolMetadata

The information about features shouldn't be a part of SymbolMetadata. It is already a part of the style metadata – a style is an association (or a collection of associations) between feature types and symbols. A symbol is not reserved for a particular feature type (+ feature property).

6. Library

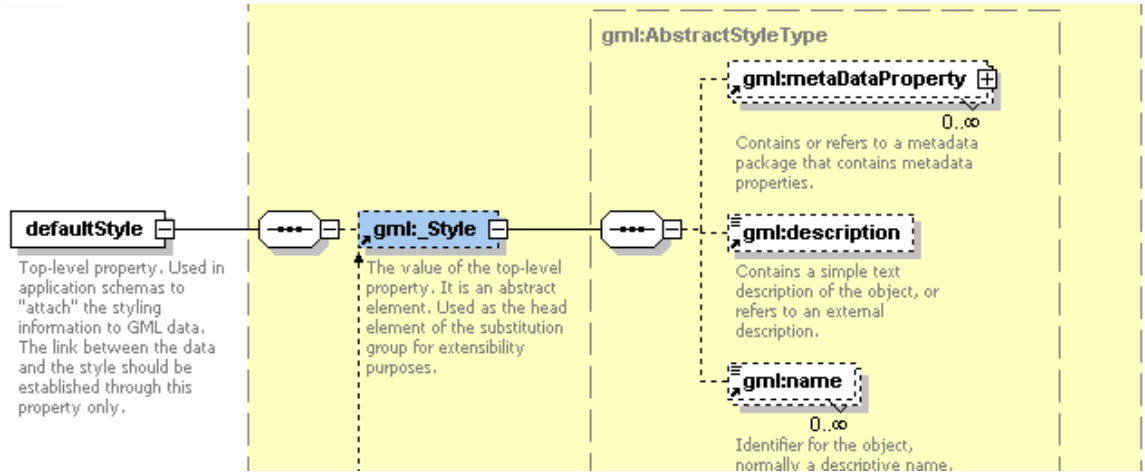
Do we specify library names in the style or do we make style an element in the library?

7. Do we have too much metadata?

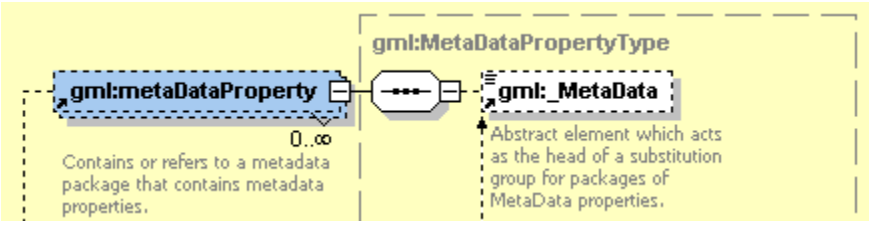
Besides the StyleMetadata fragment, metadata exists in the style itself (and overlaps to some extent), and we also have metadata infomration captured in the catalog classifications, relations and associations. We should maybe look into consolidating duplicate pieces. For example, the name and description exist both in the style (SLD) and StyleMetadata.

Metadata can be expressed as a part of the style itself. Important consideration here is whether this metadata is used and needed by the catalog component only or can it be useful in a more general scope.

Consider the way it is defined in the GML defaultStyle:



Metadata is attached to every symbol description. The metadataProperty contains the abstract Metadata object that can be supplemented by the real metadata object.



In other words, the StyleMetadata can be used inside the style instead as a separate fragment.

Related to this is the question of validating of the StyleMetadata fragment. Which component does it and when?

8. Use of StyleMetadata and SymbolMetadata

StyleMetadata and SymbolMetadata are defined as XML fragments separate from symbols and styles themselves. Perhaps there is a need to discuss their use - I am not quite sure to what extend people have an aggreement on this.

Catalog can preserve metadata fragments just as they are or can transform the information into catalog relations and objects. For example, catalog can use the metadata fragment as the extrinsic object or as the part of it. On the other hand it can use the metadata information to relations, associations etc, but not preserve the XML fragment itself. Of course it must implement recreation of the XML metadata fragments from the relations and extrinsic objects.

9. Relation to SLD discussion

Style metadata and symbol metadata should be reviewed again after the discussion on SLD change proposals take place.

C.3 Complete Metadata Examples (informative)

C.3.1 Style Metadata (HSWG Fire Station)

```
<?xml version="1.0" encoding="UTF-8"?>
<StyleMetadata xmlns="http://www.opengis.org/sms" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.org/sms http://www.opengis.org/sms/1.0.20/stylemetadata.xsd">
    <ContentURL>http://demo.cubewerx.com/sld/libraries/ers/styles/HSWG_Fire_Station_Feature_Style.xml</Con
tentURL>
    <Name>Fire Station</Name>
    <Description>A facility housing fire-fighting equipment and/or personnel</Description>
    <Contact>
        <IndividualName>Homeland Security Working Group (HSWG)</IndividualName>
        <OrganisationName>FGDC</OrganisationName>
        <ContactInfo>
            <OnlineResource>http://www.fgdc.gov/HSWG</OnlineResource>
        </ContactInfo>
    </Contact>
    <DateStamp>2004-04-07T09:30:47-05:00</DateStamp>
    <LibraryNameList>
        <LibraryName>HSWG</LibraryName>
    </LibraryNameList>
    <ClassificationList>
        <Classification
scheme="urn:csw:classificationscheme:hswg">urn:csw:classificationnode:hswg:firestation</Classification>
    </ClassificationList>
    <KeywordList>
        <Keyword>Operation</Keyword>
        <Keyword>Fire Suppression</Keyword>
        <Keyword>Fire Station</Keyword>
    </KeywordList>
</StyleMetadata>
```

C.3.2. Symbol Metadata (HSWG Fire Hydrant)

```
<?xml version="1.0" encoding="UTF-8"?>
<SymbolMetadata xmlns="http://www.opengis.org/sms" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.org/sms http://www.opengis.org/sms/1.0.20/stylemetadata.xsd">
    <ContentURL>http://demo.cubewerx.com/sld/libraries/ers/graphics/SVG_ERS_Symbols/CGM_Infrastructures_
S1/HSWG_Fire_Hydrant.svg</ContentURL>
    <Name>Fire Hydrant</Name>
    <Description>A discharge pipe with a valve and spout from which water may be drawn from a water main in
sufficient volume and at sufficient pressure for firefighting purposes </Description>
    <Contact>
        <IndividualName>HSWG</IndividualName>
        <OrganisationName>FGDC</OrganisationName>
        <ContactInfo>
            <OnlineResource>http://www.fgdc.gov/HSWG</OnlineResource>
        </ContactInfo>
    </Contact>
    <DateStamp>2004-04-07T09:30:47-05:00</DateStamp>
```

```
    <LibraryNameList>
        <LibraryName>HSWG</LibraryName>
    </LibraryNameList>
    <ClassificationList>
        <Classification scheme="HSWG">Fire Hydrant</Classification>
        <Classification scheme="String">String</Classification>
    </ClassificationList>
    <StyleParameterList>
        <StyleParameter>String</StyleParameter>
    </StyleParameterList>
    <KeywordList>
        <Keyword>Fire</Keyword>
        <Keyword>Suppression</Keyword>
        <Keyword>Operation</Keyword>
    </KeywordList>
</SymbolMetadata>
```

C.3 Style and Symbol Metadata Form (informative)

The form listed here was designed to assist subject matter experts, without knowledge of XML or XML Schema, to capture metadata for styles and symbols. The metadata content captured in the form, once encoded as XML, is used to catalog these resources so they may be published online for broader use and subsequently discovered and accessed.

1. **Name** (A short, mnemonic name for the style or symbol to which this metadata refers.)	*Mandatory.*
2. **Description** (A brief, free-text description of the style or symbol to which this metadata refers.)	*Mandatory.*
3. **Content URL** (The URL of the style or symbol instance that this metadata refers to. The URL may point to any server anywhere, not just the server where the style's existence is registered.)	*Optional.*
4. **Language** (The two letter code for the language that this metadata's free text is written in.)	*Mandatory. E.g., "en" for English.*
5. **Date** (Date and time when this metadata was last modified	*Mandatory.*

6. **Contact** (Contact information about the author or maintainer of this metadata.)	
Individual Name (Name(s) of individual(s) who maintain or published this metadata.)	*Optional.*
Organization Name (Name of organization that maintains or published this metadata.)	*Mandatory.*
Position Name (Role of the maintainer(s) of this metadata.)	*Optional.*
Phone Number	*Optional.*
Address	*Optional.*
Online Resource (A URL that can be used to reference a web page, email address, online document etc.)	*Optional. E.g., http://www.fgdc.gov/HSWG*
Contact Instructions	*Optional.*

7. **Keyword List** (List of arbitrary keywords that the author of the style may wish to list in order to facilitate searching.)	
Keyword	*Optional.*
Keyword	**May be more than one**
Keyword	…

8. **Symbol Type List** (Lists the types of symbols that this style (or symbol) contains. This list will contain strings such as "POINT", "LINE", and "POLYGON".)	
Symbol Type Identifier	**Mandatory. (“POINT”, “LINE” or “POLYGON”)**
Symbol Type Identifier	*May be more than one.*
Symbol Type Identifier	…

9. **Library Name List** (List of names of the symbol or style libraries that the object to which this metadata refers is a part of. This list will contain strings such as "HSWG", "GeoSym", etc.)	
Library Name	*Optional. e.g., "FGDC HSWG v0.1"*
Library Name	*May be more than one.*
Library Name	…

10. **Classification Scheme List** (List of symbol/style classification schemes for this style or symbol.)	
Classification Scheme Identifier (Identifies a classification scheme used to classify this symbol/style)	*Optional. e.g., "urn:x-fgdc:hswg:0.1" This identifies the "FGDC HSWG v0.1" symbol classification scheme. See discussion of Classification Scheme below.*
Classification Scheme Identifier	*May be more than one.*
Classification Scheme Identifier	…

11. **Feature Type List** (List of known feature type/class identifiers that can be styled by this style or symbol.)	
Feature Type Identifier (Identifies the semantic, structure (property names and property types) and behavior of Feature instances as defined with an Application Schema.)	*Optional. e.g., "urn:x-digest:facc:2.1:A:AL:AL201" is an identifier for "Historic Site/Point of Interest" feature types defined by the DIGEST/FACC feature classification scheme. Note that the feature type identifier "AL201" includes (is scoped by) the classification scheme identifier "urn:x-digest:facc:2.1". This classification scheme identifier identifies the "DIGEST/FACC v2.1" classification scheme. The Digital Geographic Information Exchange Standard (DIGEST) Feature and Attribute Coding Catalogue (FACC) v2.1. FACC provides a means for encoding real-world entities or objects and concepts.*
Feature Type Identifier	*May be more than one.*

Feature Type Identifier	…

12. **Application Schema Information** (Indicates the schema for the style or symbol that this metadata describes.)	
Name (A human-readable name of the schema to which this element refers.)	*Optional. E.g., "SLD", "SVG" or "CGM"*
Schema Location (A URL that can be used to retrieve the schema for the style or symbol that this metadata refers to.)	*Optional. E.g., http://www.w3.org/2000/svg*
Schema Language (A string that indicates what language the schema is written in. It is expected that this will only ever be "DTD" or "XMLSchema".)	*Optional.*

Annex D –Example Application Schema, Styles, Symbols and Classifications (informative)

This annex lists examples of the following used for development, testing and demonstration in the EMS-1 project:

1. A prototype GML Application Schema for Infrastructure and Hazard Features
2. An SLD document for Infrastructure and Hazard Features (Fire Station features)
3. A Feature Style document for the Fire Station feature type.
4. SVG encoding of HSWG Fire Station Symbol (referenced by Feature Style)
5. EMS Symbol, Feature and Service Classification Schemes

D.1 GML Application Schema for Infrastructure and Hazard Features.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xs:schema targetNamespace="http://demo.cubewerx.com/vancouver"
   xmlns:van="http://demo.cubewerx.com/vancouver"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   xmlns:gml="http://www.opengis.net/gml"
   xmlns:cwmeta="http://www.cubewerx.com/cwmeta" elementFormDefault="qualified">
   <xs:import namespace="http://www.opengis.net/gml"
      schemaLocation="http://schemas.cubewerx.com/schemas/gml/2.1.2/feature.xsd"/>
   <!-- ========================================================= -->
   <!-- Feature type definitions ...                              -->
   <!-- ========================================================= -->
   <!-- === AIRPORT ============================================= -->
   <xs:element name="Airport" type="van:Airport_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="Airport_Type">
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element ref="gml:position" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === BRIDGE ============================================== -->
   <xs:element name="Bridge" type="van:Bridge_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="Bridge_Type">
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element ref="gml:centerLineOf" minOccurs="1" maxOccurs="1"/>
```

```
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === COMMUNICATIONS ==================================== -->
   <xs:element name="Communications" type="van:Communications_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="Communications_Type">
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element ref="gml:extentOf" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === EMS1_VAN_EQE ====================================== -->
   <xs:element name="EMS1_VAN_EQE" type="van:EMS1_VAN_EQE_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="EMS1_VAN_EQE_Type">
      <xs:annotation>
         <xs:documentation>
            <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
               <Title>Earthquake Events</Title>
            </cwmeta:Metadata>
         </xs:documentation>
      </xs:annotation>
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Point Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
               <xs:element name="AREA" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:decimal">
                        <xs:totalDigits value="15"/>
                        <xs:fractionDigits value="3"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="PERIMETER" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:decimal">
                        <xs:totalDigits value="15"/>
                        <xs:fractionDigits value="3"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="QUKSIGX020" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="11"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
```

```
<xs:element name="STATE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="Y" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:decimal">
         <xs:totalDigits value="11"/>
         <xs:fractionDigits value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="X" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:decimal">
         <xs:totalDigits value="12"/>
         <xs:fractionDigits value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="DEPTH" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="DEATHS" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="5"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="DAMAGE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="11"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="MAG" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:decimal">
         <xs:totalDigits value="8"/>
         <xs:fractionDigits value="2"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="MMI" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="LOCATION" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="63"/>
      </xs:restriction>
   </xs:simpleType>
```

```
      </xs:element>
      <xs:element name="YEAR" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:integer">
               <xs:totalDigits value="5"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="MONTH" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:integer">
               <xs:totalDigits value="3"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="DAY" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:integer">
               <xs:totalDigits value="3"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="HOUR" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:integer">
               <xs:totalDigits value="3"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="MINUTE" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:integer">
               <xs:totalDigits value="3"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="SECOND" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:decimal">
               <xs:totalDigits value="8"/>
               <xs:fractionDigits value="2"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="TEXTFILE" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:string">
               <xs:maxLength value="60"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="MAPS" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:string">
               <xs:maxLength value="60"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
      <xs:element name="DAMAGE___M" minOccurs="0" maxOccurs="1">
         <xs:simpleType>
            <xs:restriction base="xs:decimal">
               <xs:totalDigits value="20"/>
               <xs:fractionDigits value="4"/>
            </xs:restriction>
         </xs:simpleType>
      </xs:element>
   </xs:sequence>
</xs:extension>
```

```
      </xs:complexContent>
   </xs:complexType>

   <!-- === FIRESTATION ======================================== -->
   <xs:element name="FireStation" type="van:FireStation_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="FireStation_Type">
      <xs:complexContent>

         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element ref="gml:position" minOccurs="1" maxOccurs="1"/>
               <xs:element name="OP_LEVEL" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="1"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
            </xs:sequence>

         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === FIRE_SHA =========================================== -->
   <xs:element name="FIRE_SHA" type="van:FIRE_SHA_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="FIRE_SHA_Type">
      <xs:annotation>
         <xs:documentation>
            <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
               <Title>Fire Stations</Title>
            </cwmeta:Metadata>
         </xs:documentation>
      </xs:annotation>
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Point Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
               <xs:element name="OBJECTID" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="ENTITY" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="16"/>
                     </xs:restriction>
                  </xs:simpleType>
```

```
</xs:element>
<xs:element name="LAYER" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="254"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="LEVELX" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="COLOR" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="LINETYPE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="254"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="TEXTX" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="254"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="GRID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="GRID_ID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_X" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_Y" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_NA" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
```

```
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_DE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ANNO_TEXT" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="100"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_TY" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_CL" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="SOURCE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="OPERATOR_I" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="24"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="UPDATED_DA" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="8"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="PROGRESS" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="24"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="MD_UID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
```

```
            </xs:element>
            <xs:element name="CO_NUM" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:integer">
                     <xs:totalDigits value="4"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="STATE" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:maxLength value="25"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="X_COORD" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:decimal">
                     <xs:totalDigits value="30"/>
                     <xs:fractionDigits value="11"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="Y_COORD" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:decimal">
                     <xs:totalDigits value="30"/>
                     <xs:fractionDigits value="11"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="SHAUID" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:integer">
                     <xs:totalDigits value="9"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<!-- === HOSPITAL ========================================== -->
<xs:element name="Hospital" type="van:Hospital_Type"
   substitutionGroup="gml:_Feature"/>
<xs:complexType name="Hospital_Type">
   <xs:complexContent>
      <xs:extension base="gml:AbstractFeatureType">
         <xs:sequence>
            <xs:element ref="gml:position" minOccurs="1" maxOccurs="1"/>
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<!-- === HOSPITAL_SHA ====================================== -->
<xs:element name="HOSPITAL_SHA" type="van:HOSPITAL_SHA_Type"
   substitutionGroup="gml:_Feature"/>
<xs:complexType name="HOSPITAL_SHA_Type">
   <xs:annotation>
      <xs:documentation>
         <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
            <Title>Hospitals</Title>
         </cwmeta:Metadata>
      </xs:documentation>
   </xs:annotation>
   <xs:complexContent>
```

```
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Point Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
               <xs:element name="OBJECTID" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="ENTITY" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="16"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LAYER" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="254"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LEVELX" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="COLOR" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LINETYPE" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="254"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="TEXTX" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="254"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="GRID" minOccurs="0" maxOccurs="1">
```

```
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="GRID_ID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_X" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_Y" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_NA" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_DE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ANNO_TEXT" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="100"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_TY" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_CL" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="SOURCE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
```

```
</xs:element>
<xs:element name="OPERATOR_I" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="24"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="UPDATED_DA" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="8"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="PROGRESS" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="24"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="MD_UID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="CO_NUM" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="STATE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="25"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="X_COORD" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:decimal">
         <xs:totalDigits value="30"/>
         <xs:fractionDigits value="11"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="Y_COORD" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:decimal">
         <xs:totalDigits value="30"/>
         <xs:fractionDigits value="11"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="SHAUID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
```

```
            </xs:element>
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<!-- === POLICESTATION ====================================== -->
<xs:element name="PoliceStation" type="van:PoliceStation_Type"
   substitutionGroup="gml:_Feature"/>
<xs:complexType name="PoliceStation_Type">
   <xs:complexContent>
      <xs:extension base="gml:AbstractFeatureType">
         <xs:sequence>
            <xs:element ref="gml:position" minOccurs="1" maxOccurs="1"/>
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<!-- === POLICE_SHA ========================================= -->
<xs:element name="POLICE_SHA" type="van:POLICE_SHA_Type"
   substitutionGroup="gml:_Feature"/>
<xs:complexType name="POLICE_SHA_Type">
   <xs:annotation>
      <xs:documentation>
         <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
            <Title>Police Stations</Title>
         </cwmeta:Metadata>
      </xs:documentation>
   </xs:annotation>
   <xs:complexContent>
      <xs:extension base="gml:AbstractFeatureType">
         <xs:sequence>
            <xs:element name="GEOMETRY" type="gml:PointPropertyType"
               minOccurs="1" maxOccurs="1">
               <xs:annotation>
                  <xs:documentation>
                     <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                        xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                        <Title>Point Geometry</Title>
                     </Metadata>
                  </xs:documentation>
               </xs:annotation>
            </xs:element>
            <xs:element name="OBJECTID" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:integer">
                     <xs:totalDigits value="9"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="ENTITY" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:maxLength value="16"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="LAYER" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:maxLength value="254"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="LEVELX" minOccurs="0" maxOccurs="1">
```

```
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="COLOR" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="LINETYPE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="254"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="TEXTX" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="254"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="GRID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="GRID_ID" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_X" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_Y" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_NA" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_DE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
```

```
            <xs:maxLength value="50"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="ANNO_TEXT" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:string">
            <xs:maxLength value="100"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="FEATURE_TY" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:integer">
            <xs:totalDigits value="4"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="FEATURE_CL" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:integer">
            <xs:totalDigits value="4"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="SOURCE" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:integer">
            <xs:totalDigits value="4"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="OPERATOR_I" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:string">
            <xs:maxLength value="24"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="UPDATED_DA" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:integer">
            <xs:totalDigits value="8"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="PROGRESS" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:string">
            <xs:maxLength value="24"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="MD_UID" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:string">
            <xs:maxLength value="50"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="CO_NUM" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
         <xs:restriction base="xs:integer">
            <xs:totalDigits value="4"/>
         </xs:restriction>
      </xs:simpleType>
   </xs:element>
   <xs:element name="STATE" minOccurs="0" maxOccurs="1">
```

```
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="25"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="X_COORD" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:decimal">
                        <xs:totalDigits value="30"/>
                        <xs:fractionDigits value="11"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="Y_COORD" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:decimal">
                        <xs:totalDigits value="30"/>
                        <xs:fractionDigits value="11"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="SHAUID" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === SCHOOL ================================================ -->
   <xs:element name="School" type="van:School_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="School_Type">
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element ref="gml:position" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === SCHOOL_SHA ============================================ -->
   <xs:element name="SCHOOL_SHA" type="van:SCHOOL_SHA_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="SCHOOL_SHA_Type">
      <xs:annotation>
         <xs:documentation>
            <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
               <Title>Schools</Title>
            </cwmeta:Metadata>
         </xs:documentation>
      </xs:annotation>
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
```

```
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Point Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
               <xs:element name="OBJECTID" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="ENTITY" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="16"/>

                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LAYER" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="254"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LEVELX" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="COLOR" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="9"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LINETYPE" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="254"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="TEXTX" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="254"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="GRID" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="4"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="GRID_ID" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
```

```
      <xs:restriction base="xs:string">
         <xs:maxLength value="3"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_X" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ACT_LOC_Y" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="9"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_NA" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_DE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="50"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="ANNO_TEXT" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="100"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_TY" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="FEATURE_CL" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="SOURCE" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:integer">
         <xs:totalDigits value="4"/>
      </xs:restriction>
   </xs:simpleType>
</xs:element>
<xs:element name="OPERATOR_I" minOccurs="0" maxOccurs="1">
   <xs:simpleType>
      <xs:restriction base="xs:string">
         <xs:maxLength value="24"/>
```

```
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="UPDATED_DA" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:integer">
                     <xs:totalDigits value="8"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="PROGRESS" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:maxLength value="24"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="MD_UID" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:maxLength value="50"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="CO_NUM" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:integer">

                     <xs:totalDigits value="4"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="STATE" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:string">
                     <xs:maxLength value="25"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="X_COORD" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:decimal">
                     <xs:totalDigits value="30"/>
                     <xs:fractionDigits value="11"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="Y_COORD" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:decimal">
                     <xs:totalDigits value="30"/>
                     <xs:fractionDigits value="11"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
            <xs:element name="SHAUID" minOccurs="0" maxOccurs="1">
               <xs:simpleType>
                  <xs:restriction base="xs:integer">
                     <xs:totalDigits value="9"/>
                  </xs:restriction>
               </xs:simpleType>
            </xs:element>
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>

<!-- === SHAROADS ========================================== -->
```

```
   <xs:element name="SHAROADS" type="van:SHAROADS_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="SHAROADS_Type">
      <xs:annotation>
         <xs:documentation>
            <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
               <Title>Maryland Roads</Title>

            </cwmeta:Metadata>
         </xs:documentation>
      </xs:annotation>
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Line Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
               <xs:element name="ENTITY" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="14"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="LAYER" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="32"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="MD_LEVEL" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:integer">
                        <xs:totalDigits value="11"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
               <xs:element name="DESCRIPTIO" minOccurs="0" maxOccurs="1">
                  <xs:simpleType>
                     <xs:restriction base="xs:string">
                        <xs:maxLength value="100"/>
                     </xs:restriction>
                  </xs:simpleType>
               </xs:element>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === SHAROADS_LP ======================================== -->
   <xs:element name="SHAROADS_LP" type="van:SHAROADS_LP_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="SHAROADS_LP_Type">
      <xs:annotation>
```

```
            <xs:documentation>
               <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
                  <Title>La Plata Area Roads</Title>
               </cwmeta:Metadata>
            </xs:documentation>
         </xs:annotation>
         <xs:complexContent>
            <xs:extension base="gml:AbstractFeatureType">
               <xs:sequence>
                  <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                     minOccurs="1" maxOccurs="1">
                     <xs:annotation>
                        <xs:documentation>
                           <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                              xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                              <Title>Line Geometry</Title>
                           </Metadata>
                        </xs:documentation>
                     </xs:annotation>
                  </xs:element>
                  <xs:element name="ENTITY" minOccurs="0" maxOccurs="1">
                     <xs:simpleType>
                        <xs:restriction base="xs:string">
                           <xs:maxLength value="14"/>
                        </xs:restriction>
                     </xs:simpleType>
                  </xs:element>
                  <xs:element name="LAYER" minOccurs="0" maxOccurs="1">
                     <xs:simpleType>
                        <xs:restriction base="xs:string">
                           <xs:maxLength value="32"/>
                        </xs:restriction>
                     </xs:simpleType>
                  </xs:element>
                  <xs:element name="MD_LEVEL" minOccurs="0" maxOccurs="1">
                     <xs:simpleType>
                        <xs:restriction base="xs:integer">
                           <xs:totalDigits value="11"/>
                        </xs:restriction>
                     </xs:simpleType>
                  </xs:element>
                  <xs:element name="DESCRIPTIO" minOccurs="0" maxOccurs="1">
                     <xs:simpleType>
                        <xs:restriction base="xs:string">
                           <xs:maxLength value="100"/>
                        </xs:restriction>
                     </xs:simpleType>
                  </xs:element>
               </xs:sequence>
            </xs:extension>
         </xs:complexContent>
      </xs:complexType>

      <!-- === TORNADO_TRACK ====================================== -->
      <xs:element name="TORNADO_TRACK" type="van:TORNADO_TRACK_Type"
         substitutionGroup="gml:_Feature"/>
      <xs:complexType name="TORNADO_TRACK_Type">
         <xs:annotation>
            <xs:documentation>
               <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
                  <Title>La Plata Tornado Track</Title>
               </cwmeta:Metadata>
            </xs:documentation>
         </xs:annotation>
         <xs:complexContent>
            <xs:extension base="gml:AbstractFeatureType">
```

```
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Point Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === TORNADO_TRACK2 ==================================== -->
   <xs:element name="TORNADO_TRACK2" type="van:TORNADO_TRACK2_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="TORNADO_TRACK2_Type">
      <xs:annotation>
         <xs:documentation>
            <cwmeta:Metadata xmlns="http://www.cubewerx.com/cwmeta">
               <Title>La Plata Tornado Track</Title>
            </cwmeta:Metadata>
         </xs:documentation>
      </xs:annotation>
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>
               <xs:element name="GEOMETRY" type="gml:PointPropertyType"
                  minOccurs="1" maxOccurs="1">
                  <xs:annotation>
                     <xs:documentation>
                        <Metadata xmlns="http://www.cubewerx.com/cwmeta"
                           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="http://www.cubewerx.com/cwmeta
http://schemas.cubewerx.com/schemas/cwmeta/1.0.0/cwmeta.xsd">
                           <Title>Point Geometry</Title>
                        </Metadata>
                     </xs:documentation>
                  </xs:annotation>
               </xs:element>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>

   <!-- === TRANSMISSIONLINE ================================== -->
   <xs:element name="TransmissionLine" type="van:TransmissionLine_Type"
      substitutionGroup="gml:_Feature"/>
   <xs:complexType name="TransmissionLine_Type">
      <xs:complexContent>
         <xs:extension base="gml:AbstractFeatureType">
            <xs:sequence>

               <xs:element ref="gml:centerLineOf" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
```

```
<!-- === TRANSMISSIONTOWER ================================== -->
<xs:element name="TransmissionTower" type="van:TransmissionTower_Type"
   substitutionGroup="gml:_Feature"/>
<xs:complexType name="TransmissionTower_Type">
   <xs:complexContent>
      <xs:extension base="gml:AbstractFeatureType">
         <xs:sequence>
            <xs:element ref="gml:extentOf" minOccurs="1" maxOccurs="1"/>
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>
```

D.2 SLD Document (styles and symbols for FIRE_SHA features).

```
<?xml version="1.0" encoding="UTF-8"?>
<StyledLayerDescriptor version="1.0.20">
   <NamedLayer>
      <RemoteOWS>
         <Service>WFS</Service>
         <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple"
xlink:href="http://demo.cubewerx.com/ems1/cubeserv/cubeserv.cgi?SERVICE=wfs&amp;DATASTORE
=EMS1"/>
      </RemoteOWS>
      <LayerFeatureConstraints>
         <FeatureTypeConstraint>
            <FeatureTypeName>van:FIRE_SHA</FeatureTypeName>
         </FeatureTypeConstraint>
      </LayerFeatureConstraints>
      <UserStyle>
         <FeatureStyle>
            <Name>Maryland Fire Stations</Name>
            <Description>
               <Abstract>Fire stations in the Maryland area.</Abstract>
            </Description>
            <Rule>
               <Name>Fire Station (PROGRESS = 1)</Name>
               <Filter>
                  <PropertyIsEqualTo>
                     <PropertyName>PROGRESS</PropertyName>
                     <Literal>1</Literal>
                  </PropertyIsEqualTo>
               </Filter>
               <PointSymbol>
                  <Name>Fire Station</Name>
                  <Graphic>
                     <Size>
                        <Literal>24</Literal>
                     </Size>
                     <ExternalGraphic>
                        <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
id=cid:32FF0DAF-7C5C-E545-D9BF-A76EE90FED57"/>
                        <Format>image/svg</Format>
                     </ExternalGraphic>
                  </Graphic>
               </PointSymbol>
            </Rule>
            <Rule>
               <Name>Fire Station (PROGRESS = 2)</Name>
               <Filter>
                  <PropertyIsEqualTo>
                     <PropertyName>PROGRESS</PropertyName>
                     <Literal>2</Literal>
                  </PropertyIsEqualTo>
               </Filter>
               <PointSymbol>
                  <Name>Fire Station</Name>
                  <Graphic>
                     <Size>
                        <Literal>24</Literal>
                     </Size>
```

```
                     <ExternalGraphic>
                        <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
id=cid:9BE79A87-4366-82DE-A876-0DD5FAF6FD36"/>
                        <Format>image/svg</Format>
                     </ExternalGraphic>
                  </Graphic>
               </PointSymbol>
            </Rule>
            <Rule>
               <Name>Fire Station (PROGRESS = 3)</Name>
               <Filter>
                  <PropertyIsEqualTo>
                     <PropertyName>PROGRESS</PropertyName>
                     <Literal>3</Literal>
                  </PropertyIsEqualTo>
               </Filter>
               <PointSymbol>
                  <Name>Fire Station</Name>
                  <Graphic>
                     <Size>
                        <Literal>24</Literal>
                     </Size>
                     <ExternalGraphic>
                        <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
id=cid:7B2846B4-30C2-FC1F-3032-67E99C025BC4"/>
                        <Format>image/svg</Format>
                     </ExternalGraphic>
                  </Graphic>
               </PointSymbol>
            </Rule>
            <Rule>
               <Name>Fire Station (PROGRESS = 4)</Name>
               <Filter>
                  <PropertyIsEqualTo>
                     <PropertyName>PROGRESS</PropertyName>
                     <Literal>4</Literal>
                  </PropertyIsEqualTo>
               </Filter>
               <PointSymbol>
                  <Name>Fire Station</Name>
                  <Graphic>
                     <Size>
                        <Literal>24</Literal>
                     </Size>
                     <ExternalGraphic>
                        <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
id=cid:C90B52F4-1D26-7048-D8F5-F63230144F36"/>
                        <Format>image/svg</Format>
                     </ExternalGraphic>
                  </Graphic>
               </PointSymbol>
            </Rule>
         </FeatureStyle>
      </UserStyle>
   </NamedLayer>
</StyledLayerDescriptor>
```

D.3 Fire Station “Feature Style” document from style repository

```
<FeatureStyle>
   <Name>Fire Stations</Name>
   <Description>
      <Abstract>Fire stations in the Maryland area</Abstract>
   </Description>
   <Rule>
      <Name>PROGRESS = 4</Name>
      <LegendGraphic>
         <Graphic>
            <ExternalGraphic>

   <OnlineResource>http://wes.compusult.net/weslet/SMSClient.direct/legendGraphics/legend
1512918959.png</OnlineResource>
               <Format>image/png</Format>
            </ExternalGraphic>
         </Graphic>
      </LegendGraphic>
      <Filter>
         <PropertyIsEqualTo>
            <PropertyName>PROGRESS</PropertyName>
            <Literal>4</Literal>
         </PropertyIsEqualTo>
      </Filter>
      <PointSymbol>
         <Name>Destroyed</Name>
         <Geometry>
            <PropertyName>GEOMETRY</PropertyName>
         </Geometry>
         <Graphic>
            <Size>
               <Literal>30</Literal>
            </Size>
            <ExternalGraphic>
               <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
amp;id=cid:02C907C6-A75F-52E8-4C93-EC6497C7D3E0"/>
               <Format>image/svg</Format>
            </ExternalGraphic>
         </Graphic>
      </PointSymbol>
   </Rule>
   <Rule>
      <Name>PROGRESS = 1</Name>
      <LegendGraphic>
         <Graphic>
            <ExternalGraphic>

   <OnlineResource>http://wes.compusult.net/weslet/SMSClient.direct/legendGraphics/legend
1512918959.png</OnlineResource>
               <Format>image/png</Format>
            </ExternalGraphic>
         </Graphic>
      </LegendGraphic>
      <Filter>
         <PropertyIsEqualTo>
            <PropertyName>PROGRESS</PropertyName>
            <Literal>1</Literal>
         </PropertyIsEqualTo>
      </Filter>
      <PointSymbol>
         <Name>Operational</Name>
```

```
         <Geometry>
            <PropertyName>GEOMETRY</PropertyName>
         </Geometry>
         <Graphic>
            <Size>
               <Literal>30</Literal>
            </Size>
            <ExternalGraphic>
               <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
amp;id=cid:FD2E5942-CE10-ED25-0BEB-8CFA7C7030C6"/>
               <Format>image/svg</Format>
            </ExternalGraphic>
         </Graphic>
      </PointSymbol>
   </Rule>
   <Rule>
      <Name>PROGRESS = 2</Name>
      <LegendGraphic>
         <Graphic>
            <ExternalGraphic>

   <OnlineResource>http://wes.compusult.net/weslet/SMSClient.direct/legendGraphics/legend
1512918959.png</OnlineResource>
               <Format>image/png</Format>
            </ExternalGraphic>
         </Graphic>
      </LegendGraphic>
      <Filter>
         <PropertyIsEqualTo>
            <PropertyName>PROGRESS</PropertyName>
            <Literal>2</Literal>
         </PropertyIsEqualTo>
      </Filter>
      <PointSymbol>
         <Name>Closed</Name>
         <Geometry>
            <PropertyName>GEOMETRY</PropertyName>
         </Geometry>
         <Graphic>
            <Size>
               <Literal>30</Literal>
            </Size>
            <ExternalGraphic>
               <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
amp;id=cid:76C57D48-BA13-C7A8-36AE-F5F1EFE9A1F5"/>
               <Format>image/svg</Format>
            </ExternalGraphic>
         </Graphic>
      </PointSymbol>
   </Rule>
   <Rule>
      <Name>PROGRESS = 3</Name>
      <LegendGraphic>
         <Graphic>
            <ExternalGraphic>

   <OnlineResource>http://wes.compusult.net/weslet/SMSClient.direct/legendGraphics/legend
1512918959.png</OnlineResource>
               <Format>image/png</Format>
            </ExternalGraphic>
         </Graphic>
      </LegendGraphic>
      <Filter>
         <PropertyIsEqualTo>
```

```
            <PropertyName>PROGRESS</PropertyName>
            <Literal>3</Literal>
         </PropertyIsEqualTo>
      </Filter>
      <PointSymbol>
         <Name>Damaged</Name>
         <Geometry>
            <PropertyName>GEOMETRY</PropertyName>
         </Geometry>
         <Graphic>
            <Size>
               <Literal>30</Literal>
            </Size>
            <ExternalGraphic>
               <onLineResource xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:type="simple"
xlink:href="http://wes.compusult.net/serviceManagerCSW/csw?request=GetRepositoryItem&amp;
amp;id=cid:5643A481-DF2A-4B58-5459-814DA1FE76A6"/>
               <Format>image/svg</Format>
            </ExternalGraphic>
         </Graphic>
      </PointSymbol>
   </Rule>
</FeatureStyle>
```

D.4 HSWG “Operational Fire Station Symbol” (SVG-encoded) from symbol repository.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!-- Generator: Adobe Illustrator 9.0, SVG Export Plug-In  -->
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20000303 Stylable//EN"
"http://www.w3.org/TR/2000/03/WD-SVG-20000303/DTD/svg-20000303-stylable.dtd" [
   <!ENTITY st0 "fill-rule:nonzero;clip-rule:nonzero;fill:none;stroke:#000000;stroke-
miterlimit:4;">
   <!ENTITY st1 "fill:#FFFFFF;">
   <!ENTITY st2 "fill:#000000;">
   <!ENTITY st3 "fill-rule:evenodd;clip-rule:evenodd;stroke:none;">
]>
<svg width="39.887pt" height="39.887pt" viewBox="0 0 39.887 39.887" xml:space="preserve">
   <g id="Layer_x0020_1" style="&st0;">
      <g style="&st3;">
         <path style="&st1;" d="M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624c0-
10.285,8.338-18.624,18.624-18.624c10.286,0,18.624,8.339,18.624,18.624c0,10.286-
8.338,18.624-18.624,18.624 M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624
            c0-10.285,8.338-18.624,18.624-
18.624c10.286,0,18.624,8.339,18.624,18.624c0,10.286-8.338,18.624-18.624,18.624
M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624c0-10.285,8.338-18.624,18.624-
18.624c10.286,0,18.624,8.339,18.624,18.624
            c0,10.286-8.338,18.624-18.624,18.624"/>
         <path d="M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624"/>
         <path d="M1.319,19.943c0-10.285,8.338-18.624,18.624-18.624"/>
         <path d="M19.943,1.319c10.286,0,18.624,8.339,18.624,18.624"/>
         <path d="M38.567,19.943c0,10.286-8.338,18.624-18.624,18.624"/>
         <path d="M19.943,38.567"/>
      </g>
      <g style="&st3;">
         <path style="&st2;" d="M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624c0-
10.285,8.338-18.624,18.624-18.624c10.286,0,18.624,8.339,18.624,18.624c0,10.286-
8.338,18.624-18.624,18.624 M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624
            c0-10.285,8.338-18.624,18.624-
18.624c10.286,0,18.624,8.339,18.624,18.624c0,10.286-8.338,18.624-18.624,18.624
```

```
M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624c0-10.285,8.338-18.624,18.624-
18.624c10.286,0,18.624,8.339,18.624,18.624
            c0,10.286-8.338,18.624-18.624,18.624
M39.887,19.943C39.887,8.93,30.958,0,19.943,0C8.929,0,0,8.93,0,19.943c0,11.015,8.929,19.94
3,19.943,19.943c11.015,0,19.943-8.929,19.943-19.943
M39.887,19.943C39.887,8.93,30.958,0,19.943,0C8.929,0,0,8.93,0,19.943
            c0,11.015,8.929,19.943,19.943,19.943c11.015,0,19.943-8.929,19.943-19.943
M39.887,19.943C39.887,8.93,30.958,0,19.943,0C8.929,0,0,8.93,0,19.943c0,11.015,8.929,19.94
3,19.943,19.943c11.015,0,19.943-8.929,19.943-19.943"/>
         <path d="M19.943,38.567c-10.286,0-18.624-8.338-18.624-18.624"/>
         <path d="M1.319,19.943c0-10.285,8.338-18.624,18.624-18.624"/>
         <path d="M19.943,1.319c10.286,0,18.624,8.339,18.624,18.624"/>
         <path d="M38.567,19.943c0,10.286-8.338,18.624-18.624,18.624"/>
         <path d="M19.943,38.567"/>
         <path d="M39.887,19.943C39.887,8.93,30.958,0,19.943,0"/>
         <path d="M19.943,0C8.929,0,0,8.93,0,19.943"/>
         <path d="M0,19.943c0,11.015,8.929,19.943,19.943,19.943c11.015,0,19.943-8.929,19.943-19.943"/>
         <path d="M19.943,39.887c11.015,0,19.943-8.929,19.943-19.943"/>
         <path d="M39.887,19.943"/>
      </g>
      <g style="&st3;">
         <path style="&st2;" d="M28.863,25.035c0-1.191,0.97-2.16,2.162-
2.16c1.19,0,2.16,0.969,2.16,2.16c0,1.191-0.97,2.161-2.16,2.161c-1.192,0-2.162-0.97-2.162-
2.161 M28.863,25.035c0-1.191,0.97-2.16,2.162-2.16c1.19,0,2.16,0.969,2.16,2.16
            c0,1.191-0.97,2.161-2.16,2.161c-1.192,0-2.162-0.97-2.162-2.161
M28.863,25.035c0-1.191,0.97-2.16,2.162-2.16c1.19,0,2.16,0.969,2.16,2.16c0,1.191-
0.97,2.161-2.16,2.161c-1.192,0-2.162-0.97-2.162-2.161"/>
         <path d="M28.863,25.035c0-1.191,0.97-2.16,2.162-2.16"/>
         <path d="M31.025,22.875c1.19,0,2.16,0.969,2.16,2.16"/>
         <path d="M33.186,25.035c0,1.191-0.97,2.161-2.16,2.161"/>
         <path d="M31.025,27.196c-1.192,0-2.162-0.97-2.162-2.161"/>
         <path d="M28.863,25.035"/>
      </g>
      <g style="&st3;">
         <path style="&st2;" d="M11.07,25.035c0-1.191,0.969-2.16,2.161-
2.16c1.191,0,2.161,0.969,2.161,2.16c0,1.191-0.969,2.161-2.161,2.161c-1.192,0-2.161-0.97-
2.161-2.161 M11.07,25.035c0-1.191,0.969-2.16,2.161-2.16c1.191,0,2.161,0.969,2.161,2.16
            c0,1.191-0.969,2.161-2.161,2.161c-1.192,0-2.161-0.97-2.161-2.161
M11.07,25.035c0-1.191,0.969-2.16,2.161-2.16c1.191,0,2.161,0.969,2.161,2.16c0,1.191-
0.969,2.161-2.161,2.161c-1.192,0-2.161-0.97-2.161-2.161"/>
         <path d="M11.07,25.035c0-1.191,0.969-2.16,2.161-2.16"/>
         <path d="M13.231,22.875c1.191,0,2.161,0.969,2.161,2.16"/>
         <path d="M15.392,25.035c0,1.191-0.969,2.161-2.161,2.161"/>
         <path d="M13.231,27.196c-1.192,0-2.161-0.97-2.161-2.161"/>
         <path d="M11.07,25.035"/>
      </g>
      <g style="&st3;">
         <path style="&st2;" d="M36.948,23.401c-1.238,0-1.238,0-1.238,0c0-3.685,0-
3.685,0-3.685c-2.767-1.52-2.767-1.52-2.767-1.52c-4.041,0-4.041,0-4.041,0c0-3.137,0-
3.137,0-3.137c-0.764,0-0.764,0-0.764,0c0,3.606,0,3.606,0,3.606c-1.073,0-1.073,0-1.073,0
            c-0.865,0-0.865,0-0.865,0c0-1.795,0-1.795,0-1.795c-0.586-0.415-0.586-0.415-
0.586-0.415c-1.165,0-1.165,0-1.165,0c0,1.947,0,1.947,0,1.947c-0.947,0.932-0.947,0.932-
0.947,0.932C5,19.336,5,19.336,5,19.336C3.854,21.19,3.854,21.19,3.854,21.19
            c0,2.211,0,2.211,0,2.211c-0.762,0-0.762,0-
0.762,0c0,1.467,0,1.467,0,1.467c7.207,0,7.207,0,7.207,0c0.086-1.541,1.369-2.77,2.932-
2.77c1.563,0,2.845,1.229,2.932,2.77c11.93,0,11.93,0,11.93,0c0.087-1.541,1.368-2.77,2.933-
2.77c1.563,0,2.845,1.229,2.931,2.77
            c2.992,0,2.992,0,2.992,0c0-1.467,0-1.467,0-1.467 M36.948,23.401c-1.238,0-
1.238,0-1.238,0c0-3.685,0-3.685,0-3.685c-2.767-1.52-2.767-1.52-2.767-1.52c-4.041,0-
4.041,0-4.041,0c0-3.137,0-3.137,0-3.137c-0.764,0-0.764,0-0.764,0c0,3.606,0,3.606,0,3.606
            c-1.073,0-1.073,0-1.073,0c-0.865,0-0.865,0-0.865,0c0-1.795,0-1.795,0-1.795c-
0.586-0.415-0.586-0.415-0.586-0.415c-1.165,0-1.165,0-1.165,0c0,1.947,0,1.947,0,1.947c-
0.947,0.932-0.947,0.932-0.947,0.932C5,19.336,5,19.336,5,19.336
            C3.854,21.19,3.854,21.19,3.854,21.19c0,2.211,0,2.211,0,2.211c-0.762,0-
0.762,0-0.762,0c0,1.467,0,1.467,0,1.467c7.207,0,7.207,0,7.207,0c0.086-1.541,1.369-
2.77,2.932-2.77c1.563,0,2.845,1.229,2.932,2.77c11.93,0,11.93,0,11.93,0
            c0.087-1.541,1.368-2.77,2.933-
2.77c1.563,0,2.845,1.229,2.931,2.77c2.992,0,2.992,0,2.992,0c0-1.467,0-1.467,0-1.467
```

```
M36.948,23.401c-1.238,0-1.238,0-1.238,0c0-3.685,0-3.685,0-3.685c-2.767-1.52-2.767-1.52-
2.767-1.52c-4.041,0-4.041,0-4.041,0
            c0-3.137,0-3.137,0-3.137c-0.764,0-0.764,0-0.764,0c0,3.606,0,3.606,0,3.606c-
1.073,0-1.073,0-1.073,0c-0.865,0-0.865,0-0.865,0c0-1.795,0-1.795,0-1.795c-0.586-0.415-
0.586-0.415-0.586-0.415c-1.165,0-1.165,0-1.165,0c0,1.947,0,1.947,0,1.947
            c-0.947,0.932-0.947,0.932-
0.947,0.932C5,19.336,5,19.336,5,19.336C3.854,21.19,3.854,21.19,3.854,21.19c0,2.211,0,2.21
1,0,2.211c-0.762,0-0.762,0-0.762,0c0,1.467,0,1.467,0,1.467c7.207,0,7.207,0,7.207,0c0.086-
1.541,1.369-2.77,2.932-2.77
            c1.563,0,2.845,1.229,2.932,2.77c11.93,0,11.93,0,11.93,0c0.087-1.541,1.368-
2.77,2.933-2.77c1.563,0,2.845,1.229,2.931,2.77c2.992,0,2.992,0,2.992,0c0-1.467,0-1.467,0-
1.467"/>
         <path d="M36.948,23.401c-1.238,0-1.238,0-1.238,0"/>
         <path d="M35.71,23.401c0-3.685,0-3.685,0-3.685"/>
         <path d="M35.71,19.717c-2.767-1.52-2.767-1.52-2.767-1.52"/>
         <path d="M32.943,18.197c-4.041,0-4.041,0-4.041,0"/>
         <path d="M28.902,18.197c0-3.137,0-3.137,0-3.137"/>
         <path d="M28.902,15.061c-0.764,0-0.764,0-0.764,0"/>
         <path d="M28.139,15.061c0,3.606,0,3.606,0,3.606"/>
         <path d="M28.139,18.667c-1.073,0-1.073,0-1.073,0"/>
         <path d="M27.065,18.667c-0.865,0-0.865,0-0.865,0"/>
         <path d="M26.2,18.667c0-1.795,0-1.795,0-1.795"/>
         <path d="M26.2,16.872c-0.586-0.415-0.586-0.415-0.586-0.415"/>
         <path d="M25.614,16.457c-1.165,0-1.165,0-1.165,0"/>
         <path d="M24.449,16.457c0,1.947,0,1.947,0,1.947"/>
         <path d="M24.449,18.404c-0.947,0.932-0.947,0.932-0.947,0.932"/>
         <path d="M23.502,19.336C5,19.336,5,19.336,5,19.336"/>
         <path d="M5,19.336C3.854,21.19,3.854,21.19,3.854,21.19"/>
         <path d="M3.854,21.19c0,2.211,0,2.211,0,2.211"/>
         <path d="M3.854,23.401c-0.762,0-0.762,0-0.762,0"/>
         <path d="M3.092,23.401c0,1.467,0,1.467,0,1.467"/>
         <path d="M3.092,24.868c7.207,0,7.207,0,7.207,0"/>
         <path d="M10.299,24.868c0.086-1.541,1.369-2.77,2.932-2.77"/>
         <path d="M13.231,22.099c1.563,0,2.845,1.229,2.932,2.77"/>
         <path d="M16.163,24.868c11.93,0,11.93,0,11.93,0"/>
         <path d="M28.093,24.868c0.087-1.541,1.368-2.77,2.933-2.77"/>
         <path d="M31.025,22.099c1.563,0,2.845,1.229,2.931,2.77"/>
         <path d="M33.956,24.868c2.992,0,2.992,0,2.992,0"/>
         <path d="M36.948,24.868c0-1.467,0-1.467,0-1.467"/>
      </g>
      <g style="&st3;">
         <path style="&st2;" d="M23.051,14.02v-
1.219H4.311v1.219h1.945v2.965H4.311v1.218h18.74v-1.218h-1.944V14.02h1.944 M23.051,14.02v-
1.219H4.311v1.219h1.945v2.965H4.311v1.218h18.74v-1.218h-1.944V14.02h1.944 M23.051,14.02v-
1.219H4.311v1.219h1.945v2.965H4.311
            v1.218h18.74v-1.218h-1.944V14.02h1.944 M14.032,14.02h2.835v2.965h-2.835V14.02
M14.032,14.02h2.835v2.965h-2.835V14.02 M14.032,14.02h2.835v2.965h-2.835V14.02
M13.33,16.984h-2.836V14.02h2.836v2.965 M13.33,16.984h-2.836V14.02h2.836v2.965
M13.33,16.984
            h-2.836V14.02h2.836v2.965 M6.957,14.02h2.835v2.965H6.957V14.02
M6.957,14.02h2.835v2.965H6.957V14.02 M6.957,14.02h2.835v2.965H6.957V14.02
M20.404,16.984h-2.835V14.02h2.835v2.965 M20.404,16.984h-2.835V14.02h2.835v2.965
M20.404,16.984h-2.835V14.02h2.835
            v2.965"/>
         <path d="M23.051,14.02v-1.219H4.311v1.219h1.945v2.965H4.311v1.218h18.74v-1.218h-
1.944V14.02h1.944z"/>
         <path d="M14.032,14.02h2.835v2.965h-2.835V14.02z"/>
         <path d="M13.33,16.984h-2.836V14.02h2.836v2.965z"/>
         <path d="M6.957,14.02h2.835v2.965H6.957V14.02z"/>
         <path d="M20.404,16.984h-2.835V14.02h2.835v2.965z"/>
      </g>
   </g>
</svg>
```

D.5. EMS Classification Schemes

These classification schemes (taxonomies) are published to CSW implementations so that associations of features, styles and symbols to classification nodes can be constructed and subsequently queried for discovery of associated resources.

D.5.1 HSWG Symbol Classification Scheme

The HSWG Symbol Classification derived from http://www.fgdc.gov/HSWG/.

```
<?xml version="1.0" encoding="UTF-8"?>
<rim:ClassificationScheme xmlns:rim="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:oasis:names:tc:ebxml-
regrep:rim:xsd:2.5
    urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5" objectType="ClassificationScheme" id="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1" nodeType="UniqueCode" isInternal="true">
    <rim:Name>
        <rim:LocalizedString xml:lang="en" value="Emergency Management Symbology Taxnonomy"/>
    </rim:Name>
    <rim:Description>
        <rim:LocalizedString xml:lang="en" value="A work-in-progress taxonomy for use with emergency
management systems. Initially developed by the Federal Geographic Data Committee (FGDC) Homeland Security
Working Group.  See http://www.fgdc.gov/HSWG/ for more information."/>
    </rim:Description>
    <rim:ClassificationNode xmlns:rim="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1" id="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:1" code="1">
        <rim:Name>
            <rim:LocalizedString xml:lang="en" value="Incidents Symbology Reference"/>
        </rim:Name>
        <rim:Description>
            <rim:LocalizedString xml:lang="en" value="Category of eight Themes and forty-two Features that
symbolize a ???cause of action??? or ???source of disaster.???"/>
        </rim:Description>
        <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:1" code="1.1" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:1:1">
            <rim:Name>
                <rim:LocalizedString xml:lang="en" value="Incidents Stage 01 Background Symbol
(Background)"/>
            </rim:Name>
            <rim:Description>
                <rim:LocalizedString xml:lang="en" value="The background fill shape for the Incidents symbol,
level one."/>
            </rim:Description>
        </rim:ClassificationNode>
        <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:1" code="1.2" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:1:2">
            <rim:Name>
                <rim:LocalizedString xml:lang="en" value="Incidents Stage 01 Frame Symbol (Frame)"/>
            </rim:Name>
            <rim:Description>
                <rim:LocalizedString xml:lang="en" value="The frame shape for the Incidents symbol, level one."/>
            </rim:Description>
        </rim:ClassificationNode>
        <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:1" code="1.3" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:1:3">
            <rim:Name>
                <rim:LocalizedString xml:lang="en" value="Civil Disturbance Incident (Theme)"/>
```

```
            </rim:Name>
            <rim:Description>
                <rim:LocalizedString xml:lang="en" value="Human activities resulting in the disrupting of services
or requiring varying levels of support, law enforcement or attention."/>
            </rim:Description>
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:1:3" code="1.3.1" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:1:3:1">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Civil Demonstrations (Civil Disturbance Feature)"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="A public display of group feelings toward a person
or cause. (Source: Merriam-Webster Online Dictionary definition)"/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:1:3" code="1.3.2" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:1:3:2">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Civil Displaced Population (Civil Disturbance
Feature)"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="Persons or groups of person who have been forced
or obliged to flee or to leave their homes or places of habitual residence, in particular as a result of or in order to
avoid the effects of armed conflict, violations of human rights, or natural or human-ma"/>
                </rim:Description>
            </rim:ClassificationNode>
        </rim:ClassificationNode>

...
        <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:3" code="3.5" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:3:5">
            <rim:Name>
                <rim:LocalizedString xml:lang="en" value="Fire Suppression Operation (Theme)"/>
            </rim:Name>
            <rim:Description>
                <rim:LocalizedString xml:lang="en" value="The extinguishing of a burning (and flaming) object by
means of applying an agent, such as water. (Source: Modified from www.firewise.org glossary of terms)"/>
            </rim:Description>
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:3:5" code="3.5.1" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:3:5:1">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Fire Hydrant (Fire Suppression Feature)"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="A discharge pipe with a valve and spout from which
water may be drawn from a water main in sufficient volume and at sufficient pressure for firefighting purposes.
(Source: Adapted from Merriam-Webster Online Dictionary definition of hydrant:)"/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:3:5" code="3.5.2" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:3:5:2">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Other Water Supply Location (Fire Suppression
Feature)"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="Any source of water other than a fire hydrant that is
sufficient for the purpose of fire fighting."/>
                </rim:Description>
            </rim:ClassificationNode>
```

```
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-opengis:ip:ems-
1:taxonomy:ems:0.1:3:5" code="3.5.3" id="urn:x-opengis:ip:ems-1:taxonomy:ems:0.1:3:5:3">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Fire Station (Fire Suppression Feature)"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="A facility housing fire-fighting equipment and/or
personnel."/>
                </rim:Description>
            </rim:ClassificationNode>
        </rim:ClassificationNode>

...
    </rim:ClassificationNode>
</rim:ClassificationScheme>
```

D.5.2 Digest/FACC Feature Classification Scheme

The DIGEST/FACC (2.1) feature taxonomy derived from
http://www.digest.org/html/DIGEST_2-1_Part4.pdf

```
<?xml version="1.0" encoding="utf-8"?>
<?xml-stylesheet type="application/xml" href="rimToSQL.xsl"?>
<rim:ClassificationScheme xmlns:rim="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:oasis:names:tc:ebxml-
regrep:rim:xsd:2.1 rim.xsd" id="urn:digest:facc:2.1" objectType="ClassificationScheme" isInternal="true"
nodeType="UniqueCode">
    <rim:Name>
        <rim:LocalizedString rim:lang="en" value="DIGEST FACC v2.1"/>
    </rim:Name>
    <rim:Description>
        <rim:LocalizedString rim:lang="en" value="Digital Geographic Information Exchange Standard (DIGEST)
Feature and Attribute Coding Catalogue (FACC) v2.1. FACC provides a means for encoding real-world entities or
objects and concepts."/>
    </rim:Description>
    <rim:ClassificationNode id="urn:digest:facc:2.1:0" objectType="ClassificationNode"
parent="urn:digest:facc:2.1" code="0">
        <rim:Name>
            <rim:LocalizedString rim:lang="en" value="Review"/>
        </rim:Name>
        <rim:ClassificationNode id="urn:digest:facc:2.1:0:00" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:0" code="00">
            <rim:Name>
                <rim:LocalizedString rim:lang="en" value="Review-General Comments"/>
            </rim:Name>
        </rim:ClassificationNode>
    </rim:ClassificationNode>
    <rim:ClassificationNode id="urn:digest:facc:2.1:A" objectType="ClassificationNode"
parent="urn:digest:facc:2.1" code="A">
        <rim:Name>
            <rim:LocalizedString rim:lang="en" value="Culture"/>
        </rim:Name>
        <rim:ClassificationNode id="urn:digest:facc:2.1:A:AA" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:A" code="AA">
            <rim:Name>
                <rim:LocalizedString rim:lang="en" value="Culture-Extraction"/>
            </rim:Name>
            <rim:ClassificationNode id="urn:digest:facc:2.1:A:AA:AA010" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:A:AA" code="AA010">
                <rim:Name>
```

```
                    <rim:LocalizedString rim:lang="en" value="Mine"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="An excavation made in the earth for the purpose of
extracting natural deposits. (See also AQ090)"/>
                </rim:Description>
            </rim:ClassificationNode>
        </rim:ClassificationNode>

. . .
        <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z" code="ZD">
            <rim:Name>
                <rim:LocalizedString rim:lang="en" value="General-Miscellaneous"/>
            </rim:Name>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD001" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD001">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Network"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="A system of inter-connected real world objects of
the same type or directly related types."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD003" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD003">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Artifact Location"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="An indicator which identifies incomplete or illogical
data at a specific location (node), used for data processing only."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD012" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD012">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Geographic Information Point"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="A location where geographic information or
statistics may apply."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD015" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD015">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Point of Change"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="The location/position on a linear feature where the
characteristics of the feature change significantly."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD020" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD020">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Void Collection Area "/>
                </rim:Name>
                <rim:Description>
```

```
                    <rim:LocalizedString rim:lang="en" value="An area lacking suitable source coverage, or where
data is not required."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD040" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD040">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Named Location "/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="A geographic place on the earth, not normally
appearing as a feature on a map, but having a name that is required to be placed on a map."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZD:ZD045" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZD" code="ZD045">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Text Description "/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="An area in which a characteristic or an activity
pertaining to the area can be described and possibly labeled on a product if deemed important at the time the
product is being produced."/>
                </rim:Description>
            </rim:ClassificationNode>
        </rim:ClassificationNode>
        <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZE" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z" code="ZE">
            <rim:Name>
                <rim:LocalizedString rim:lang="en" value="General-Background Features"/>
            </rim:Name>
            <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZE:ZEXXX" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z:ZE" code="ZEXXX">
                <rim:Name>
                    <rim:LocalizedString rim:lang="en" value="Feature Place Holder ZE"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString rim:lang="en" value="Place Holder - ZE"/>
                </rim:Description>
            </rim:ClassificationNode>
        </rim:ClassificationNode>
        <rim:ClassificationNode id="urn:digest:facc:2.1:Z:ZM" objectType="ClassificationNode"
parent="urn:digest:facc:2.1:Z" code="ZM">
            <rim:Name>
                <rim:LocalizedString rim:lang="en" value="General-Metadata"/>
            </rim:Name>
        </rim:ClassificationNode>
    </rim:ClassificationNode>
</rim:ClassificationScheme>
```

D.5.3 ISO 19119 Service Classification Scheme

ISO 19119 (with OGC extensions)services taxonomy ISO 19119 (http://www.opengis.org/docs/02-112.pdf)

```
<?xml version="1.0" encoding="UTF-8"?>
<rim:ClassificationScheme xmlns:rim="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
objectType="ClassificationScheme" nodeType="UniqueCode" isInternal="true" id="urn:x-iso-tc211:19119(2003)">
    <rim:Name>
        <rim:LocalizedString xml:lang="en" value="ISO 19119 Geographic services taxonomy"/>
    </rim:Name>
```

```
    <rim:Description>
        <rim:LocalizedString xml:lang="en" value="The geographic services taxonomy is an ISO/OGC standard for
categorizing geographic services. The taxonomy consists of the name of the category and the description of the
category."/>
    </rim:Description>
    <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-iso-tc211:19119(2003)" code="1"
id="urn:x-iso-tc211:19119(2003):1">
        <rim:Name>
            <rim:LocalizedString xml:lang="en" value="Geographic human interaction services"/>
        </rim:Name>
        <rim:Description>
            <rim:LocalizedString xml:lang="en" value="Human interaction services are services for management of
user interfaces, graphics, multimedia, and for presentation of compound documents."/>
        </rim:Description>
        <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-iso-tc211:19119(2003):1"
code="1.1" id="urn:x-iso-tc211:19119(2003):1.1">
            <rim:Name>
                <rim:LocalizedString xml:lang="en" value="Catalogue viewer"/>
            </rim:Name>
            <rim:Description>
                <rim:LocalizedString xml:lang="en" value="Client service that allows a user to interact with a
catalogue to locate, browse, and manage metadata about geographic data or geographic services."/>
            </rim:Description>
        </rim:ClassificationNode>
        <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-iso-tc211:19119(2003):1"
code="1.2" id="urn:x-iso-tc211:19119(2003):1.2">
            <rim:Name>
                <rim:LocalizedString xml:lang="en" value="Geographic viewer"/>
            </rim:Name>
            <rim:Description>
                <rim:LocalizedString xml:lang="en" value="Client service that allows a user to view one or more
feature collections or coverages. This viewer allows a user to interact with map data, e.g., displaying, overlaying
and querying. An example is the viewer client generator defined in ISO 19128."/>
            </rim:Description>
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-iso-tc211:19119(2003):1.2"
code="1.2.1" id="urn:x-iso-tc211:19119(2003):1.2.1">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Geographic viewer - animation"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="Geographic viewer that allows a human to
sequence views of the same geographic location at different times."/>
                </rim:Description>
            </rim:ClassificationNode>
            <rim:ClassificationNode objectType="ClassificationNode" parent="urn:x-iso-tc211:19119(2003):1.2"
code="1.2.2" id="urn:x-iso-tc211:19119(2003):1.2.2">
                <rim:Name>
                    <rim:LocalizedString xml:lang="en" value="Geographic viewer - mosaicing"/>
                </rim:Name>
                <rim:Description>
                    <rim:LocalizedString xml:lang="en" value="Geographic viewer that allows combination of
views of geographic data for adjacent areas into a single view."/>
                </rim:Description>
            </rim:ClassificationNode>
        </rim:ClassificationNode>

•••

    </rim:ClassificationNode>
</rim:ClassificationScheme>
```

Annex E – Web-Mapping Module for CSW (informative)

1. Overview

This section defines an application module for CSW to accommodate web mapping resources, notably styles and symbols. The **Web Mapping** module currently includes the items listed below.

NOTE: Highlighted items are defined in the **Common** application module (OGC Project Document 03-094), but are also used here.

Classification nodes (new object types)

Dataset – a description of a geographic dataset (i.e., a collection of feature instances)

FeatureStyle (or FeaturePortrayal after 19117?) – a resource that specifies feature styling rules (i.e. a feature portrayal object in ISO 19117)

MapSymbol – a map symbol (i.e., a graphic element)

MapSymbolLib – a cohesive collection of map symbols

MapContext – a resource that encapsulates directives for creating a map <http://www.opengis.org/docs/03-036r2.pdf>

Classification schemes:

- the DIGEST/FACC feature classification scheme <http://www.digest.org/html/DIGEST_2-1_Part4.pdf>
- the FGDC classification scheme for emergency management symbology <http://www.fgdc.gov/HSWG/>

Associations:

- [RegistryObject] *PointOfContact* [User | Organization]
- [FeatureStyle] *Uses* [MapSymbol | MapSymbolLib]
- [Dataset] *PortrayalInfo* [RegistryPackage | FeatureStyle]
- [FeatureStyle] *Portrays* [XmlSchema] (or use *DefinedFor* as in 19117?)

All the feature styles that apply to an app schema may be aggregated into a “Feature portrayal” package (i.e. a “portrayal catalogue package” in ISO 19117) . This is a RegistryPackage with FeatureStyle members, where the package has a “subject” slot containing the value “Feature portrayal” (or just “portrayal”?).

Slots:

- Slot[@name="subject" and slotType="http://purl.org/dc/elements/1.1/subject"]
- Slot[@name="modified" and slotType="http://purl.org/dc/terms/modified"]
- Slot[@name="language" and slotType="http://purl.org/dc/elements/1.1/language"]
- Slot[@name="fragment" and slotType=" http://www.w3.org/TR/xptr-framework/"]

Fine-grained associations and classifications may be created using a "fragment" slot. The slotType tells us how to interpret the values; in this case we use the xpointer framework as described in the W3C specification. If the fragment in the target XML resource has an information item that is a schema-determined ID, then the value of that information item may be used to simply identify the fragment using a shorthand pointer. Otherwise, a full scheme-based pointer must be used to reference the fragment(s).

Example (shorthand):

```
Slot.name="fragment"
Slot.slotType="http://www.w3.org/TR/xptr-framework/"
Slot.values="Quarry"
```

Example (scheme-based):

```
Slot.name="fragment"
Slot.slotType="http://www.w3.org/TR/xptr-framework/"
Slot.values="xmlns(xsd=http://www.w3.org/2001/XMLSchema)
xpointer(/xsd:schema/xsd:element[@name='Quarry')"
```

where the app schema looks like this:

```
<xsd:schema>
  <!-- stuff -- >
  <xsd:element name="Quarry" type="foo:QuarryType"
    id="Quarry">
  <xsd:complexType name="QuarryType" id="QuarryType">
     <!-- type def -->
  </xsd:coomplexType>
</xsd:schema>
```

2. FeatureStyle and MapSymbol metadata – Basic CSW/RIM mappings

This section describes how to map Style/Symbol metadata elements as described in the SMS document (OGC 03-031) to the CSW catalogue information model. All required elements are mapped (there is only one: StyleMetadata/ContentURL), but only a subset of the optional elements are considered here.

Table 1 – FeatureStyle/MapSymbol metadata mappings

Style metadata element	RIM element
StyleMetadata/ContentURL	**CSWExtrinsicObject[@objectType="FeatureStyle"]/ repositoryItem**
StyleMetadata/Name	**CSWExtrinsicObject/Name**
StyleMetadata/Description	CSWExtrinsicObject/Description
StyleMetadata/Language	CSWExtrinsicObject/Slot[@name="language" and slotType="http://purl.org/dc/elements/1.1/language"] NOTE: Dublin Core metadata term
StyleMetadata/DateStamp	CSWExtrinsicObject/Slot[@name="modified" and slotType="http://purl.org/dc/terms/modified"] NOTE: Dublin Core metadata term
StyleMetadata/Contact	Association/@associationType="PointOfContact", where: [FeatureStyle] PointOfContact [User | Organization] User(RegistryObject) properties: Address, PersonName, TelephoneNumber, EmailAddress, url
StyleMetadata/ ApplicationSchemaInfo	Association/@associationType="Portrays", where: `[FeatureStyle] Portrays [XmlSchema]` NOTE: use relationship from 19117 (Portrayal)

Style metadata element	RIM element
StyleMetadata/ ClassificationList/ Classification	CSWExtrinsicObject/Classification (0..*) (i.e. FeatureStyle/MapSymbol may be classified according to some conceptual scheme)
StyleMetadata/ LibraryNameList/LibraryName	Association/@associationType="Uses", where: `[FeatureStyle] Uses [MapSymbolLib|MapSymbol]`
StyleMetadata/KeywordList	CSWExtrinsicObject/Slot[@name="subject" and slotType="http://purl.org/dc/elements/1.1/subject"] NOTE: Dublin Core term for unstructured keywords or key phrases; use a Classification to reference a term from a controlled vocabulary (e.g., a thesaurus)

Overview

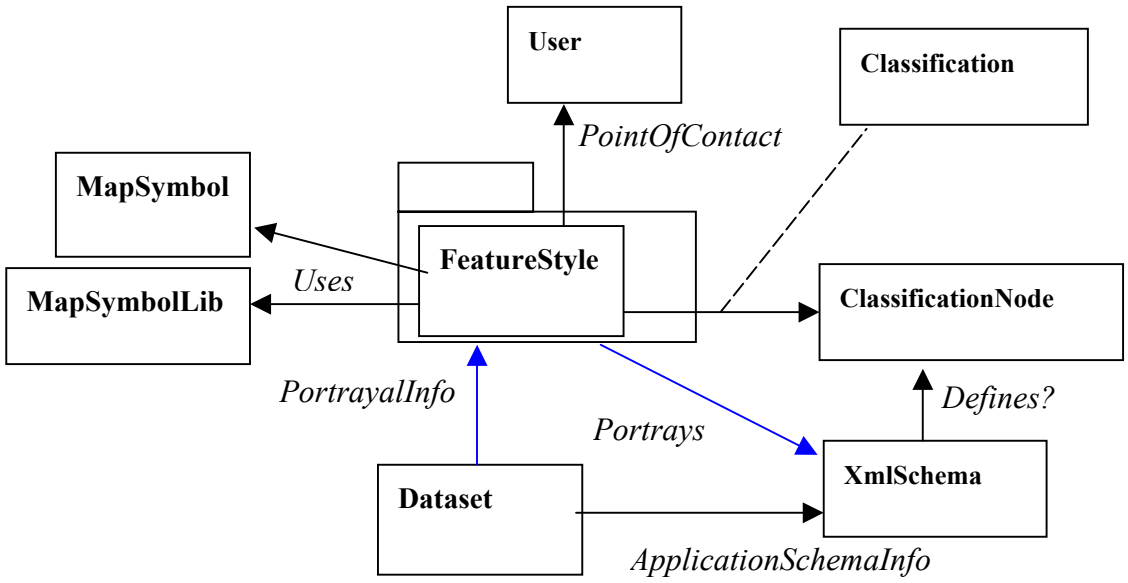


Figure 1. RIM Feature Style Metadata Associations

NOTES:

- associations in blue are based on ISO/DIS 19117 (Portrayal)
- also see Fig. 7 in ISO 19117 (Portrayal catalogue package)

Sample XML representation of

```
<csw:CSWExtrinsicObject
  xml:base="http://www.opengis.net/cat/csw/object-types/"
  xmlns:csw="http://www.opengis.net/cat/csw"
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
  id="uri-1"
  objectType="FeatureStyle"
  userVersion="1.0"
  expiration="2004-03-31T12:00Z"
  mimeType="application/xml; charset=UTF-8"
  isOpaque="false" >
  <rim:Name>
    <rim:LocalizedString xml:lang="en"
      value="ACME Hydro feature styles." />
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en"
      value="Defines styling rules for hydrographic features."/>
  </rim:Description>
  <rim:Slot name="subject"
    slotType=" http://purl.org/dc/elements/1.1/subject">
    <rim:ValueList>
      <rim:Value>cartography</rim:Value>
      <rim:Value>styling rules</rim:Value>
    <rim:ValueList>
  </rim:Slot>
  <rim:Slot name="modified"
    slotType="http://purl.org/dc/terms/modified">
    <rim:ValueList>
      <rim:Value>2004-02-10T14:25-08:00</rim:Value>
    <rim:ValueList>
  </rim:Slot>
  <rim:Slot name="language"
    slotType="http://purl.org/dc/elements/1.1/language">
    <rim:ValueList>
      <rim:Value>en</rim:Value>
    <rim:ValueList>
  </rim:Slot>
  <rim:Classification
    id="urn:uuid:4b581ee6-8e0c-48fe-9246-e00e22aa2b4b"
    objectType="Classification"
    classifiedObject="uri-1"
    classificationNode="urn:x-digest:facc:2.1:B"
    classificationScheme="urn:x-digest:facc:2.1">
    <rim:Name>
      <rim:LocalizedString xml:lang="en"
        value="Hydrography" />
    </rim:Name>
  </Classification>

  <repositoryItem>some-uri</repositoryItem>
</csw:CSWExtrinsicObject>
```

NOTES:

- the xml:base attribute specifies a base URI for interpreting objectType values specified as relative URIs

3. Publication use cases

Publish Feature Classification Scheme (taxonomy)

The CSW should use the classification object to classify resources instead of the association with the type "Describes" as specified in the use case. The context (i.e, role) of the classification may be described by using the "context" slot, which is defined within the Common application module. At the end of this use case, the relevant registry objects are depicted in Figure 1, where a feature taxonomy is used to classify a dataset. Basically, we propose to use the DIGEST/FACC feature classification scheme for this purpose.

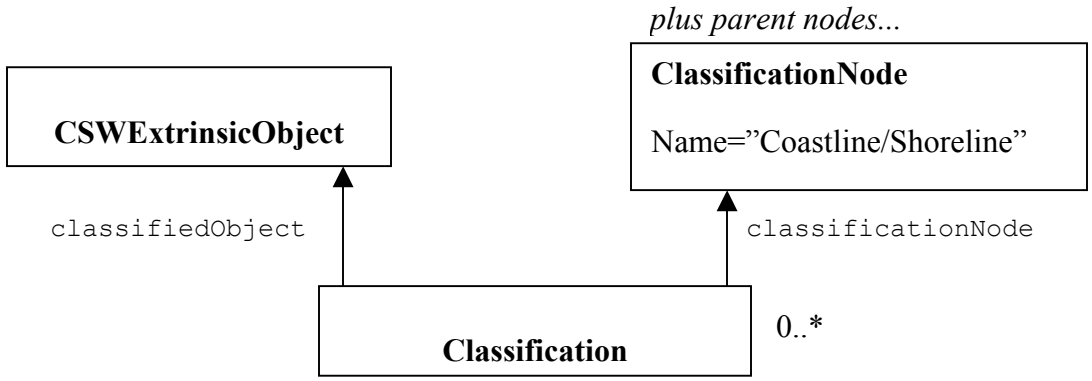


Figure 1: Classifying a dataset by feature type(s)

Publish Application Schema

The "Describes" association between an application schema and metadata (i.e. “Dataset”) should be changed to employ the "ApplicationSchemaInfo" role name specified in the ISO19115 standard. The "HasSchema" association between the "Dataset" and the "ExternalLink" isn’t really necessary (unless we misunderstand exactly what resource is referenced by the the external link). Also, a *targetNamespace* slot could be attached to the app schema as a convenience.

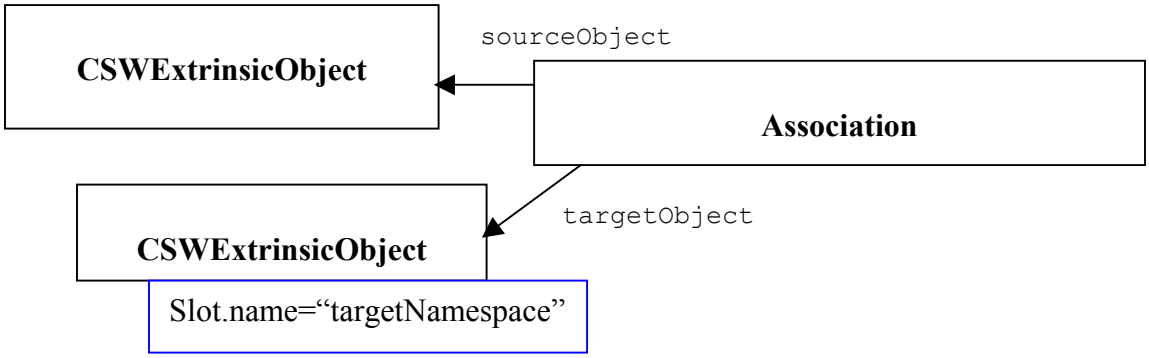


Figure 2: Associating a datasets with an application schema

Publish Dataset Description

The CSW catalogue need not store each feature type definition as a separate registry object—these are already embodied in published application schemas; however, declared feature types could be summarized by a slot named "featureTypes" which is attached to the "XmlSchema" extrinsic object. A service offer may be associated with a dataset description via the "OperatesOn" association (from 19119—see figure below). In addition, the "HasSchema" and "Describe" association types are not necessary since the relationship between the "Dataset" and the "XmlSchema" is established by the "ApplicationSchemaInfo" association type (from 19115).

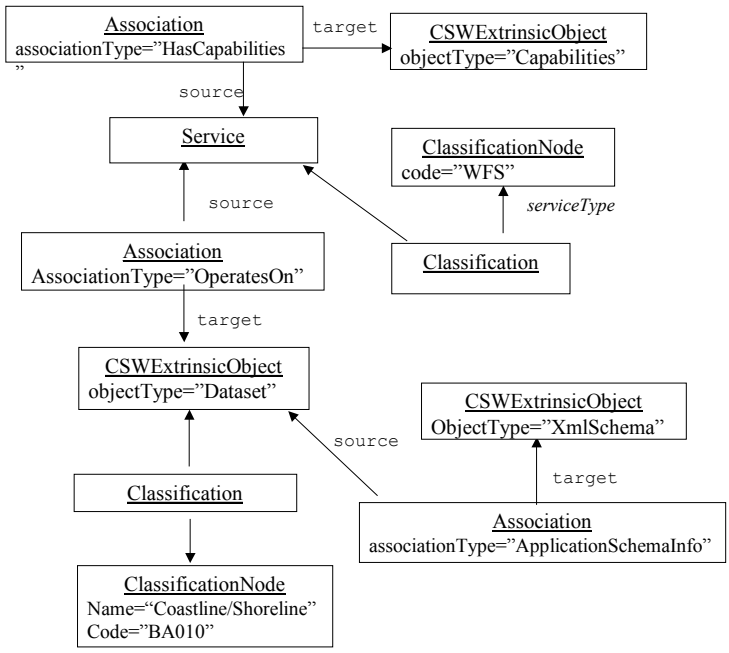


Figure 11. Relating Services, Datasets and Application Schemas

Fine-grained classification (i.e. at the level of schema components) can be supported using a "fragmentId" slot; doing this also allows us to identify schema components that are classified in the same way by including multiple values that reference specific schema components (i.e., instead of classifiying the entire application schema, we classify the schema components).

Classification ("Hydrography")	Slot.name="fragment" Slot.slotType=http://www.w3.org/TR/xptr-framework/ Slot/ValueList/Value="River" Slot/ValueList/Value="Watershed"

Figure 12 Using the fragment slot for fine-grained classification

Publish FeatureStyle

We can distinguish different categories of metadata: general, lifecycle, rights, technical, classification, etc[7]. Generic metadata can be captured as RIM objects (perhaps adorned with additional slots where necessary);

[7] Such a categorization in introduced in section 4.1 of the IEEE Learning Object Metadata standard: <http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf>.

detailed technical metadata—like app schemas, feature styling rules, dataset descriptions, video clips, etc.—can be stored as “extrinsic” metadata.

A FeatureStyle object is then handled as an extrinsic object where the styling rules are embodied in a repository item “bound” to the extrinsic object that conveys the more general metadata (see the example in section 2).

Note that ISO/DIS 19117 (Portrayal) appears to shed some light on these relationships.

Annex F – CSW Request/Response Examples (informative)

1. Introduction

This document describes the CSW interface from the aspect of a client developer. At the moment it solely consists of the CSW Interface operations examples with explanations

2. Discovery

2.1. Find styles by application schema

The initial query has to be made to obtain the UUIDs of the objects that satisfy the conditions.

Query:

```
<?xml version='1.0' encoding='UTF-8'?>
<GetRecords xmlns='http://www.opengis.net/cat/csw'
  xmlns:ogc='http://www.opengis.net/ogc'
  elementSet='full'
  startPosition='1'
  maxRecords='50'
  outputFormat='application/xml; charset=UTF-8'>

  <Query typeNames='FeatureStyle Association'>
    <SelectObjectType>FeatureStyle</SelectObjectType>
    <QueryConstraint>
      <ogc:Filter>
        <ogc:And>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/Association/@associationType</ogc:PropertyName>
            <ogc:Literal>Portrays</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/Association/@targetObject</ogc:PropertyName>
            <ogc:Literal>urn:x-galdosinc-com:van:schema:vancouver:v1.00</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/FeatureStyle/@id</ogc:PropertyName>
```

```
            <ogc:PropertyName>/Association/@sourceObject</ogc:PropertyName>
          </ogc:PropertyIsEqualTo>
        </ogc:And>
      </ogc:Filter>
    </QueryConstraint>
  </Query>
</GetRecords>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetRecordsResponse xmlns="http://www.opengis.net/cat/csw" version="0.8.3">
    <requestInfo>
        <GetRecords xmlns:ogc="http://www.opengis.net/ogc" elementSet="full"
startPosition="1" maxRecords="50" outputFormat="application/xml; charset=UTF-8">
            <Query typeNames="FeatureStyle Association">
                <SelectObjectType>FeatureStyle</SelectObjectType>
                <QueryConstraint>
                    <ogc:Filter>
                        <ogc:And>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/Association/@associationType</ogc:PropertyName>
                                <ogc:Literal>Portrays</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/Association/@targetObject</ogc:PropertyName>
                                <ogc:Literal>urn:x-galdosinc-
com:van:schema:vancouver:v1.00</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>/FeatureStyle/@id</ogc:PropertyName>

    <ogc:PropertyName>/Association/@sourceObject</ogc:PropertyName>
                            </ogc:PropertyIsEqualTo>
                        </ogc:And>
                    </ogc:Filter>
                </QueryConstraint>
            </Query>
        </GetRecords>
    </requestInfo>
    <searchStatus status="complete" timestamp="2004-03-16T09:13:14-0800"/>
    <searchResults resultSetId="urn:uuid:a5cd1652-6224-4f74-92b8-bd104dee696e"
expiration="2004-03-16T09:28:14-0800" hitCount="5" elementSet="full" recordSchema=""
numberOfRecords="5" nextRecord="0">
```

```
        <CSWExtrinsicObject id="urn:uuid:371e680a-258b-474c-8367-fad4cfdacc73"
objectType="FeatureStyle" status="Submitted" majorVersion="1" minorVersion="0"
userVersion="1.0" mimeType="text/xml" isOpaque="false">
            <ns1:Name xmlns:ns1="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns1:LocalizedString lang="en" value="SCHOOL:EMS1"/>
            </ns1:Name>
            <ns2:Description xmlns:ns2="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns2:LocalizedString lang="en" value="EMS_SVG_Symbols"/>
            </ns2:Description>
            <repositoryItem>/FeatureStyle/urn:uuid:371e680a-258b-474c-8367-
fad4cfdacc73</repositoryItem>
        </CSWExtrinsicObject>
        <CSWExtrinsicObject id="urn:uuid:3b6a2ee7-f1e6-4f45-b753-d3543e91fbb5"
objectType="FeatureStyle" status="Submitted" majorVersion="1" minorVersion="0"
userVersion="1.0" mimeType="text/xml" isOpaque="false">
            <ns1:Name xmlns:ns1="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns1:LocalizedString lang="en" value="ROADL_1M:Foundation"/>
            </ns1:Name>
            <ns2:Description xmlns:ns2="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns2:LocalizedString lang="en" value="GeoSym"/>
            </ns2:Description>
            <repositoryItem>/FeatureStyle/urn:uuid:3b6a2ee7-f1e6-4f45-b753-
d3543e91fbb5</repositoryItem>
        </CSWExtrinsicObject>
        <CSWExtrinsicObject id="urn:uuid:6b7eeabf-96b1-4343-8c22-9e4f5461e7bb"
objectType="FeatureStyle" status="Submitted" majorVersion="1" minorVersion="0"
userVersion="1.0" mimeType="text/xml" isOpaque="false">
            <ns1:Name xmlns:ns1="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns1:LocalizedString lang="en" value="geosym-vmap0-hydro-inwatera"/>
            </ns1:Name>
            <ns2:Description xmlns:ns2="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns2:LocalizedString lang="en" value="Inland Water Areas"/>
            </ns2:Description>
            <repositoryItem>/FeatureStyle/urn:uuid:6b7eeabf-96b1-4343-8c22-
9e4f5461e7bb</repositoryItem>
        </CSWExtrinsicObject>
        <CSWExtrinsicObject id="urn:uuid:8007279f-8442-45cc-bd53-b0e92dedec89"
objectType="FeatureStyle" status="Submitted" majorVersion="1" minorVersion="0"
userVersion="1.0" mimeType="text/xml" isOpaque="false">
            <ns1:Name xmlns:ns1="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns1:LocalizedString lang="en" value="WATRCRSL_1M:Foundation"/>
            </ns1:Name>
            <ns2:Description xmlns:ns2="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns2:LocalizedString lang="en" value="Water Courses"/>
            </ns2:Description>
            <repositoryItem>/FeatureStyle/urn:uuid:8007279f-8442-45cc-bd53-
b0e92dedec89</repositoryItem>
        </CSWExtrinsicObject>
        <CSWExtrinsicObject id="urn:uuid:90b15980-837f-43f0-88b5-bfb23b1a3cf0"
objectType="FeatureStyle" status="Submitted" majorVersion="1" minorVersion="0"
userVersion="1.0" mimeType="text/xml" isOpaque="false">
            <ns1:Name xmlns:ns1="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
```

```
                <ns1:LocalizedString lang="en" value="FIRESTATION:EMS1"/>
            </ns1:Name>
            <ns2:Description xmlns:ns2="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns2:LocalizedString lang="en" value="EMS_SVG_Symbols"/>
            </ns2:Description>
            <repositoryItem>/FeatureStyle/urn:uuid:90b15980-837f-43f0-88b5-
bfb23b1a3cf0</repositoryItem>
        </CSWExtrinsicObject>
    </searchResults>
</GetRecordsResponse>
```

The subsequent action is to retrieve the styles by their IDs.

Query by ID:

```
urn:uuid:371e680a-258b-474c-8367-fad4cfdacc73
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<FeatureStyle xmlns="http://www.opengis.net/sld" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" version="1.0.20" xsi:schemaLocation="http://www.opengis.net/sld
http://schemas.cubewerx.com/schemas/sld/1.0.20/FeatureStyle.xsd">
    <Name>SCHOOL:EMS1</Name>
    <Description>
        <Title>EMS_SVG_Symbols</Title>
    </Description>
    <Rule>
        <Name>Level1</Name>
        <Description>
            <Title>School</Title>
        </Description>
        <MaxScaleDenominator>8e6</MaxScaleDenominator>
        <PointSymbol>
            <Graphic>
                <ExternalGraphic>
                    <OnlineResource xlink:type="simple"
xlink:href="http://demo.cubewerx.com/sld/libraries/ers/graphics/SVG_ERS_Symbols/CGM_Infra
structures_S1/Edu_Schools_S1.svg"/>
                    <Format>image/svg+xml</Format>
                </ExternalGraphic>
            </Graphic>
        </PointSymbol>
    </Rule>
</FeatureStyle>
```

2.2. Find symbols by data set

Query:

```
<?xml version='1.0' encoding='UTF-8'?>
<GetRecords xmlns='http://www.opengis.net/cat/csw'
  xmlns:ogc='http://www.opengis.net/ogc'
  elementSet='summary'
  startPosition='1'
  maxRecords='5'
  outputFormat='application/xml; charset=UTF-8'>

  <Query typeNames='Association:$a,$a2 FeatureStyle MapSymbolLib'>
    <SelectObjectType>MapSymbolLib</SelectObjectType>
    <QueryConstraint>
      <ogc:Filter>
        <ogc:And>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/$a/@sourceObject</ogc:PropertyName>
            <ogc:Literal>urn:uuid:35668c06-fdc6-4516-8da9-bd0d1cd4b2b0</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/$a/@associationType</ogc:PropertyName>
            <ogc:Literal>PortrayalInfo</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/$a/@targetObject</ogc:PropertyName>
            <ogc:PropertyName>/FeatureStyle/@id</ogc:PropertyName>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/FeatureStyle/@id</ogc:PropertyName>
            <ogc:PropertyName>/$a2/@sourceObject</ogc:PropertyName>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/$a2/@associationType</ogc:PropertyName>
            <ogc:Literal>Uses</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/$a2/@targetObject</ogc:PropertyName>
            <ogc:PropertyName>/MapSymbolLib/@id</ogc:PropertyName>
          </ogc:PropertyIsEqualTo>
        </ogc:And>
      </ogc:Filter>
    </QueryConstraint>
  </Query>
</GetRecords>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetRecordsResponse xmlns="http://www.opengis.net/cat/csw" version="0.8.3">
    <requestInfo>
        <GetRecords xmlns:ogc="http://www.opengis.net/ogc" elementSet="summary"
startPosition="1" maxRecords="5" outputFormat="application/xml; charset=UTF-8">
            <Query typeNames="Association:$a,$a2 FeatureStyle MapSymbolLib">
                <SelectObjectType>MapSymbolLib</SelectObjectType>
                <QueryConstraint>
                    <ogc:Filter>
                        <ogc:And>
                            <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>/$a/@sourceObject</ogc:PropertyName>
                                <ogc:Literal>urn:uuid:35668c06-fdc6-4516-8da9-
bd0d1cd4b2b0</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/$a/@associationType</ogc:PropertyName>
                                <ogc:Literal>PortrayalInfo</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>/$a/@targetObject</ogc:PropertyName>
                                <ogc:PropertyName>/FeatureStyle/@id</ogc:PropertyName>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>/FeatureStyle/@id</ogc:PropertyName>
                                <ogc:PropertyName>/$a2/@sourceObject</ogc:PropertyName>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/$a2/@associationType</ogc:PropertyName>
                                <ogc:Literal>Uses</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>
                                <ogc:PropertyName>/$a2/@targetObject</ogc:PropertyName>
                                <ogc:PropertyName>/MapSymbolLib/@id</ogc:PropertyName>
                            </ogc:PropertyIsEqualTo>
                        </ogc:And>
                    </ogc:Filter>
                </QueryConstraint>
            </Query>
        </GetRecords>
    </requestInfo>
    <searchStatus status="complete" timestamp="2004-03-16T09:22:08-0800"/>
```

```
    <searchResults resultSetId="urn:uuid:b76460d3-de4b-4862-960a-a2ab0ec94ea1"
expiration="2004-03-16T09:37:08-0800" hitCount="1" elementSet="summary" recordSchema=""
numberOfRecords="1" nextRecord="0">
        <CSWExtrinsicObject id="urn:uuid:072cff83-2937-4b88-aa36-d59cd89eb769"
objectType="MapSymbolLib" status="Submitted" majorVersion="1" minorVersion="0"
userVersion="1.0" mimeType="text/xml" isOpaque="false">
            <ns1:Name xmlns:ns1="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns1:LocalizedString lang="en" value="VancouverSymbols"/>
            </ns1:Name>
            <ns2:Description xmlns:ns2="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5">
                <ns2:LocalizedString lang="en" value="VancouverSymbols"/>
            </ns2:Description>
            <repositoryItem>/MapSymbolLib/urn:uuid:072cff83-2937-4b88-aa36-
d59cd89eb769</repositoryItem>
        </CSWExtrinsicObject>
    </searchResults>
</GetRecordsResponse>
```

Again, we have obtained the IDs of the desired repository objects, so we proceed to retrieve them.

Query by ID:

```
urn:uuid:072cff83-2937-4b88-aa36-d59cd89eb769
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<symbolCollection Name="Lib1">
    <symbol id="Lib1_1">
        <geometry>Point</geometry>
        <style/>
        <g id="Lib1_1">
            <path style="fill-rule:nonzero;fill:#27AECE;stroke:#44B8CD;stroke-
width:3;stroke-miterlimit:4;" d="M74.429,75.506H3V5.863h71.429v69.643z"/>
            <path style="fill-rule:nonzero;fill:#F6F623;stroke:#EBF322;stroke-
width:3;stroke-miterlimit:4;" d="M13.333,72.967l19.563-15.879v-
46.8l20.774,32.593L35.257,27.002"/>
            <path style="fill:none;stroke:#EDF33C;stroke-width:6;stroke-miterlimit:4;"
d="M35.143,55.863L53,75.506"/>
            <path style="fill-rule:nonzero;fill:#F8F95F;stroke:#E4F120;stroke-
width:3;stroke-miterlimit:4;" d="M29.786,4.077L15.5,41.577l16.072-19.643"/>
            <path style="fill:none;stroke:#D8ED1A;stroke-width:6;stroke-miterlimit:4;"
d="M61.436,41.577c0,19.932-12.171,36.089-27.186,36.089S7.065,61.509,7.065,41.577c0-
19.932,12.171-36.09,27.185-36.09s27.186,16.158,27.186,36.09z"/>
        </g>
        <!-- Airport -->
    </symbol>
…
…
…
```

```
</symbolCollection>
```

2.3. Find portrayal package by dataset

Query:

```
<?xml version='1.0' encoding='UTF-8'?>
<GetRecords xmlns='http://www.opengis.net/cat/csw'
  xmlns:ogc='http://www.opengis.net/ogc'
  elementSet='full'
  startPosition='1'
  maxRecords='50'
  outputFormat='application/xml; charset=UTF-8'>

  <Query typeNames='RegistryPackage Association'>
    <SelectObjectType>RegistryPackage</SelectObjectType>
    <QueryConstraint>
      <ogc:Filter>
        <ogc:And>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/Association/@associationType</ogc:PropertyName>
            <ogc:Literal>PortrayalInfo</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/Association/@sourceObject</ogc:PropertyName>
            <ogc:Literal>urn:uuid:35668c06-fdc6-4516-8da9-bd0d1cd4b2b0</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/Association/@targetObject</ogc:PropertyName>
            <ogc:PropertyName>/RegistryPackage/@id</ogc:PropertyName>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/RegistryPackage/Slot/@name</ogc:PropertyName>
            <ogc:Literal>subject</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>/RegistryPackage/Slot/ValueList/Value</ogc:PropertyName>
            <ogc:Literal>portrayal</ogc:Literal>
          </ogc:PropertyIsEqualTo>
        </ogc:And>
      </ogc:Filter>
    </QueryConstraint>
  </Query>
```

```
</GetRecords>
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetRecordsResponse xmlns="http://www.opengis.net/cat/csw" version="0.8.3">
    <requestInfo>
        <GetRecords xmlns:ogc="http://www.opengis.net/ogc" elementSet="full"
startPosition="1" maxRecords="50" outputFormat="application/xml; charset=UTF-8">
            <Query typeNames="RegistryPackage Association">
                <SelectObjectType>RegistryPackage</SelectObjectType>
                <QueryConstraint>
                    <ogc:Filter>
                        <ogc:And>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/Association/@associationType</ogc:PropertyName>
                                <ogc:Literal>PortrayalInfo</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/Association/@sourceObject</ogc:PropertyName>
                                <ogc:Literal>urn:uuid:35668c06-fdc6-4516-8da9-
bd0d1cd4b2b0</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/Association/@targetObject</ogc:PropertyName>

    <ogc:PropertyName>/RegistryPackage/@id</ogc:PropertyName>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/RegistryPackage/Slot/@name</ogc:PropertyName>
                                <ogc:Literal>subject</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                            <ogc:PropertyIsEqualTo>

    <ogc:PropertyName>/RegistryPackage/Slot/ValueList/Value</ogc:PropertyName>
                                <ogc:Literal>portrayal</ogc:Literal>
                            </ogc:PropertyIsEqualTo>
                        </ogc:And>
                    </ogc:Filter>
                </QueryConstraint>
            </Query>
        </GetRecords>
    </requestInfo>
    <searchStatus status="complete" timestamp="2004-03-16T09:28:08-0800"/>
```

```
    <searchResults resultSetId="urn:uuid:1f57afb6-d4fa-440c-9890-ab42151ea312"
expiration="2004-03-16T09:43:08-0800" hitCount="1" elementSet="full" recordSchema=""
numberOfRecords="1" nextRecord="0">
        <RegistryPackage xmlns="urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5"
id="urn:uuid:cb9b5bb4-456a-4226-80c4-800f2a4cc524" objectType="RegistryPackage"
status="Submitted" majorVersion="1" minorVersion="0">
            <Name/>
            <Description>
                <LocalizedString lang="en" value="Portrayal package for Vancouver
dataset."/>
            </Description>
            <Slot name="subject" slotType="http://purl.org/dc/elements/1.1/subject">
                <ValueList>
                    <Value>portrayal</Value>
                </ValueList>
            </Slot>
            <RegistryObjectList>
                <RegistryObject id="urn:uuid:6b7eeabf-96b1-4343-8c22-9e4f5461e7bb"
objectType="FeatureStyle" status="Submitted"/>
                <RegistryObject id="urn:uuid:8007279f-8442-45cc-bd53-b0e92dedec89"
objectType="FeatureStyle" status="Submitted"/>
                <RegistryObject id="urn:uuid:3b6a2ee7-f1e6-4f45-b753-d3543e91fbb5"
objectType="FeatureStyle" status="Submitted"/>
                <RegistryObject id="urn:uuid:90b15980-837f-43f0-88b5-bfb23b1a3cf0"
objectType="FeatureStyle" status="Submitted"/>
                <RegistryObject id="urn:uuid:371e680a-258b-474c-8367-fad4cfdacc73"
objectType="FeatureStyle" status="Submitted"/>
            </RegistryObjectList>
        </RegistryPackage>
    </searchResults>
</GetRecordsResponse>
```

Note that this result contains multiple feature style IDs. The following request retrieves one of them.

Query by ID:

```
urn:uuid:6b7eeabf-96b1-4343-8c22-9e4f5461e7bb
```

Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<FeatureStyle xmlns="http://www.opengis.net/sld" xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" version="1.0.20" xsi:schemaLocation="http://www.opengis.net/sld
http://schemas.cubewerx.com/schemas/sld/1.0.20/FeatureStyle.xsd">
    <Name>geosym-vmap0-hydro-inwatera</Name>
    <Description>
        <Title>Inland Water Areas</Title>
    </Description>
    <SemanticTypeIdentifier>nima:vmap0:hydro:inwatera</SemanticTypeIdentifier>
```

```
    <Rule>
        <Name>BH000</Name>
        <Description>
            <Title>Inland Water</Title>
        </Description>
        <ogc:Filter>
            <ogc:PropertyIsEqualTo>
                <ogc:PropertyName>F_CODE</ogc:PropertyName>
                <ogc:Literal>BH000</ogc:Literal>
            </ogc:PropertyIsEqualTo>
        </ogc:Filter>
        <PolygonSymbol>
            <BaseSymbol>
                <OnlineResource xlink:type="simple"
xlink:href="http://demo.cubewerx.com/ows12/wos/cwwos.cgi?oid=SYMBOLS.54"/>
            </BaseSymbol>
        </PolygonSymbol>
    </Rule>
    <Rule>
        <Name>BH090</Name>
        <Description>
            <Title>Land Subject to Inundation</Title>
        </Description>
        <ogc:Filter>
            <ogc:PropertyIsEqualTo>
                <ogc:PropertyName>F_CODE</ogc:PropertyName>
                <ogc:Literal>BH090</ogc:Literal>
            </ogc:PropertyIsEqualTo>
        </ogc:Filter>
        <PolygonSymbol>
            <BaseSymbol>
                <OnlineResource xlink:type="simple"
xlink:href="http://demo.cubewerx.com/ows12/wos/cwwos.cgi?oid=SYMBOLS.51"/>
            </BaseSymbol>
        </PolygonSymbol>
    </Rule>
</FeatureStyle>
```

3. Publication

When a resource such as an SLD stylesheet is being submitted as part of a CSW transaction request it is included as part of a multi-part message. Two variant encodings are possible, as described in the following RFC documents:

- RFC 2387, "The MIME Multipart/Related Content-type" <http://ietf.org/rfc/rfc2387>
- RFC 2388, "Returning Values from Forms: multipart/form-data" <http://ietf.org/rfc/rfc2388>

The "cid" URL scheme is used to reference other message parts. The scheme is documented in RFC 2392:

- RFC 2392, "Content-ID and Message-ID Uniform Resource Locators"
  http://ietf.org/rfc/rfc2392

3.1. Publish a style

Using the multipart/related content type, the HTTP request message looks like this, per RFC 2387:

```
POST /some/path HTTP/1.1
Host: www.some-host.com:8080
Accept: application/xml
Content-Type: Multipart/Related; boundary=part-boundary;
        type=application/xml;
        start="<requestId@galdosinc.com>"

--part-boundary
Content-Type: application/xml; charset=UTF-8
Content-ID: <requestId@galdosinc.com>


<?xml version='1.0' encoding='UTF-8'?>
<Transaction xmlns='http://www.opengis.net/cat/csw'
  xmlns:rim='urn:oasis:names:tc:ebxml-regrep:rim:xsd:2.5'>

  <Insert handle='insertSLD01'>
    <CSWExtrinsicObject
      id='some-uri'
      objectType='http://www.opengis.net/catalogue/csw/objectType/Stylesheet'
      status='Submitted'
      expiration='2002-12-31T12:00:00'
      mimeType='application/xml'
      isOpaque='false'>
      <rim:Name>
        <rim:LocalizedString value='A sample style sheet.'/>
      </rim:Name>
      <rim:Description>
        <rim:LocalizedString value='A feature style sheet that uses the OGC SLD
syntax.'/>
      </rim:Description>

      <repositoryItem>cid:sldtest1.xml@galdosinc.com</repositoryItem>
    </CSWExtrinsicObject>
  </Insert>
```

```
</Transaction>

--part-boundary

Content-Type: application/xml; charset=UTF-8
Content-Description: SLD style sheet
Content-ID: <sldtest1.xml@galdosinc.com>

<?xml version='1.0' encoding='UTF-8'?>
<FeatureTypeStyleCollection
  xmlns="http://www.opengis.net/sld"
  xmlns:ogc="http://www.opengis.net/ogc">
  <FeatureTypeStyle>
    <Name>Road Style</Name>
    <FeatureTypeName>Road</FeatureTypeName>
    <Rule>
      <LineSymbolizer>
        <Geometry>
            <ogc:propertyName>lineStringProperty</ogc:propertyName>
        </Geometry>
        <Stroke>
          <CssParameter name="stroke">red</CssParameter>
          <CssParameter name="stroke-width">4</CssParameter>
        </Stroke>
      </LineSymbolizer>
    </Rule>
  </FeatureTypeStyle>
  <FeatureTypeStyle>
    <Name>River Style</Name>
    <FeatureTypeName>River</FeatureTypeName>
    <Rule>
      <LineSymbolizer>
        <Geometry>
            <ogc:propertyName>lineStringProperty</ogc:propertyName>
        </Geometry>
        <Stroke>
          <CssParameter name="stroke">blue</CssParameter>
          <CssParameter name="stroke-width">6</CssParameter>
        </Stroke>
      </LineSymbolizer>
    </Rule>
  </FeatureTypeStyle>
</FeatureTypeStyleCollection>
```

If the transaction is successfully completed something like the following is returned:

```
<?xml version='1.0' encoding='UTF-8'?>
<TransactionResponse
xmlns='http://www.opengis.net/cat/csw'xmlns:ogc="<http://www.opengis.net/ogc>">

    <TransactionSummary requestId="urn:uuid:e9354092-b041-4d97-8299-8f582f1fbb6a">
        <objectsInserted>1</objectsInserted>
    </TransactionSummary>
    <!-- optional content follows -->
    <InsertResult>
        <ogc:RecordId>some-uri</ogc:RecordId>
    </InsertResult>
</TransactionResponse>
```

Otherwise, returned is a service exception reporting that the transaction failed.

Annex G – SMS Client Interactions (informative)

This annex describes the inner-workings of two representative Client Applications as "consumers" of SMS resources.

1. Galdos Client

1.1. Overview

Galdos provides three clients for the EMS-1 initiative: Symbol Editor, Map Style Editor and Feature Schema Editor.

There is a common purpose to all of these three clients: provide a visual tool for an easy discovering, editing and publishing their specific artefacts (symbols, styles and feature schemas respectively).

The separation of the clients comes from the toolkit-oriented design. The clients represent components with distinct and clearly separated functionality that allow for easy integration in an arbitrary custom client.

EMS-1 system with the Galdos client applications is shown in the following picture:

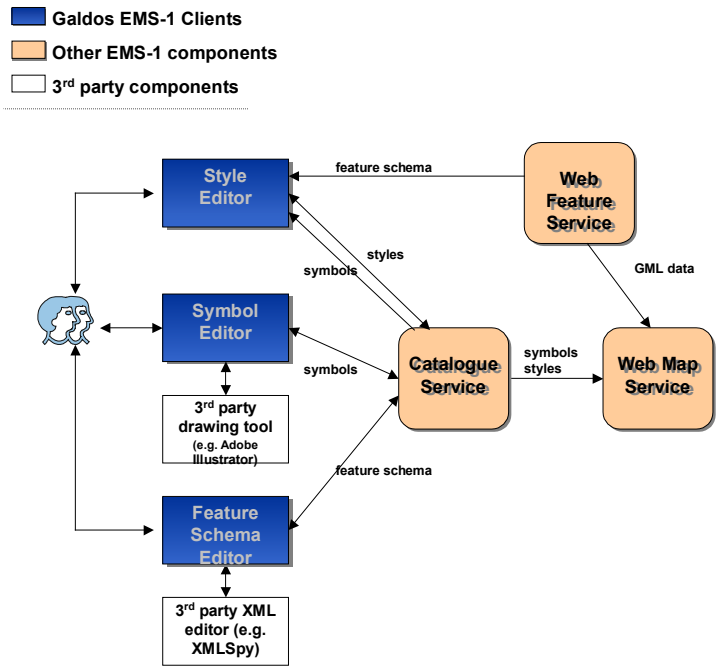


Figure 1: Galdos clients in the EMS-1 system

1.2. The Clients

Map Style Editor (MSE)

Map Style Editor is a client that allows for creating, editing and publishing feature styles. It supports two different style formats: OGC Styled Layer Descriptor (SLD) and W3C Extensible Stylesheet Language Transformations (XSLT).

Creating a style essentially means creating an association between a feature type (e.g. Road) and a symbol. Therefore, the MSE requires loading a desired feature schema and desired symbol collection prior to creating styles. It can use Catalogue for that, but can also use Web Feature Service, arbitrary Uniform Resource Identifier (URI) or a local drive.

Map Style Editor also allows for storing of styles to various locations; however, in the EMS-1 initiative we have focused on its capability to publish styles to the Catalogue.

Map Style Editor is used according to the following action sequence:

1. User loads a desired feature schema (from Catalogue or some other source).
2. User loads a set of symbols (from Catalogue or some other source)
3. User creates a number of feature styles by associating feature types from the schema with symbols. Alternatively, user loads an existing style from the Catalogue and modifies it.
4. User enters the metadata about styles.
5. User sends the style to the Catalogue for publishing.

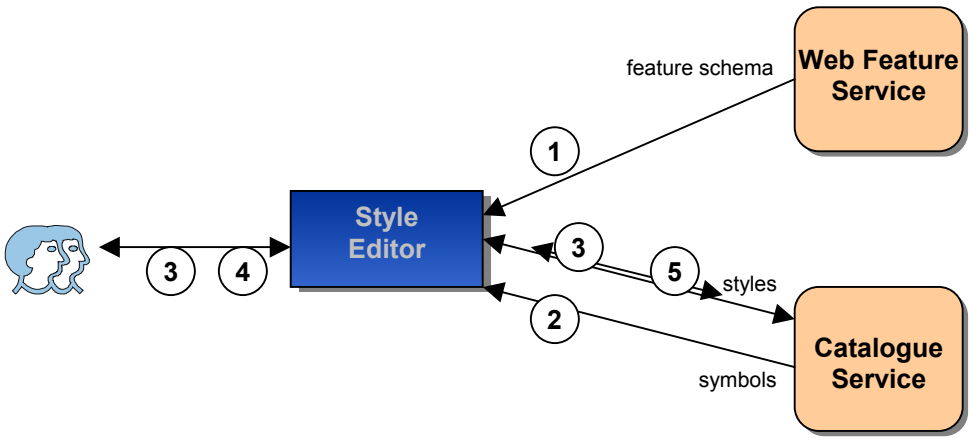


Figure 2: Map Style Editor action sequence

Symbol Editor

Symbol Editor provides the visual capability for retrieving, creating/modifying and publishing of symbols. A symbol is a graphical object, expressed using Scalable Vector Graphic grammar.

Symbol Editor uses a 3rd party drawing application to perform the graphical creation of a symbol. Any drawing application conforming to simple start-up rules (described in the extended documentation of the Symbol Editor) can be used.

The following action sequence describes how the Symbol Editor is used:

1. User creates a new symbol artefact in the client or loads an existing one from the Catalogue.
2. User selects an option in the client that opens a desired graphical drawing tool. User draws the symbol and saves it on the local drive.
3. User types in the symbol metadata.
4. User browses the symbol classification schemes that exist on the target Catalogue service.
5. User selects a classification scheme and classifies the symbol according to it.
6. User sends the symbol with the metadata to the Catalogue for publishing.

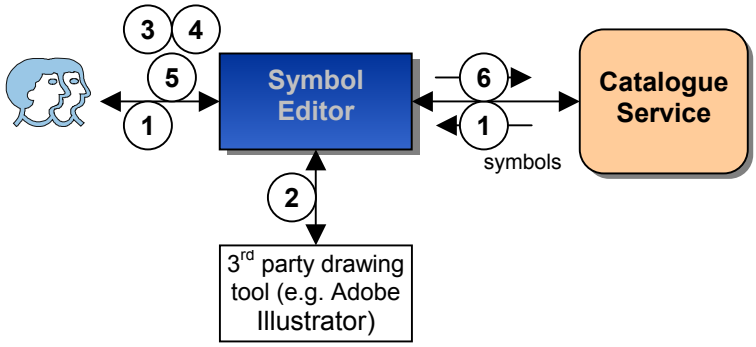


Figure 3: Symbol Editor action sequence

Feature Schema Editor

Feature Schema Editor provides the capability for retrieving, creating/modifying and publishing of feature schemas.

Feature Schema Editor uses a 3rd party XML Editor application to perform the editing of the schemas. (Just like in the case of symbols, the editing capability is very common and provided by many different commercial or even free tools)

Any XML Editing application conforming to simple start-up rules (described in the extended Schema Editor documentation) can be used. Presently, the Schema Editor is integrated with couple of well-known commercial XML Editor applications, but the list can be easily extended.

The following action sequence describes how the Schema Editor is used:

1. User browses the schemas in the Catalogue service and loads one of them. Alternatively, user wants to create a new schema in which case proceeds immediately to the step 2.
2. User selects an XML Editor from the list of editors available in the client. User selects an option in the client that opens the selected XML Editor.

3. User modifies the schema, and, depending on the capability of the XML Editor, either saves the schema to the local drive or simply closes the XML Editor (in which case the modified schema is transferred to the Schema Editor automatically).
4. User types in the schema metadata.
5. User sends the schema with the metadata to the Catalogue for publishing.

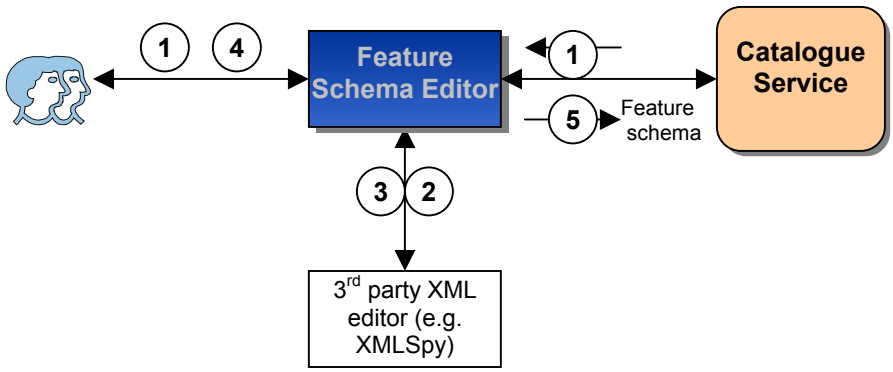
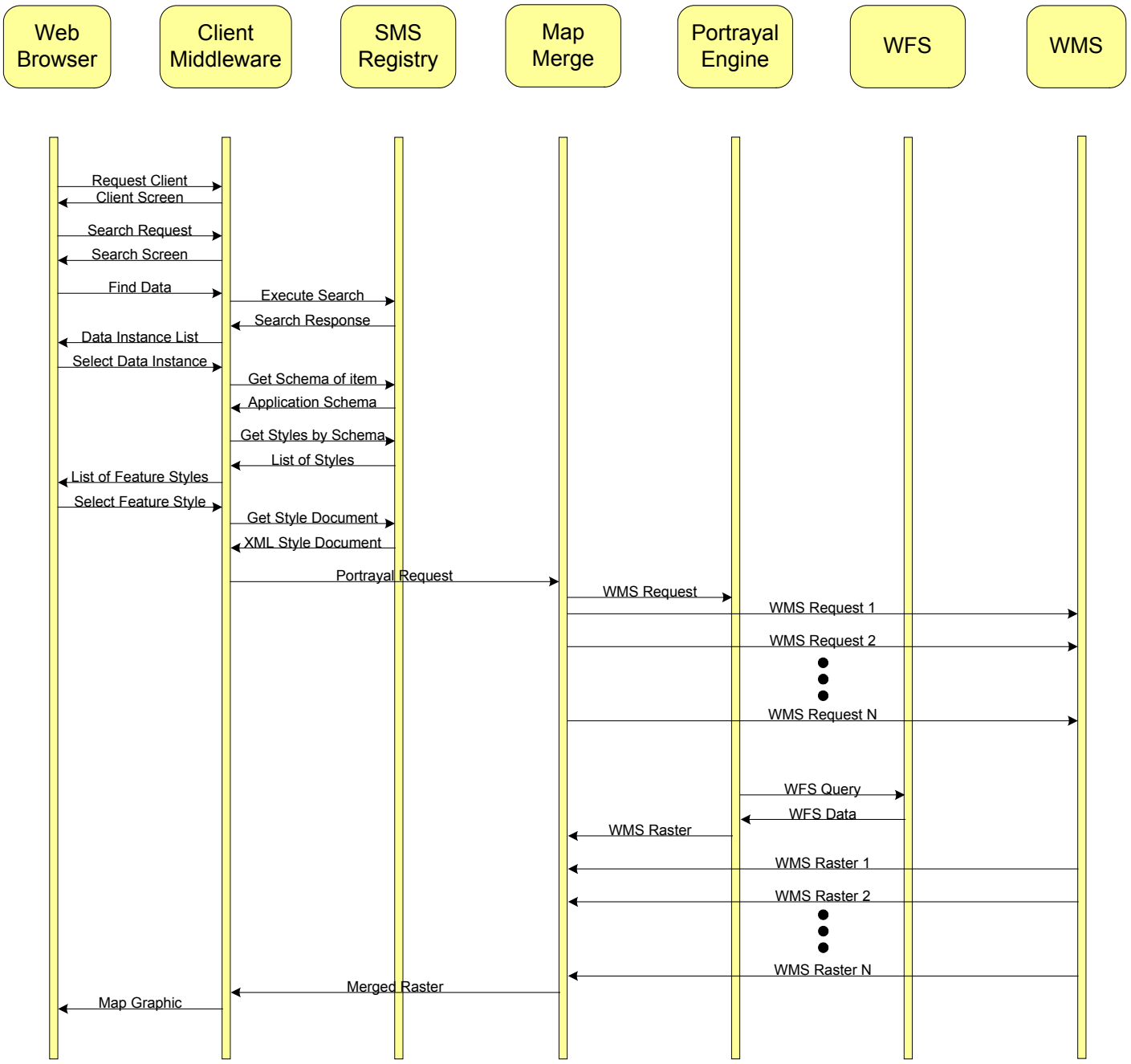


Figure 4: Schema Editor action sequence

2. Compusult Client

2.1. Overview

The following sequence diagram shows the flow of a web mapping system using an SMS Registry to find WFS data instances and generates a map using a portrayal engine. This example includes accessing layers from other WMS services and assumes the web mapping system uses a map-merging module to generate a single graphic on the server from multiple WMS requests.



Bibliography

Refer to the OGC website (http://www.opengis.org/specs/?page=baseline) for the authoritative listing of adopted documents.

Note: Please contact the OGC Tech Desk if you need assistance in gaining access to these documents (techdesk@opengis.org).

OGC Specifications and Supporting Documents Relevant to EMS-1:

[1] Style Management Services (SMS) Discussion Paper, OpenGIS® Project Document 03-031. Available at: http://www.opengis.org/specs/?page=discussion

[2] Style and Symbol Management Services Requirements Interoperability Program Report, OpenGIS® Project Document 03-030. Available at: http://portal.opengis.org/index.php?m=projects&a=view&project_id=82&tab=2&artifact_id=1238

[3] OpenGIS® Style Layered Description (SLD) Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***

[4] OpenGIS® Geography Markup Language (GML) Implementation Specification (version 3.0), available at: http://www.opengis.org/specs/?page=specs

[5] OpenGIS® Web Feature Server (WFS) Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***

[6] OpenGIS® Filter Encoding Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***

[7] OpenGIS® Web Map Service (WMS) Implementation Specification, version 1.2, available at: ***http://www.opengis.org/specs/?page=specs***

[8] OpenGIS® Map Context Documents Implementation Specification, version 1.0, available at: ***http://www.opengis.org/specs/?page=specs***

[9] Coverage Portrayal Service Specification (CPS) Interoperability Program Report. OpenGIS Project Document 02-019r1, available at: http://portal.opengis.org/index.php?m=projects&a=view&project_id=82&tab=2&artifact_id=1235

[10] OpenGIS® Catalog Service Implementation Specification, version 2.0, available at: http://portal.opengis.org/files/?artifact_id=5929&version=1 (approval by electronic vote pending)

[11] Catalog Services – Web Application Profile (CSW). OpenGIS® Project Document 03-094 (version 0.8.3), available at: http://portal.opengis.org/files/files/?artifact_id=4412

[12] OpenGIS® Web Registry Service Interoperability Program Report. OpenGIS® Project Document 03-024, available at: http://portal.opengis.org/index.php?m=projects&a=view&project_id=82&tab=2&artifact_id=1238

[13] Registry Service Requirements Interoperability Program Report. OpenGIS® Project Document 03-027. Available at: http://portal.opengis.org/index.php?m=projects&a=view&project_id=82&tab=2&artifact_id=1238

[14] Web Object Service Discussion Paper, OpenGIS® Project Document 03-013. Available at http://www.opengis.org/specs/?page=discussion

[15] Critical Infrastructure Collaborative Environment (CICE) Architecture: Enterprise Viewpoint Discussion Paper, OpenGIS Project Document 03-061. Available at http://www.opengis.org/specs/?page=discussion

[16] Critical Infrastructure Collaborative Environment (CICE) Architecture: Information Viewpoint Discussion Paper, OpenGIS Project Document 03-062r1. Available at http://www.opengis.org/specs/?page=discussion

[17] Critical Infrastructure Collaborative Environment (CICE) Architecture: Computation Viewpoint Discussion Paper, OpenGIS Project Document 03-063r1. Available at http://www.opengis.org/specs/?page=discussion

[18] Critical Infrastructure Collaborative Environment (CICE) Architecture: Engineering Viewpoint Discussion Paper, OpenGIS Project Document 03-055r1. Available at http://www.opengis.org/specs/?page=discussion

Other OGC Specifications and Supporting Documents

[19] OpenGIS® Abstract Specification, Topics 1-17), available at: http://www.opengis.org/specs/?page=abstract

[20] OpenGIS® Reference Model, version 0.1.2, available at: http://www.opengis.org/specs/?page=orm

[21] OpenGIS® Web Service Architecture Discussion Paper. OpenGIS Project Document 03-025, available at: http://www.opengis.org/specs/?page=discussion

[22] OpenGIS® Grid Coverages Implementation Specification, version 1.0, available at: http://www.opengis.org/specs/?page=specs

[23] OpenGIS® Coordinate Transformation Services Implementation Specification, version 1.0, available at: http://www.opengis.org/specs/?page=specs

[24] Web Coverage Service Implementation Specification, Version 0.0. OpenGIS® Project Document 02-076r3, available at http://www.opengis.org/specs/?page=specs

[25] OpenGIS® Location Services (OpenLS): Core Services [Parts 1-5], OpenGIS® Project Document 03-006r1. Available at: http://www.opengis.org/specs/?page=specs

[26] OpenGIS® Location Services (OpenLS): Navigation Service [Part 6], OpenGIS® Project Document 03-007r1. Available at: http://www.opengis.org/specs/?page=specs

[27] Service Information Model Discussion Paper, OpenGIS® Project Document 03-026. Available at: http://www.opengis.org/specs/?page=discussion

[28] Web Terrain Service Discussion Paper, OpenGIS® Project Document 01-061. Available at: http://www.opengis.org/specs/?page=discussion

[29] XML for Imagery and Map Annotations (XIMA) Discussion Paper, OpenGIS® Project Document 01-019. Available at: http://www.opengis.org/specs/?page=discussion

[30] Web Coordinate Transformation Service Discussion Paper, OpenGIS® Project Document 02-061r1. Available at http://www.opengis.org/specs/?page=discussion

[31] Recommended Definition Data for Coordinate Reference Systems and Coordinate Transformations, OGC Recommendation Paper, version 1.1.0, available at: http://www.opengis.org/specs/?page=recommendation

ISO Specifications

[32] ISO 19101:2002 (Reference Model): http://webstore.ansi.org/ansidocstore/product.asp?sku=ISO+19101:2002

[33] ISO 19107 (Spatial Schema): http://www.isotc211.org/protdoc/DIS/ISO_DIS_19107_(E).pdf

[34] ISO 19108 (Temporal Schema): http://www.isotc211.org/protdoc/DIS/DIS19108.pdf

[35] ISO 19109 (Rules for Application Schema): http://www.isotc211.org/protdoc/DIS/ISO_DIS_19109_(E).pdf

[36] ISO 19110 (Methodology for Feature Cataloguing): http://www.isotc211.org/protdoc/DIS/ISO_DIS_19110_(E).pdf

[37] ISO 19111 (Spatial Referencing by Coordinates): http://www.isotc211.org/protdoc/DIS/DIS19111.pdf

[38] ISO 19112 (Spatial Referencing by Geographic Identifiers): http://www.isotc211.org/protdoc/DIS/ISO_DIS_19112_(E).pdf

[39] ISO 19115 (Metadata) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19115_(E).pdf

[40] ISO 19117 (Portrayal) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19117_(E).pdf

[41] ISO 19119 (Services) : http://www.isotc211.org/protdoc/DIS/ISO_DIS_19119_(E).pdf

[42] ISO 19123 (Schema for Coverage Geometry and Functions): http://www.isotc211.org/protdoc/211n1227/readme.htm

[43] ISO 19125-1 (Simple Features Access - Part 1: Common Architecture): http://www.isotc211.org/protdoc/DIS/DIS19125-1.pdf

[44] ISO 19125-2 (Simple Features Access - Part 2: SQL option): http://www.isotc211.org/protdoc/DIS/DIS19125-2.pdf

[45] ISO 19128 (Web Mapping): http://www.isotc211.org/protdoc/211n1331/211n1331.pdf

Other Related Specifications:

[46] Uniform Resource Identifiers (URI): Generic Syntax (RFC 2396) T. Berners-Lee, R. Fielding, L. Masinter, available at: http://www.ietf.org/rfc/rfc2396.txt

[47] Extensible Markup Language (XML) 1.0, Second Edition, Tim Bray et al., eds., W3C, 6 October 2000. See http://www.w3.org/TR/2000/REC-xml-20001006

[48] XML Schema Part 1: Structures. World Wide Web Consortium (W3C).. W3C Recommendation (2 May 2001). Available [online]: http://www.w3.org/TR/xmlschema-1/

[49] XML Linking Language (XLink) Version 1.0, DeRose, S., Maler, E., Orchard, D., available at http://www.w3.org/TR/xlink/

[50] Web Services Description Language (WSDL) Version 1.2. W3C Working Draft (9 July 2002). World Wide Web Consortium (W3C). Available [online]: http://www.w3.org/TR/wsdl12/

[51] Simple Object Access Protocol (SOAP) 1.1, Box, D., et. al., available at http://www.w3.org/TR/SOAP/

[52] UDDI – Universal Description, Discovery, and Integration, see http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=uddi-spec

[53] Registry Information Model v2.1, OASIS/ebXML Registry Technical Committee (Approved Committee Specification, June 2002). See http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=regrep

[54] Registry Services Specification v2.1. OASIS/ebXML Registry Technical Committee (Approved Committee Specification, June 2002). See http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=regrep