A User Comment Draft of an NTCIP Information Report

NTCIP 9010 v01.07

National Transportation Communications for ITS Protocol

XML in ITS Center-to-Center Communications

October 2003

This is a draft document, which is distributed for review and comment purposes only. You may reproduce and distribute this document within your organization, but only for the purposes of and only to the extent necessary to facilitate review and comment to the NTCIP Coordinator. Please ensure that all copies reproduced or distributed bear this legend. This document contains preliminary information that is subject to change.

Published by

American Association of State Highway and Transportation Officials (AASHTO)
444 North Capitol Street, N.W., Suite 249
Washington, D.C. 20001

Institute of Transportation Engineers (ITE)
1099 14th Street, N.W., Suite 300 West
Washington, D.C. 20005-3438

National Electrical Manufacturers Association (NEMA)
1300 North 17th Street, Suite 1847
Rosslyn, Virginia 22209-3801

INFORMATION REPORT CONTRIBUTORS

Technical Editors:

Manny Insignares, ConSysTec
Warren Tighe, Siemens ITS

Editors and Contributors:

Mike Brown, Southwest Research
Jeff Brummond, Iteris
Blake Christie, Mitretek
Steve Dellenback, Southwest Research
David Kelley, SubCarrier Systems
Scott Lee, Illinois DOT
Jeff Mayo, TransCore ITS
Brian Pecus, CalTrans
Robert Rausch, TransCore ITS
Bruce Schopp, NEMA
Ed Seymour, Texas Transportation Institute
Ray Starr, Minnesota DOT
Jeff Thorn, New York State DOT
Warren Tighe, Siemens ITS
Walt Townsend, Siemens ITS
Ken Vaughn, Trevilon
Jeris White, Mitretek
David Zavattero, Illinois DOT

DOCUMENT HISTORY

XML in ITS Center-to-Center Communications White Paper, September 30, 2002
NTCIP C2C XML-based Standard Development Concept Paper, April 8, 2003
Internal WG Draft, XML in ITS Center-to-Center Communications Information Report, June 6, 2003
WG Review Draft, XML in ITS Center-to-Center Communications Information Report, June 30, 2003
NTCIP 9010 v01.05 -- JC Review Draft, XML in ITS Center-to-Center Communications Information Report, August 8, 2003
NTCIP 9010 v01.06 -- User Comment Draft, XML in ITS Center-to-Center Communications, October 28, 2003
NTCIP 9010 v01.07 – Revised title page and copyright statement, November 11, 2003.

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

ACRONYMS

AASHTO	American Association of State Highway and Transportation Officials
ADUS	Archived Data User Systems
ASCII	American Standard Code for Information Interchange
ASN	Abstract Syntax Notation
ASP	Active Server Pages (by Microsoft)
ATIS	Advanced Traveler Information Systems
Axis	Apache's Implementation of SOAP
C2C	Center to Center
CARS	Conditional Acquisition Reporting System
CCTV	Closed Circuit Television
CDR	Common Data Representation
CGI	Common Gateway Interface
CORBA	Common Object Request Broker Architecture
DATEX	DATa Exchange
FTP	File Transfer Protocol
GRM	Generic Reference Model
HTTP	HyperText Transfer Protocol
HTTPS	Secure HyperText Transfer Protocol
ID	Identification
IDL	Interface Definition Language
IEEE	Institute of Electrical and Electronic Engineers
IIOP	Internet Inter-ORB Protocol
IM	Incident Management
IP	Internet Protocol
ISPs	Information Service Providers
ITE	Institute of Transportation Engineers
ITS	Intelligent Transportation Systems
JSP	Java Server Pages
MSETMCC	Message Sets for External Traffic Management Center Communications
NEMA	National Equipment Manufacturers Association
NTCIP	National Transportation Communication for ITS Protocol
Oasis	Oasis Group
OER	Octet Encoding Rules
ORB	Object Request Broker
PERL	Practical Extraction and Report Language
SAE	Society of Automotive Engineers
SDOs	Standards Development Organizations
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SSL	Secure Sockets Layer
Sync	Synchronized
TCIP	Transit Communications Interface Profiles
TCP/IP	Transmission Control Protocol and the Internet Protocol
TMDD	Transportation Management Data Dictionary
TREX	Metro Denver Transportation Expansion Project
UDDI	Universal Description Discovery and Integration

ACRONYMS

UDP	User Datagram Protocol
UML	Unified Modeling Language
W3C	World Wide Web Consortium
WG	Working Group
WSDL	Web Services Description Language
XML	eXtensible Markup Language
XSL	eXtensible Stylesheet Language
XSLT	eXtensible Stylesheet Language Transform

1 INTRODUCTION

This NTCIP Information Report provides an overview of the issues involved in using XML-based technologies for Intelligent Transportation Systems (ITS) data exchange. It was developed by the NTCIP Center-to-Center Working Group (NTCIP C2C WG) to identify the opportunities and needs XML creates for the National Transportation Communications for ITS Protocol (NTCIP) effort.

This paper is written for the technical transportation system professional. An appendix includes technical information regarding XML intended for a purely technical audience. This document is not intended to be a tutorial on XML or XML-based standards, but rather provides sufficient background to support the discussion related to the development of an XML-based C2C standard consistent with the working group's mission.

This report also attempts to minimize the use of jargon, and uses deliberately abbreviated or simplified descriptions of many items and concepts in order to make the information useful for the non-technical professional. Any number of other papers, books, and web sites offer more detailed and comprehensive treatment of most of the subjects described herein.

1.1 NTCIP C2C Working Group Mission

The core mission of the NTCIP C2C WG is to: develop and maintain standards, develop application guidance for open systems, non-proprietary-based product development, and facilitate the procurement, implementation, and testing of integrated ITS center systems.

A standards-based approach can assist system applications, regardless of operating system or programming language, to communicate using simple encoded messages that both applications understand. Specifically, the goal related to development of an XML-based C2C standard is to support the communication of information and/or control commands between ITS centers.

This information report will outline two approaches for implementation of XML-based communication in center-to-center systems: first, an approach to support robust command and control leveraging the existing standards of the World Wide Web Consortium (W3C), and second, a file-based sharing approach with a focus on information sharing and aggregation. The C2C WG has coined the term "XML Direct" to refer to this file-based approach.

1.2 Current Applications Using XML for Center-to-Center Exchange

One of the driving factors in the development of this paper is the number of on-going or planned efforts that will implement XML for data exchange between transportation centers. The following are representative examples:

- Gary-Chicago-Milwaukee Corridor (planning XML traveler information to Information Service Providers (ISPs) and agencies)
- TravInfo (traveler information dissemination - San Francisco Bay area)
- CARS (rural traveler information – various western and mid-west states)
- TREX (I-25 in Denver – traveler information to ISPs and agencies over the web)

These implementations have been developed separately; using conventions customized for each project, and therefore are not guaranteed to be interoperable. There is no standard protocol or message set for exchanging data encoded in XML. XML in itself does not define dialogues, delivery mechanism/protocol, subscribe-once/publish-many mechanisms, login/logout mechanism, security protocol, discovery mechanism, or other services or functionality provided by protocols such as DATEX-ASN and CORBA. Such services may not be necessary for simple data distribution networks, but are needed to support the full range of center-to-center communications applications which can involve functions such as dynamic subscriptions, dynamically configured data filtering at the source, remote command and control, and security checks prior to sending sensitive data or responding to commands.

In addition, other ITS Standards efforts (outside of the NTCIP C2C WG) are using XML to document message formats and dialogs, including:

- NTCIP Electrical Lighting Management System Standards
- SAE Advanced Traveler Information System Messages
- IEEE 1512 Messages
- TMDD Message Sets

1.3 Report Organization

This information report is organized into 7 sections as follows:

Section 1 – Introduction. Covers general information as to the purpose and audience for this report.

Section 2 – Overview of XML. Provides a high-level introduction to XML concepts geared towards a non-technical audience.

Section 3 – C2C Communications Requirements. This section focuses on a discussion of the basic needs, requirements, and concepts related to center-to-center communications and system interoperability.

Section 4 – Technical Approach Discussion. Provides a general discussion related to the development approach for an XML-based center-to-center communications standard.

Section 5 – NTCIP C2C Working Group Recommendation. This section outlines 2 recommended approaches for standardization: 1) a technical approach based on the World Wide Web's Web Services Architecture, and 2) an XML file-based messaging approach.

Section 6 – Appendix. The appendix discusses a number of technical issues that may be included in the standard or that may be useful to those planning to deploy XML-based solutions.

2 XML OVERVIEW

XML, the eXtensible Markup Language, is a standard of the World Wide Web Consortium (W3C). XML is a means by which one computer can encode some information (data) so that another computer receiving that encoded information will be able to understand its contents and act on that content (e.g., process the information, display the information to a human, store the information in a database, issue a command to a field device, etc.). Unlike most computer encoding standards (e.g., Basic Encoding Rules, Octet Encoding Rules, Packed Encoding Ruled, Common Data Representation, Hypertext Markup Language, etc.), there is no single set of encoding rules for XML. Instead, XML encoding rules are customized for different applications. Furthermore, XML encoding rules include a mechanism for identifying each element of an XML document or message.

Following is an example of some information encoded using XML.

```
<SignalController ID=45 Owner=”City of Utopia”>
  <HardwareType>Wiz Bang 250</HardwareType>
  <FirmwareType>Independent</FirmwareType>
  <FirmwareVersion>1.5</FirmwareVersion>
  <Group>5</Group>
  <Detector ID=23>
    <HardwareType>InductiveLoop</HardwareType>
    <Lane>2</Lane>
    <Dimensions>
      <Width Units=”cm”>240</Width>
      <Length Units=”cm”>600</Length>
    </Dimensions>
  </Detector>
</SignalController>
```

This example describes a traffic signal controller, number 45, owned by the city of Utopia. It is a Wiz Bang 250 controller with Independent firmware, version 1.5. It is a member of signal group 5. It is associated with one detector, which is an inductive loop in lane 2, with dimensions of 240 by 600 centimeters.

Notice that the encoded information is comprised of English-language characters, or text. This is different from many encoding schemes that encode to a binary format.

2.1 XML Terminology

Once information is described (encoded) using XML, it is called a **document**. An XML document is comprised primarily of **elements**. An element is delimited by **tags**. Usually, a tag is comprised of the name of the element it delimits, enclosed between a pair of angle brackets (“<” and “>”). Tags usually come in pairs – a start-tag and an end-tag. The element name in the start and end tag are identical except that the latter has a forward slash ("/") at its beginning. A start-tag may optionally include **attributes** of the element. An attribute name is followed by an equal sign and a value for that attribute. The **content** of an element is the text that appears between its start-tag and its end-tag. Elements may be **nested** – the content of an element can include other elements.

Nesting allows for complex **information structures**. In the above example, nesting is used to show that an element (a piece of information) called "SignalController" is made up of other elements (HardwareType, FirmwareType, FirmwareVersion, Section, and Detector), and that Detector information in turn is comprised of other elements (HardwareType, Lane, Dimensions), and that Dimension information in turn is comprised of still other elements (Width, Length).

The encoding rules, or grammar, for a particular application are usually defined using an **XML Schema**[1]. This is a definition of things such as the set of allowable tags (e.g, element types), attributes of each element type (if any), default values (if any), and how elements can be nested (data structure). A message or document describing a traffic signal would use a different XML schema than a message or document describing an incident. If desired, the schema can be included within the document, to enable a receiving computer to validate the encoding. XML schemas are defined using the **XML schema language**.

An XML schema is **extensible**. Additional element types can be added at any time in the future without invalidating earlier versions of the schema. The XML standard requires software that reads and interprets an XML document (called an **XML parser**) to ignore any tags (and elements thus delimited) that are not included in the version of the schema currently used by that parser.

The **eXtensible style sheet language (XSL)** can be used to enable a computer to automatically translate the information content of an XML document, into another formatted document (must be in text format) including XML.

[1] XML Schema is a standard of the World Wide Web Consortium and becoming the preferred method of defining the grammar for an XML document type, replacing the original method called Document Type Definition (DTD).

3 CENTER-TO-CENTER COMMUNICATIONS STANDARDS REQUIREMENTS

ITS communications requirements are often divided into two categories according to the general environment in which the communications take place. One category is center-to-field communications, such as occurs for remote management of field devices and fleet vehicles. The other category is center-to-center communications.

For the purpose of this discussion, a “center” can be thought of as a computer. Center-to-center communications involve data exchange between computers. A computer may be in a traffic management center, a transit management center, a traveler information center, an incident or emergency management center, etc. The data sent from one center or computer to another may be any of a wide variety.

To avoid wasting bandwidth and overloading the communications networks that link computers, data are often sent between computers only in response to a request for information. One center may request another center to send a particular type of data just once, or might request that the data be sent repeatedly, say every time there is a change. Many centers, such as those that provide traveler information generate relatively small quantities of data over time, say an average of one new piece of information every few seconds or minutes. But other centers, such as those that manage large traffic signal systems, can generate literally thousands of new data values every second.

Some centers use received data for human consumption only, while at other centers much of the received data may be used not by a human, but by a computer that processes them in real-time to perform automated functions (e.g., change a traffic signal timing pattern, adjust a ramp metering rate, notify a human of an alarm condition, etc.).

Centers most often exchange data with other centers of the same type, but there is increasing call for centers of different types to exchange data. For example, a transit or traffic management center has information about congestion and service delays of value to a traveler information center, an incident management center has incident status information of value to a traffic or transit management center, a transit system may request a traffic management system to provide signal priority for a bus that is running behind schedule, etc.

Centers most often exchange data with adjacent centers, often in the same metropolitan area. But ITS centers are rapidly spanning the nation. For example, regional traveler information systems, rural traffic management systems, and weather monitoring systems often span large geographic areas that may include multiple metropolitan areas. In some parts of the nation, population increases and suburban sprawl are leading to the merging of metropolitan areas. What starts out as a small group of centers exchanging information among themselves can grow to a much larger network of connected centers, and adjacent center-to-center networks may need to merge.

Sometimes a single ITS project will install the same software at multiple centers and create a center-to-center communications network between those centers all at once, ensuring they can all successfully communicate with each other. But after that project is completed, other centers may wish to join the network, or an adjacent center-to-center network using quite different software may need to merge with it. If the communications protocols and standards used in each network are different, a large and expensive effort may be needed.

In an ideal world, any transportation-related center could get any information it needs from any other center, without having to modify its software or support multiple interfaces to communicate with different centers or groups or centers in different ways. This is the goal of center-to-center ITS communications standards.

3.1 C2C Communications Standards Concepts and Interoperability

Successful data exchange between centers requires the involved centers to agree on several key items such as the following:

- The mechanism, or **message patterns**, by which a message is requested or triggered (e.g., The Incident Message is sent only when a new incident is first created and thereafter when a change in status occurs, and only to those centers that previously sent a subscription message requesting incident data). Message patterns, also referred to as **message dialogs**, define the relationships between messages.
- The structure of the **message**, which defines the data elements that make up a message, (e.g., Message 1 contains three elements – Incident ID, Incident Type, and Incident Status in that order) are catalogued in a **message set**. Like data dictionaries, the SDOs are developing message sets based on transportation functional area. Message sets can be defined by an XML Schema.
- The definition of **data elements** in the requested message (e.g., Incident Type 3 means an accident involving at least one fatality). Data elements are defined in a **data dictionary**. ITS Standards Development Organizations, or SDOs, have defined data dictionaries for various transportation functional areas (transit, traffic management, traveler information, archived data, and emergency/incident management).
- The rules used to **encode the data** into computer readable format (e.g., Incident Type is encoded as an enumerated short integer, with valid values lying between 0 and 8 and a default value of 0 if the element is missing from the message). XML is encoded using the ASCII format. The valid tags that can be used in an XML document for a specific area of application are specified in the XML Schema.
- The transmission **protocol** used to transmit the message between one computer and another. (e.g., The Incident Message is transmitted using the Transmission Control Protocol and the Internet Protocol (TCP/IP)).

These are the types of things center-to-center communications standards specify. There are more issues that are not mentioned here, such as security (e.g., login/logout, user ID, password, and encryption), compression algorithms, concatenation of messages in one transmitted packet, naming convention, etc. **Separate implementations will not be interoperable until a standard is in place for all components.**

3.2 XML-based Communications Standards Development

To date, two families of standards have been defined for center-to-center communications. One is based on the International Standards Organization's DATEX-ASN protocol, and the other is based on the Object Management Group's CORBA standard. A third family of standards will be defined to support XML-based implementations. Within the XML family a set of communications profiles will be defined that are based on the World Wide Web Consortium's XML Activity and Web Service Architecture, and another profile will define a simple file-sharing approach.

The following table summarizes the major standards used to specify the various components for each of the existing two center-to-center communications families (DATEX and CORBA), and equivalent standards for the new XML family. Notice that all three families can share a common set of standards for data element definitions (data dictionaries), description of message dialogs, and transmission protocols.

Table 1. Center-to-Center Protocol Comparison

Communications Protocol	DATEX-ASN	CORBA	W3C-based XML Protocol	XML Direct Protocol
Data Dictionaries	Functional Area Data Dictionaries (TMDD, TCIP, ATIS, IM, ADUS, etc.)			
Message Dialogs	Defined in sequence diagrams in the functional area standards.			
Rules for Defining Messages/Objects	ASN.1 (Abstract Syntax Notation) e.g., MS-ETMCC, TCIP, ATIS, IM, etc.	CORBA IDL (Interface Definition Language)	• XML Schema Language • SAE ASN.1 to XML Encoding Rules • WSDL (Web Services Description Language)	• XML Schema Language • SAE ASN.1 to XML Encoding Rules
Encoding Rules for Transmission of Data/Objects	OER (Octet Encoding Rules)	CDR (Common Data Representation)	• XML is encoded as ASCII text • Functional area XML Schemas define valid tags	
Application Protocol (handshaking, message framing, etc.)	DATEX-ASN	IIOP (Internet Inter-ORB Protocol)	• SOAP (Simple Object Access Protocol) over HTTP for the W3C Approach.	• XML Direct - FTP and HTTP
Other Services (e.g., discovery, security, aggregation, subscriptions, etc.)	DATEX-ASN supports a robust subscription-based service and administrative messages	CORBA services, Near Real-Time Data Service, Administrative Objects, Naming Service, Trader Service	• Possibly UDDI (Universal Description, Discovery and Integration). An OASIS Group Standard, for Discovery.	
Transport Protocol	TCP/IP and UDP/IP			

Note: CORBA = Common Object Request Broker Architecture (see www.omg.org)
DATEX-ASN = Data Exchange Using ASN.1 (ISO 14827 Parts 1 & 2)

4 TECHNICAL APPROACH DISCUSSION

4.1 Methodology: Systems Engineering Approach

The recommended approach for development of the XML-based standard is to continue using the Systems Engineering Methodology already in use by the C2C WG. The following table depicts (though certain steps and detail are missing) the Systems Engineering Methodology (as outlined in a Mitretek Report: *Building Quality Intelligent Transportation Systems Through Systems Engineering*, April 2002) as it applied to current SDO C2C activities.

Table 2. Mapping of Systems Engineering Methodology to SDO Standards

Systems Engineering Step	Resource Documents and Content
Needs Analysis	Concept of Operations (ConOps), part of the GRM • Use Case Diagrams • Use Case Specifications
Requirements	Requirements Specification
System Architecture	National ITS Architecture for high-level standards requirements Regional ITS Architecture for system implementations
Design	Sequence Diagrams, part of the GRM Class Diagrams and specifications, part of the GRM
Testing	Requirements Traceability Matrix Test Plan and Procedures

Developing an XML-based standard will be based on the Concept of Operations and Functional Requirements, Sequence Diagrams, and XML Schemas defined by other transportation functional area SDOs. Specifically, the XML-based standard developers will require:

- Information Model – to describe messages and message elements (primitives) from which can be defined an XML Schema
- Message Dialogs – that shows the relationships and sequencing of messages
- Processing Requirements – to define what a center-system must do when it receives a message. The dialogs should identify the message inputs and outputs.
- Requirements Traceability – to allow development of test procedures to show that the system does what the system is intended to do

4.2 XML-Standards Development Approach across Multiple Center Types

It would be advantageous to be able to use the same XML-based C2C standard for use across centers of different types – for example, traffic management, emergency management, traveler information service provider, transit, and archived data management centers. Once the individual functional area standards bodies have developed requirements, messages and dialogs, much of the approach and technologies used in implementing a center-based communications standard would be the same.

Close collaboration among the various SDOs developing XML-based protocols (e.g., SAE ATIS, IEEE IM, TCIP) is warranted as interoperability across ITS centers will require a similar XML-based approach for C2C communications.

4.3 Benefits and Challenges of XML

4.3.1 Existing Support

XML and related protocols commonly used for web-based communications are very broadly supported in the general computing and information technology industry. Experienced personnel, off-the-shelf software, and support tools are readily available and relatively inexpensive.

4.3.2 Human Readable

Because XML data are exchanged in a tagged text format, it is possible to directly read and understand the message content. This can help during system development and debugging, and allows direct review of the system interfaces at any time.

4.3.3 Bandwidth and Latency

The bytes in an XML transmission use ASCII encoding and human readable names for data types and attributes. Since the data are transmitted in text format, it takes more bandwidth to transmit the data than binary encoding formats.

In addition, there is overhead involved in the encoding and decoding (parsing) of the textual format. This can add latency to the delivery and processing of a message.

Compression schemes are available to reduce message size but add complexity and cost, and have interoperability considerations. In addition, the time taken to compress and de-compress information content will increase communications latency. The degree to which this is a significant issue depends on the application.

Bandwidth and latency issues are not a problem for many C2C applications, but may be a problem for some real-time applications such as second-by-second traffic signal status monitoring, and remote control of pan-tilt-zoom CCTV cameras.

4.3.4 Security

Different center-to-center applications have different security requirements. Security in this context relates to issues such as prevention of unauthorized access to the data being transmitted, and prevention of unauthorized introduction of spurious data or commands. XML itself has no provision for security, but various off-the-shelf security services can be used to deliver XML messages and for private use of the Internet. It should be noted that use of security measures such as Secure Sockets Layer (SSL) for encryption adds further complexity and overhead, and have interoperability considerations.

On the other hand, if security is not a concern, a potential benefit of using XML over a protocol such as Hypertext Transfer Protocol is that the messages will pass through a standard firewall, whereas protocols used with CORBA and DATEX require special measures (for example, DATEX would require the network administrator to allow communications via port 355). This can be particularly significant when using the Internet for center-to-center communications.

4.3.5 Connectionless and Stateless

Web protocols such as Hypertext Transfer Protocol commonly used with XML, are connectionless. This is not a problem for typical one-way data transmission, but can add complexity for transaction-type exchanges. Depending on the application context, this can require all messages to maintain any and all information required to carry out a request. Although this is a logical requirement, the technique to maintain the state of information across a number of requests can add complexity and overhead.

4.3.6 Scalability

Some protocols or web services commonly used with XML have constraints that can impact the scalability of a center-to-center network. For example, the Hypertext Transfer Protocol (HTTP) relies on a well-known single port for the start of all messages, potentially causing an overload of this port as network traffic increases. There are techniques for managing this problem (e.g., load balancing).

4.4 Interoperability Issues

4.4.1 Compatibility vs. Interoperability

Although two systems may use XML to distribute data, this does not automatically produce interoperability. Systems must be compatible in two different ways to achieve interoperability.

These two definitions come from the IEEE Std. 610.12-1990 – IEEE Standard Glossary of Software Engineering Terminology

- Compatibility: “The ability of two or more systems or components to exchange information.”
- Interoperability: “The ability of two or more systems or components to exchange information and use the information that has been exchanged.”

4.4.2 Protocol compatibility

Systems with protocol compatibility use common data exchange protocols to move data from one center to another. They can connect to each other and communicate information through some common medium.

XML may be transferred between systems using a variety of protocols. Protocols for exchanging XML that are known to the committee include:

- DATEX-ASN
- Plain HTTP
- SOAP/HTTP
- HTTPS
- Plain TCP/IP sockets
- CORBA

Section 5 of this report recommends two protocols for standardization efforts.

4.4.3 Interoperability using standard XML schemas

Systems with interoperable XML schemas use a common set of rules for encoding data into XML. These rules are the XML schemas described in section 2.1. When systems use a common XML schema, they agree on how to represent data in XML. These schemas must be defined before the systems are written and agreed upon by all users of the systems.

An XML schema roughly corresponds to one or more message sets.

Define common functional area schemas
The first necessary step to achieve compatibility is to define a common set of rules for describing data. This document does not cover the proper method for defining these rules; other standards bodies are currently doing this (IEEE 1512, TMDD MSETMCC, SAE ATIS, etc.).

Publishing schemas
After the schema is defined, a standards development organization must publish the schema for use by all interested parties.

4.4.4 System interoperability

System interoperability requires that centers use both common schemas for encoding data into XML, and common protocols to exchange the data.

Two approaches to achieve system interoperability
Using common schemas and common protocols is the least expensive approach, but requires all parties to agree on common schemas and protocols in advance.

Nonstandard XML-based systems may be integrated by writing software that translates data from nonstandard formats into standard XML schemas over standard protocols. This is more expensive than using common schemas and protocols, and new translation software must be written and maintained for each nonstandard system.

4.5 Supporting Translation between CORBA, DATEX, and XML

Two nearby groups of centers may adopt different families of standards for their center-to-center communications, and later they may need to interconnect their networks and exchange data. Rather than one of the groups having to change their center's software to add support for the other group's standard, it is desirable that a bridge or translation mechanism be provided to enable the two different networks to interoperate. For example, one or more server computers could be set up to act as gateways or translators between the two systems. Any data exchange between centers in different groups would be routed via a translator server. Such a server would have a separate interface for each family of standards. It could be integrated with a center's computer system to also provide that center with direct access to both networks, or it could be completely separate from any existing center.

Translation can be greatly simplified if considered and facilitated as the standards are developed based on a common model – this is the intent behind the development of the Generic Reference Model (GRM) for Traffic Management Centers. Ensuring all standards families use the same data dictionaries is critically important. If they all use the same naming convention, similar data structures (message content), the same dialogues, and equivalent subscription mechanisms, then translation becomes quite feasible. Converting between encoding schemes, or repackaging a message in a different packet format, are quite achievable if the fundamental information and interaction procedures are aligned.

5 NTCIP C2C WORKING GROUP RECOMENDATIONS

The NTCIP C2C WG is favoring the development of: 1) an XML-standard application protocol based on the Web Services Architecture defined by the W3C – the World Wide Web Consortium. Namely, this includes: SOAP – the Simple Object Access Protocol, and WSDL – Web Service Description Language; and 2) a File-based approach, referred to as XML Direct.

5.1 W3C Web Services Approach - WSDL and SOAP

There is a large installed base of computers that implement the W3C organizations standards, and a large number of transportation organizations that have invested and support a communications infrastructure based on W3C standards, most notably, HTTP – the Hypertext Transfer Protocol. Web Services represent the next generation implementation of the World Wide Web. If the first generation of the Web was characterized by static web pages, then the second generation brought forth dynamic page content. Dynamic web pages are created in a variety of ways – CGI/PERL (Common Gateway Interface/PERL Scripting Language), ASP (Microsoft Active Server Pages), and JSP (Java Server Pages). The next generation builds on the second generation experience, and fully integrates XML to support web-based services. Dynamic web content software providers all support an upgrade path to the newer XML web services as PERL, ASP, and JSP all of which have extension modules for SOAP.

The concept behind using W3C XML-based ITS in C2C applications is in figure 1 below.

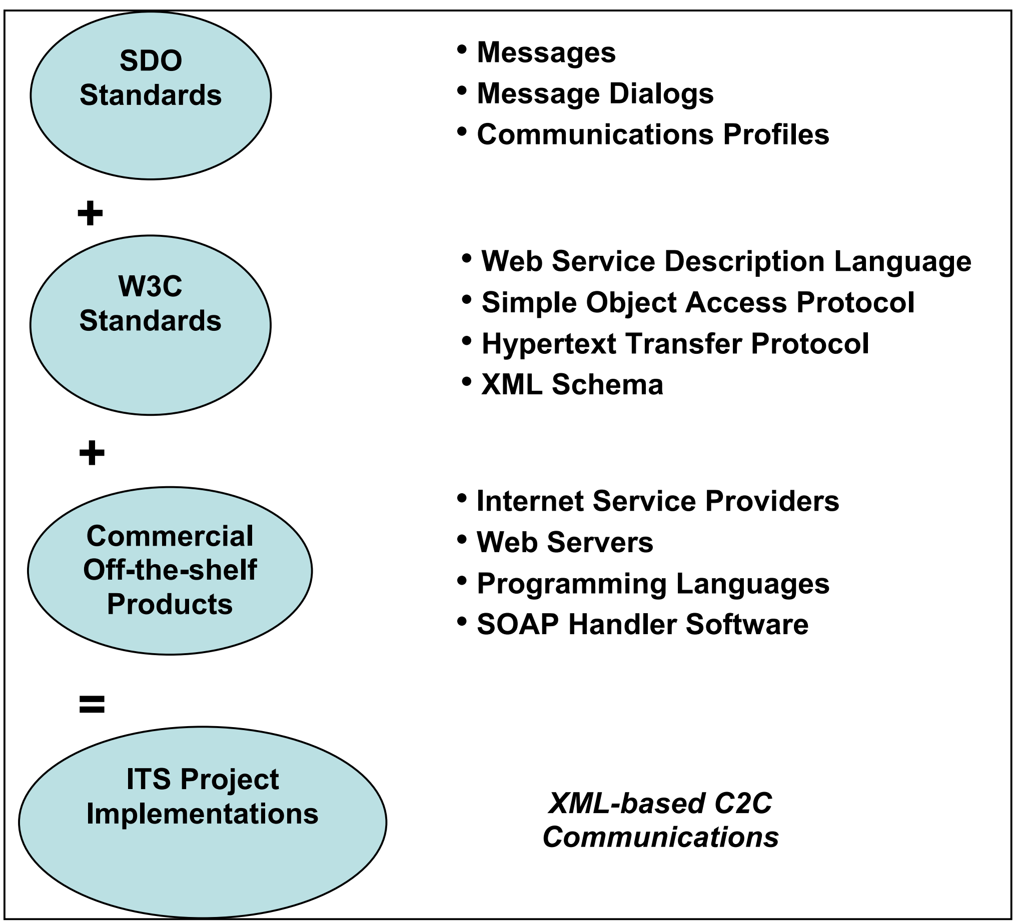


Figure 1. Building XML-based C2C ITS Applications

5.1.1 Methodology: Define Messaging Dialogs in WSDL / Implement Dialogs using SOAP

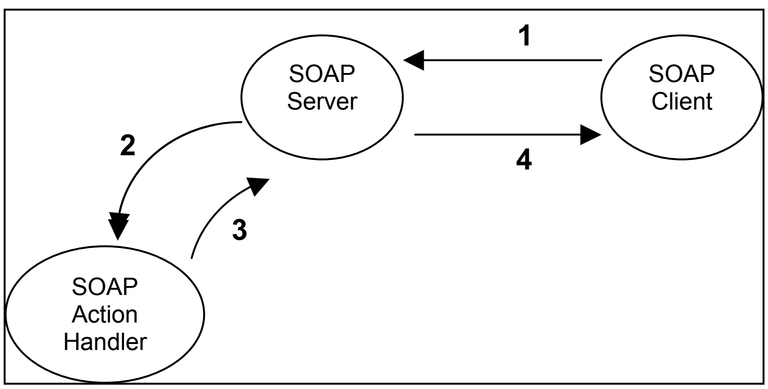
WSDL defines a web service. It includes:

- Applicable validation schemas
- Message definitions
- Dialogs or message inputs and outputs (including processing faults) for each web service "operation"
- Bindings or transport protocols over which the service communicates

SOAP defines packaging of messages and a framework for passing of the messages over a transport protocol. Currently (2003), the W3C has standardized a framework for passing SOAP over HTTP (World Wide Web), and SMTP (e-mail), for example.

Returning to the definition of interoperability as "the basic idea that two applications, regardless of operating system or programming language, can communicate … using simple encoded messages." SOAP provides a standardized way to structure XML messages, and provides agreed upon conventions for defining the type of information being exchanges, how the information is expressed in XML, and how to send the information.

The basic SOAP framework (over HTTP) is shown in figure 2 below.



Step1: SOAP Client sends a *request* to the Soap Server. Over HTTP, for example.

Step 2: The SOAP Server transfers control to a SOAP Action Handler.

Step 3: The SOAP Action Handler processes the request for the client.

Step 4: The SOAP Server sends the *response* to the SOAP Client.

Figure 2. Basic SOAP Framework over HTTP

NOTE: Over HTTP the connection between the SOAP Client and Server is open only during processing and then is closed. HTTP is said to be a connection-less protocol.

5.1.2 Methodology: Technical Approach for Developing the WSDL and SOAP

Deriving WSDL from UML Sequence Diagrams

The general approach involves converting the inputs and outputs of any boundary class object described in a UML Sequence Diagram to a WSDL node.

Therefore, given a set of UML Sequence Diagrams, one can determine the message patterns that would fulfill the message transport requirements of center-to-center operations (operations here is intended to be consistent with the WSDL meaning).

Implementing SOAP

The following lists, though preliminarily, a set of message patterns that could potentially support many of the expected messages that would be transported within the overall framework of the National ITS Architecture (from the Center-to-Center perspective), and found in the typical operation of center systems. These are:

1. Fire and Forget
2. Request / Response
3. Publish and Subscribe
    a. Periodic Publish and Subscribe
    b. Event-Driven Publish and Subscribe

Special cases of these basic message patterns would support information queries, information dissemination (either periodic update or event-driven), global updates (sync messages, heart-beat messages), short lived and long lived transaction updates (as in a database), and short lived and long lived command and control sequences. An adaptation of the basic SOAP framework over the connectionless HTTP to support Publish and Subscribe is shown below.

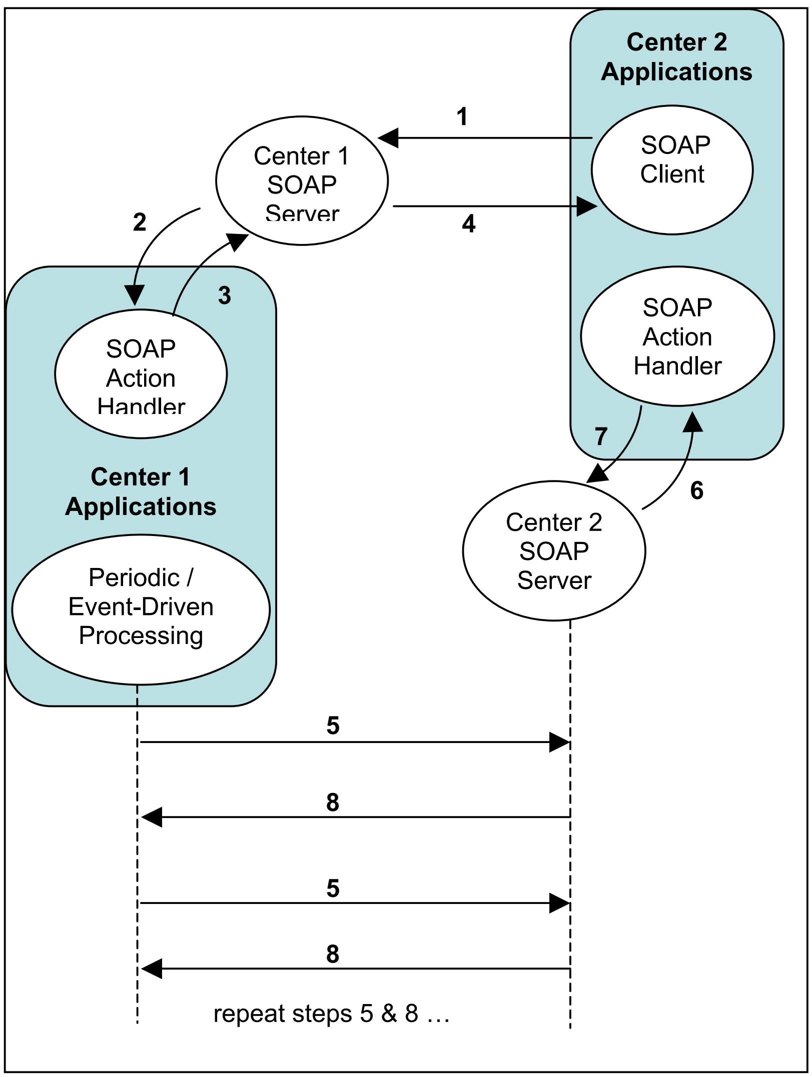


Figure 3. SOAP-Based Publish/Subscribe Example

Step 1: The SOAP Client in Center 2 initiates a subscription request for information in Center 1. One of the parameters sent in the SOAP message is information to allow the Center 1 application to contact the Center 2 application either periodically, or when updates occur (i.e., event-driven).

Step 2: The Center 1 SOAP Server receives the subscription message and transfers control to the appropriate SOAP Action Handler. The implementation details of the subscription process are left to the Center2 developers. The functional requirements, however, are specified and dictate what the system must do in response to a subscription message.

Step 3: The SOAP Action Handler in Center 1 completes processing the subscription message.

Step 4: The SOAP Client (Center 2) that initiated the subscription request is notified that the subscription has been received, processed, and that the client is now on a subscription list.

Step 5: The 'Periodic/Event-Driven Processing' in Center 1 is also a SOAP Client. Whenever it is time to notify subscribers of updates, Center 1 sends a SOAP Message to the Center 2 SOAP Server.

Step 6: The Center 2 SOAP Server transfers control to a SOAP Action Handler to handle to periodic or event-driven updates.

Step 7: Center 2 SOAP Action Handler finishes processing the message.

Step 8: The Center 1 SOAP Server is notified of completion or an error is returned.

Again, the communication between Clients and Servers over HTTP is connectionless.

5.2 XML DIRECT – File-based Approach

This section outlines some of the technical issues involved in development of an XML file-based messaging approach, called XML Direct by the C2C WG.

5.2.1 Challenges

There is no agreed upon way for sharing XML files. Some of the issues that need to be resolved include:

- Standardizing where files are located. For example, what directories do files go in?
- What goes in the file is also not currently standardized, as there are no standardized encoding schemes for putting messages into files.
- How should files be named (i.e., there are no naming conventions for XML files)?
- How often should files (or the information in the files) be updated?
- How should multiple file themes be organized, e.g. are weather, and incidents, and transit in the same file, or as separate files?
- What protocol is to be used to perform the file transfer – FTP, HTTP?
- What should be the rules of access to the file-based information?

5.2.2 Benefits

Many agencies are implementing file-based approaches for exchange of XML-formatted information. Benefits of the file-based approach include:

- It is inexpensive; that is, there is a low development cost associated with a file-based approach.
- Interconnected systems are very loosely coupled and there is a low complexity of implementation. In addition, loose coupling brings relatively few institutional hurdles.
- Security can be handled through SSL for HTTP and FTP
- A file-based approach is straight forward to implement for information dissemination. There is no cost for adding additional users. That is, there is no additional per user cost from the stand point of the information disseminator institution.
- Many organizations are already familiar with file sharing using text files. The advantage of XML encoding is providing a standardized approach to describing the information and ease of decoding of the information by the requester.

5.2.3 Considerations

While a number of benefits exist with the file-based approach, it may not be applicable for all situations, including the following:

- There is no inherent command and control capability
- The approach is a uni-directional mechanism where the responsibility for receiving the data is on the requester not the publisher of the information.

APPENDIX A

This appendix deals with issues that may be included in the standard, or that may be useful to those planning to deploy XML-based solutions.

A.1 Development Tools

Requirements Management Tools

- Rational Requisite Pro

UML Tools

- Rational Rose
- Microsoft Visio Professional Edition

XML and XML Schema Development Tools

- XML Spy
- XRay2

SOAP Servers

- SOAP Lite: PERL - programming language - based SOAP Server, runs under Windows/Unix, compatible with Apache and Microsoft Internet Information Server web server.
- Microsoft SOAP – BizTalk Server, runs only under Microsoft Internet Information Server – supports Visual Basic, C++, C#.
- Tomcat: java-based programming, JavaServer Pages, AXIS is a SOAP handler add-on, runs under Windows/Unix, compatible with Apache and Microsoft Internet Information Server web server.
- Apache SOAP – runs under Windows/Unix/Linux and Apache Web Server

A.2 Where to find the W3C Standards

XML – eXtensible Markup Language

Version 1.o of the specification is available at http://www.w3.org/tr/2000/REC-xml-20001006.html

HTTP – Hypertext Transfer Protocol

Version 1.1 of the specification is available at http://www.w3.org/Protocols/rfc2616/rfc2616.html

W3C Web Services Architecture discussion at http://www.w3.org/tr/ws-arch/ (notable is the discussion on usage scenarios at http://www.w3.org/tr/ws-arch-scenarios/)

SOAP – Simple Object Access Protocol

Version 1.1 of the specification is available at http://www.w3.org/tr/soap

Version 1.2 working draft is available at http://www.w3.org/tr/soap12

WSDL – Web Service Description Language

Version 1.1 of the WSDL specification is available at http://www.w3.org/tr/wsdl

XSLT – eXtensible Stylesheet Language Transformations

Version 1.0 of the XSLT specification is available at http://www.w3.org/tr/1999/REC-xslt-19991116.html.