A Recommended Information Report of the
Joint Committee on the NTCIP

NTCIP 9001 version v04

National Transportation Communications for ITS Protocol

The NTCIP Guide

published in July 2009

This Adobe® PDF copy of an NTCIP information report is available at no-cost for a limited time through support from the U.S. DOT / Research and Innovative Technology Administration, whose assistance is gratefully acknowledged.

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 1752
Rosslyn, Virginia 22209-3806

file version 9001v0406r rita

NOTICES

Copyright Notice

PDF File License Agreement

Content and Liability Disclaimer

The information in this publication was considered technically sound by the consensus of persons engaged in the development and approval of the document at the time it was developed. Consensus does not necessarily mean that there is unanimous agreement among every person participating in the development of this document.

AASHTO, ITE, and NEMA standards and guideline publications, of which the document contained herein is one, are developed through a voluntary consensus standards development process. This process brings together volunteers and seeks out the views of persons who have an interest in the topic covered by this publication. While AASHTO, ITE, and NEMA administer the process and establish rules to promote fairness in the development of consensus, they do not write the document and they do not

independently test, evaluate, or verify the accuracy or completeness of any information or the soundness of any judgments contained in their standards and guideline publications.

AASHTO, ITE, and NEMA disclaim liability for any personal injury, property, or other damages of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, application, or reliance on this document. AASHTO, ITE, and NEMA disclaim and make no guaranty or warranty, express or implied, as to the accuracy or completeness of any information published herein, and disclaims and makes no warranty that the information in this document will fulfill any of your particular purposes or needs. AASHTO, ITE, and NEMA do not undertake to guarantee the performance of any individual manufacturer or seller's products or services by virtue of this standard or guide.

In publishing and making this document available, AASHTO, ITE, and NEMA are not undertaking to render professional or other services for or on behalf of any person or entity, nor are AASHTO, ITE, and NEMA undertaking to perform any duty owed by any person or entity to someone else. Anyone using this document should rely on his or her own independent judgment or, as appropriate, seek the advice of a competent professional in determining the exercise of reasonable care in any given circumstances. Information and other standards on the topic covered by this publication may be available from other sources, which the user may wish to consult for additional views or information not covered by this publication.

AASHTO, ITE, and NEMA have no power, nor do they undertake to police or enforce compliance with the contents of this document. AASHTO, ITE, and NEMA do not certify, test, or inspect products, designs, or installations for safety or health purposes. Any certification or other statement of compliance with any health or safety-related information in this document shall not be attributable to AASHTO, ITE, or NEMA and is solely the responsibility of the certifier or maker of the statement.

Trademark Notice

NTCIP is a trademark of AASHTO / ITE / NEMA. All other marks mentioned in this standard are the trademarks of their respective owners.

< This page is intentionally left blank. >

ACKNOWLEDGEMENTS

The NTCIP development effort is guided by the NTCIP Joint Committee, which consists of six representatives each from the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE) and the National Electrical Manufacturers Association (NEMA). NTCIP 9001 v04, *The NTCIP Guide,* is one of the many NTCIP documents developed under a cooperative agreement among these organizations.

Members of the NTCIP Joint Committee include:

- Kleinjan Deetlefs
- Steve Dellenback
- Gary Duncan
- Michael Forbis
- Mark Hudgins
- Manny Insignares
- Jeff McRae
- Jeffrey Morgan
- Bryan Mulligan
- Raman Patel
- Peter Ragsdale
- Ed Roberts
- Edward Seymour (Chair)
- Rick Stalowski
- Ray Starr
- John Thai
- Warren Tighe

The NTCIP Guide v04 Project Team updated NTCIP 9001 v03 to the new major version v04 under the direction of the Joint Committee on the NTCIP, and with input from a peer review team selected for this project. The following individuals were members of *The NTCIP Guide v04* Project Team:

- Ralph Boaz
- Steve Dellenback
- Robert DeRoche
- Manny Insignares (Editor-in-Chief)
- Jeff McRae
- Jeffrey Morgan
- Bryan Mulligan
- Edward Seymour
- Ray Starr

Other individuals providing Peer Review and input to NTCIP 9001 v04 include:

- Patrick Chan
- Sean Coughlin
- Bruce Eisenhart
- Jean Johnson
- Raman Patel
- Robert Rausch
- Bruce Schopp
- Warren Tighe

In addition to the many volunteer efforts, recognition is also given to those organizations that supported efforts associated with NTCIP 9001 v04 by providing guidance, comments and funding for the effort:

- U.S. Department of Transportation, Research and Innovative Technology Administration, ITS Joint Program Office
- Battelle Memorial Institute
- Bi Tran Systems
- California Department of Transportation
- City of Atlanta, Georgia
- City of Mesa, Arizona
- Consensus Systems Technologies Corp.
- Econolite Control Products, Inc.
- Florida Department of Transportation
- Georgia Department of Transportation
- Image Sensing Systems, Inc.
- Intelligent Devices, Inc.
- McCain Traffic
- Minnesota Department of Transportation
- New York City Transit Authority
- New York Department of Transportation
- Iteris, Inc.
- Ohio Department of Transportation
- Ontario Ministry of Transportation
- Oregon Department of Transportation
- Peek Traffic Systems, Inc.
- Pillar Consulting, Inc.
- Robert DeRoche Consulting
- Siemens ITS, Inc.
- Southwest Research Institute
- Telvent Farradyne, Inc.

- Texas Department of Transportation
- Texas Transportation Institute
- Trevilon Corp.
- TransCore ITS, Inc.
- University of Tennessee
- Virginia Department of Transportation
- Washington State Department of Transportation
- Wisconsin Department of Transportation

IN MEMORIAM

The Joint Committee on the NTCIP dedicates NTCIP 9001 v04, *The NTCIP Guide*, to the memory of G. Curtis Herrick.

FOREWORD

NTCIP 9001 v04 is an NTCIP standards publication, and NTCIP 9001 v04 is known colloquially as "The NTCIP Guide." NTCIP 9001 v04 is an educational tool, created to assist planners, specification writers, and implementers in understanding the various NTCIP standard publications and how to use them. NTCIP 9001 v04 also explains the motivations behind the use of NTCIP. NTCIP 9001 v04 is an informative NTCIP publication, but NTCIP 9001 v04 is not an NTCIP standard and is, therefore, not to be considered binding.

The following keywords apply to NTCIP 9001 v04: NTCIP, Guide.

For more information about NTCIP standards publications, visit the NTCIP Web site at www.ntcip.org .

User Comment Instructions

The term "User Comment" includes any type of written inquiry, comment, question, or proposed revision, from an individual person or organization, about any part of this standard publication's content. A "Request for Interpretation" of this standard publication is also classified as a User Comment. User Comments are solicited at any time. In preparation of this NTCIP standards publication, input of users and other interested parties was sought and evaluated.

All User Comments are be referred to the committee responsible for developing and/or maintaining this standards publication. The committee chairperson, or their designee, may contact the submitter for clarification of the User Comment. When the committee chairperson or designee reports the committee's consensus opinion related to the User Comment, that opinion is forwarded to the submitter. The committee chairperson may report that action on the User Comment may be deferred to a future committee meeting and/or a future revision of the standards publication. Previous User Comments and their disposition may be available for reference and information at www.ntcip.org.

A User Comment should be submitted to this address:

NTCIP Coordinator
National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, Virginia 22209-3806
e-mail: ntcip@nema.org

A User Comment should be submitted in the following form:

Standard Publication number and version:
Page:
Section, Paragraph, or Clause:
Comment:
Editorial or Substantive?:
Suggested Alternative Language:

Please include your name, organization, and address in your correspondence.

History

From 1996 to 1999, predecessors of NTCIP 9001 v04 were referenced as "The NTCIP Guide." However, to provide an organized numbering scheme for the NTCIP documents, standard publication numbers were assigned, including the 9000-series of NTCIP information reports. The NTCIP Guide was assigned 9001 as the first information report.

The major version revisions and acceptance are noted in the following development history:

NTCIP Guide version v01, 1997. Written and edited by the Joint Committee on the NTCIP.

NTCIP 9001 v02.06, December 2000. New major version prepared by project team.

NTCIP 9001 v03.02, December 2002. New major version prepared by project team and accepted as a recommended information report by the Joint Committee on the NTCIP.

NTCIP 9001 v04.03, August 2008. New major version prepared by project team and issued as a User Comment Draft.

NTCIP 9001 v04.04, December 2008. Accepted as a Recommended Information Report by the Joint Committee on the NTCIP.

NTCIP 9001 v04.06, July 2009. Edited and published as an NTCIP Recommended Information Report.

Compatibility of Versions

To distinguish NTCIP 9001 v04 (as published) from previous drafts, NTCIP 9001 v04 also includes NTCIP 9001 v04.06 on each page header. All NTCIP Standards Publications have a major and minor version number for configuration management. The version number syntax is "v00.00a," with the major version number before the period, and the minor version number and edition letter (if any) after the period.

NTCIP 9001 v04 is designated, and should be cited as, NTCIP 9001 v04. Anyone using NTCIP 9004 v04 should seek information about the version number that is of interest to them in any given circumstance.

CONTENTS

Page

FIGURES

Page

TABLES

Page

Section 1
INTRODUCTION

The National Transportation Communications for ITS Protocol (NTCIP) family of standards defines protocols and profiles that are open, consensus-based data communications standards. When used for remote control of roadside and other transportation management devices, NTCIP-based devices and software can help achieve interoperability and interchangeability. When used between transportation and emergency management centers, NTCIP standards facilitate agency coordination and information sharing.

Why are NTCIP standards needed? How are NTCIP standards used? What is interoperability and interchangeability? NTCIP 9001 v04, referenced as "The NTCIP Guide" here, explains these terms, and gives you the "why" and the "how" to specify and implement NTCIP standards in your ITS devices and systems.

1.1 PURPOSE OF *THE NTCIP GUIDE*

The NTCIP Guide is an educational tool, created to assist planners, specification writers, and implementers in understanding the various NTCIP standard publications and how to use them. The NTCIP Guide also explains the motivations behind the use of NTCIP. The NTCIP Guide is an informative NTCIP publication, but *The NTCIP Guide* is not an NTCIP standard and is, therefore, not to be considered binding.

The reader should understand that, in writing specifications or implementing systems, only the actual NTCIP standards govern and take precedence, not The NTCIP Guide. For updated information on NTCIP standards publications, please see the NTCIP website at www.ntcip.org .

1.2 PURPOSE OF NTCIP

The transportation industry has had a history of deploying systems with unique data definitions and proprietary communications protocols. Field devices and systems from one manufacturer or developer were not interoperable with those of other manufacturers or developers. As a result, expansion of the system after initial deployment can generally only be done using equipment of the same type and usually the same brand as in the initial deployment, unless there are investments in major systems integration efforts.

With proprietary protocols, there is little to no opportunity for realistic competitive bidding as additional field devices are added to the system, due to the lack of interchangeability. Nor, is there any opportunity for realistic competitive bidding to add additional types of field devices to the system, due to the lack of interoperability.

NTCIP is a family of **open** standards, defining common communications protocols and data definitions.

The NTCIP standards define common data definitions and open protocols. The proper use of NTCIP open-standards in an ITS deployment allows future expansion of the system to benefit from true competitive bidding, as well as allowing other types of field devices to be added. NTCIP is an entire family of standards designed to meet the communications needs of various fixed-asset roadside devices and traffic management centers. The open-standards approach is illustrated in Figure 1.

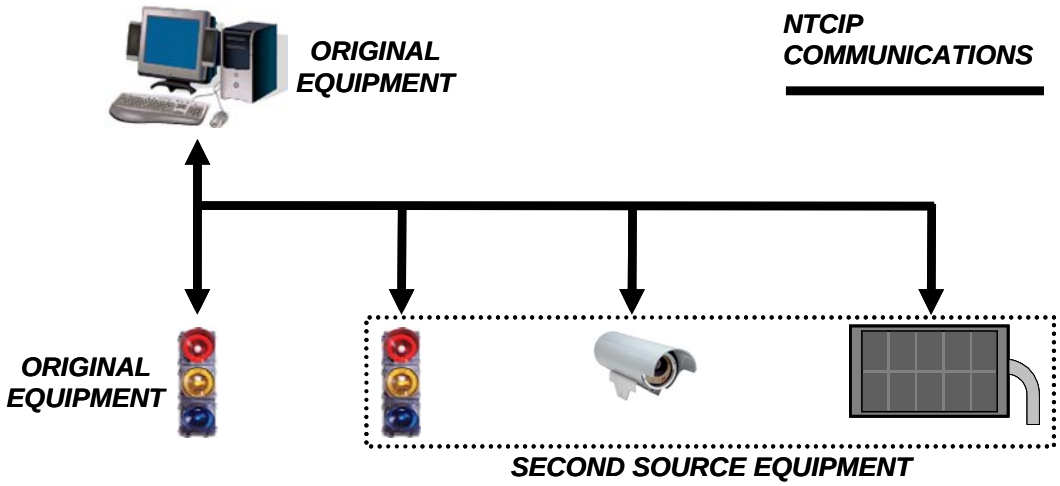

Figure 1 NTCIP Facilitates Interoperability and Interchangeability of Field Equipment

Interoperability and interchangeability are two key goals of the NTCIP open-standards effort. The terms interoperability and interchangeability generally reflect the ability to use multiple brands of a device on the same communications channel, along with the ability to swap them out. For example, the ability to put any brand of NTCIP-conformant traffic signal controller in the same system at the same time reflects interchangeability for that device type. It is for this reason that the NTCIP family of protocols is being widely embraced and specified in many new system deployments.

Interoperability and ***Interchangeability*** help to reduce the total life system expenditures (procurement, operations, and maintenance).

1.2.1 Interoperability

Interoperability reflects the ability of multiple center systems and devices of different types to exchange information for some common purpose. Interoperability allows system components from different vendors to communicate with each other to provide system functions and to work together as a whole system. For example, using the same communications infrastructure to interconnect a management system with traffic signal controllers, dynamic message signs, video surveillance controls and other devices to manage traffic reflects a real-world example of interoperability.

1.2.2 Interchangeability

Interchangeability reflects the capability to exchange devices of the same type on the same communications channel and have those devices interact with others devices of the same type using standards-based functions. With interchangeability, system components can be changed out (switched) with similar components from different vendors because they possess common functional and physical characteristics. An example of interchangeability is a signal controller from different manufacturers interacting with each other to provide traffic signal coordination along an arterial throughway.

1.3 NTCIP OVERVIEW

NTCIP is a family of communications standards for transmitting data and messages between computer systems used in Intelligent Transportation Systems (ITS). A communications standard specifies a set of rules for how messages are coded and transmitted between electronic devices. The equipment at each end of a data transmission uses the same specification to successfully communicate. It is a bit like human languages that have an alphabet, vocabulary and grammar rules used by everyone speaking that language.

1.3.1 Center to Field (C2F) Communications

NTCIP provides communications standards for two fundamentally different types of ITS communications. The first type is between a center system and multiple control or monitoring devices managed by that center. An example of a center system is a computer at city hall monitoring and controlling the operation of microprocessor-based roadside controllers at traffic signals within a city. The center system may send instructions to the traffic signal controllers to change signal timings as traffic conditions change and the controllers send status and traffic flow information to the computer.

Other examples of this type of communications include:

a) An on-board vehicle transit system communicating with a traffic signal device to facilitate transit priority;
b) A freeway management system communicating with detectors and ramp meters on freeways; and
c) A traffic management system controlling roadway lighting, closed-circuit television (CCTV) cameras, dynamic message signs, advisory radio transmitters, environmental sensors and traffic count stations on roadways.

Since most applications of this type involve a center system communicating with various devices at the roadside or on agency vehicles, this type of communication is referred to as “center to field” (C2F). The NTCIP protocols intended for this communications application is often used in an environment where a center system routinely polls each field device, as in the most common case of multiple field devices sharing a communications channel.

1.3.2 Center to Center (C2C) Communications

The second type of communication involves messages sent between two or more center systems. This type of communication is referred to as center to center (C2C) communications, although two or more of the various systems may in fact be located within the same “center” or building, they are logically separate. C2C involves peer-to-peer communications between any number of center systems in a many-to-many network. This type of communication is similar to the Internet, in that any center can request information from, or provide information to, any number of other centers.

An example of C2C communications is two traffic management centers that exchange real-time information about the inventory and status of traffic control devices. This allows each center system to know what timing plan, for example, the other center system is running to allow traffic signal coordination across center geographic boundaries. Other examples of this type of communication include:

a) Two or more traffic signal systems exchanging information (including second-by-second status changes) to achieve coordinated operation of traffic signals managed by the different systems and to enable personnel at one center to monitor the status of signals operated from another center;
b) A transit system reporting schedule adherence exceptions to a transit customer information system and to a regional traveler information system, while also asking a traffic signal management system to instruct its signals to give priority to a behind-schedule transit vehicle;
c) An emergency management system reporting an incident to a freeway management system, to a traffic signal management system, to two transit management systems and to a traveler information system;
d) A freeway management system informing an emergency management system of a warning message just posted on a dynamic message sign on the freeway in response to its notification of an incident; and
e) A weather monitoring system informing a freeway management system of ice forming on the roadway so that the freeway management system is able to post warning messages on dynamic message signs as appropriate.

1.4 NTCIP BENEFITS

NTCIP standards offer increased flexibility and choices for agencies operating transportation management systems. NTCIP standards usage removes barriers to interagency coordination and allows

equipment of different types and different manufacturers to be mixed on the same communications line. For these reasons, operating agencies benefit from specifying that NTCIP be included in all future acquisitions and upgrades, even if NTCIP is not initially used.

1.4.1 Avoiding Early Obsolescence

While retrofitting legacy equipment and systems with NTCIP support is not practical in most situations, most manufacturers offer NTCIP support in their ITS devices. It is possible to migrate a system gradually, since it is possible to operate a mixture of NTCIP and non-NTCIP devices in the same system, though not on the same communications line. Equipment may also continue to use a current protocol even though the device may also support NTCIP as a second protocol. Integrating legacy equipment and systems with NTCIP-conformant upgrades in this manner ensures that an operating agency's systems and equipment remain useful and compatible long into the future.

Buying a field device or central control system that has no software available to support NTCIP is like buying a computer that has no software available to access the Internet. Even if agencies do not use the Internet now, Internet use is surely likely during the lifetime of the computer.

1.4.2 Providing a Choice of Vendor

Since a computer system that supports NTCIP can communicate with any device from other vendors that are NTCIP-conformant, the number of vendors and systems, field devices, or software that can be considered for procurement increases greatly.

While vendor-specific features may only be available to other software and ITS devices from the same vendor, the basic functionality described in an NTCIP standard is available regardless of vendor. This requires that agency specifications (procurement documents) adequately specify the mandatory and optional conformance requirements that support the agency's functional requirements. However, NTCIP makes it easier for an agency to gradually change its software, controllers and other field devices from one vendor to supporting multiple vendors for the entire system.

Agencies should also consider issues beyond the interoperability/interchangeability aspects inherent to the NTCIP family of standards. These include issues such as stocking replacement parts that are different from provider to provider, and the knowledge-base of ITS technicians who would have to become familiar with multiple vendors' devices.

1.4.3 Phased Procurement and Deployment

Specifying NTCIP allows agencies to procure devices and center systems in phases, over several financial cycles. For example, many agencies procure a few signs one year, then a few more the next year, and so on. Sometimes devices are procured from one vendor, and sometimes from multiple vendors. Specifying NTCIP standards means that multiple deployment phases, over multiple years, can be integrated, with little difficulty. The initial deployment establishes an ITS communications infrastructure that can be leveraged by future deployment phases resulting in improved cost/benefit for ITS projects.

1.4.4 Enabling Interagency Coordination

NTCIP allows agencies to exchange information and (with authorization) basic commands that enable any agency to monitor conditions in other agencies' systems, and to implement coordinated responses to incidents and other changes in field conditions when needed. Such data exchange and coordinated response can be implemented either manually or automatically. One agency can monitor, and issue basic commands, if authorized, to field devices operated by another agency, even though those devices may be from a different vendor than those used by the monitoring agency. Potential applications of interagency coordination include:

a) Coordinating timed transfers at a shared transit center,
b) Coordinating traffic signals across jurisdictional boundaries,
c) Providing traffic signal priority for selected, e.g., behind schedule, transit vehicles,
d) Providing real-time information to a shared traveler information center,

e) Monitoring traffic volumes on another agency's roadway,
f) Coordinating the operation of a freeway ramp meter with an adjacent traffic signal, or
g) Posting a warning message on another agency's dynamic message sign.

1.4.5 Use One Communications Network for All Purposes

NTCIP allows a management system to communicate with a mixture of device types on the same communications channel. For example, with the addition of appropriate application software in the system computer, a dynamic message sign could be installed near a signalized intersection, and the computer could communicate with the sign controller using the communications line or channel already in place for the traffic signal controller, if certain aspects of the communications protocols, that is, the Data Link and Physical layer protocols are the same. Similarly, a wide area network interface installed for communications with a system operated by another agency can be used for communications with any number of other systems, of any type, if NTCIP and the C2C Data Dictionaries and Message Sets of other efforts, such as the Traffic Management Data Dictionary (TMDD), are used. The communications network is usually one of the components of a transportation management system that requires the most resource investment. NTCIP ensures flexibility in the future use of that component.

1.5 U.S. DOT RITA ITS STANDARDS PROGRAM OVERVIEW

The U.S. DOT Research and Innovative Technology Administration (RITA) ITS Standards Program has supported the development and widespread use of standards to encourage the interoperability of ITS systems. Through cooperative agreements with five standards development organizations (SDOs), the U.S. DOT RITA ITS Standards Program accelerated the development of non-proprietary, industry- and consensus-based open ITS standards, and has encouraged public-sector participation in the development process. The five original cooperative agreement SDOs are: AASHTO, ASTM, ITE, IEEE, and SAE. NEMA and APTA joined the program through other agreements.

ITS practitioners are encouraged to use SDO-approved ITS standards when deploying ITS projects in their region. The use of ITS standards is necessary to provide integrated, open systems.

U.S. DOT continues to encourage stakeholders to test and evaluate developing standards, and where available, to use ITS standards in their deployment. In support of early deployment, the U.S. DOT RITA ITS Standards Program offers information and resources to those ITS project managers who decide to use ITS standards. A website, located at *www.standards.its.dot.gov/* , provides background information, testing results, and guides to deploying specific standards. In addition, links to contacts, training, and technical assistance resources are also located on this site.

1.6 ORGANIZATION OF THE NTCIP GUIDE

The NTCIP Guide is divided into seven sections as follows:

a) **Section 1—Introduction** provides a brief overview of NTCIP as well as a discussion of the motivations that led to the development of NTCIP standards. It also discusses the issues associated with NTCIP use.
b) **Section 2—Understanding NTCIP** provides a general purpose technical overview of NTCIP and the NTCIP Framework. It's a good starting point for anyone wishing to become better informed on the various technical aspects of the NTCIP approach.
c) **Section 3—Procuring NTCIP** is intended principally for agency specification writers. This section presents an overview of the procurement process and issues related to procurement.
d) **Section 4—Agency Requirements and Specifications** is intended principally for agency specification writers. This section outlines elements of NTCIP standards and how to use them in developing agency requirements and specifications.
e) **Section 5—Designing NTCIP** is intended principally for those faced with the task of designing the communications element of transportation systems that use NTCIP protocols. This section, together with the material in Annex E, provides a detailed discussion on bandwidth analysis and system timing.

f) **Section 6—Implementing NTCIP** is intended for systems implementers, including software and hardware developers for field equipment, traffic management center software and hardware developers and systems integrators. In particular, some of the lessons learned and common pitfalls encountered during actual deployments are discussed, with suggested solutions.
g) **Section 7—NTCIP Testing** is intended principally for test documentation developers. The information presented is a summary of NTCIP 9012 v01, Testing Guide for NTCIP Center-to-Field Communications.

1.7 ADDITIONAL NTCIP INFORMATION

More information about NTCIP standards can be found on the NTCIP Website at www.ntcip.org . Those without access to the World Wide Web may contact:

NTCIP Coordinator
National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, Virginia 22209-3801
e-mail: ntcip@nema.org

NTCIP standards are developed with input from users and other interested parties. Such input was also sought and evaluated for the development of *The NTCIP Guide*. Anyone interested in making written inquiries, comments, and proposed or recommended revisions should submit them to the NTCIP Coordinator at the above address. Please include the following information in your correspondence:

a) NTCIP Document Number:
b) Version Number:
c) Section Number:
d) Paragraph:
e) Comment (including Suggested Alternative Language):
f) Your Name:
g) Your Address:
h) Your Organization

1.8 NTCIP STYLE CONVENTIONS

Within *The NTCIP Guide*, as well as within other NTCIP standards publications, certain conventions are used with respect to document self-reference, acronym and abbreviation usage, and normative and informative references.

1.8.1 Self-Reference Within NTCIP Standards Publications

When an NTCIP standards publication references itself or is referenced in another standard, the style of that reference is: NTCIP XXXX vZZ. Specifically, “XXXX” is the NTCIP publication designation number, and “ZZ” refers to the major version number.

NOTE—Prior to 2008, NTCIP standards used the following reference style: NTCIP XXXX:YYYY, where “XXXX” was the NTCIP publication number, and “YYYY” refered to the year. For those NTCIP standards published prior to 2008, the cover of the published NTCIP standard or document governs the style of the self-reference, and these publications should be cited as: NTCIP XXXX:YYYY.

NOTE—Following this convention, NTCIP 9001 v04 should refer to itself as NTCIP 9001 v04; however, to make NTCIP 9001 v04 more “user-friendly,” NTICP 9001 v04 refers to itself as *“The NTCIP Guide.”*

1.8.2 Abbreviation and Acronym Usage

Various abbreviations and acronyms are used throughout the family of NTCIP standards and publications. For a list of abbreviations and acronyms commonly used in NTCIP standards and publications, as well as a glossary of common terms, see Annex A.

Within NTCIP standards and publications, abbreviations or acronyms are usually spelled out on first use, with the abbreviation or acronym following in parentheses. On subsequent use, the abbreviation or acronym is used alone, without parentheses.

Section 2
UNDERSTANDING NTCIP

2.1 TYPES OF SYSTEMS AND DEVICES SUPPORTED BY NTCIP

NTCIP defines a family of general-purpose communications protocols and transportation-specific data dictionaries/message sets that support most types of computer systems and field devices used in transportation management. Applications for NTCIP are generally divided into two categories: C2F and C2C. The former, C2F, normally involves devices at the roadside, communicating with management software on a central computer. The latter, C2C, usually involves computer-to-computer communications where the computers can be in the same room, in management centers operated by adjacent agencies, or across the country.

For both C2F and C2C applications, NTCIP supports systems and devices used in traffic, transit, emergency management, traveler information, and planning (data archiving) systems. Figure 2 illustrates how various transportation management systems and devices can be integrated using NTCIP.

The following are examples of systems and devices that can take advantage of NTCIP:

a) C2F

- Dynamic message signs
- Traffic signals
- Field masters (closed loop systems)
- Data collection and monitoring devices such as traffic counter, traffic classifiers and weigh-in-motion stations
- On-board sensors and controllers
- Environmental sensors
- Ramp meters
- Vehicle detectors
- Closed circuit television cameras (camera control only)
- Video switches
- Highway lighting control

b) C2C

- Traffic management (freeway/surface street, urban/rural)
- Transit management (bus/rail/other)
- Incident management
- Emergency management
- Parking management
- Traveler information (all modes)
- Commercial vehicle operations regulation
- Any mix of these

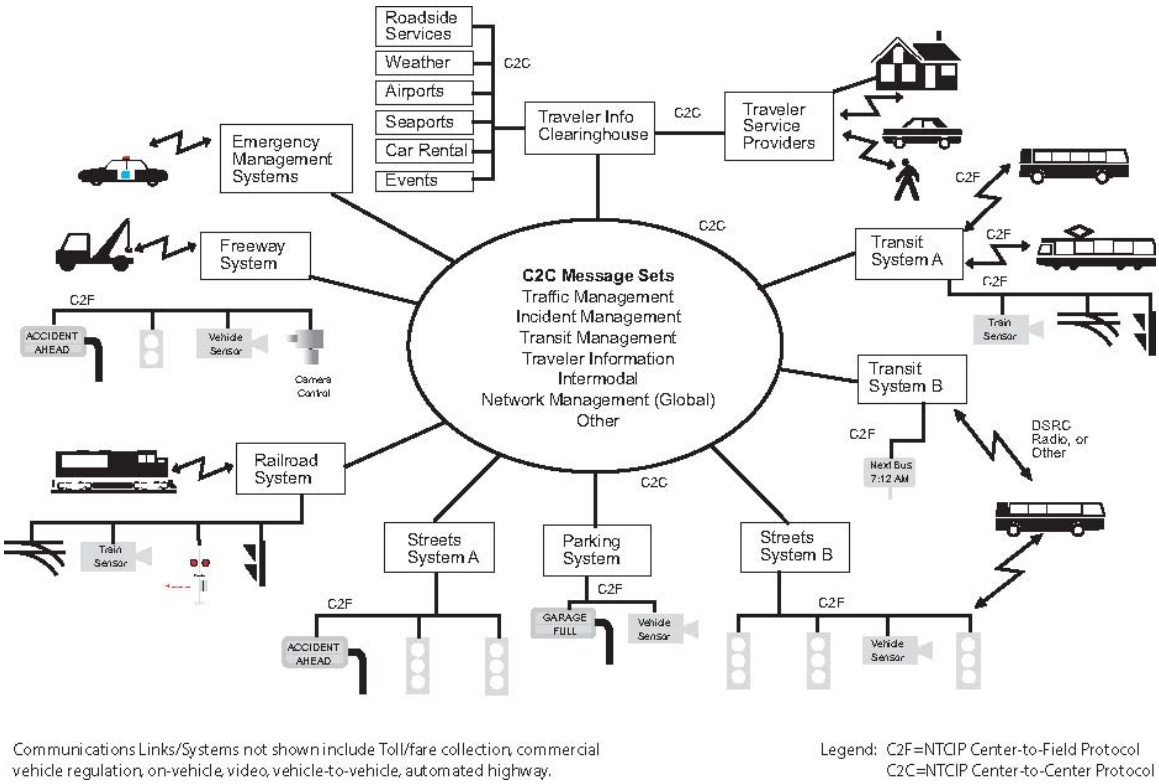

Figure 2 Example—ITS Integration Using NTCIP

Many applications of NTCIP are related to near real-time communications and involve continuous, automated transmissions of data or commands. NTCIP also supports human to-remote-machine/system transmissions. Historical data can also be sent using NTCIP, but other communication standards, especially electronic mail and file transfer protocols developed for the Internet, may also be suitable for this purpose. Human-to-human communications are generally better served by fax/telephone and Internet protocols, for example, e-mail, but basic support is also provided in the NTCIP C2C protocols.

2.2 APPLICATIONS NOT ADDRESSED BY NTCIP

The NTCIP family of standards is intended for use in all types of management systems dealing with the transportation environment, including those for freeways, traffic signals, transit, emergency management, traveler information, and data archiving. NTCIP is intended for fixed-point to fixed-point communications between computers in different systems or different management centers, and between a computer and devices at the roadside. Current NTCIP standards are not intended for use in devices owned by individual travelers; other standards either currently exist or are in development for those purposes.

Some of the data transfers involved in ITS have special needs that are the subject of other standards development efforts. The NTCIP effort is coordinating with the activities of these other groups to the extent practical. These other standards efforts include:

a) A roadside device reading and/or writing to an electronic tag on a vehicle. This involves very fast and compact wireless data transfers over short distances of a few meters during the few milliseconds that a passing vehicle's tag is within that reception range. However, NTCIP is suited to C2F communications between the roadside tag reader and a central computer;
b) Full motion video images transmitted from a camera or recorded media. This involves specialized protocols able to accommodate the large volume of continuous streaming information making up a video signal, and several such industry standards already exist, for example, NTSC. However, NTCIP

is suited to C2F transmission of video camera control commands and switch control data using a separate communications channel;

c) Transmission of traveler information data to privately owned vehicles. This involves special broadcast and limited bandwidth protocols such as those that work in conjunction with the FM radio standards or cellular radio. However, NTCIP is suited to sending the information from various data sources to the traveler information service provider, using C2C communications;
d) Communications for financial transactions. This involves special security measures not currently supported in NTCIP;
e) In-vehicle communications for operations monitoring, advanced vehicle control, and safety. This involves specialized protocols for very high speed and fail-safe transmissions between devices housed on the same vehicle; and
f) In-cabinet communications between a controller and other electronic devices in a roadside cabinet. This involves specialized protocols for very fast high-volume data transmissions over short distances. The ITS industry is currently addressing these requirements in the Advanced Transportation Controller (ATC) efforts, via three standards: the ATC Cabinet standard, the ATC Controller standard, and the ATC Application Programming Interface (API) standard.

Other communications standards are available, or under development, to serve each of these specialized needs.

2.3 RELATIONSHIP TO THE NATIONAL ITS ARCHITECTURE

The U.S. National ITS Architecture defines a common framework for ITS integration, and the ITS standards define how the system components operate within this framework. By specifying how systems and components interconnect, standards allow for interoperability. To expedite deployment of nationally interoperable ITS systems and services, the U.S. DOT supports specific ITS standards initiatives, especially in those areas that have significant public benefit.

The role of NTCIP in the National ITS Architecture is illustrated in Figure 3.

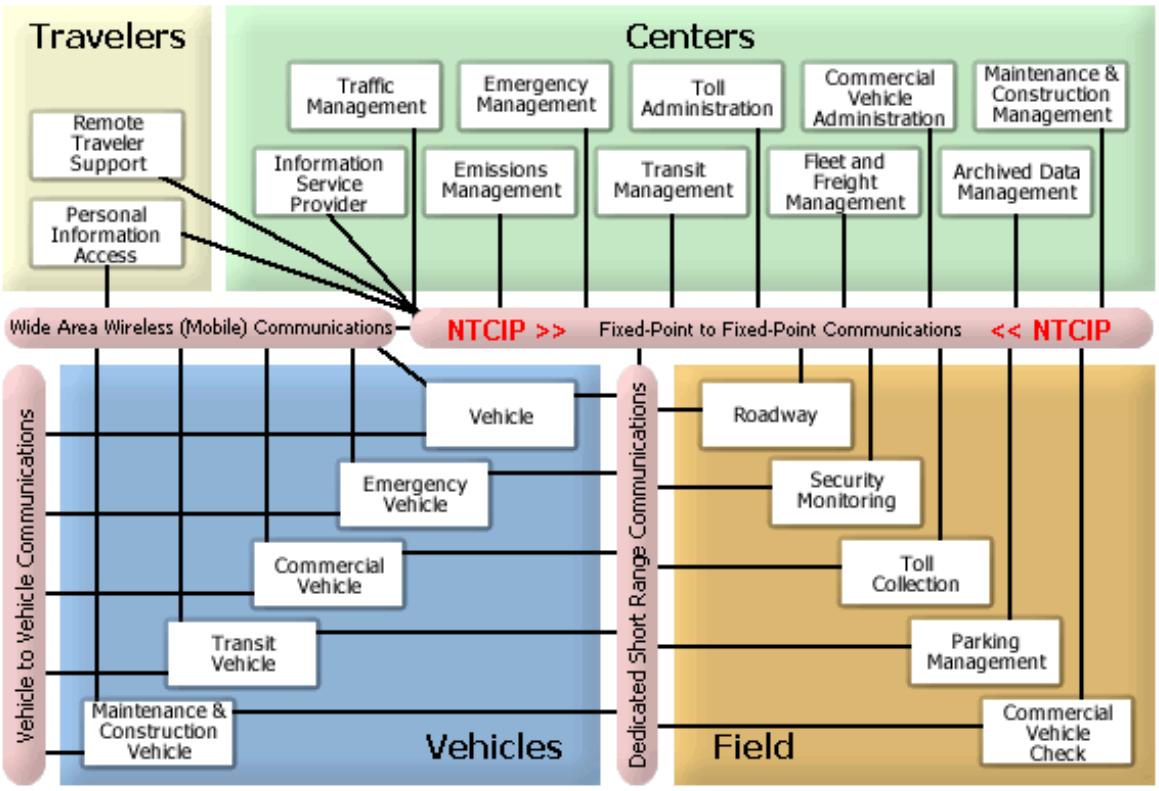

Figure 3 NTCIP and the U.S. National ITS Architecture

2.4 NTCIP FRAMEWORK

The NTCIP Framework, shown in Figure 4, uses a layered or modular approach to communications standards, similar to the layering approach adopted by the Internet and the International Organization of Standards (ISO). In general, data communications between two computers or other electronic devices can be considered to involve the following primary layers, called “levels” in NTCIP, to distinguish them from those defined by the International Organization for Standardization (ISO) and the Internet. The five levels are: information level, application level, transport level, subnetwork level, and plant level.

The levels in the NTCIP framework are somewhat different from communication stack layers defined by ISO’s Open Systems Interconnect (OSI) seven-layer reference model and other standards developing organizations. The OSI model breaks the communications process into seven well-defined layers. Each layer has a defined purpose, generally independent of adjacent layers. Although OSI communications protocols are not widely used, the layered model remains.

With the many diverse requirements of NTCIP, it is not surprising that the NTCIP family of standards looked at the ISO OSI Basic Reference model to help define its framework. However, the NTCIP stack extends beyond the communications stack to include informational data and interfaces to the physical communications infrastructure. The levels and terminology used in NTCIP were chosen for simplicity and ease of understanding by lay readers, and relevance to typical applications in the transportation industry. The OSI layers and terminology are often referenced in later technical sections of this publication and in many of the standards defined by NTCIP. Figure 5 shows how the NTCIP Information, Application, Transport, Subnetwork, and Plant Levels loosely relate to the OSI model.

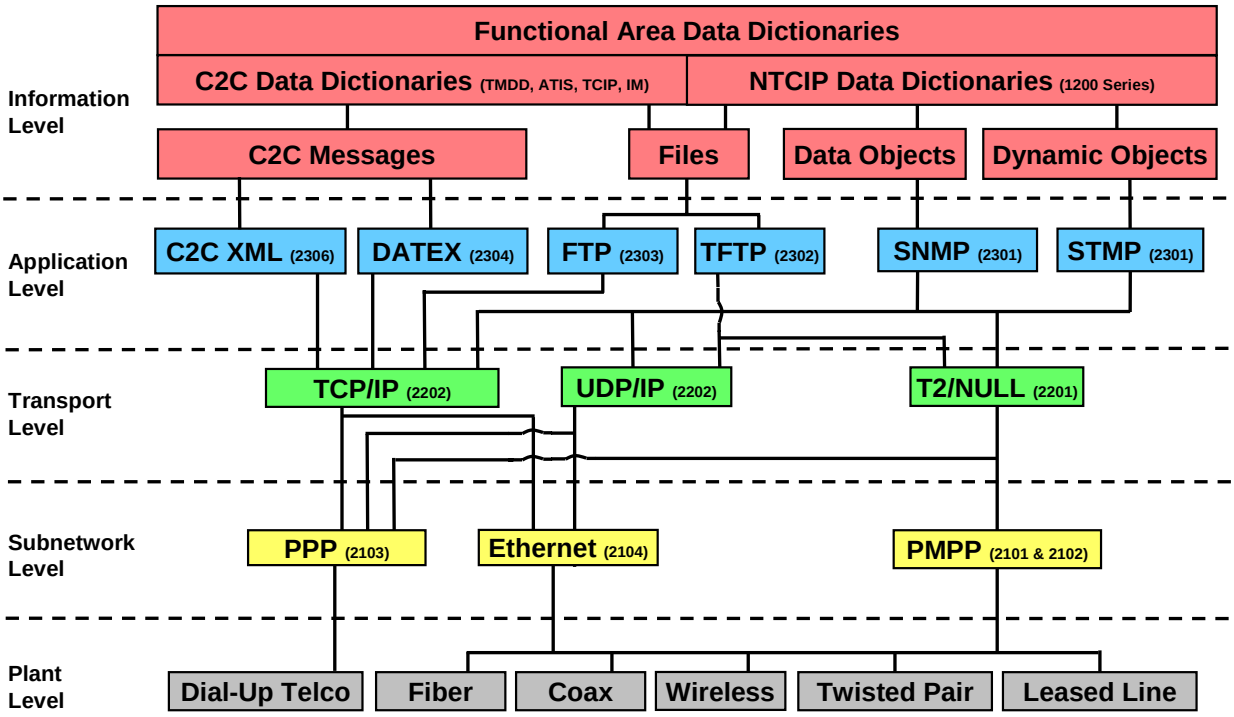

Figure 4 NTCIP Framework

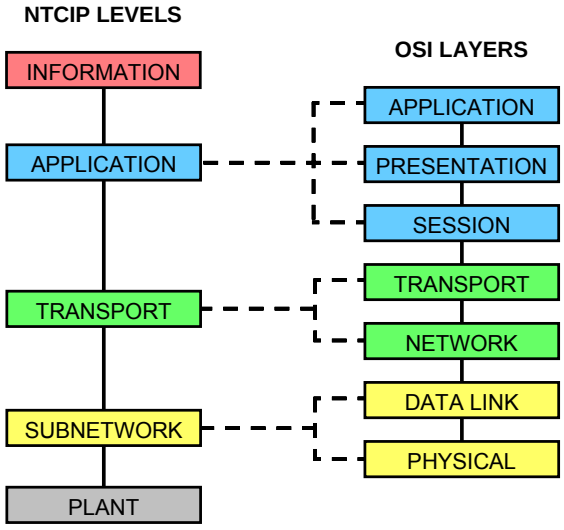

Figure 5 OSI Layer to NTCIP Level Mapping

a) **NTCIP Information Level—**Information standards define the meaning of data and messages and generally deal with ITS information (rather than information about the communications network). This is similar to defining a dictionary and phrase list within a language. These standards are above the traditional ISO seven-layer model. Information level standards represent the functionality of the system to be implemented.
b) **NTCIP Application Level—**Application standards define the rules and procedures for exchanging information data. The rules may include definitions of proper grammar and syntax of a single statement, as well as the sequence of allowed statements. This is similar to combining words and phrases to form a sentence, or a complete thought, and defining the rules for greeting each other and exchanging information. These standards are roughly equivalent to the Session, Presentation and Application Layers of the OSI model.
c) **NTCIP Transport Level—**Transport standards define the rules and procedures for exchanging the Application data between point 'A' and point 'X' on a network, including any necessary routing, message disassembly/re-assembly and network management functions. This is similar to the rules and procedures used by the telephone company to connect two remotely located telephones. Transportation level standards are roughly equivalent to the Transport and Network Layers of the OSI model.
d) **NTCIP Subnetwork Level—**Subnetwork standards define the rules and procedures for exchanging data between two 'adjacent' devices over some communications media. This is equivalent to the rules used by the telephone company to exchange data over a cellular link versus the rules used to exchange data over a twisted pair copper wire. These standards are roughly equivalent to the Data Link and Physical Layers of the OSI model.
e) **NTCIP Plant Level—**The Plant Level is shown in the NTCIP Framework only as a means of providing a point of reference to those learning about NTCIP. The Plant Level includes the communications infrastructure over which NTCIP communications standards are to be used and has a direct impact on the selection of an appropriate Subnetwork Level for use over the selected communications infrastructure. The NTCIP standards do not prescribe any one media type over another. In most cases, communications media selections are made early in the design phase.

Deployers should select and specify at least one NTCIP protocol or profile at the information, application, transport and subnetwork level for a system.

The NTCIP Framework does not preclude combinations beyond those expressly indicated on the diagram. For example, some C2C deployments are exchanging eXtensible Markup Language (XML) messages across Data Exchange Protocol (DATEX). Also, device deployments have implemented a protocol stack of Data Objects → SNMP→ T2/Null → PMPP → Dial-Up Telco.

To ensure a working system, deployers should select and specify at least one NTCIP protocol or profile at each level. A discussion of each level, and NTCIP standards that apply at that level, follows.

A complete and current list of the protocols and profiles defined by the NTCIP family of standards can be found at www.ntcip.org .

2.5 PROTOCOL STANDARDS—NTCIP 1100 SERIES

The NTCIP 1100 series of standards are the protocol, cross-cutting foundation standards that may be referenced by any other NTCIP standard. These standards are not shown on the NTCIP Framework.

2.6 INFORMATION PROFILES—NTCIP 1200 SERIES

The NTCIP 1200 series of standards defines the information content between a management center and a field device, or other center. The data element definitions may include syntax, allowable ranges, and may also include valid sequences for transmitting data elements.

NOTE—NTCIP 1201 v03, Global Object (GO) Definitions, is considered a supporting standard, not a primary standard, because NTCIP 1201 v03 can be used for multiple types of field devices.

2.7 APPLICATION PROFILES—NTCIP 2300 SERIES

NTCIP application profiles can be categorized into three areas: application profiles specific to C2F communications, application profiles specific to C2C communications, and application profiles adopted directly from the Internet (per the Internet Engineering Task Force (IETF)).

2.7.1 Application Profiles for Center to Field (C2F) Communications

As of the publication of NTCIP 9001 v04, two application profiles are supported by NTCIP for C2F communications, Simple Network Management Protocol (SNMP) and Simple Transportation Management Protocol (STMP).

2.7.1.1 SNMP Overview

SNMP is a communication protocol widely used in computer networks for managing network devices. For transportation communications, SNMP is the essential protocol used by all NTCIP compliant field devices. SNMP uses a client-server communications model where the central computer acts as the client and the field devices act as servers. The client uses four types of messages: "Set," "Get," "Get Next," and an NTCIP-defined Trap to configure, control, and monitor data elements in field devices. SNMP uses Abstract Syntax Notation—1 (ASN.1) notation to specify and encode data elements in a Management Information Base (MIB) for each compliant field device. A copy of the field device's MIB is on the central system so that the central system may access the field device appropriately.

Stacks based on SNMP provide a simple, but bandwidth inefficient, protocol for C2F applications, based on the Internet protocol of the same name (SNMP). It is suitable only for networks with high bandwidth, or low volumes of messages. SNMP has been designed by the Internet community to run over User Datagram Protocol/Internet Protocol (UDP/IP), but SNMP can be forced to run over Transmission Control Protocol/Internet Protocol (TCP/IP) or T2/NULL.

2.7.1.2 STMP Overview

STMP is an extension of SNMP in which the central system and the field device dynamically establish composite messages (typically when they are first connected) made up of a set of the field device data elements. The composite messages are still sent using SNMP but with greatly reduced overhead over sending the data elements individually. STMP is ideal for low-bandwidth communication media such as 9600 serial communication.

STMP was developed specifically for use in the transportation industry. It is an extension of SNMP that allows C2F messages to be sent more efficiently using dynamic composite objects. Stacks based on this protocol are suitable for networks with low bandwidth and high volumes of messages, including such traffic signal systems where a central computer is directly connected to field devices, without the need to route the information through some other device such as an on-street master in a closed loop system. STMP has been designed to run over T2/Null since STMP supports low bandwidth links, but STMP could also be used over UDP/IP or TCP/IP if sufficient bandwidth is available.

2.7.1.3 Comparison of SNMP and STMP

NTCIP provides two application level protocol choices for C2F communications: the Internet's SNMP and STMP. These base protocols use the get/set messaging paradigm used in SNMP. These choices use the same base data elements, as defined in the NTCIP 1200 series of standards. They differ in the level of complexity to implement and the types of services offered. Table 1 summarizes the services offered and implementation requirements.

Table 1 Center to Field (C2F) Protocol Comparison

	SNMP	STMP
Can send any base data element?	Yes	Yes
Bandwidth Efficiency – inverse of packet overhead	Worse	Better (uses dynamic objects)
Supports routing & dial-up	Options	Options
Message Set	Supported	Limited to 13
Ease of Implementation	Easier	Harder

STMP is the most bandwidth efficient option currently available and includes full support of SNMP for infrequent messaging demands. It includes SNMP as a subset, so that any management system that implements STMP can also communicate with a device that supports only SNMP. It also requires the use of SNMP to define dynamic objects. Occasional messages requiring additional security can be sent using SNMP. The greatest advantage of STMP is its support for dynamic objects which, when combined with a more efficient encoding scheme, dramatically reduce the packet overhead relative to SNMP. Dynamic objects also enable users to define custom messages that are composed of any number of individual data elements. However, these data elements have to be defined in both the central computer and the field devices to work properly. STMP is the most flexible and bandwidth efficient option.

2.7.1.4 Communications Patterns

Communications patterns are used to describe the general sequence of communications between two entities on a network. Three basic communications patterns, or simple dialogs, can handle a wide variety of situations. The three basic communications patterns are:

a) **Request-Response—**This communications pattern supports sending of data followed by a response. This pattern implements a synchronous pattern of message communications.
b) **Dynamic Objects—**This pattern supports a subscriber application (center) performing an initial request-response to set up future asynchronous responses from a device.
c) Traps—The main feature of traps is that communication is initiated by the field device to the center system when the field device has something to report. This means traps are much more efficient than polled-response communication, but traps have a key disadvantage, in that it is not immediately apparent when field communications have a problem.

See Annex C for more information on traps and exception based reporting.

2.7.1.5 Data Encoding

Data encoding refers to the procedures for representing the bits and bytes representation of information content to be transferred. For C2F communications, data encoding of information is governed by:

a) **Basic Encoding Rules (BER)—**BER is a series of procedures for the representation of data octets to be transferred. BER is used only in conjunction with the SNMP Application Profile; and
b) **Octet Encoding Rules (OER)—**OER is a variation of BER developed for use on low bandwidth communications links. OER can be used in conjunction with the SNMP and STMP Application Profiles.

2.7.1.6 Data Transmission

Data transmission for C2F communications is handled by the Transport and Subnetwork profiles.

2.7.2 Information Profiles for Center to Center (C2C) Communications

To date, NTCIP has not developed an information profile for C2C communications. Other standards development organizations have developed functional area data dictionaries that can use NTCIP

standards for communications. The functional area data dictionaries (TMDD, ATIS, TCIP, IM) are shown on the NTCIP framework diagram and described briefly as:

a) TMDD—The TMDD Standard for Traffic Management Center-to-Center Communications covers the functional area of traffic management, and is a joint standard of ITE and AASHTO.
b) ATIS—The Message Sets for Advanced Traveler Information Systems, SAE J2354, is a standard of the Society of Automotive Engineers.
c) TCIP—The Transit Communications Interface Profile (TCIP) is an American Public Transit Association (APTA) standard to allow transit agencies and transit suppliers to create standardized tailored interfaces. TCIP is based on the earlier NTCIP 1400-series standards.
d) IM—The incident management (IM) standards of IEEE, IEEE 1512 series, have been developed to standardize communications between transportation and emergency management centers.

2.7.3 Application Profiles for Center to Center (C2C) Communications

C2C communications require a peer-to-peer network connection between the involved computers. This is typically a local area network, a wide area network, or a dial-up connection. Local area networks typically use agency-owned twisted pair cable or fiber optic cable. Wide area networks typically use commercial telecommunications links such as frame-relay, fractional T1 leased lines, packet radio, or leased "virtual private networks." Dial-up connections typically use ISDN, V.90, or similar modems over "plain-old telephone" lines. Any type of communication link can be used, as long as it enables use of the Internet transport and routing protocols (TCP/IP and UDP/IP) and has sufficient bandwidth for the planned communications load to achieve the desired operational performance—based upon frequency, size of messages to be exchanged, and latency issues encountered when using C2C systems).

At publication, NTCIP supports two application profiles for C2C communications; DATEX and XML. NTCIP previously supported a third application profile, CORBA; however, that standard was withdrawn.

2.7.3.1 DATEX Overview

DATEX, commonly referenced as "Application Profile for DATEX-ASN (AP-DATEX)," is referenced in ISO 14827. ISO 14827 defines the rules for message exchanges, encoding, and transport for ASN.1-based communications definitions. DATEX is based on the assumption that two centers are always connected. Therefore, common dialogs for connecting, logging in, and disconnecting are defined and required. Once connected, two centers share information using a request-response message pattern or subscription-publication pattern.

DATEX was designed to provide simple, resource-effective solutions for basic needs. It is especially well suited for:

a) Systems requiring real-time, fast data transfer, for example, traffic signal status data;
b) Systems with limited communications bandwidth, but high data transfer load;
c) Systems with infrequent event-driven exchanges over dial-up links; and
d) Non-object oriented systems.

C2C networks allow each system to request any available information from any or all other systems. Each system can be configured to either accept or reject any request. The "data" sent can be informational or can constitute a "command" to take some action. Consider a message sent from one traffic signal system to another and containing a signal timing pattern number. In DATEX, for example, depending on the message type, the message could represent a command to implement that timing pattern at a particular traffic signal or group of signals, or the message could represent a status report indicating that this timing pattern was just implemented at a particular traffic signal or group of signals.

The user can also establish standing subscriptions for data, if the user wants the same data sent repeatedly. In DATEX, these subscriptions can specify that data be sent one-time-only, periodically, or repeatedly on occurrence of some event as defined in the subscription. Each subscription message has a corresponding publication message. Unless the subscription is a one-time request, the data continue to

be automatically "published" repeatedly until the subscription is cancelled, or until a predefined end date specified in the subscription.

2.7.3.2 C2C XML Overview

The application profile for C2C XML is based on the rules of message encoding and transport of the W3C's (World Wide Web Consortium) Web Services Architecture. C2C XML provides a way to define messages (using XML Schema) and dialogs (using the Web Services Definition Language (WSDL)).

C2C XML provides a way to specify WSDL for the following combinations of message encoding and transport:

a) **Simple Access Application Protocol (SOAP)—**Using SOAP encoded messages over the hypertext transfer protocol (HTTP), centers are able to describe and deploy center interfaces that support the request-response and subscription-publication message patterns.
b) **eXtensible Markup Language (XML)—**Using XML encoded messages over HTTP, centers are able to describe and deploy interfaces that support the request-response (via HTTP POST), AND request-only message patterns (via HTTP GET and file transfer protocol (FTP)).

2.7.3.3 Comparison of DATEX and Center to Center (C2C) XML

NTCIP provides two alternative application level protocol choices for C2C communications, DATEX-ASN and C2C XML. These two different protocols were found necessary to meet the variety of requirements for inter-system data exchanges. Table 2 provides a comparison between the two application profiles.

Table 2 Center to Center (C2C) Protocol Comparison

	DATEX	C2C XML
Support for Message Sets	Yes	Yes
Bandwidth Efficiency – inverse of packet overhead	Better	Worse
Ease of Implementation	Harder	Easier

2.7.3.4 Communications Patterns

Communications (or message) patterns are used to describe the general sequence of communications between two entities on a network. Three basic communications patterns, or simple dialogs, can handle a wide variety of situations. The three basic communications patterns are:

a) **Request-Response—**This communications pattern supports sending a message followed by a response. This pattern implements a synchronous pattern of message communications.
b) **Subscription-Publication—**This communications pattern supports a subscriber application performing an initial request-response to set up future asynchronous responses from an information publisher application.
c) **One-way—**This communications pattern reflects a concept intended for bulk data transfer. This pattern implements a request of a file by name.

2.7.3.5 Data Encoding

Data encoding refers to the procedures for representing the bits and bytes representation of information content to be transferred. In C2C communications, messages are encoded into the bit-byte representation prior to the start of network transfer regardless of how the information is represented in the originating and destination systems. The message encoding formats of C2C XML and DATEX are described as:

a) **C2C XML—**specifies two information encoding formats: XML and SOAP, both standards of the W3C, and an encoding format for compression, GZip, a standard of the IETF.

b) **DATEX—**specifies information encoding of ASN.1. These include Basic Encoding Rules (BER) and NTCIP 1102:2004, Octet Encoding Rules (OER) Base Protocol. In addition, projects have been encoding message content as XML.

2.7.3.6 Data Transmission

Data transmission for DATEX and C2C XML is supported by TCP/IP. In both cases a well-known socket is defined and used by TCP/IP. The data transmission mechanisms (TCP/IP applications) used in DATEX and C2C XML are described below:

a) **C2C XML**—describes two message transport mechanisms: HTTP and FTP, both standards of the IETF.
b) **DATEX**—specifies a transport mechanism based on the TCP Socket API. Clients and servers exchange messages in DATEX over TCP Socket 355.

2.7.4 Application Profiles Adopted from the Internet

2.7.4.1 File Transfer Protocol (FTP)

File Transfer Protocol (FTP) is a widely used protocol to exchange files between computing devices. It uses TCP.

2.7.4.2 Trivial File Transfer Protocol (TFTP)

Trivial File Transfer Protocol (TFTP) is a protocol used to exchange files between computing devices. It is less capable than FTP. TFTP uses UDP.

2.8 TRANSPORT PROFILES—NTCIP 2200 SERIES

NTCIP C2F protocol stacks can be used in management systems of any configuration or complexity. If the Transmission Control Protocol/User Datagram Protocol Internet Protocol (TCP/UDP IP) transport standards are implemented, then support for message routing through intermediate communications hubs or field masters is inherently included. However, a particular implementation of a C2F protocol stack may not provide support for such immediate or future options unless specifically requested at the time of procurement.

2.8.1 Transmission Control Protocol/Internet Protocol (TCP/IP)

TCP/IP is made up of two protocols, Transmission Control Protocol (TCP) and Internet Protocol (IP). TCP/IP is the most widely used protocol for internet communications. TCP/IP is used for routed networks that require a reliable protocol. A reliable protocol, in this context, means that the protocol attempts to detect and recover from transmission errors. The additional reliability also results in reduced efficiency due to overhead within the packet and more processing required. SNMP performs error handling at the application level, making UDP/IP sufficient for most NTCIP applications that use routed networks.

2.8.2 User Datagram Protocol/Internet Protocol (UDP/IP)

UDP/IP is made up of two protocols, User Datagram Protocol (UDP) and Internet Protocol (IP). UDP/IP is used for routed networks that do not require a reliable protocol (also known as non-reliable). A non-reliable protocol, in this context, means that the protocol does not make any attempt to detect or recover from transmission errors. Any detection and error recovery should be done at a higher layer. Because of this, UDP/IP communications are more efficient than TCP/IP due to reduced overhead and processing requirements. UDP/IP is recommended for NTCIP in routed networks unless the application explicitly requires TCP/IP.

2.8.3 T2/NULL

T2/NULL is a non-routed serial communication protocol. T2/Null provides multiplexing on a single serial channel and works with half or full duplex.

2.9 SUBNETWORK PROFILES—NTCIP 2100 SERIES

Devices that use any particular subnetwork protocol can share the same communications line with other devices using the same subnetwork protocol. It doesn't matter whether such devices are from different manufacturers or are totally different devices, for example, a traffic signal and a dynamic message sign. Each device is assigned an address that is unique on that line or channel. The management system can communicate with any of the devices at any time by sending a message addressed to that device. However, when using Point-to-MultiPoint Protocol, the management system can communicate with only one of the devices on the line or channel at a time. As a function of SNMP and STMP, devices can only send a message to the management system when requested to do so by the management system. NTCIP protocols enable broadcast messages intended for all devices, for example, a time clock update. No devices can reply to a broadcast message. At present, NTCIP devices cannot communicate peer-to-peer with each other exclusive of a central facility. The identification of needs and requirements for this capability are under consideration.

2.9.1.1 Ethernet

This subnetwork profile specifies the provisions for a connectionless and connection-oriented data link service and the physical interface between an end system and other compatible end systems. It has specific reference when these services are used through the Internet Protocol (IP) connectionless network service. "Ethernet" is somewhat of a misnomer. More precisely, NTCIP network-type communications are based on IEEE 802 network communications, which are similar to Ethernet but also include Logical Link Control (LLC) and Media Access Control (MAC) layers.

2.9.1.2 Point-to-Point Protocol (PPP)

Point-to-Point Protocol (PPP) is a protocol that operates in a point-to-point configuration where exactly two devices (called peers) are connected by a communications link. PPP is intended to provide an interoperability standard for transportation related devices for dialed-up circuits using V Series Modems.

2.9.1.3 Point-to-Multipoint Protocol (PMPP)

Point-to-Multipoint Protocol (PMPP) is a protocol that operates in a primary/secondary configuration where one device is the designated primary while one or more other devices are connected to one communication channel acting as secondaries. PMPP is intended to provide an interoperability standard for transportation related devices using frequency shift keying (FSK) modems.

2.9.1.4 Communication Links Using Tunneling

Tunneling describes the encapsulating or embedding of one protocol within another. A common example is tunneling of PMPP through an Ethernet network. In this example, both the central management station and the field device are "talking" PMPP, but at least part of the link between them involves encapsulating the PMPP packets within Ethernet packets. A device called a terminal server (also called an Ethernet-serial converter, port server, or other name) is used to convert the PMPP serial data to and from Ethernet. At the management station, this conversion may be performed within software on a computer in a process often called (serial) port replication. This avoids the need for multiple serial ports and terminal servers at the management station.

When serial data are tunneled through an Ethernet network, each field device (e.g., controller) may have its own terminal server (the Ethernet network extends to each cabinet). Some field devices may even have a terminal server built into the device. Alternatively, multiple field devices may share a serial multi-drop channel with one terminal server for the channel. In any case, field devices still communicate via a serial port.

As long as the protocol conversion devices or processes and the Ethernet network have adequate buffering, reliability, and low latency, such tunneling can be considered part of the physical layer and has no impact on the NTCIP-related project requirements. Both the management station and the field devices operate as if using a direct Point-to-Multipoint Protocol serial channel without the intermediate protocol conversion.

The NTCIP Framework (see Figure 4) does not attempt to illustrate the use of tunneling.

Tunneling of serial communications via an Ethernet network is quite different from using an end-to-end Ethernet link between the management station and a field device. The latter does not involve serial communications or conversion between protocols. In this case, the field device has a built-in Ethernet port and native support for the Ethernet protocol and related protocols such as TCP, UDP, and IP.

2.10 OTHER NTCIP SERIES STANDARDS

The NTCIP 8000-series documents cover Process, Control & Information Management Policy. The NTCIP 9000-series documents are informational reports.

NOTE—The previously-designated NTCIP 1400-series standards are transit standards, which were moved to APTA and are no longer maintained by or under NTCIP.

2.11 USING THE NTCIP FRAMEWORK

When a user wants to deploy an NTCIP-based system, the user chooses the protocols the user wants to deploy. A stack is a subset of the overall NTCIP framework—a selected route through the levels, given the choices available. Some stacks include two standards at some levels, which usually mean the protocol can use either of the optional standards. NTCIP protocols generally offer further options within most of the standards. The highlighted portion of Figure 6 illustrates an example of a C2F protocol stack choice that can be defined using NTCIP standards. Additional examples are included in Annex D.

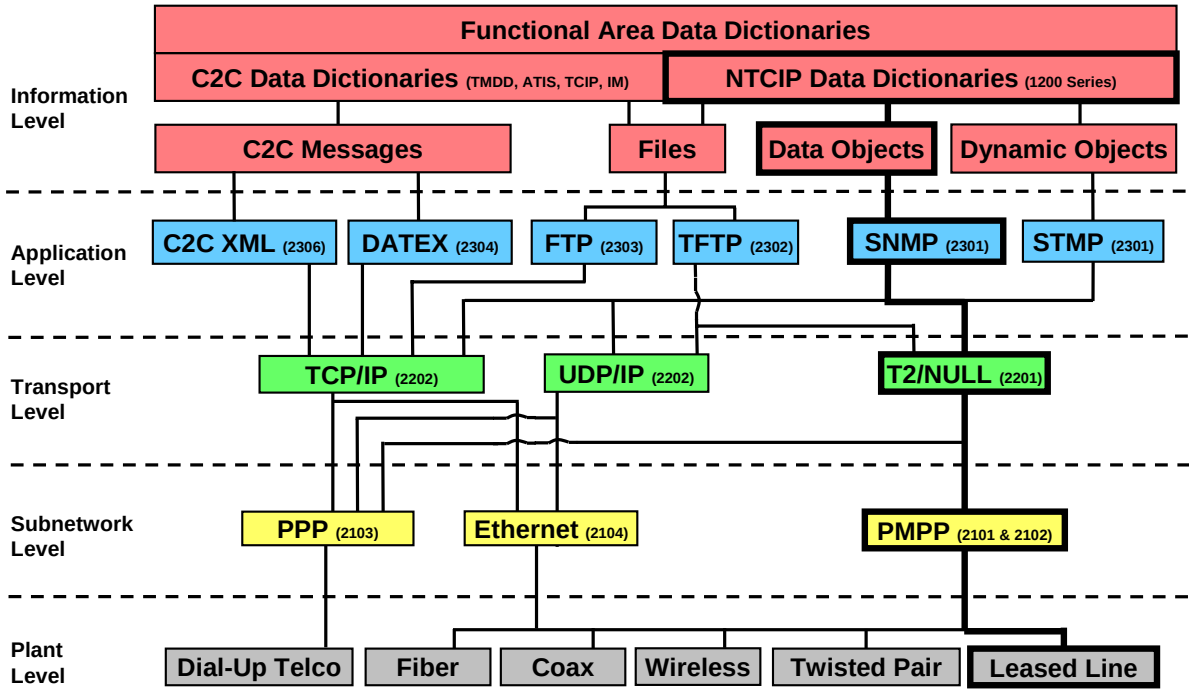

Figure 6 Example—Center to Field (C2F) Stack

The process flowchart defined in Figure 7 can help determine which NTCIP communication profiles should be used in different systems and different communications networks. The colors and document numbers in Figure 7 correspond with the NTCIP Framework diagram (see Figure 4).

The flowchart in Figure 7 can guide the user, specification writer, or developer when selecting the most appropriate NTCIP standard to use when specifying different communications infrastructure and field devices. In Figure 7, diamond shapes identify the decision points. The progression of decision points are identified by a number in brackets, for example **[1].** Boxes with a curved bottom line represent documents, namely the NTCIP standards publications. These boxes are color-coded to correspond with the colors of the NTCIP Framework diagram (Figure 4): blue for application level standard, green for transport level, and blue for subnetwork level. Agency specifications should identify the major version number for each NTCIP standards publication they wish to use. The ovals represent the start point and end point of the decision path taken for selecting the NTCIP standards publications.

Further explanation of the questions and considerations within each decision point follows:

a) **Device Uses Ethernet [1]?**—Is the primary communications port on the device an RJ-45 connector? If the answer is YES, it should be verified that the device is not using an internal terminal server, which could accept an Ethernet data packet, but internally strips the Ethernet header/footer to deliver a serial data packet to the field device application. If the device does not use Ethernet communications, then the answer is NO. If the answer is NO, then proceed with the question "Device Uses Serial **[2]**?".

b) **Device Uses Serial [2]?**—If the answer to the first question (Ethernet) is NO, then the device uses a serial interface, a dial-up interface, or a non-NTCIP-defined interface. If the device is supposed to conform to NTCIP, then the answer is YES. If the device is something different, then the answer is NO. If the answer to the first question (Ethernet) is NO, and the answer to this question (Serial) is NO, then there are no NTCIP standards for the communications configuration in question.

c) **Using Dial-Up [3]?—**If the answer to the second question (Serial) is YES, the device uses either a serial interface or a dial-up interface. If the device uses an RS-232 interface, then it is likely that the answer is NO. If the device has an internal dial-up modem or connects to an external dial-up modem, then the answer is YES.

d) **FSK Modem [4]?—**If the interface to the device is serial, the user decides whether an FSK modem (also known as Bell 202) is to be used. Even though an external FSK modem, which typically contains a regular RS-232 port into the device, is used, a user could decide to answer this question NO. If external FSK modems are used on both ends, the answer is also NO. It is a matter of where NTCIP conformance is measured, on the outside or the inside of the external FSK modem.

e) **File Transfer Required [5]?**—There is currently no NTCIP-conformant device type that solely operates using a file transfer mechanism. Agencies deploying a system should ask the central system developer and field device vendors to determine whether a particular system requires file transfers. If the system requires file transfers, the answer is YES; otherwise, the answer is NO.

f) **Ack Required [6]?**—A decision has to be made whether UDP/IP or TCP/IP is to be supported. If the file transfer protocol is FTP, then the answer is YES. If TFTP is to be supported, then the answer is NO. FTP is more commonly used but FTP requires more communication bandwidth and more processing capabilities. If neither FTP nor TFTP are used, either TCP or UDP may be used; however, UDP is recommended due to the improved efficiency.

g) **STMP Required [7]?**—STMP is a protocol that is only supported by Actuated Signal Control (ASC) devices at this time. The need to use STMP is based on system bandwidth requirements and the media used.

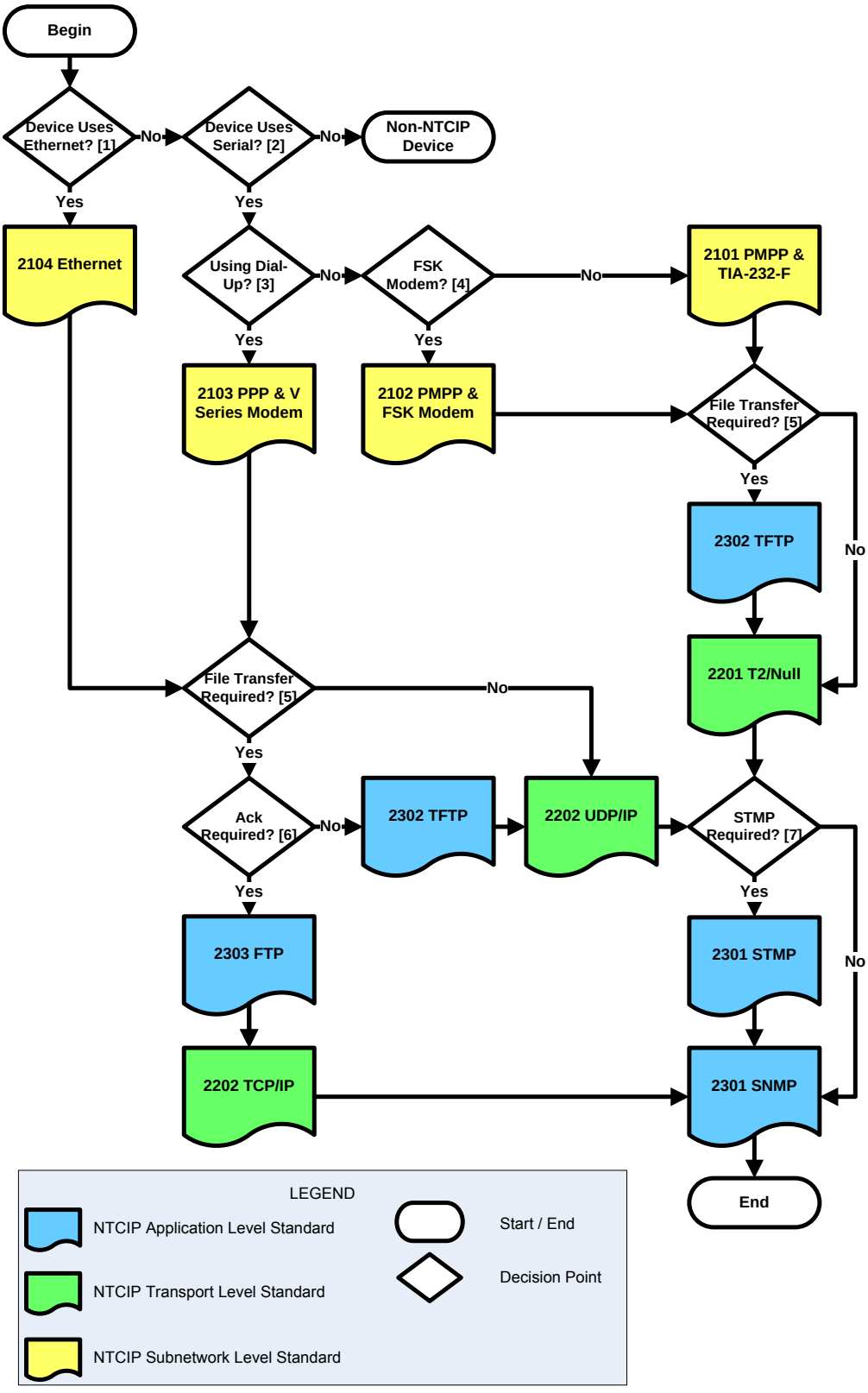

Figure 7 NTCIP Profile Selection Flowchart

Figure 8 illustrates an example C2C protocol stack choice that can be defined using the NTCIP standards. Additional examples are included in Annex D.

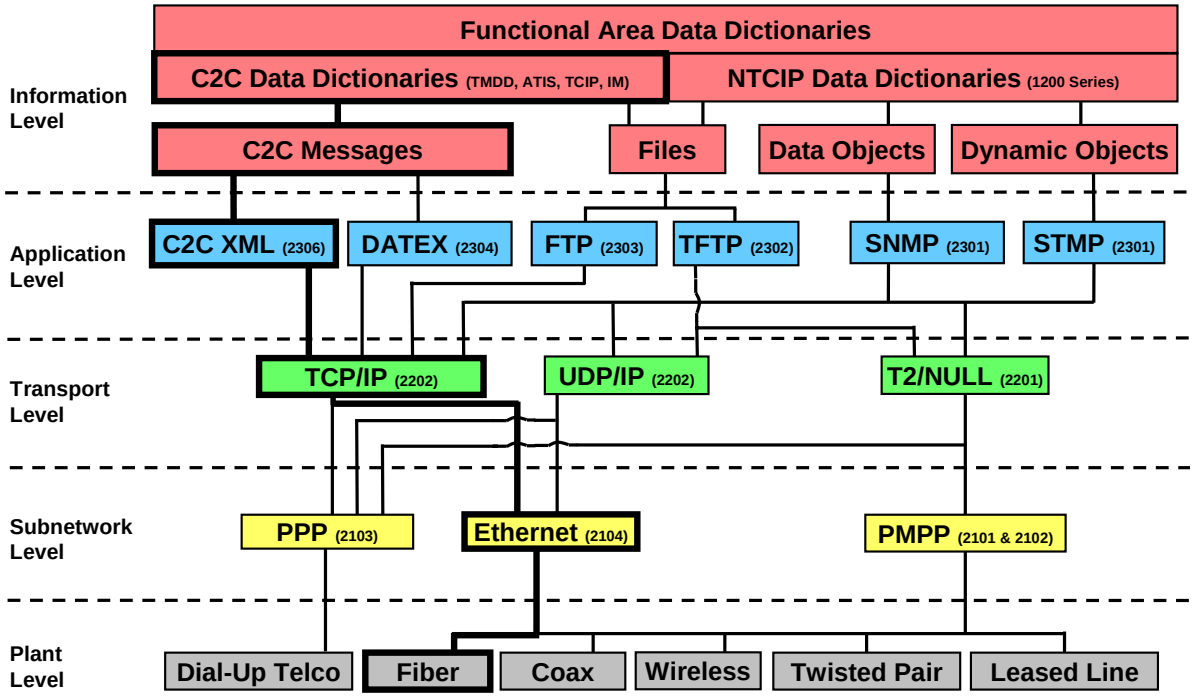

Figure 8 Example—Center to Center (C2C) Stack

2.12 CONFORMANCE

Each NTCIP standard contains a section addressing "Conformance" which clearly states the requirements for conformance to the standard by implementations that embody that standard. The ease or difficulty of determining conformance depends on the complexity of the standard and the resulting implementation.

- **Standards Conformance—**When an item fulfills all of the mandatory requirements as defined by a standard.
- **Specification Compliance—**When an item fulfills all the requirements of an agency specification.

Conformance differs from compliance. The term compliance is most often used in contractual language to assess the legal determination of meeting or not meeting an agency's specifications of the contract. An agency creating a specification, based on NTCIP standards, is necessary to ensure that all optional elements in the standard are required for your deployment. An agency specification allows for the explicit removal of optional elements of the NTCIP standards that do not apply. Therefore, while a vendor may state that their devices are "NTCIP conformant," meeting conformance with the minimal mandatory requirements of an NTCIP standard may not be sufficient to fulfill agency project requirements, or to achieve compliance with an agency specification.

2.13 OPTIONS AND CONFORMANCE LEVELS

In addition to specifying a protocol stack, the system designer should also choose between alternatives available in the selected stack. These options exist in both C2C and C2F protocol stacks. Major options, such as which protocol(s) to support at each level in the communications stack, are sometimes grouped according to conformance levels, while others are individually selectable. Most manufacturers and system

suppliers typically offer features that go beyond the standard. To make use of such features, it is necessary to specify the inclusion of manufacturer-specific data elements or messages as extensions of the standards when procuring a management system.

An agency decision to use features above and beyond the standard should be made only with the understanding of the potential impacts. These impacts could be considerable in the long term. These options may, in effect, result in the acquisition of proprietary systems. Part of the agency decision should include the number of such features allowed.

Details on options and conformance levels, and how to specify your selection, are presented in later sections of *The NTCIP Guide*.

Section 3
PROCURING NTCIP

3.1 INTRODUCTION

Procuring transportation related systems that use NTCIP communications involves additional considerations beyond what is required for non-NTCIP systems. When procuring systems that use proprietary non-NTCIP communications, the system vendor is normally responsible for ensuring that both ends of the communications link can communicate and support system features. In contrast, when procuring NTCIP systems, the agency is responsible for specifying the appropriate NTCIP communications requirements. It is not sufficient to simply say, "The device shall be NTCIP compliant." Section 3 describes the additional procurement considerations for specifying NTCIP communications requirements.

Agencies acquiring transportation transportation-related systems want the system to have specific features that meet agency requirements. For a system feature to be available across a communication link, the components on each end of the link should support that feature and also support communications related to that feature. The goal of the communications portion of an agency specification is to ensure that communications support the desired features and that both ends of the communications link are compatible with each other.

Section 3 describes how to identify which NTCIP standards are applicable and how to generate the communications portion of an agency specification. Section 3 also highlights some agency specification considerations that are not unique to NTCIP.

3.2 DETERMINING APPLICABLE NTCIP STANDARDS

NTCIP is a family of standards addressing different levels or layers of the communication hierarchy. This hierarchy, with associated series of NTCIP standards, is described in Section 2. For C2F applications, a good starting place for defining NTCIP requirements is at the information level using an NTCIP 1200-series standard. For C2C applications, the information level standards are defined outside of the NTCIP program, and an agency should refer to the appropriate standard for information on how to specify the information level. Therefore, the NTCIP portion of an agency specification for C2C should begin at the application level.

3.3 DEVELOPING NTCIP SPECIFICATIONS

If an agency is planning to prepare detailed specifications for NTCIP-based systems, the agency should select the appropriate standards from each level within the NTCIP Framework. To effectively make these selections, a good understanding of agency resources and operational capabilities is needed, as well as any existing communications constraints, such as existing communications infrastructure and equipment. System requirements describe a set of desired functionality that satisfies an agency's operational and user needs plus communications and infrastructure constraints.

Specifications for NTCIP communications are just one part of a complete agency specification.

NTCIP communications specifications are just one necessary part of the specification an agency produces to procure a transportation-related system. Other specifications include: the physical construction, environmental requirements, installation requirements, testing requirements, warranties, etc.

Section 4 describes the systems engineering process (SEP) that is helpful to follow when procuring a complex system. NTCIP communication requirements derive from an agency's operational and user needs. If an agency intends to procure a commercial off the shelf (COTS) system rather than specify a

custom designed system, the agency should compare their communications and non-communications requirements with the capabilities of commercially available ITS devices before finalizing an agency specification.

Many NTCIP standards are designed around SEP. Section 4 describes how to produce an agency specification for NTCIP communications standards. Other NTCIP standards are not designed around SEP. The following process describes how to produce an agency specification for NTCIP communications standards.

Some of the newer NTCIP standards are designed around a systems engineering process. Section 4 addresses how to specify those NTCIP standards using the systems engineering process.

To effectively prepare detailed agency specifications for an NTCIP-based system, the systems planner/specification writer should consider and document the agency's functional and operational needs, so that a deployed system satisfies those needs.

The next step in the process is to identify the mandatory and optional elements within each NTCIP standard that are needed to ensure that an agency's identified functionality needs are met. An optional element in an NTCIP standard that is required for an agency project becomes mandatory in an agency specification. The optional and mandatory elements of an NTCIP standard are documented in its Profile Requirements List (PRL), which an agency should tailor to satisfy a particular project's needs.

NOTE—Earlier versions of NTCIP standards contained conformance groups, which PRLs replace.

PRLs, in turn, define the "features" of NTCIP standards. These features of NTCIP standards enable the requested functionality within an implementation. Using a top-down approach, the system's functional requirements determine which elements of the PRL (previously conformance groups) are mandatory to provide the features that enable that stated functionality.

Once the applicable requirements are selected, and their included data element/object sets are identified, realistic value ranges are defined for each of the data elements, or data objects. These value range choices are based on the functional requirements of agency device specifications. It is important to remember that detailed NTCIP design elements can, and should be, tailored to meet the intended needs of the system being implemented. Otherwise, an agency may expend resources unnecessarily to acquire more functionality than is truly needed.

An entire NTCIP *stack* should be defined, with identification of NTCIP (or other) standards required at each level. This process should be repeated for each level of the NTCIP framework. NTCIP standards that are not designed around SEP contain an information profile listing the conformance groups with their objects. An agency can use this profile to select which conformance groups and objects are required. This completed form is then called a Profile Implementation Conformance Statement (PICS). An agency should use this completed form to specify the NTCIP communications requirements of the system.

If the system or ITS device being procured are to interface with an existing system, the agency may want to include a copy of the existing MIB as part of the agency specification to ensure compatibility of both ends of the communications link. Examples of this situation include adding new field devices to an existing central system or upgrading a central system to operate with existing field devices.

The process for NTCIP standards that are designed around SEP is slightly different and is described in Section 4.

3.4 ADDITIONAL PROCUREMENT CONSIDERATIONS

The project scope, deliverables and agency specifications should also include information related to hardware and/or devices, systems integration, testing and device configuration. Agency specifications

should also address the ownership, re-distribution and/ or re-use rights of the MIB, as well as requirements for documenting and obtaining the MIB.

Agency specifications should address ownership, re-distribution, documentation and/or re-use rights of the MIB.

Agencies should obtain final electronic and paper copies of the MIB(s) used in their system, including both standardized data elements and manufacturer-specific data elements. Re-distribution and/or re-use rights should be clearly spelled out at the onset of the project so that future expansion and integration issues can be minimized. Overall, a comprehensive set of system requirements, as well as design and other elements of agency specifications, helps the system/hardware implementation proceed more smoothly and with less ambiguity.

3.4.1 Software Acquisition

When agencies move to implement or upgrade center-based systems, the bulk of the work involves the acquisition of software. In the case of C2C communications, software is developed to provide a means of communicating between two central systems.

The National Highway Institute (NHI) has developed an ITS software acquisition course, based on the U.S. Department of Transportation Federal Highway Administration (FHwA) *The Road to Successful ITS Software Acquisition,* that lays out a successful approach to acquiring software. The ITS software acquisition course presents a variety of themes that are useful in any project, but are especially important in a software acquisition project. The main themes presented in the NHI course include:

a) System Themes—relating directly to the final product
   1) Break the project up into “bite-size” pieces
   2) Consider COTS software whenever possible
b) Management Themes—managing the acquisition
   1) Up-front planning is essential
   2) Maintain flexibility throughout the process
   3) There are no “silver bullets” or magical cures for troubled software projects
c) People Themes – partnering and team building
   1) Maintain active customer involvement
   2) Maintain good collaboration
   3) Open communications is essential
   4) Team building is important

The NHI—ITS software acquisition course also presents the concept of a requirements walkthrough as a means of describing essential elements of the acquisition that should be well documented at the onset of the project.

Important factors affecting ITS standards-based implementations include:

a) **Property Rights and Permissions—**Who owns the software being implemented? What rights do I have as a user? It is important to understand the difference between ownership and right to use. Right to use licensing implies restricted rights of use and distribution, while ownership may only be minimal or nonexistent. There are important implications to consider when evaluating which approach is right for your agency. Remember that intellectual property rights issues should be resolved prior to signing a contract.
b) **Delivery—**What is my timeline for system delivery? Does the schedule adequately reflect time needed for software development (hurried software development can lead to implementation problems)? What is my method of acquisition and delivery (for example, software delivery may be acquired via a professional services arrangement)?
c) **Acceptance Testing—**How is the system tested? How can an agency determine whether agency specifications are met? How can an agency determine whether the requirements of the pertinent standards are met? Are staff resources used to do system testing or is this outsourced? Is a good test

plan generated from agency specifications and the pertinent NTCIP (or other) standards? What tools are available within the agency, and what tools are needed to perform adequate testing?

Other important factors to consider in the procurement of NTCIP or other standards-based systems that may have a profound affect on post implementation include:

a) **Maintenance and Support—**Once the system is built, how does an agency maintain the system in good working order? What kind of support is needed to maintain the system (e.g. additional staff, vendor contracts, etc.)? Standards are often undergoing initial development or revision, and the use of draft standards brings risks of substantive change prior to acceptance and publication—how can an agency upgrade to the final standard if the agency was involved in a lead deployment based on draft standards?
b) **Documentation and Training—**What documentation does an agency need to have on hand to ensure that staff adequately understands the system? What are agency training needs (consider both operations and maintenance)? Are additional agency staff resources needed with specialty training?
c) **Warranty Considerations—**What is the warranty period? What resources and benefits are associated with extended warranties?

For NHI course information, see www.nhi.fhwa.dot.gov. FHWA's *The Road to Successful ITS Software Acquisition* is at www.fhwa.dot.gov/tfhrc/safety/toc.htm .

3.4.2 Procurement Methods

When procuring NTCIP standards-based transportation system, agencies typically use existing procurement processes, selecting one depending on what the agency is procuring. Agency procurement processes for hardware may differ from those for software, field installation, or construction, as examples. For software development, the agency process for procuring technical services may be used.

When procuring ITS systems that include software development, an agency's information technology section may provide assistance.

When procuring ITS systems, the agency needs to determine the most appropriate process or combination of processes for the particular procurement at hand. If the traffic engineering section of the agency is not familiar with procuring software, the agency's information technology section may have such experience and could provide assistance.

As one example, the agency may wish to procure additional field devices to operate with an existing central system. In this case, the agency develops specifications for the field devices, including NTCIP-based communications requirements.

If the agency wants a contractor to install the devices in the field, along with electrical wiring and support structures, an agency could include the field devices with the field construction to be let as part of a construction project; however, it is often desirable for the agency to procure field devices separately from the construction aspects of a project. Separating field device procurement from the construction project allows the agency to deal directly with the field device vendor to resolve any issues, and avoids making a construction contractor responsible for installation tasks that are outside of the normal construction skill set. The agency's options include:

a) The construction contractor installs agency-provided field devices or
b) The agency installs field devices into cabinets and onto supports that the electrical contractor has installed.

As another example, the agency may wish to procure software for a center system or for C2C communications. Considerations include:

a) If COTS software exists that meets most of the agency's needs, the agency may want to license the software.

b) If no COTS software exists, the agency can either develop custom software internally or procure the development of custom software. Development of custom software often has high risks for achieving desired functionality and reliability, meeting the project schedule, and staying within project budget.
c) For a large center with many software applications, it may be more effective to develop and maintain software using agency employees.
d) For smaller operations without full time agency employees with software expertise, the agency may choose to procure outside technical services for software development. See 3.4.1.

Two alternative approaches can be used when preparing to procure NTCIP systems and devices.

One agency procurement approach is for the agency to solicit proposals and allow respondents (the manufacturer, vendor, developer, or systems integrator) to present detailed information on how their proposed system or device meets agency requirements or conforms to elements of NTCIP standards. This relies on the preparation of requirements during the initial stages. These requirements should be as detailed as possible to ensure that resulting responses provide sufficient information for fair and even comparisons among competing alternatives. At some point , either during selection or after award, the agency solicits (for approval), a detailed proposal from the respondents (systems developer or integrator) that presents information as to which standards, conformance groups (PRL or PICS), data elements and range values can be provided to meet the agency specification requirements. Other pertinent information should be included. This approach might be somewhat iterative, depending on agency requirements. This approach, however, is not appropriate for the procurement of additional field devices to operate with pre-existing central system software.

The second agency procurement approach involves the preparation of an agency specification, which includes detailed NTCIP system (or device) based and agency functional requirements. This approach requires knowledge of both NTCIP standards and the device or system functionality an agency requires to meet agency needs. This approach requires the agency to make appropriate selections at all levels within the NTCIP Framework and identify an NTCIP stack that meets agency needs. The agency identifies specific NTCIP standards, PRLs, PICS, data elements and range values. This approach requires agency expertise in communications and systems design, as well as identification of agency-specific needs and requirements.

While either approach may be used for initial procurement, the second approach is appropriate when existing central system software is to be re-used without modification.

3.4.3 Procurement Request

The procurement request can take many forms. Additional steps may be added to traditional procurement processes to ensure an adequate understanding of the requirements, and means used to evaluate conformance to elements of NTCIP standards and agency specifications.

3.4.4 Procurement Response

To ensure a thorough understanding of agency specification requirements, the agency should request a proposal (for agency approval) that reflects an understanding of and acknowledges the various issues addressed through the development of detailed design plans and requirements using the systems engineering approach. This proposal should address issues such as conformance requirements statements and range values, where appropriate, and any requirements that are unique to agency specification for a project.

If proposals deviate from agency specifications, the agency decides whether or not to allow such exceptions. If exceptions are allowed, the agency should identify the impact of these exceptions. Exceptions should be reflected within agency specifications to ensure that future procurements are interoperable and/or interchangeable. A good practice includes review of exceptions by all parties involved—both internal and external to the agency.

Exceptions may reflect options provided within a standard, or exceptions may render an implementation non-conformant. Working within the context of what is allowed within a standard promotes and enhances

future interoperability and interchangeability. Exceptions that detract from the intent of the standard should be avoided.

3.4.5 Maintenance

Agency specifications may need to address operational maintenance, version maintenance and subsequent device and/or software upgrades. Operational maintenance requirements accommodate the reality that ITS devices and component subsystems are often deployed over a wide geographical area. Agency specifications should consider addressing needs or requirements for fault detection, remote troubleshooting and diagnosis, and availability of service personnel and replacement parts.

3.5 OTHER PROCUREMENT CONSIDERATIONS

Other issues may merit consideration during the agency specification and procurement process.

3.5.1 Using Newly Adopted Standards

All standards are subject to future revision and amendment, for various purposes, including development of new technology, addition of functionality, improvements in the efficiency or “elegance” or system design.

Revisions and amendments may also reflect the experience of developers who have implemented the standard and identify inconsistency within a standard, lack of completeness, ambiguous wording, or advances in technology or implementation. “New” standards, the first version of a standard to be adopted and published, may be more frequently revised or amended to address these issues. Standards typically become more mature and stable over time.

If an agency pursues early adoption and implementation of standards (particularly prior to adoption and publication), the agency should work closely with manufacturers, systems integrators and developers to make informed choices and minimize associated risk.

3.5.2 Support of NTCIP Standards Amendments or Revisions

An agency should develop an approach to accommodate future revision or amendment of NTCIP (or other) standards cited in an agency specification, should the revision or amendment occur during the life of the project. It may be difficult to require developers to support amendments or revisions that are made late in the systems engineering life cycle. However, proposed revisions or draft amendments may be available during the initial procurement stages, and the agency should require developer proposals to address the existence of proposed revisions or amendments.

If additional (or replacement) field devices are procured based on conformance to standards revisions or amendments, some new functional capabilities of the new device may not be available when using existing central system software that conforms to a previous version of a standard. If such “backward compatibility” is required, those requirements need to be clearly stated in the agency specification. In any case, backward compatibility may, or may not, be achievable depending on revision or amendment of the standard, or availability of advances in related hardware or software technologies.

3.5.3 Performance Specifications

NTCIP standards, and their associated flexibility, may trend toward more sophisticated processors or better communication facilities than traditional systems to provide similar performance levels (such as, operational response times). If the agency overlooks these issues early in its requirement development process, significant implementation issues or delays could arise later in the project.

3.5.4 Extensions to NTCIP Standards

NTCIP standards do not define standardized data elements for every technology or every functional feature of every device. Some special features or requirements in the agency specification may not yet be included in an NTCIP standard. To accommodate this situation, NTCIP standards allow extensions.

If such features or requirements are present, then the systems developer or integrator should determine precisely how special features or requirements can be supported without conflicting with other aspects of NTCIP standards, and agencies should be aware that manufacturer-specific extensions might lock an agency into a specific manufacturer's solution.

An agency should only accept an extension in response to an agency-specific requirement, and extensions should be included in the agency specification and documentation deliverables.

3.6 MANAGEMENT INFORMATION BASE (MIB) ISSUES

In his book, *Understanding SNMP MIBs*, Perkins defines MIBs as follows:

> MIBs are specifications containing definitions of management information so that networked systems can be remotely monitored, configured, and controlled.

Early in the procurement process, the agency should obtain redistribution or re-use rights to the entire MIB, including any extensions, as well as both electronic and paper copies of the entire MIB (including any extensions). The MIBs should be provided in American Standard Code for Information Exchange (ASCII) format on the medium of the agency's choice. The MIBs should include all SNMP/STMP data elements that the device(s) support(s).

Section 4
AGENCY REQUIREMENTS AND SPECIFICATIONS

4.1 INTRODUCTION

Section 4 is written specifically for the systems planner or agency specification writer, or that person responsible for preparing system requirements and specifications for NTCIP-based devices and systems.

Agency specifications should not include over-simplified statements such as *"All components shall be NTCIP compliant,"* or *"The system shall use NTCIP as the communications protocol."* Neither of these statements, nor those that simply list the NTCIP publication numbers, provide sufficient information to manufacturers or systems integrators on the type, scope and functionality of the system or hardware an agency wishes to implement. While manufacturers can derive matching NTCIP requirements from the agency specifications, agencies should identify means to determine whether proposed systems or hardware address agency requirements. User needs that have *no* matches in NTCIP standards are likely to be addressed on an implementation-specific basis, which may not serve the agency's best interest.

There is no specific wording that can simply be copied into agency specifications, because there is no single system design that is standard across the transportation industry. Available resources and needs and requirements vary from agency to agency and as a result, system designs vary from agency to agency. While defining a system that encompasses all available functionality and options is an unwarranted burden on any agency, defining a system with minimal functionality may not meet the needs and requirements of any agency. Section 4 focuses on a process to follow to develop an agency specification for systems and equipment implementing NTCIP communications.

4.2 SYSTEMS ENGINEERING PROCESS (SEP) OVERVIEW

SEP is an approach to designing projects that employs a process to develop the concept of operations, user needs and requirements, design, build, testing, evaluation, and implementation of a system. SEP requires that a project team consider all phases of a system's life cycle from the moment of system conception through installation, operation, and maintenance.

The SEP model shown in Figure 9, also known as the "V" diagram, identifies the various stages in the acquisition of an ITS device or system.

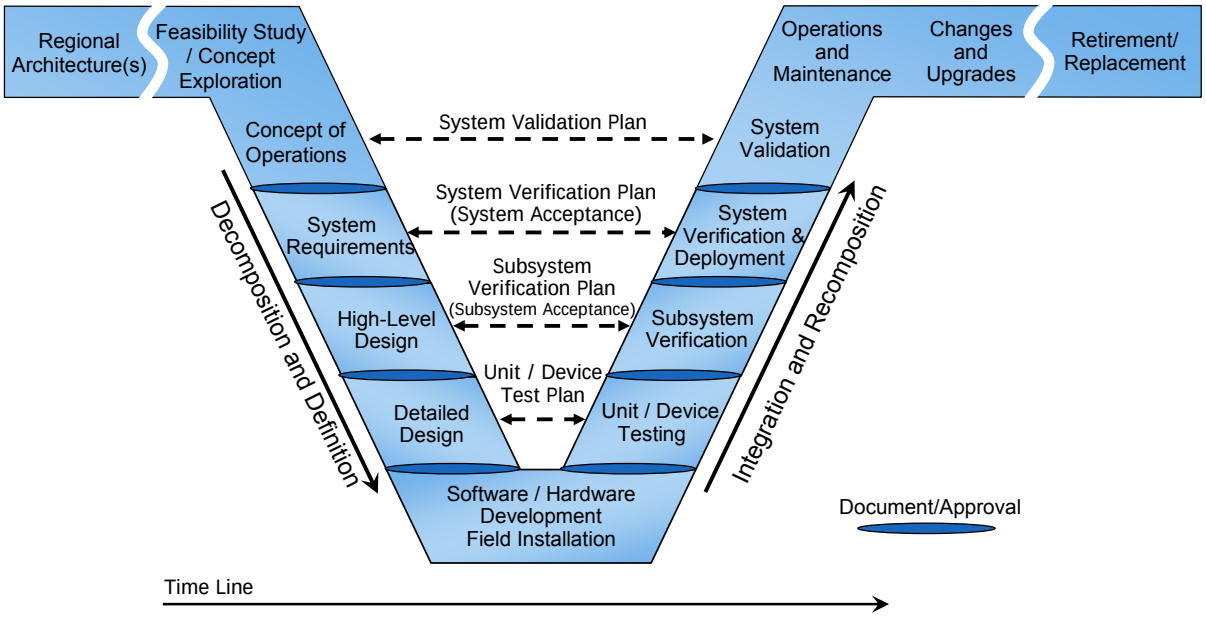

Figure 9 Systems Engineering Process (SEP) "Vee" Diagram

The major process steps comprising the SEP, starting from concept of operations through testing, include:

a) **Develop a Concept of Operations (ConOps)—**Concept definition is an early step in SEP. During this phase, the project should document what the problem is from a user's perspective. This leads to the development of a ConOps, describing the intended operation of a system from the user's point of view.
b) **Define Agency System Requirements—**Requirements definition is a key SEP activity. A set of system requirements defines the functions a system performs, how well a system should perform, and under what conditions performance takes place. System requirements are the result of the definition of need, the operational concept, and the system analysis. System requirements are a description of what the system's customers expect it to do for them. See IEEE Std 1233-1998.
c) **Develop Agency Specifications.** Based on agency system requirements, agency specifications are developed to guide system design phases. Agency specifications may be based on a tailoring of the ITS hardware and environment standards (e.g., NEMA TS 4) and the NTCIP communications standards (e.g., NTCIP 1203 v02). Standards-based agency specifications are not a system design, but rather specify design requirements.
d) **Design to the Requirements and Specifications—**Once there is a clear understanding of the project requirements, the developer can investigate ways to design a solution that fulfills the information and performance requirements of the system.
e) **Implement the Design—**This phase of development is where coding and unit testing of communications software occur. Implementation translates the detailed design of the NTCIP communications software into code capable of running on a computer. Each piece of the system is installed at a center, in the field, or in a vehicle.
f) **Prepare Test Documentation—**Test documentation is prepared based on an agency's specifications and is traceable to requirements. Therefore, once an agency specification is developed, the agency should develop test documentation, which specifies what and how to verify that delivered ITS devices fulfill agency specifications and associated requirements. Test documentation specifies the extent of testing that is required for the ITS device.
g) **Execute Tests Based on Test Documentation—**All testing is conducted based on the test documentation. NTCIP 9012 v01 references IEEE 829 and NTCIP 8007 v01 to describe the content of test documentation. Test plans are executed and the results documented in test reports (test logs,

test incident reports, and test plan summary reports). Testing progresses through a series of test phases, from prototype test through burn-in and observation test.

4.3 APPLICATION OF SYSTEMS ENGINEERING PROCESS (SEP) IN NTCIP

Recent versions of NTCIP standards are organized to align with SEP, and generally include:

a) Section 2—Concept of Operations (ConOps)
b) Section 3—Requirements (focusing on Functional Requirements, as well as Architectural, Data Exchange, and Supplemental, and including a Profile Requirements List (PRL))
c) Section 4—Dialogs and Sequence Diagrams
d) Section 5—Data Dictionary; the MIB(s) and Object Definitions
e) Annex A—Requirements Traceability Matrix (RTM)
f) Annex C—Test Procedures and Test Cases (including a Requirements to Test Case Traceability Matrix (RTCTM))

NTCIP standards contain traceability threads to:

a) Validate that user needs in the ConOps are satisfied by the functional requirements,
b) Verify that requirements are fulfilled by the design elements of the standard contained in the dialogs and data dictionary, and
c) Verify that requirements are testable and fully accounted for by the test cases and procedures.

These traceability threads are documented in Figure 10.

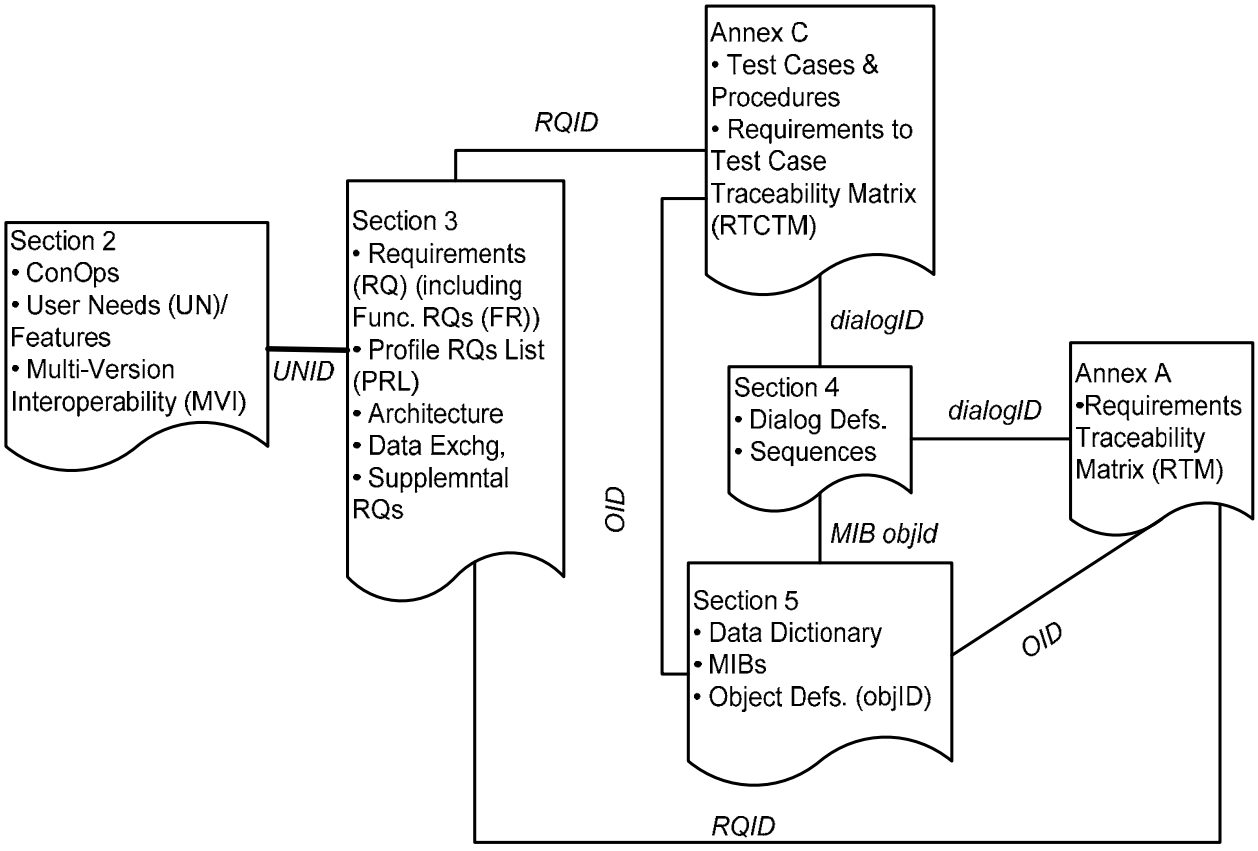

Figure 10 Traceability Threads in NTCIP Standards

4.3.1 Concept of Operations (ConOps) and User Needs

4.3.1.1 Concept of Operations (ConOps)

A Concept of Operations (ConOps) is a document, written from the user and agency's point of view, that clearly defines the situation or problem scope, identifies user needs, and the operational context for the information exchanges that the system interface is expected to support. The ConOps, therefore, should be developed with participation from all users that benefit from or are impacted by the system.

4.3.1.2 Attributes of Well-Written User Needs

Criteria for well-written user needs include:

a) **Uniquely Identifiable—**Each user need should be uniquely identified, i.e., each need is assigned a unique number and title.
b) **Major Desired Capability—**Each user need should express a major desired capability in the system, regardless of whether the capability exists in the current system or situation or is a gap.
c) **Solution Free—**Each user need does not include a stated solution, thus giving designers flexibility and latitude to produce a feasible solution.
d) **Capture the Intent and Rationale—**Each user need should capture the rationale or intent as to why the capability is needed in the system.

4.3.1.3 User Needs Contained in NTCIP Standards

In the ConOps section, many NTCIP standards identify and describe user needs that users may want the device (or system) to accommodate. Why focus on user needs? Because user needs tend to remain stable over time (if needs changed frequently, it would be impossible to build a system interface to satisfy those needs). It is this inherent stability in user needs that bounds the scope of the system interface. Well-written needs describe one or more system features and the intent of the said need in addressing a user problem or responsibility. User needs then drive the requirements definition and allow development of complete and correct requirements.

User needs identified in NTCIP standards do not reflect all possible user needs that may be desired. User needs only reflect those features that are commonly desired by stakeholders and thus are supported by the standard. Each agency may have additional user needs not identified by an NTCIP standard, and those user needs should be included in an agency specification.

See Figure 11 for an example user need from NTCIP 1203 v02.

2.5.1.2 Determine Sign Display Capabilities

This feature allows the operator to retrieve the necessary information to produce a rendering of a suggested or active message. This feature also allows the system to ensure that a message can be displayed on the DMS. The feature will allow the operator to determine the detailed physical limitations of the DMS as well as details regarding the current fonts and any graphics that are stored.

Figure 11 Example—User Need

4.3.2 Functional Requirements

4.3.2.1 Requirements Overview

One definition of a requirement is a condition or capability needed by a user to solve a problem or achieve an objective (see IEEE Std 1233-1998). Requirements comprise the basis of agency specifications and testing and play a cross-cutting role in governing the expectations of a system across the entire system life cycle.

Everyone involved in system acquisition should share an understanding of what capabilities the system should have. These capabilities should be described at a functional level and not used to prescribe a solution. Required functions should use “shall” in the sentence, and above all, the requirement should be testable. Functions should be defined in a manner reflective of the nature of the operation, such as being manual, automated, or semi-automated.

When considering the implementation of a project, it is a good practice to understand the requirements of the devices and/or systems being implemented. Knowing these requirements early in the project life-cycle can alleviate potential problems during subsequent phases. Successful projects rely on the understanding of functional, design, and testing requirements before any procurement, development or implementation.

4.3.2.2 Attributes of Well Written Requirements

The Federal Highway Administration Systems Engineering Guidebook for ITS provides an excellent summary of the attributes of good requirements (see www.fhwa.dot.gov/cadiv/segb/ or Table 3

Table 3 Summary of Quality Attributes for Requirements

Quality Attribute	Validate by:
Necessary	Make sure that each requirement traces to either a user need in the ConOps or a parent requirement. A computer can check that the traceability is complete, but people have to verify that the identified traces are valid.
Clear	Some requirements management tools can help with this by looking for red-flag words and constructs in the requirements (e.g., “user friendly”, “optimum”, “real-time”, pronouns, and complex sentences). Most of this aspect of validation relies on walkthroughs and other reviews to make sure the requirements aren’t subject to different interpretations. The main culprit here is ambiguity in the English language.
Complete	Does every stakeholder or organizational need in the ConOps trace to at least one requirement? If you implement all of the requirements that trace to the need, is the need satisfied? A computer can answer the first question, but only stakeholder(s) can answer the second.
Correct	In general, it takes a walkthrough to verify that the requirements accurately describe the functionality and performance that should be delivered. The stakeholders should validate that the highest-level system requirements are correct. Traceability can assist in determining the correctness of lower-level requirements. If a child requirement is in conflict with a parent requirement, then either the parent or the child requirement is incorrect.
Feasible	Again, this should be determined by review and analysis of the requirements. A computer can help with the analysis and possibly even flag words like “instant” or “instantaneous” that may be found in infeasible requirements, but a person ultimately makes the judgment of whether the requirements are feasible. In this case, it is the developer who can provide a reality check and identify requirements that may be technically infeasible or key expenditure drivers early in the process. Since system performance is dependent on system design and technology choices, requirements feasibility continues to be monitored and addressed as the system design is developed.
Verifiable	Does the requirement have a verification method assigned? (This is something a computer can check.) Is the requirement really stated in a way that is verifiable? (This much more difficult check can only be performed by people.) For example, ambiguous requirements are not verifiable.

4.3.2.3 Functional Requirements Contained in NTCIP Standards

Many NTCIP standards identify and define the functional requirements for a communications interface based on the user needs identified in the ConOps. These requirements satisfy user needs.

Example requirements from the NTCIP 1203 v02 standard are shown in Figure 12. These particular requirements trace backward to the user need **2.5.1.2 Determine Sign Display Capabilities.**

3.5.1.2.1 Determine Basic Message Display Capabilities

Requirements for determining the basic message display capabilities of the sign face are provided in the following subsections.

3.5.1.2.1.1 Determine the Size of the Sign Face

The DMS shall allow a management station to determine the height and width of the sign face.

3.5.1.2.1.2 Determine the Size of the Sign Border

The DMS shall allow a management station to determine the size of the horizontal and vertical border around the sign face.

3.5.1.2.1.3 Determine Beacon Type

The DMS shall allow a management station to determine the configuration of any beacons attached to the DMS, which may be 'none'.

3.5.1.2.1.4 Determine Sign Access and Legend

The DMS shall allow a management station to determine the access mechanism to the sign internal components and whether the DMS has a legend.

Figure 12 Example—Requirements

4.3.3 Dialogs and Sequences

A dialog describes a sequence of message exchanges between two entities. For example, a request-response message sequence uses a request message and a response message to accomplish information sharing. In the first part of the sequence, the request message is sent from an external center to an owner center. In the second part of the sequence, the owner center returns a response message to the external center. Some dialogs are simple and include one or two exchanges, while complex dialogs include a larger number of steps and alterations of sequence steps based on some criteria, for example, special error handling.

Many NTCIP standards identify and define the dialogs that fulfill the functional requirements supported by the NTCIP standard. While NTCIP standards do allow two entities to exchange information in any sequence to fulfill a requirement, this flexibility presents a challenge to ensuring interoperability. Thus, standardized dialogs are presented in the NTCIP standards to provide a base level of interoperability. So long as the devices support the standardized dialogs in the NTCIP standards, a particular level of interoperability is provided.

Some dialogs are simple, such as the request-response message sequence. Other dialogs are more complicated and thus are presented in the NTCIP standard for clarification. Some dialogs may also be accompanied by an informative figure that provides a graphical depiction of the normative text.

An example dialog from NTCIP 1203 v02, **Dialog 4.2.3.1 Activate Message**, is shown in Figure 13.

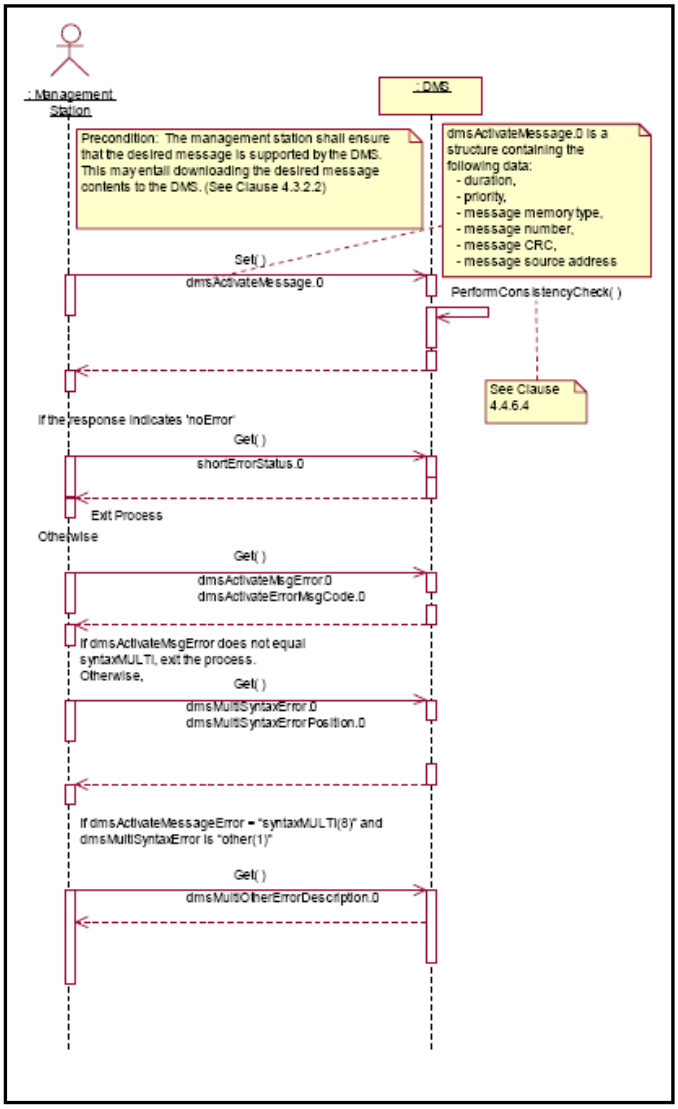

Figure 13 Example—Dialog

4.3.4 Data Dictionary

NTCIP information level standards provide documentation and registration of the data passing through an ITS interface in a data dictionary. Among the data dictionary elements are dialogs, messages, data frames, and data elements. These data dictionary elements are written using the standardized ASN.1.

An example data element from NTCIP 1203 v02 is shown in Figure 14.

5.7.19 Position of MULTI Syntax Error Parameter

```
dmsMultiSyntaxErrorPosition OBJECT-TYPE
SYNTAX INTEGER (0..65535)
ACCESS read-only
STATUS mandatory
DESCRIPTION
"<Definition> This is the offset from the first character (e.g. first
character has offset 0, second is 1, etc.) of the MULTI string where the
SYNTAX error occurred.
<Unit>character
<Object Identifier> 1.3.6.1.4.1.1206.4.2.3.6.19"
::= { signControl 19 }
```

Figure 14 Example—Data Element (NTCIP Object)

4.3.5 Requirements Traceability

Requirements traceability refers to the ability to follow the life of a requirement from its origins, through its planning phases and specification, to its subsequent deployment and use. The process of requirements traceability traces user needs to requirements that satisfies those needs, then traces each requirement to the place where it is fulfilled in the agency specification, design, implementation, and operation.

Requirements-based verification at each SEP phase provides a method of system quality improvement by helping to:

a) Find and eliminate defects early on;
b) Find requirements gaps and inconsistencies (i.e., conflicting requirements);
c) Find requirements redundancies; and
d) Uncover poorly-structured relationships among system elements.

4.3.5.1 Profile Requirements List (PRL)

Many NTCIP standards contain a PRL, which serves to trace each user need supported by the NTCIP standard to the requirement(s) that satisfies that need, ensuring that all needs are satisfied.

The PRL is also a list of all mandatory, optional, and conditional requirements that satisfies each user need supported by the NTCIP standard. If a user need is selected by an agency in its specification, all the mandatory requirements that trace to that user need should be supported by the implementation. Optional requirements may also be selected by the agency in its specification, if the agency deems the requirement necessary for its implementation. Conditional requirements may be mandatory or optional, but apply only when the feature or features identified are supported.

A partial PRL example from NTCIP 1203 v02 is shown in Table 4.

Table 4 Example—Protocol Requirements List (PRL)

User Need Section Number	User Need	FR Section Number	Functional Requirement	Conformance	Support / Project Requirement	Additional Project Requirements
		3.5.2.2	Reset the Sign Controller	M	Yes	
2.5.2.3	Control the Sign Face			M	Yes	
2.5.2.3.1	Activate and Display a Message			M	Yes	
		3.5.2.3.1	Activate a Message	M	Yes	
		3.5.2.3.3.5	Retrieve Message	M	Yes	
		3.5.2.3.6	Activate a Message with Status	Drum:M	Yes / NA	
		3.6.5 †	Supplemental Requirements for Message Activation Request	M	Yes	
		3.6.7 †	Supplemental Requirements for Locally Stored Messages	M	Yes	
2.5.2.3.2	Prioritize Messages			M	Yes	
		3.5.2.3.1	Activate a Message	M	Yes	
		3.5.2.3.3.3	Define a Message	VMS:M	Yes / NA	
		3.5.2.3.6	Activate a Message with Status	Drum:M	Yes / NA	
		3.6.5.4 †	Supplemental Requirements for Message Activation Priority	M	Yes	
		3.6.6.4 †	Priority to Maintain a Message	M	Yes	
2.5.2.3.3	Define a Message			VMS:M	Yes / NA	
		3.5.1.2.1.3	Determine Beacon Type	M	Yes	
		3.5.1.2.3.1	Determine Maximum Number of Pages	M	Yes	
		3.5.1.2.3.2	Determine Maximum Message Length	M	Yes	
		3.5.1.2.3.3	Determine Supported Color Schemes	M	Yes	

4.3.5.2 Requirements Traceability Matrix (RTM)

A Requirements Traceability Matrix (RTM) traces system requirements to the solution elements of an NTCIP standard, namely the data concepts such as the dialogs and data dictionary elements. If a requirement is included in an agency specification, the RTM describes data concepts, including the dialogs, that should be implemented to fulfill that requirement and conform to the NTCIP standard.

A partial RTM example from the NTCIP 1203 v02 is shown in Table 5.

Table 5 Example—Requirements Traceability Matrix (RTM)

FR Clause Number	Functional Requirement	Dialog ID	Object Clause Number	Object	Additional Specifications
			5.8.7	dmsIllumBrightnessValues	
			5.8.8	dmsIllumBrightnessValuesError	
3.5.1.6	Configure Current Speed Limit	G.3			
			5.11.1.4	dmsCurrentSpeedLimit	
3.5.1.7	Configure Low Fuel Threshold Value	G.3			
			5.11.3.2	lowFuelThreshold	
3.5.2	Control the DMS				
3.5.2.1	Manage Control Source	G.3			
			5.7.1	dmsControlMode	
3.5.2.2	Reset the Sign Controller	G.3			
			5.7.2	dmsSWReset	
3.5.2.3	Control the Sign Face				
3.5.2.3.1	Activate a Message	4.2.3.1			
			5.7.3	dmsActivateMessage	
			5.7.17	dmsActivateMsgError	
			5.7.24	dmsActivateErrorMsgCode	
			5.7.18	dmsMultiSyntaxError	
			5.7.19	dmsMultiSyntaxErrorPosition	
			5.7.20	dmsMultiOtherErrorDescription	

4.3.6 Conformance Section

Each NTCIP standard has a section defining how a vendor may claim "conformance" to the NTCIP standard. For those standards that use SEP, the vendor should minimally satisfy the mandatory requirements for the user needs as indicated in the PRL. The conformant system may offer additional (optional) features, as long as they are conformant with the requirements of the NTCIP standard and identified standards that may be referenced. That is, the conformant system fulfills the requirements by using all of the data concepts that trace to those requirements in the RTM.

4.3.7 Test Procedures

Some NTCIP standards define detailed, but generic, test procedures to test an implementation of the NTCIP standard.

4.4 AGENCY SPECIFICATIONS—TAILORING NTCIP STANDARDS TO PROJECT NEEDS

An agency specification defines, in a complete, precise, verifiable manner, the requirements, design, behavior, or other characteristics of a system or component, and, often, the procedures for determining whether these provisions have been satisfied [Dorfman, Software Engineering, 2000]. An agency specification references the relevant and required portions of NTCIP standards that apply to that ITS device or system and the media that are used (or planned for use) for the communications infrastructure.

Agency specifications should include the following items:

a) Reference to NTCIP standards and conformance statements. This list should specify version and date.
b) An agency specification compliance statement
c) An agency-completed PICS for each specified NTCIP standard
d) List of agency requirements traceable to the PICS. This list is the agency-tailored list based on NTCIP standards requirements.

e) An RTM for each agency-tailored NTCIP standard that reflects the agency's solution choices from options that may be included in the NTCIP standard
f) List of Dialogs and Object Definitions traceable to the RTM
g) Value ranges for all of the objects to clearly identify such parameters as the size of event logs, the number of messages to be supported, and the number of special functions managed.

Over time, NTCIP standards have enhanced their approach to facilitate the tailoring necessary to develop an agency specification based on SEP.

The PRL presents an entry point for tailoring an NTCIP standard so as to include only those portions that apply to an agency project. Table 4 is an example of a PRL. A PICS is a PRL that an agency has tailored to its project requirements.

The format of the PRL allows an agency to distinguish between optional and mandatory NTCIP standards elements. Optional elements in the NTCIP standard can be either removed from the agency specification or selected to be mandatory for a specific implementation. Making sure that the PICS includes all mandatory requirements of the NTCIP standard (that is, those that are beyond the agency's requirements) ensures the possibility of conformance with the NTCIP standard.

To obtain interoperable ITS devices from an agency specification, the agency should examine applicable NTCIP standards and resolve the following items:

a) Which optional conformance groups for the ITS device should be supported;
b) Which optional objects for the ITS device should be supported;
c) Specify minimum support values for certain capabilities (i.e. the minimum number of plans in a traffic signal controller, the minimum number of phases, size of event logs, number of fonts supported, etc).

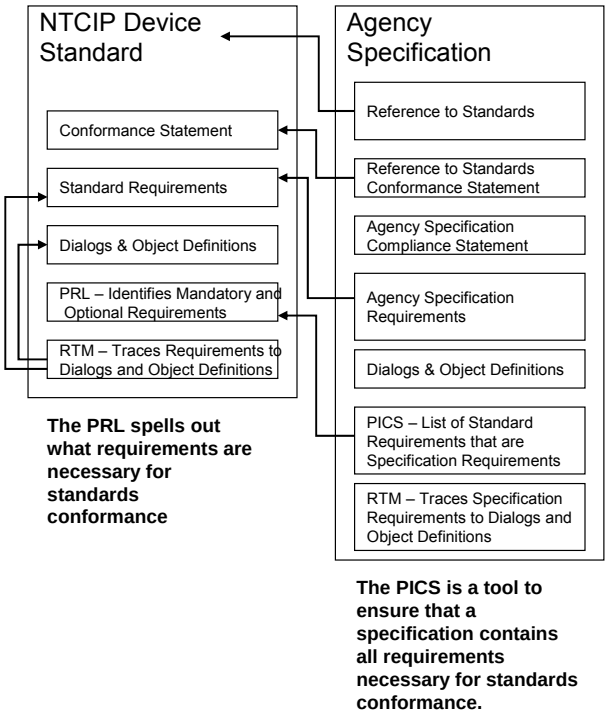

Figure 15 Relationship of an Agency Specification to an NTCIP Standard

Figure 15 shows the relationship between an agency specification and NTCIP standards. Elements of an NTCIP standard shown in Figure 15 include:

a) **Standard Conformance Statement—**Governs how conformance with the standard is fulfilled. The PRL supports conformance with the standard.
b) **Standard Requirements—**Are the standard's functional requirements that the dialogs and object definitions fulfill. Generally, only a subset of the standard's requirements is mandatory, the remainder being optional, or optional under certain conditions.
c) **Dialogs and Object Definitions—**Are the solution elements of the standard and fulfill the standard's requirements.
d) **PRL (Profile Requirements List)—**Lists the mandatory, optional, and conditionally optional requirements.
e) **Standard RTM (Requirements Traceability Matrix)—**Traces a requirement to the standards dialogs and object definitions.

Elements of an agency specification in Figure 15 include:

a) **Reference to Standards—**References the governing ITS standard for the agency specification.
b) **Reference to Standards Conformance Statement—**Addresses how the agency specification fulfills the standard's conformance statement.
c) **Agency Specification Compliance Statement—**Governs how compliance with the agency specification is fulfilled. The PICS may support compliance with the agency specification.
d) **Agency Specification Requirements—**Are project- and agency-specific requirements, the results of tailoring of the standards, mandatory for the project. The PICS may support traceability to mandatory standards requirements and serve to support standards conformance.
e) **Dialog and Object Definitions—**Are project specific solution elements and fulfill agency requirements.
f) **PICS (Profile Implementation Conformance Statement)—**Is a tailored list of standards requirements that applies to the agency specification. The PICS reflects both a standard's optional and mandatory requirements and when completed, reflects only those requirements included in the agency specification.
g) **Agency Specification RTM (Requirements Traceability Matrix).** Traces an agency specification requirement to a standard's dialogs and object definitions.

Section 5
DESIGNING NTCIP

5.1 INTRODUCTION

Section 5 provides an overview of communication bandwidth calculations and some of the issues that should be considered when designing an NTCIP communications system. While the title is *Designing NTCIP*, the material presented focuses on how to design a communications system for use with NTCIP. Section 5 is intended for those system designers, system integrators and manufacturers who are tasked with determining communication system design and performance criteria.

5.2 DESIGN ALTERNATIVES

Once there is a clear understanding of the project requirements, the developer can investigate ways to design a solution that fulfills the information and performance requirements of the system. For example, the developer may be able to acquire COTS software to minimize the effort required to implement features. If this needs to be platform- (computer) or operating system- or database-specific to be compatible with existing site infrastructure or support staff capabilities—that needs to be specified in the requirements for the implementation.

NTCIP has used widely recognized standards whenever possible. For example, NTCIP standards reference TCP, IP, SNMP, and High Level Data Link Control Protocol (HDLC) standards to name just a few. In many cases, private industry has developed COTS tools to aid system developers in implementing these protocols. Being aware of what is available COTS, inherent in an operating system or browser, allows the agency and developer to set a reasonable schedule and budget. For example, some developers may use a COTS implementation of TCP/IP rather than creating their own. Standards for which there are known COTS implementations include:

a) FTP
b) TFTP
c) SNMP
d) TCP/IP and UDP/IP
e) PPP
f) Ethernet

In cases where the NTCIP protocols do not provide the necessary services, a technique referred to as tunneling can often be used. Tunneling describes the encapsulating or embedding of one protocol within another. Tunneling is different from layering in that two (or more) protocols at the same layer are used. Sets of protocols at different layers can also be used. One widely used example is TCP/IP over ATM SONET. It is possible to tunnel PMPP over Ethernet. RS-232 over RS-485 is another example of tunneling.

5.3 COMMUNICATIONS INFRASTRUCTURE FOR CENTER TO FIELD (C2F)

When planning a C2F communications network using NTCIP that involves continuous polling of field devices, for example, a traffic signal system or transit fleet Automatic Vehicle Location (AVL) system, it is important to consider the relationship among the following key variables:

a) Transmission rate (bit rate);
b) Transmission method, for example, full or half duplex, sequential or overlapping;
c) Transmission delay (including any modem/radio set-up/turn-around time);
d) Response delay in the field device (time from receipt of request to sending response);
e) Time between devices or between polling cycles (if needed);
f) Length of message(s) to be sent (dynamic object definitions);

g) Frequency of each type of message (per second, per minute, per day);
h) Number of devices sharing the same line or channel; and
i) Frequency of communication, for example, polling period.

The first seven of these variables determine the total time needed to communicate once with each device. If this time is then treated as fixed (T), the number of devices sharing the same line or channel is (N) and the frequency of communication with each device is (P for polling period, the inverse of frequency), these variables relate by the following equation.

$$P = N * T$$

This is a very simplified explanation of what can be a rather complex design issue that should be addressed early in project planning.

Although STMP is designed for use with communications channels that use a slow transmission rate, as low as 1200 bits per second (bps), STMP is not as bandwidth efficient as most proprietary protocols used in the past. With existing communications infrastructure, it may not be possible to maintain the same polling period with the same number of devices per channel, because proprietary protocols are optimized for each manufacturer's equipment and consist of very few fixed short messages without any flexibility in terms of changing these messages. In contrast, standard protocols are flexibly designed to accommodate all needs and a wide variety of information, as well as messages in a multi-manufacturer environment. However, careful design can usually find a reasonable compromise between the principal variables. Higher available bandwidth or bit rates yield fewer compromises or required trade-offs. If new communications infrastructure can be provided, it should allow for additional channels and/or higher transmission rates.

Such implementation issues are discussed in more detail later in *The NTCIP Guide*.

5.4 RETROFITTING OR MIGRATION OF EXISTING CENTER TO FIELD (C2F) SYSTEMS

It may not be feasible to retrofit or migrate legacy versions of controllers or controller software to make them NTCIP conformant. Constraints such as computing power, memory availability, expense of modification may well preclude such migrations. If such controllers or software cannot be upgraded or replaced, traffic control systems that continue to use older equipment or older software versions are likely to have to continue using the protocols unique to communications with those devices. However, current version controllers and software within the system may be capable of modification to use NTCIP, and agency specifications should indicate conformance with appropriate NTCIP standards elements. If in doubt, the agency may wish to ask the manufacturer whether upgrades to NTCIP are available.

The inability to update older equipment should never stop an agency from replacement or migration strategies that make full use the benefits of NTCIP-based-implementations. For example, a central system whose current field devices cannot be updated might be expanded to run NTCIP protocols on some communications channels while older equipment is maintained on others.

Figure 16 illustrates a model for a three-step migration from legacy systems to NTCIP. As shown, initially the details of a proprietary interface may or may not be known (indicated by a cloud showing proprietary ownership of system details). Next, there is an intermediate state and some time when the operational system consists of a mixture of legacy systems and newer NTCIP hardware. There may be shared use of a common communications channel or not for legacy and NTCIP devices—the figure illustrates these as separate. The central control system may be separate or combined; it may run on the same computer or on separate computers—this is determined by the scope of the project to accomplish these migration steps. Pursuit of a migration strategy towards the use of open standards starts to minimize the use of proprietary communications and begins to maximize the use of NTCIP (as shown by the cloud now being dotted as it starts to fade away). Lastly, at some future point, the migration is completed and NTCIP is fully deployed, having retired all legacy systems (no proprietary cloud at all).

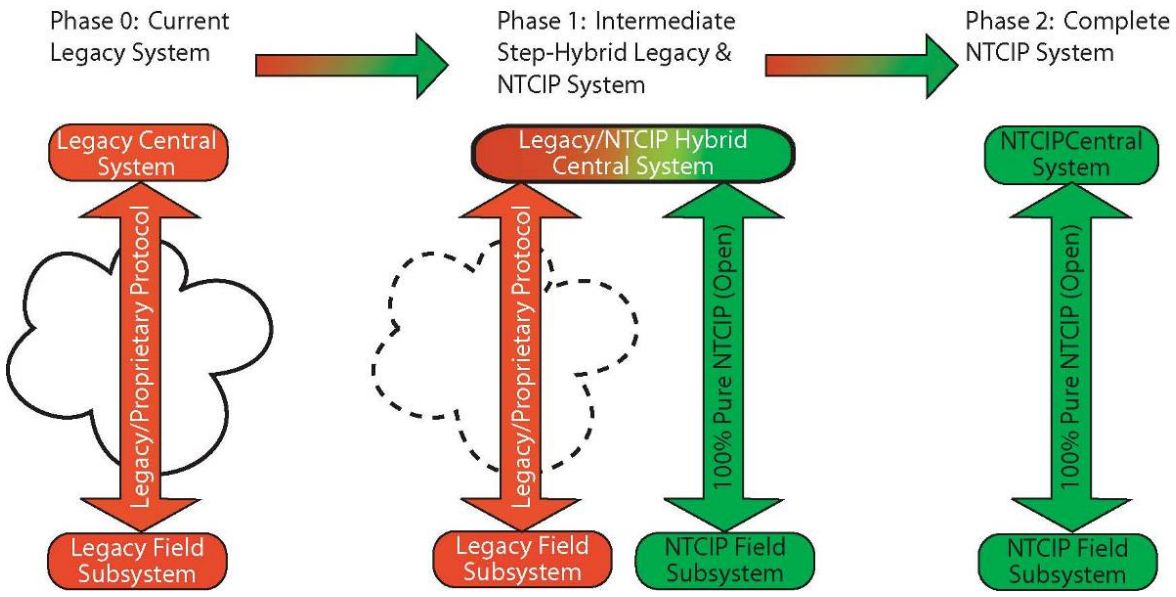

Figure 16 Example—Three-Phase Migration Process

In general, NTCIP and non-NTCIP devices cannot be mixed on the same communications channel. Therefore, all devices sharing a channel should be upgraded simultaneously. A central computer or on-street master that communicates with both NTCIP and non- NTCIP devices needs to use a different communications port for NTCIP devices and for non-NTCIP devices, and needs to support both protocols. Commensurately, the mixture of devices listening on the shared communications channels should recognize and react only to those data elements and commands intended for them individually, and should not produce unpredictable results in response to any other data traffic on the channel.

A specific example: in traditional closed-loop traffic signal systems, the most likely and simplest solution is to limit each field master to one protocol. Only field masters with NTCIP-compatible controllers would be upgraded to support NTCIP. This avoids the need for field masters to simultaneously support two protocols on two separate ports. In closed-loop traffic signal systems, the central computer could communicate with field masters using a different protocol than that used by the field master to communicate with controllers. As with the controllers and the field master, the central computer software needs to be modified to add support for an NTCIP protocol, if NTCIP is to be used for communications with field masters.

An upgrade for an existing system to add support for NTCIP is probably best designed in consultation with the system provider. Each provider is likely to adopt an upgrade or migration strategy that is most efficient for the majority of its customers.

One approach to the introduction of NTCIP in a C2F system is to operate two separate systems—one NTCIP and one non-NTCIP—during a transition period, as shown in Figure 16. Field devices can gradually be switched over as they are replaced or their software is upgraded. This may be the only choice for old systems, where upgrading to NTCIP is not practical. Such a transition would logically be done as part of a general system upgrade.

5.5 LEGACY ISSUES AND SYSTEMS MIGRATION

Migration from proprietary legacy systems to those that are NTCIP standards based can follow many paths. While this is primarily associated with initial installation of an NTCIP standards based system,

many concepts discussed here should also be considered during the life-cycle of an NTCIP standards based system.

Any migration should consider both hardware and software. While NTCIP standards are primarily focused on software, hardware can also be a major consideration. Many older existing hardware platforms may not be powerful enough to support the demands of NTCIP protocols. An agency should take a close look at hardware and work with suppliers to determine whether existing hardware can meet those demands. If the existing hardware platform is more than ten years old, serious consideration should be given to its replacement, with changes to, or most likely, a complete replacement of the operating software.

Given that migration impacts both system software and hardware, and that future changes/improvements in technology are a fact of life, consideration should be given to separating hardware and software decisions. In many older legacy systems, hardware and software have been provided as an integral package. Just like in the desktop PC world, users typically go through several software upgrades before the hardware is finally replaced. The separation of software from hardware also gives the user flexibility and a larger choice of potential applications software. Many manufacturers and system developers are beginning to structure systems in a manner that allows such separation.

One caveat, however, is that one should not overlook the condition and capability of the communications infrastructure that connects these systems. Performing a bandwidth analysis as described in *The NTCIP Guide* helps to determine your migration strategy.

Many paths can be taken in migrating legacy systems to current NTCIP standards.

The two most likely scenarios are:

a) ***Replace the entire system at once*—**For smaller agencies with only a handful of signals, this approach may be possible. If the agency previously installed field devices that accept new software, users could easily load the new software during a single off-peak period. However, for many agencies this strategy may prove successful for smaller subsystems, but not an entire system.
b) ***Migrate parts of the system*—**Larger systems likely need to be broken down into manageable chunks. The size and shape of these chunks needs to be carefully selected. Some constraints on selection are:
    1) **Communications channels available—**Bandwidth analysis identifies how many communications channels are needed. Devices are then assigned to available channels. All devices on the same channel should talk the same language (i.e. use the same communications protocol). The result of this approach is that two separate systems should be operated, often independently, until the migration is complete. One system operates using the new communications protocols and the other operates using the legacy communications protocols.
    2) **Communications channel capacity—**This constraint is closely related to the previous one. The new communications protocols may not allow as many devices on a channel as legacy systems. In this case, the communications infrastructure should be altered to accommodate new channel loading.
    3) **Staffing—**Staff time is be needed for fieldwork, as well as data entry, for the new system. In many agencies, limited available staff time suggests that upgrades are deferred to emergency maintenance requirements. Contracting strategies also need to consider the most expedient approach to performing these activities.
    4) **Operational considerations—**An agency may want to consider selecting less critical locations for first deployments. As such, any issues not uncovered during system testing can be corrected with minimal impact. For traffic signal systems, consideration should be given to selecting coordinated groups of signals.

Often overlooked items that can have a big impact on system migration include:

a) **Data Migration—**System data from the old system may not be easily translated to the new system. Time and resources are needed to input and configure data for the new system to operate efficiently. This also includes any graphics.
b) **Two Systems?** During this migration period, **two** systems are in operation. If you can't shut down and replace an entire system at once, a plan is needed for the care and feeding of the legacy system. Do not forget that system migration may take several years, depending on the size of the system and resource availability. At the end of the system migration is a standards based system that is likely to require lower expenditures and be less difficult for the next system migration.

Even if a system continues to use a proprietary protocol, new controllers and masters, or new software packages should include the appropriate NTCIP protocol stack as an option, to facilitate migration to a NTCIP standards-based system in future. However, few providers have implemented support for both existing protocols and NTCIP in the same software package, while most require a change of software to switch from one protocol to the other. Regardless of how it is done, the agency should ensure that an appropriate NTCIP protocol stack is available for future use, to maximize the useful service life of new equipment and enable migration to NTCIP in future while minimizing hardware and software upgrades.

Section 6
IMPLEMENTING NTCIP

6.1 INTRODUCTION

Section 6 describes various issues related to implementing NTCIP in a device or central system. The primary audiences for Section 6 are system/device developers and system implementers. Section 1 through Section 4 provide background, and concepts in Section 3 and Section 4, related to system design and specification, should be useful.

Section 6 is germane to all of NTCIP standards. As such, concepts are presented at a high level. However, one detailed example is given at the end of Section 6to explain concepts more fully.

6.2 NTCIP DATABASE

Transportation devices need to store and communicate constants, parameters and collected data. Examples include minimum cycle time for a traffic signal, text for a dynamic message sign, vehicles per hour for a count station, and current wind speed from a weather station.

The ***Management Information Base*** (or ***MIB***) is the database where data element (or object) values are stored.

NTCIP is intended for many types of transportation devices, each with different database requirements. NTCIP relies on the SNMP approach to database management. SNMP uses an industry-standard GET/SET paradigm to read and write data into a database. Data elements in the database are called “managed objects” or just “objects” for short. The database is a group of related data elements and is called a “management information base” or MIB. The entity that manages the MIB within a device is called an agent. The concept is that a management application sends messages to the agent to fetch or modify the values of data elements stored within the MIB. When MIB values change, the transportation device responds as defined in its programming. The agent works at the application layer in the protocol, but is itself not the application. The agent does not care whether the MIB represents a traffic controller or dynamic message sign or any other device.

STMP works in the same way as SNMP and uses the same data elements. STMP simply provides a more bandwidth-efficient, but more processor-intensive, solution to the same problem.

A data element is a type of data. Within an agent, one or more instances of the data element may exist. Different instances are identified by their index. A data element instance should be transported as a whole. As an example, a data element may define the current time, say 12:15:30. Since this data element is defined as hours, minutes and seconds, it cannot be used to transport just seconds. Alternatively, time could be defined as three separate data elements, one each for hours, minutes and seconds; in this case, a request could deal with any one of the three data elements or simultaneously access all three. A third alternative would be to define both of these representations of time (for a total of four data elements). It may be convenient to have access to different forms of the same data in the same MIB, but it is not very efficient, since extra data elements require more computer memory. Thus, when designing data elements, the designer should consider the trade-offs of each approach. Additionally, one needs to consider that NTCIP only addresses the transport of this data, not its presentation to the user; that is, while time could be sent in three data elements, it could be presented to the user in the 12:15:30 format.

To standardize some of the commonly needed data elements, NTCIP defines a global data element’s MIB module. The global MIB module contains definitions of commonly needed data elements that are basic to NTCIP, such as the name and versions of the MIBs supported by an agent, the representation of time, logging of events, or the definition of a scheduler.

A MIB and its data elements are defined using ASN.1. “Abstract syntax” means that the manner used to define the data is independent of the procedure for encoding the data into binary form. The MIB is a text document that can be read by a human and compiled by computer. A management application should have a copy of the MIB employed by the managed agent. The MIB is transferred to a management application via a text file on a floppy disk. This is usually performed when the device is first installed. Other methods could be used for MIB transfer; however, it is possible that the management application does not have a complete copy of the MIB employed by the managed agent. This scenario would limit the possible data exchange between management application and the agent to the data elements supported within the management application, which is sometimes desired when the field device includes many features not (currently) used by the management application, and when the goal is to avoid software modifications to the management application.

6.2.1 Example—Device Includes a Clock

Consider the example in Section 6.2 where the device includes a clock. The management application needs to set the clock periodically to ensure that all devices are synchronized. The MIB defined in Global Object Definitions (NTCIP 1201:1996, including Amendment 1), includes a data element called “globalTime.” Below is the ASN.1 definition of this data element.

```
globalTime OBJECT-TYPE
SYNTAX Counter
ACCESS read-write
STATUS mandatory
DESCRIPTION “The current time in seconds since the epoch 00:00:00
(midnight) January 1, 1970 UTC (a.k.a. Zulu).”
::={globalTimeManagement 1}
```

The first line is the name of the data element, followed by the invocation of the **OBJECT-TYPE** macro.

The **SYNTAX** line defines the variable type. In this example, the type is a Counter. SNMP standards define a Counter to be an INTEGER with a range of 0 to 4294967295 that performs a counting operation. Upon reaching the maximum value, the Counter rolls over and starts at zero.

The **ACCESS** line defines the access permission for the object. Values include read only, read-write, or none.

The **STATUS** line is provided to simplify conformance statements. At the end of each data element standard, conformance groups are defined. For a device to claim conformance to a conformance group, it should support all “mandatory” data elements within that group. It may also support “optional” data elements within the group. For a device to claim conformance to a standard, it should support all “mandatory” conformance groups defined by the standard, and may optionally support “optional” conformance groups.

NOTE—NTCIP 1201:2005 (v02) revises this scheme for global data elements (only) in that the definition of conformance groups and contained mandatory data elements are left up to NTCIP device-specific standards, such as NTCIP 1202:2005 (actuated traffic signal controllers).

The **DESCRIPTION** line provides a human readable description of the data element. This data element represents the current time as represented in seconds since midnight January 1, 1970 in Greenwich, England. Many data elements specify some sort of functionality; in this case, this data element requires the device to increment the value of this data element by one at a rate of exactly once a second.

The last line indicates the location of this data element on the ISO Global Naming tree. In this case, the data element is the first node under the globalTimeManagement node. Earlier within NTCIP 1201:1996, the globalTimeManagement node was defined to be underneath the global node. This referencing continues within NTCIP 1201:1996 all the way back to the ISO root node.

Table 6 lists some common ASN.1 terms used for data element definitions. Since this is an incomplete list, refer to ISO 8824 and various NTCIP standards for more information. NTCIP also defines additional limitations to ASN.1. These limitations are defined in a section called the NEMA Structure and Identification of Management Information (NEMA_SMI). The NEMA_SMI is not a MIB but is added to all NTCIP MIBs.

The ASN.1 macro language is very powerful, even with the restrictions imposed by SNMP. A MIB may define a new syntax by combining basic (primitive) data types. Likewise, a MIB can define a one or multi-dimensional table using the SEQUENCE operator. Default values and ranges for data elements can also be defined in the MIB. The robustness of the ASN.1 language allows modeling of virtually any database likely to be encountered in ITS field devices.

MIB data elements are related by device type, for example, a traffic signal controller MIB or a message sign MIB. These device MIBs are called modules. NTCIP has MIB modules for actuated signal controllers (ASC), dynamic message signs (DMS), CCTV control, and environmental sensor stations (ESS), to name a few. It is desirable to use a standard MIB wherever possible, but as new device features require additional data elements, new versions of a MIB can be created. NTCIP also supports proprietary and experimental MIBs. Experimental MIBs are kept under a separate node and users know that the MIB is subject to change. Proprietary MIBs can exist under nodes registered to either private firms or public agencies.

Table 6 Some Common ASN.1 or NTCIP Terms for Data Element Definitions

ASN.1 Tag	Description	Some Options
OBJECT-TYPE	Alphanumeric string that names the data element	
SYNTAX	Object data type	INTEGER (-128..127) INTEGER (0..255) INTEGER (0..4294967295) OCTET STRING—a string of bytes, such as ASCII text
ACCESS	Determines read/write capabilities	read-only read-write not-accessible
STATUS	Determines whether this data element is required.	Mandatory—support required if a conformance group containing this data element is supported. Optional—not mandatory. Deprecated—use of the data element is discouraged, however, management stations should support the data element as it may be encountered in a deployed device; future releases of the standard may mark the data element obsolete. Obsolete—the data element has been deleted or replaced; management stations and agents are not required to implement it.
DESCRIPTION	Explanation of what the data element represents and how to interpret it	Anything you want to write to unambiguously describe the purpose of the data element including any limitations, the units (i.e., seconds or tenths of seconds)

Data elements in the MIB are arranged in a tree structure, and data elements are named by the path along the branches of the tree to the data element. The path starts at the trunk of the tree, and a node identifier is added at each branch until the data element is reached. The node identifiers are unsigned integers and are frequently documented in text as dot-separated, for example, “1.3.6.1.4.1.1206”. The

tree structure is defined by ISO and CCITT. An entire MIB module hangs off this global name tree. Table 7 is a diagram showing the tree structure from its root to the NEMA node. All standardized NTCIP MIB modules are attached to the NEMA node.

All data elements in the NEMA NTCIP MIB modules start with 1.3.6.1.4.1.1206 or (*iso.org.dod.internet.private.enterprises.nema*). NEMA has further divided the 1206 node into four subgroups: mgmt(1), experimental(2), private(3) and devices(4).

Everything below the transportation node constitutes the Transportation MIB (TMIB). Within the TMIB there are protocol, devices and tcip data element groups. The devices group has subtrees for each of the supported devices: asc, ramp meter, dms, cctv, ess, global, cctv, cctvSwitch, dcm, ssm, scp, networkCamera, and elms. Branches are added as new devices are included in the NTCIP family of standards.

6.2.2 Example—globalTime Data Element

Consider the globalTime data element in the example in Section at 6.2.1. The object identifier for globalTime is defined as globalTimeManagement 1. GlobalTimeManagement is previously defined as global 3. Finally, the header of the MIB contains:

```
nema OBJECT IDENTIFIER::= {iso(1) org(3) dod(6) internet(1)
private(4) enterprises(1) 1206}
transportation OBJECT IDENTIFIER::= {nema 4}
devices OBJECT IDENTIFIER::= {transportation 2}
global OBJECT IDENTIFIER::= {devices 6}
```

Thus, the Object Identifier (OID) for global is 1.3.6.1.4.1.1206.4.2.6 or (iso.org.dod.internet.private.enterprises.nema.transportation.devices 6). The OID for globalTimeManagement is 1.3.6.1.4.1.1206.4.2.6.3 and the OID for globalTime is 1.3.6.1.4.1.1206.4.2.6.3.1. As mentioned, each data element is instantiated by the agent. In the case of globalTime, there is only one instance; that is, the data element is not contained in a table, and thus, its instance number is zero (0). Thus, the full OID for the instance of globalTime within our DMS would be 1.3.6.1.4.1.1206.4.2.6.3.1.0.

When using STMP, this 13-node identifier (OID) can be pre-configured to minimize bandwidth consumption on frequently transmitted messages.

The databases for NTCIP devices are defined using a subset of ASN.1, which provides a standard method for data element definition, organization and identification. This discussion examined how to define a data element and identify the path to the data element using this standard. Next, discussion examines how the object identifier or path and the value of the data element are encoded into binary data for transmission.

6.2.3 Example—Encoding a Data Element Value

To transmit a data element, first, select the protocols to be used for transmission. In this example, we use an application layer of SNMP.

SNMP uses a standard set of rules for encoding called, Basic Encoding Rules (BER) (ISO 8825-1). Basic Encoding Rules (BER) define that each data element is encoded in variable binding list. A variable binding list consists of a separate encoding for the OID of the data element and the encoding for the value of the data element. The encoding for both types (OID and value) follows the same setup: data type, length and value.

6.2.3.1 Example—Encoding Object Identifier and Value

Using the data element from 6.2.1, we need to encode the object identifier and value:

To encode object identifier:

Type OBJECT IDENTIFIER
Length The number of octets (i.e., bytes) used to encode the value of the identifier
Value 1.3.6.1.4.1.1206.4.2.6.3.1.0

To encode value:

Type Counter
Length The number of octets used to encode the value of the identifier
Value 915148800 (i.e., 00:00:00, 1 January 1999 UTC)

SNMP defines the values for allowed types. The hexadecimal values for the most common types are:

a) INTEGER 0x02
b) OCTET STRING 0x04
c) NULL (Placeholder) 0x05
d) OBJECT IDENTIFIER 0x06
e) SEQUENCE 0x30
f) Counter 0x41
g) Gauge 0x4x
h) Opaque 0x4x

Per Basic Encoding Rules (BER), the first two components of an OBJECT IDENTIFIER are combined using the formula (40X)+Y to form the first subidentifier. Each subsequent component forms the next subidentifier. Each subidentifier is encoded as a non-negative integer using as few seven bit blocks as possible. The blocks are packed into octets, with the first bit of each octet set to a 1 except for the last octet of each subidentifier. Thus, the object identifier (OID), {1.3.6.1.4.1.1206.4.2.6.3.1.0} is encoded as shown in Table 7:

Table 7 Data Element Component, Subidentifier and Octet Sequence Hex

Component	Subidentifier	Octet Sequence Hex
1.3 (iso org)	43	[2B]
6 (dod)	6	[06]
1 (internet)	1	[01]
4 (private)	4	[04]
1 (enterprises)	1	[01]
1206 (nema)	1206	10010110110 bin [89][36]
4 (transportation)	4	[04]
2 (devices)	2	[02]
6 (global)	6	[06]
3(globalTimeManagement)	3	[03]
1 (globalTime)	1	[01]
0 (instance 0)	0	[00]

NOTE—The notation [xx] represents a number in hexadecimal format.

Adding the OBJECT IDENTIFIER type of [06] and a length of [0D] or (13 decimal) yields the byte sequence:

[06][0D][2B][06][01][04][01][89][36][04][02][06][03][01][00]

Next, encode the data value, which is 915148800. Basic Encoding Rules (BER) encode Counters in the same way BER encodes INTEGERs, with a two's complement representation using the minimum number of octets.

NOTE—This means the value 255 would be encoded in two bytes, [00][FF], so that the high order bit is set to zero indicating a positive number.

Thus, the byte sequence is:

[36][8C][10][00]

A Counter has a type code of [41].

Thus, our data element value with a type of [41] and a length of [04] would become:

[41][04][36][8C][10][00]

The combined byte SEQUENCE (Type [30]) for the OID and value has a length of 21 ([15]).

Thus, the entire encoding for this data element is:

[30][15][06][0D][2B][06][01][04][01][89][36][04][02][06][03][01][00][41][04][36][8C][10][00]

If STMP was used, a dynamic object could be configured to include this data element. In this case, only the object identifier would not be transmitted (because each end of the link would already be aware of what data to expect next). Further, the data value would be encoded using OER as defined in NTCIP 1102:2004, which is more efficient than Basic Encoding Rules (BER). Thus, an STMP dynamic object would encode the preceding in 4 bytes instead of the 23 bytes shown, that is, 21 plus type and length of the sequence. Some occasional messages should still be sent via SNMP, including the message used to tell the field device of a change in a dynamic message definition.

The example in Section 6.2.3.1 examined only two data types that can be encoded with BER. BER contains encoding rules for all ASN.1 types.

6.2.4 Example—Encoding the SNMP Data Packet

Thus far, a data element was created using ASN.1. Then, it was placed in a tree structure, given a real value, and finally the data element and its value were encoded using BER. Now, this information should be given in a context. For example, is this a request to set the time, or is it a response to a get request? The context is given by the surrounding structure of the data packet, as defined by SNMP rules.

SNMP uses a get/set and an NTCIP defined trap paradigm. Table 8 lists SNMP message types, purposes and originators.

Table 8 SNMP Message Type, Purpose, and Originator

Message Type	Purpose	Originator
Get Request	Contains a list of data elements, the agent is to return the values	Management Application
Get Next Request	Contains a list of data elements, the agent is to return the values of the next sequential data element from those indicated.	Management Application
Set Request	Contains a list of data elements and values, the agent is to set the values in its MIB per this message	Management Application
'Get' Response	Agent response to either a Get or a Set request	Agent Application
Trap	An Agent initiated transmission to indicate that a defined event has occurred.	Agent Application

The SNMP Message structure is given by the following ASN.1 structure:

```
Message::= SEQUENCE {
    version INTEGER { version-1(0)},
    communityOCTET STRING,
    data CHOICE {
    get-request GetRequest-PDU (with a data type value of 0xA0),
    get-next-request GetNextRequest-PDU (data type value = 0xA1),
    get-response GetResponse-PDU (data type value = 0xA2),
    set-request SetRequest-PDU (data type value = 0xA3
    }
}
```

All PDU structures have essentially the same structure, as follows, with a different Tag.

```
GetRequest-PDU::= [0] IMPLICIT SEQUENCE {
    request-id RequestID,
    error-status ErrorStatus,
    error-index ErrorIndex,
    variable-bindings SEQUENCE OF SEQUENCE {
        name OBJECT IDENTIFIER,
        value ObjectSyntax -- i.e., the SYNTAX of the selected data element
    }
}
```

This 23 byte data stream forms the following sub-structure in the preceding structure:

```
SEQUENCE {
    name OBJECT IDENTIFIER,
    value ObjectSyntax -- i.e., the SYNTAX of the selected data element
}
```

Thus, the rest of the components of the data packet now need to be added. In this case, it is assumed that the data packet is a get response for a get request of both the globalTime data element as well as the globalDaylightSavings data element, as shown in Table 9.

Table 9 Example—Get Response

Field	Byte Stream
SEQUENCE - Type and Length (Value is below)	[30][45]
version - INTEGER of 1 byte, value 0	[02][01][00]
community - OCTET STRING of 6 bytes ("Public")	[04][06][50][75][62][6C][69][63]
data - Type and Length (Value below)	[A2] [38]
request-id - In this case we use 1	[02][01][01]
error-status	[02][01][00]
error-index	[02][01][00]
variable-bindings SEQUENCE OF	[30][27]
SEQUENCE	[30][15]
name	[06][0D][2B][06][01][04][01] [89][36][04][02][06][03][01][00]
value	[41][04][36][8C][10][00]
SEQUENCE	[30][12]
name	[06][0D][2B][06][01][04][01] [89][36][04][02][06][03][02][00]
value	[02][01][03]

The Get Request would be nearly identical. The data type would be [A0] rather than [A2] and the value fields would be NULL, that is, Type 5 and zero length, [05][00].

6.3 MIB EXTENSIONS

AASHTO, ITE and NEMA have defined an open and expandable suite of protocols. NTCIP permits completely open database definitions without precluding completely proprietary (closed) ones. NTCIP serves both open and closed databases on the same network. Users are encouraged to review the existing MIB data element definitions before attempting to add new ones to avoid allowing the definition of data elements whose functions are already defined in standard NTCIP data elements.

The creation of a new MIB module can be quite easy. This is especially true if the device to be supported already has a list of defined requirements and database. To start, define the necessary data elements for the device using ASN.1 and attempt to organize them in a subtree. Obtain from NEMA a root node for the subtree under the NEMA private or experimental node. Seek comments from NEMA, manufacturers, and users of similar devices. In the early stages of NTCIP development, it may be sufficient to list the needed data elements by name and proposed data types (submit them to the Joint Committee of the NTCIP for further development). Above all, try to use existing data element definitions as much as possible to further compatibility between devices.

6.4 PROTOCOL-RELATED ISSUES

Implementing an NTCIP standard requires careful examination of a large amount of text within the NTCIP standards family. A number of problems discovered during the integration work related to invalid interpretation of the NTCIP standards, especially relating to those sections that reference other NTCIP standards without providing significant detail. To minimize the number of these conflicts in future, a discussion of some of these issues follows.

6.4.1 Bit and Byte Order

Bit and byte order in a computer are not necessarily the same as the bit and byte order on the transmission medium. The transmission order varies in accordance with the guidelines of international standards. Implementations should ensure that the representation of the most and least significant bits and bytes in the computer accurately reflects what is sent and received on the transmission media.

6.4.2 Extended Addresses

There has been some confusion about how large a supported HDLC address should be. For both of the PMPP and PPP Subnetwork Profiles, NTCIP devices are required to fully support one-byte addresses and to accept incoming frames with two-byte addresses to the device address. Production of frames with two-byte addresses is optional, as is support for configuring the device to an address greater than 63.

All addresses are odd; if the first byte of an address is even, then the address is multi-byte.

6.4.3 Maximum Duration Between Successive Bytes

Many existing field devices use proprietary protocols that expect incoming messages to be a series of bytes with minimal delay between the bytes. These messages time-out as an ‘end of message’ when a byte is not received in 15 ms for 1,200 bps communications. This means that consecutive messages should be separated by about 30 ms at 1,200 bps. With simultaneous use of ‘soft carrier turnoff ’, this gap should be increased by the ‘soft carrier turnoff time.’ At higher transmission rates, these times are proportionally reduced.

Because of the desire for full-duplex communications, devices conforming to the PMPP Profile should be designed to support much greater durations between successive bytes as suggested in NTCIP standards. In short, the only distinguishing limit of a message is the 0x7E flag.

6.4.4 Response Time

The public domain code that is available was developed using DOS and Windows; in both cases the serial port interrupt is only checked every 50 ms. Thus, these systems do not perform quite as well as desired; however, this was not an issue for demonstration.

An ITS device should use a serial port driver to achieve the desired performance. The desired performance is system-dependent and is stored in the T2 [the maximum time that a device is allowed to take before starting to send a response] data element. In a multi-drop environment, it is desirable to minimize the duration of the T2 timer. A T2 value of 40 ms or less is desirable, but not always achievable. According to NTCIP standards, the secondary is not allowed to respond after its T2 timer expires.

6.4.5 Control Byte

PMPP includes support for three control byte values. A primary can transmit an unnumbered poll (0x33), an unnumbered information command with the poll bit set (0x13), or an unnumbered information command without the poll bit set (0x03). The secondary should respond to every frame received with the poll bit set, that is, either 0x33 or 0x13, and the response frame from the secondary should be an unnumbered information response with the final bit set (0x13). The secondary may not transmit at any other time.

NOTE—These values are presented according to Internet encoding rules.

PPP devices are peer devices, that is, either the management station or the field device may communicate at any time, because they are the only devices on the 4-wire link at any given time. In this environment, there is no need to give permission to the other device, nor is there a need to force a data link response. As such, the PPP Profile only uses the unnumbered information command without the poll bit set for both ends of the link. This byte may be omitted, if such operation has previously been negotiated (see the PPP Subnetwork Profile and the PPP Request for Comment (RFC)).

6.4.6 Frame Handling

In PMPP systems, the primary may constantly poll each device to determine whether it has any information to report. If the primary station has information to transmit with this poll, for example, a request, it encapsulates this data in an unnumbered information command with the poll bit set. If there is not any information to send, it sends an “empty” frame of six bytes called an unnumbered poll frame. If the primary station has information to send, but does not want to give the opportunity for the receiving device to respond, for example, a broadcast, it sends the data in an unnumbered information frame without the poll bit set.

When a secondary station receives an unnumbered information frame with the poll bit set or an unnumbered poll, it responds with an unnumbered information frame with the final bit set. If the secondary has any data to send with this frame, it is encapsulated within the frame. If the secondary does not have any data to send, it should send an empty frame. When a secondary station receives an information command without the poll bit set, for example a broadcast message, it does not respond.

NOTE—These rules only deal with the data link layer. For example, a central system may send a broadcast message without a poll at the data link layer, while requesting a response at the application layer. The remote device would prepare a response at the application layer that would then be stored in the device’s data link layer. This response could only be sent out after the device has received a frame with the poll bit set.

For example, if dynamic object 1 had been defined to be the time data element, the primary (central) station could send the following byte stream:

```
Flag Addr Ctrl IPI STMP ----SET Time---- CRC Flag
    7E FF 03 C1 91 31 E6 E7 00 XX XX 7E
```

The address indicates that everyone receives it and the control byte prevents anyone from responding on the channel; however, the STMP byte is a SET with response. Thus, all of the devices generate responses at the application layer and are sent to the data link layer to be transmitted. Then, the data link layer waits until permission is granted for the device to speak, for example, an unnumbered poll frame. In this way, a central system can broadcast the time and then go back and ensure that all the devices received the message.

If the secondary has a pending response waiting at the data link layer, it should send the response immediately upon receiving a frame with the poll bit set. If the incoming frame contains information, the information should be processed. This might entail producing a new response sent to the data link layer, where it resides until the next poll is received.

A device is responsible for storing one response frame at the data link layer. If a second response is generated before the first is sent, the first response should be overwritten.

6.4.7 Cyclical Redundancy Check (CRC) Algorithm

Both PMPP and PPP use the same cyclical redundancy check (CRC) algorithm. When a device receives an invalid frame, it should just discard the frame. Invalid frames include those with invalid CRCs, and invalid initial protocol identifiers. Devices should not provide any response to invalid frames. A discussion and example code for the CRC algorithm can be found in Annex F.

6.4.8 Length Values for Variable Message Fields

The exact meaning of this field, that is, which bytes are included in the count, has led to some confusion. The count value does not include the count byte in the count, that is, the count starts the byte after the count byte.

6.5 SYSTEMS INTEGRATION ISSUES

In addition to those issues raised about the interpretation of NTCIP standards, there were also issues over how systems should be designed and what should be required in agency specifications to achieve the goal of systems interoperability. Section 6.5 provides some guidance on how to approach these issues.

6.5.1 Carriers

It is very important that secondary stations on multi-drop lines turn off their modem carriers when not sending data. After responding to a poll, the carrier should be removed from the line so that other stations may respond.

6.5.2 Number of Devices on a Channel

The input impedance of the transmission output circuit in field device modems limits the maximum number of field devices that can reliably be supported on a single modem channel. Each of these outputs is a load on the field device transmission line. A practical upper limit is somewhere around 15 field devices for the current technology 202 modems. Advanced 202 modems can be used that isolate the individual transmission circuits unless the modem is actively transmitting. In a system with only four to seven field devices per channel, this consideration can be ignored without detrimental effects, since communications timing is usually the determining factor in the maximum number of devices per channel.

Section 7
NTCIP TESTING

The Joint Committee on the NTCIP has taken steps toward facilitating effective and efficient testing for real-world implementations of NTCIP C2F and C2C standards. The Joint Committee on the NTCIP believes that the highest priority item is the development of standards-based test cases, which can be reused to construct testing procedures and plans. The lower level test cases help in the quality design of the NTCIP standard itself. The test procedures and plans assist in the ITS device and systems testing done by the manufacturers and users of ITS. The Joint Committee on the NTCIP affirmed the following recommendations for how test procedure evolution and packaging:

a) **Testing and Conformity Assessment Working Group (TCA)—**The TCA was established and tasked with responsibility for creating a framework within which test cases can be developed by individual functional area working groups. TCA was also tasked with the responsibility of investigating suitable testing tools for use in conjunction with the implementation of NTCIP test cases, and resultant testing procedures.

b) **NTCIP 8007 v01—**A separate document, NTCIP 8007 v01 defines the structure, content, and format of test cases. NTCIP 8007 v01 is not for end users. NTCIP 8007 v01 is a standard for NTCIP standards developers, and defines the rules and guidelines to be used by NTCIP working groups when they produce NTCIP test documentation. NTCIP 8007 v01 is intended to promote a consistent look and feel for NTCIP test documentation throughout NTCIP standards. These tests cases are for NTCIP standards testing only, and not for agency acceptance of devices.

c) **NTCIP 9012 v01—**A separate document, NTCIP 9012 v01 provides guidance for agencies to define their NTCIP device testing process and program. Similarly, NTCIP 9012 v01 can help an agency understand other NTCIP testing should the agency decide to rely on vendor testing, independent lab testing, or testing done previously by other agencies.

7.1 NTCIP TESTING OVERVIEW

Delivery and acceptance testing is an important aspect of SEP. The agency should be aware of any time constraints that might be required for development, testing, and refinement of new software that comes as a result of implementing a standard. Before any testing begins, there should be a clear statement and understanding of requirements that should be fulfilled and the minimum acceptable performance levels. All testing should then be based upon, and derived *only* from, these agreed upon requirements. Each requirement has a test, and each test traces to a requirement. In the case of NTCIP, test procedures should be aligned with the accepted NTCIP requirements and designed to exercise a range of valid values within all objects that are supported by the device under test. Test procedures should also ensure that the device under test tolerates invalid values or improperly formatted messages and functions properly under those circumstances.

Testing aspects of SEP are illustrated in Figure 17.

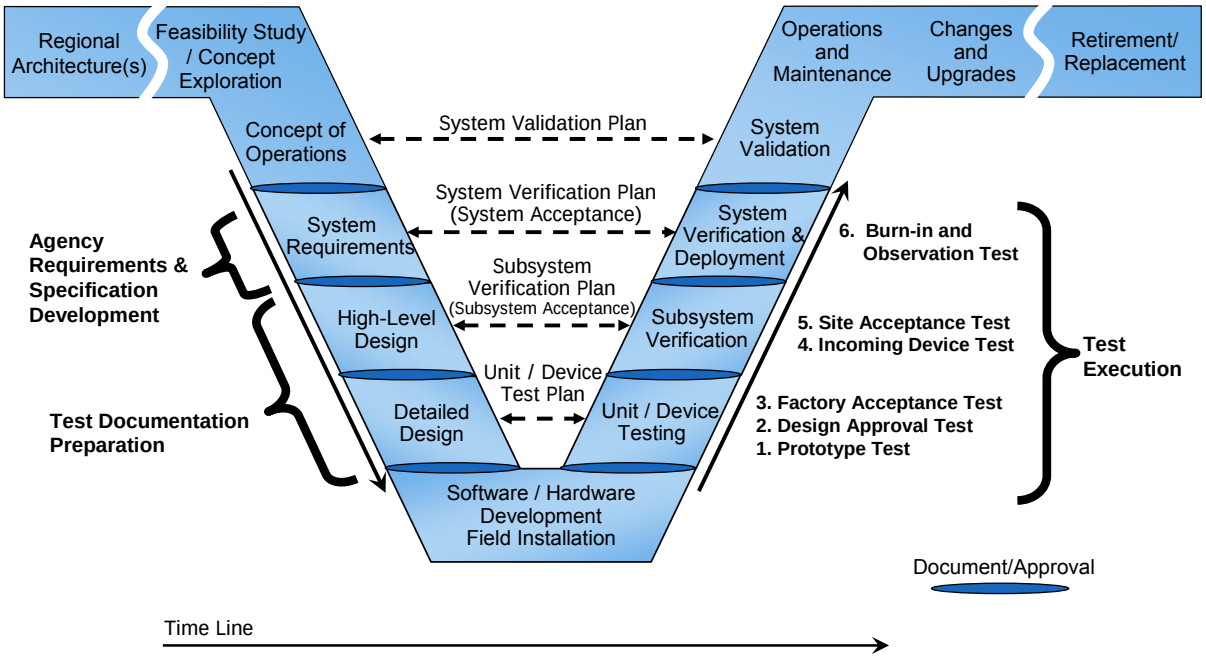

Figure 17 Testing Aspects of the Systems Engineering Process (SEP)

7.2 TESTING PHASES

An agency's approach to testing for any deployment should consider the maturity of the ITS device, the number of units being acquired and installed, the agency's ability to test, available expertise, and the relative significance of each agency specification requirement, among other factors.

The approach to testing should be tempered with the number of units to be acquired, unique agency or project implementation requirements, and the history of the ITS device and NTCIP standards involved. Device testing is generally divided into the following phases:

a) Prototype test and inspection
b) Design approval test and inspection
c) Factory acceptance test
d) Incoming device test
e) Site acceptance test
f) Burn-in and observation test

Table 10 summarizes the relationship of these testing phases with the testing phases defined by SEP.

Table 10 NTCIP Testing Phases

Test Phase	Purpose	Number of Units	Test Location
Prototype Test and Inspection	Verify the electrical and mechanical design.	One prototype.	Test Laboratory
Design Approval Test and Inspection	Verify the final design.	Pre-production or a small percentage of the production units	Laboratory
Factory Acceptance Test	Verify production units are identical to the final design and production quality	A percentage of the production unit.	Production factory.
Incoming Device Test	Inspect for damage due to shipping and handling.	All delivered units, including spares	Agency.
Site Acceptance Test	Full functionality of the entire system.	All installed units.	Final location for operation.
Burn-in and Observation Test	Monitor proper operation of the installed unit.	All installed units.	Final location for operation.

7.3 TEST DOCUMENTATION

Test documentation is a key element of a testing program. Test documentation includes test plans, test cases and test procedures. Test documentation may be developed by the vendor, the agency, a test laboratory, a consultant, or perhaps it is based on test documentation used by another agency as part of their qualified products program. Testing is conducted by a combination of vendor, agency, and possibly an independent laboratory to verify that an ITS device complies with the agency specification.

NTCIP 9012 v01 discusses the following test documentation:

a) **Test Plan**—Describes the scope, approach, resources, and schedule of testing activities
b) **Test Design—**References the test cases applicable to a particular test plan associated with the test design. The test design also references the features (requirements) to be tested.
c) **Test Cases and Procedures—**Describes the inputs, outputs, expected results, and procedures used to verify one or more requirements.
d) **Test Reports—**Document the test plan execution.

Developing agency test documentation can take a significant amount of time and require coordination of many parties. It is recommended that test plan development begin after system interface requirements have been completed and approved. Test Design and development or Test Cases can begin after agency specification requirements have been approved and signed-off. Test Plan execution occurs throughout implementation, according to the suggested testing phases summarized in Figure 17. Test reports document test plan execution. Test documentation, as outlined, ensures that testing is thoroughly documented. In addition, test designs, test cases, and test procedures should be regularly reviewed based on past experience and results.

NTCIP 9012 v01 identifies two standards that cover the test documentation needs for ITS device testing—IEEE 829 and NTCIP 8007 v01.

a) **IEEE 829, IEEE Standard for Software Test Documentation—**standardizes test documentation content and includes content descriptions for test plans, test design specifications, test case specifications, test procedure specifications, and test reports.
b) **NTCIP 8007 v01, Testing and Conformity Assessment Documentation within NTCIP Standards Publications—**focuses on system interfaces for C2F communications. NTCIP 8007 v01 does not include provisions for IEEE 829 test plans and test designs, and combines test case and test procedure aspects, which IEEE 829 maintains as separate.

Annex A
ACRONYMS AND GLOSSARY

A.1 ACRONYMS AND ABBREVIATIONS

See Table 11 for acronyms and abbreviations used throughout the NTCIP family standards.

Table 11 Acronyms

Acronym/Abbreviation	Definition
AASHTO	American Association of State Highway and Transportation Officials
AP	Application Profile
API	Application Program Interface
APTA	American Public Transit Association
ASC	Actuated Signal Control
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation—1
ASTM	American Society for Testing and Materials
ATC	Advanced Transportation Controller
ATMS	Advanced Traffic Management System
AVL	Automatic Vehicle Location
BER	Basic Encoding Rules or Bit Error Rate
bps	bits per second
C2C	Center to Center
C2F	Center to Field
CCTV	Closed-circuit television
CMS	Changeable Message Sign
COTS	Commercial Off The Shelf
CRC	Cyclical Redundancy Check
CVO	Commercial Vehicle Operations
DATEX	DATa EXchange Protocol
DCM	Data Collection and Monitoring
DMS	Dynamic Message Sign
EBR	Exception-Based Reporting
ELMS	Electrical and Lighting Management Systems
DSRC	Dedicated Short-Range Communications
ESS	Environmental Sensor Systems
FCC	Federal Communications Commission
FCS	Frame Check Sequence (see Cyclic Redundancy Check)

Acronym/Abbreviation	Definition
FHWA	Federal Highway Administration
FMS	Field Management Station
FSK	Frequency Shift Keying
FTP	File Transfer Protocol
GIS	Geographic Information System
GPS	Global Positioning System
HAR	Highway Advisory Radio
HDLC	High-Level Data Link Control
HTTP	Hyper-text Transfer Protocol
IANA	Internet Assigned Numbers Authority
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISO	International Organization for Standardization
ISTEA	Intermodal Surface Transportation Efficiency Act of 1991
ITE	Institute of Transportation Engineers
ITS	Intelligent Transportation Systems
ITSA	Intelligent Transportation Society of America
MIB	Management Information Base
MULTI	Mark-Up Language for Transportation Information
NEMA	National Electrical Manufacturers Association
NHI	National Highway Institute
NTCIP	National Transportation Communication for ITS Protocol
OER	Octet Encoding Rules
OET	Outreach, Education and Training
OID	Object Identifier
OSI	Open Systems Interconnection
PER	Packed Encoding Rules
PICS	Protocol Implementation Conformance Statement
PMPP	Point-to-Multi Point Protocol
PRL	Profile Requirements List
PPP	Point-to-Point Protocol
RFC	Request For Comment
RMC	Ramp Metering Control
RSE	Roadside Equipment
RTM	Requirements Traceability Matrix
RWIS	Road Weather Information System (see also ESS)

Acronym/Abbreviation	Definition
SAE	Society of Automotive Engineers
SCP	Signal Control and Prioritization
SDO	Standards Development Organization
SEP	Systems Engineering Process
SLIP	Serial Line Internet Protocol
SNMP	Simple Network Management Protocol
SOAP	Simple Object Access Protocol
STMP	Simple Transportation Management Protocol
TCIP	Transit Communications Interface Protocol
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TEA-21	Transportation Equity Act for the 21st Century (successor to ISTEA)
TFTP	Trivial File Transfer Protocol
TMC	Transportation Management Center
TMDD	Traffic Management Data Dictionary
UDP	User Datagram Protocol
UDP/IP	User Datagram Protocol/Internet Protocol
VMS	Variable Message Sign
WAN	Wide Area Network
WIM	Weigh In Motion
XML	eXtensible Markup Language

A.2 GLOSSARY

See Table 12 for a glossary of terms, and their definition, used throughout the NTCIP family standards

Table 12 Glossary

Term	Definition
ANSI	American National Standards Institute, a standardization group that develops or adopts standards for the United States.
Application Services	The services collectively offered by the upper four layers of the OSI model.
Application Program Interface (API)	A set of calling conventions defining how a service is invoked through a software package.
ASCII	American Standard Code for Information Interchange. A 7-bit binary code representation of letters, numbers and special characters. It is universally supported in computer data transfer.
ASN.1	Abstract Syntax Notation One, a language for describing information to be processed by computer, an ISO standard.
Asynchronous	Data transmission in which the actual data is preceded by a start bit and followed by a stop bit since the time between transmitted characters varies. Compare with Synchronous.

Term	Definition
ATC	Advanced Transportation Controller, transportation field control equipment standards under development. FHWA, NEMA and the ATC Joint Standards Committee are spearheading the development effort.
Authentication	The process whereby a message is associated with a particular originating entity.
Authorization	The process whereby an access policy determines whether an entity is allowed to perform an operation.
Bandwidth	The range of frequencies that can be used for transmitting information on a channel, equal to the difference in Hertz (Hz) between the highest and lowest frequencies available on that channel. Indicates the transmission-carrying capacity of a channel.
Basic Encoding Rules (BER)	A series of procedures for describing transfer syntax of types specified with ASN.1. Transfer syntax is the actual representation of octets to be sent from one network entity to another. Used in conjunction with SNMP.
Baud Rate	The number of discrete signal events per second occurring on a communications channel. It is often interchanged with bits per second (bps), which is technically inaccurate but widely accepted for slower bit rates.
Bit	Binary digit. A single basic computer signal consisting of a value of 0 or 1, off or on.
Bit Error Rate (BER)	The number of bits transmitted incorrectly. In digital applications it is the ratio of bits received in error to bits sent.
Bps	Bits per second, transmission rate (speed) of data
Bridge	A means for connecting two networks at the data link layer.
Broadcast Address	An address referring to all stations on a medium.
BYTE and UBYTE	A group of bits acted upon as a group, which may have a readable ASCII value as a letter or number or some other coded meaning to the computer. It is commonly used to refer to 8-bit groups. Octet sized (8 bits) integers where BYTE is signed (range -128 to 127) and UBYTE is unsigned (range 0 to 255).
Carrier	A continuous frequency capable of being either modulated or impressed with another information-carrying signal. Carriers are generated and maintained by modems via the transmission lines of the telephone companies.
Changeable Message Sign	Changeable Message Sign (this terms has 2 common definitions: a.) in NTCIP 1203 v02, it defines Drum signs, and b.) certain public agencies use this term to describe VMS signs.
Checksum	An arithmetic sum used to verify data integrity.
CHOICE	As defined by ITU-T X.680, Abstract Syntax Notation One Specification of Basic Notation, a choice type is defined by referencing a list of distinct types; each value of the choice type is derived from the value of an object or data element.
Component	The closely related functions of a system. A component produces an information product.
Cyclical Redundancy Check (CRC)	Cyclical Redundancy Check. An error-detection technique consisting of a cyclic algorithm performed on each “block” of data at the sending and receiving end of the transmission. As each block is received, the CRC value is checked against the CRC value sent along with the block.
Data	Information before it is interpreted.
Data Dictionary	An organized listing of dialogs, messages, data frames, data elements, and their properties that are required so that both the user and the system developer have a common understanding of input, output, components of storage and intermediate calculations.
Data Flow	The description of information movement and the transforms that are applied as the data moves from input to output.

Term	Definition
Data Interface	The connection between two or more components through which information (e.g., data element or message) is passed.
Data Link Layer	Layer 2 of the OSI Reference Model; it is responsible for transmission, framing and error control over a single communications link.
Datagram	A self-contained unit of data transmitted independently of other datagrams.
Dialog	An ordered grouping of messages exchanged between at least two components.
Data Terminal Equipment (DTE)	Data Terminal Equipment. The device that is the originator or destination of the data sent by a modem. (A TIA-232-F signal)
Data Terminal Ready (DTR)	Data Terminal Ready. A signal generated by most modems indicating a connection between the DTE (computer) and the modem. When DTR is high, the computer is connected. (A TIA-232-F signal)
End-to-End Services	The services collectively offered by the lower three layers of the OSI model.
Flow Control	A mechanism that compensates for differences in the flow of data to and output from a modem or computer. Either hardware or software can be used for this control to prevent data loss. Hardware flow control using the modem makes use of a buffer to store data to be sent and data received. Flow control is necessary if the communications port is locked at a higher rate than the connection rate.
FSK modem interface	Typical method of traffic control system communications, phone line, or twisted wire based.
Full Duplex	Signal flow in both directions at the same time. It is sometimes used to refer to the suppression of on-line local echo and allowing the remote system to provide a remote echo.
Gateway	A router and translator between protocols; also, (imprecise usage) an entity responsible for complex topology mappings.
Half Duplex	Signal flow in both directions, but only one way at a time. It is sometimes used to refer to activation of local echo that causes a copy of sent data to be displayed on the sending display.
High-Level Data Link Control (HDLC)	Generalized network approach: high-level data link control
Highway Advisory Radio (HAR)	Low-powered AM or FM stations that broadcast brief messages to standard car radios from small transmitters placed near highways.
Host	(Internet usage) an end system.
Internet Engineering Task Force (IETF)	A group chartered by the IAB to develop certain RFCs for standardization.
Indirect Routing	The process of sending a network message to a router for forwarding.
Infrastructure	This refers to all fixed components in a transportation system such as rights of way, tracks, equipment, stations, parking/park-n-ride lots, signalization equipment and maintenance facilities.
Informative	Non-prescriptive information that provides context to a standard.
Intelligent Transportation Systems (ITS)	A major national initiative to improve information, communication and control technologies to improve the efficiency of surface transportation. Technological innovations that apply direct communications and information processing to improve the efficiency and safety of surface transportation systems. These include on-board navigation for vehicles, emergency communications systems, electronic toll/fare collections, traffic management centers, etc.

Term	Definition
Interchangeability	A condition which exists when two or more items possess such functional and physical characteristics as to be equivalent in performance and durability, and are capable of being exchanged one for the other without alteration of the items themselves, or adjoining items, except for adjustment, and without selection for fit and performance. (See National Telecommunications and Information Administration, U.S. Department of Commerce.)
Interoperability	The ability of two or more systems or components to exchange information and use the information that has been exchanged. (See IEEE 610.12-1990.)
Intermediate System	A network device performing functions from the lower three layers of the OSI model. Intermediate systems are commonly thought of as routing data for end systems.
Intermodal Surface Transportation Efficiency Act of 1991 (ISTEA)	Federal authorizing legislation for highways, transit and other surface transportation programs. Established intermodal objectives for national transportation system to achieve efficiency, air quality and environmental quality.
Intermodalism	The use and coordination of more than one mode of transportation.
International Organization for Standardization (ISO)	An international standards organization. ANSI is the primary interface to ISO within the United States. Often thought to be International Standards Organization because of the usage ISO for short.
Internet	A large collection of connected networks, primarily in the United States, running the Internet suite of protocols. Sometimes referred to as the *DARPA Internet, NSF/DARPA, Internet, or* the *Federal Research Internet.*
Internet Activity Board	Internet Activities Board, group in charge of authorizing RFCs for the purpose of standardizing Internet operations.
Internet Assigned Numbers Authority (IANA)	, group in charge of assigning Internet addresses.
Internet Protocol (IP)	The network protocol offering a connectionless mode network service in the Internet suite of protocols.
IP address	A 32-bit quantity used to represent a point of attachment in an internet. An Internet Protocol Address.
Internet suite of protocols	A collection of computer-communication protocols originally developed under DARPA sponsorship.
Local Area Network (LAN)	Any one of a number of technologies providing high speed, low-latency transfer and being limited in geographic size.
LONG and ULONG	Four byte (32 bits) integers where LONG is signed (range -2,147,483,648 to 2,147,483,647) and ULONG is unsigned (range 0 to 4,294,967,295).
Management Information Base (MIB)	A collection of data elements or objects defined using Abstract Syntax Notation One (ASN.1) that can be accessed via a network management protocol. (See Structure of Management Information.)
Manager	The entity that sends commands to entries and processes their responses.
Maximum Transmission Unit (MTU)	The largest amount of user data that can be sent in a single frame on a particular medium.
Message	A grouping of data elements that encapsulate an idea, concept or thing, or convey information. A basic message encapsulates an idea, concept or thing, and a compound message embeds one or more basic messages and other data elements to convey information.
Message Set Catalog	A list of messages and the functional requirements needed to support the exchange of information among components within a system, or between systems.
Message Set Template	The format used to transmit messages among components or between systems.

Term	Definition
Network	A collection of subnetworks connected by intermediate systems and populated by end systems.
Network Identifier	The portion of an IP address corresponding to a network in an internet.
Network Layer	That portion of an OSI system responsible for data transfer across the network, independent of both the media comprising the underlying subnetworks and the topology of those subnetworks.
Network management	The technology used to manage a network. Usually referring to the management of networking specific devices such as routers. In the context of the NTCIP, refers to all devices including end systems that are present on the network or inter network.
Normative	Prescriptive requirements for the use of this standard.
NTCIP	The National Transportation Communication for ITS Protocol, a joint standardization effort of AASHTO, ITE, and NEMA, with support from the U.S. DOT's RITA.
NTCIP Home Page	Site on the World Wide Web where one may obtain the latest NTCIP information. The address is www.ntcip.org/ .
NTCIP Joint Committee	An 18-member committee of ITS experts appointed by AASHTO, ITE, and NEMA who guide the development of the NTCIP and recommend acceptance actions to the SDOs. (referred to as the Joint Committee on the NTCIP).
Object	A representation of a data element that is managed. The definition of a data element or message including its name, object identifier, description and syntax.
OBJECT IDENTIFIER (OID)	A unique name (identifier) that is associated with each type of data element in a MIB. This is a defined ASN.1 type. "A value (distinguishable from other such values) that is associated with an object identifier type. A simple type whose distinguished values are the set of all object identifiers allocated in accordance with the rules of [ASN.1]." The number or address by which a data element may be located on the NTCIP or TCIP object tree.
OBJECT-TYPE	The macro defined in RFC-1212 that is the format used to define SNMP objects or data elements. The OBJECT-TYPE macro consists of five fields: Object Name Syntax Description Access Status.
OCTET	An ordered sequence of eight bits.
OER	Octet Encoding Rules, a variation BER developed for use on low bandwidth communications links. OER is based on Octet boundaries (opposite of PER, which is based on bit boundaries).
Open Systems Interconnection (OSI)	An international effort to facilitate communications among computers of different manufacture and technology.
Parity	A simple error detection method used in both communications and computer memory checking to determine character validity.
PER	Packed Encoding Rules, a variation of BER developed for use on low bandwidth communications links, specified in ISO 8825. The original version of NEMA TS 3.2 used this term for a transportation industry-specific set of encoding rules that has since been renamed to OER (as defined in NTCIP 1102:2004).
Physical Address	The address of a physical interface.
Physical Layer	That portion of an OSI system responsible for the electro-mechanical interface to the communications media.
Point-to-Point Protocol (PPP)	Transmission of data between two and only two stations on a point-to-point link.

Term	Definition
Point-to-Multi-Point Protocol (PMPP)	Transmission of data between multiple stations or nodes (i.e., one primary and multiple secondaries).
Port Number	Identifies an application-entry to a transport service in the Internet suite of protocols. The concept of ports are often present in OSI literature, however, ports are not Internet standard, but exists as local network conventions only.
Presentation Layer	That portion of an OSI system responsible for adding structure to the units of data that are exchanged.
Primary	A node on a link that controls the polling to and from secondary nodes on that link and controls the communications from the secondary nodes on that link.
Profile	The defined protocol at each of the seven OSI layers. A standard that combines one or more base standards and selects appropriate options or functions within them. (A base standard may be a “standard” or another profile that references standards).
Protocol	A set of conventions governing the format and relative timing of message exchange between two communicating processes. A system of rules and procedures governing communications between two devices.
Protocol Data Unit (PDU)	A part of transmitted data that contains information used by the protocol at a particular layer in the OSI stack.
Proxy agent	A device that receives and responds to network management commands on behalf of another entity.
RFC	Request for Comments, the name given to correspondence and standards by the IAB.
Router	A level 3 (network layer) relay
Secondary	A node on a link that is controlled by the primary node in terms of polling and communications.
SEQUENCE and SEQUENCE OF	An ordered record or array (respectively) of data elements or objects. “Types defined by referencing an ordered list of types (some of which may be declared to be optional)”
Service Primitive	An artifact modeling how a service is requested or accepted by a user
Session Layer	That portion of an OSI system responsible for adding control mechanisms to the data exchange.
SHORT and USHORT	Double octet sized (16 bits) integers where SHORT is signed (range -32,768 to 32,767) and USHORT is unsigned (range 0 to 65,535).
Simple Transportation Management Protocol (STMP)	Simple Transportation Management Protocol, a variation of SNMP developed by NEMA to address low bandwidth communication links and real time device monitoring.
Structure of Management Information (SMI)	A definition of how to create management data element or objects and a hierarchical (tree like) definition of nodes where management objects or data elements attach for unique identification.
Simple Network Management Protocol (SNMP)	A communications protocol developed by the IETF, used for configuration and monitoring of network devices.
Simple Network Management Protocol version 2 (SNMPv2)	A modification of SNMP that is undergoing evaluation by the Internet community.
Socket	A pairing of IP address and a port number
Transit Communications Interface Protocol (TCIP)	Protocols specific to the transit community developed and maintained by APTA.

Term	Definition
Transmission Control Protocol/Internet Protocol (TCP/IP)	A protocol addressing both the network and transport layers.
TIA-232-F	Telecommunications Industries Association standard that defines the serial port on a PC (formerly designated as EIA/TIA)
TLV	Tag, Length, Value: the form used in SNMP encoding.
Transaction	See Dialog.
Transmission Control Protocol (TCP)	The transport protocol offering a connection-oriented transport service in the Internet suite of protocols.
Transport Layer	That portion of an OSI system responsible for reliability and multiplexing of data transfer across the network (over and above that provided by the network layer) to the level required by the application.
Transport Level	The combination of protocols, at the transport layer and below, used in a given context.
User data	Conceptually, the part of a protocol data unit used to transparently communicate information between the users of the protocol. Prefixed by the protocol control information.
User Datagram Protocol (UDP)	The transport protocol offering a connectionless mode transport service in the Internet suite of protocols.
Wide Area Network (WAN)	Any one of a number of technologies that provide geographically distant transfer.
eXtensible Markup Language (XML)	eXtensible Markup Language is a standard of the World Wide Web Consortium (W3C). XML is a means of encoding information (data) so that another computer receiving that encoded information understands its contents and acts on that content.

Annex B
BIBLIOGRAPHY

See Table 13 for a selected list of additional references on various topics.

Table 13 Selected Additional References

Subject	Reference
Communications	Communication Handbook for Traffic Control Systems, Federal Highway Administration Report FHWA-SA-93-052, April 1993
Object Definition	Understanding SNMP MIBS, 1997, D. Perkins and E. McGinnis, Prentice Hall, Inc., ISBN 0-13-437708-7 ASN.1: The Tutorial & Reference, 1993, D. Steedman, Technology Appraisals Ltd., ISBN 1-871802-06-7 ISO/IEC 8824: Abstract Syntax Notation One (ASN.1) (ITU-T X.680-X.690) (various dates circa 2008) IEEE Std 1488-2000, IEEE Trial-Use Standard for Message Set Template for Intelligent Transportation Systems IEEE Std 1489-1999, IEEE Standard for Data Dictionaries for Intelligent Transportation Systems
OSI	The Open Book: A Practical Perspective on OSI, 1990, M.T. Rose, Prentice Hall, Inc. Understanding OSI, 1994, J. Larmouth, www.isi.salford.ac.uk//books/osi/osi.html ,1997
OSI Network Management	Telecommunications Network Management into the 21st Century, 1994, S. Aidarous and T. Plevyak, IEEE Press, New York, NY
Profiles	Guide to Open System Specifications, European Workshop for Open Systems, 1997
SNMP Protocol	SNMP, SNMPv2 and CMIP The Practical Guide to Network Management Standards, 1993, W. Stallings, Addison-Wesley Publishing Company, Inc., ISBN 0-201-63331-0 SNMP, SNMPv2 and RMON, 1996, W. Stallings, Addision-Wesley Publishing Company, Inc., ISBN 0-201-63479-1 Managing Internetworks with SNMP, 1995, M. Miller, M&T Books, ISBN 1-5581-304-3 SNMP: A Guide To Network Management, 1995, S. Feit, McGraw Hill, Inc., ISBN 0-07-020359-8 Perkins, D. and McGinnis, E., Understanding SNMP MIBS, 1997, Upper Saddle River, New Jersey: Prentice Hall, Inc., 1997.

Subject	Reference
Systems Engineering Process (SEP)	Federal Highway Administration Systems Engineering Guidebook for ITS, (see www.fhwa.dot.gov/cadiv/segb/ IEEE Std 1233-1998, IEEE Guide for Developing System Requirements Specifications INCOSE (International Council on Systems Engineering) Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activiteis, Version 3.1, 2007 www.incose.org
TCP,UDP, IP and PPP Protocols	Internetworking with TCP/IP, 1995, D. Comer, Prentice Hall, Inc., ISBN 0-13-216987-8 TCP/IP: Architecture, Protocols and Implementation, 1993, S. Feit, McGraw Hill, Inc., ISBN 0-07-020346-6 TCP/IP Illustrated, Volume 1, The Protocols, W. R. Stevens, 1994, Addison Wesley Publishing Co. TCP/IP Illustrated, Volume 2, The Implementation, G. R. Wright and W. R. Stevens, 1995, Addison Wesley Publishing Co. TCP/IP Tutorial and Technical Overview, International Technical Support Center, Raleigh, NC, 1990, Document Number GC24-3376-01, IBM Corp.
Training Courses	Center to Center Communications, ITS Standards Outreach, Education and Training Program, Institute of Transportation Engineers Intelligent Transportation Systems (ITS) Software Acquisition, National Highway Institute Intelligent Transportation Systems (ITS) Procurement, National Highway Institute ITS Standards Overview, ITS Standards Outreach, Education and Training Program, Institute of Transportation Engineers Management and Operations of Intelligent Transportation Systems Using the National ITS Architecture for Deployment, National Highway Institute

Annex C
TRAPS AND EXCEPTION-BASED REPORTING (EBR)

The next major version after NTCIP 1103 v02 will include the concept of traps, or exception based reporting (EBR), primarily for use on CSMA/CD networks (e.g. Ethernet). Traps are designed to preserve the fidelity of data while consuming significantly reduced amounts of bandwidth. Traps have been used in wireless environments where constant polling creates bandwidth bottlenecks and where lost packets compromise the fidelity of the monitored status.

Basically, the management station configures “triggers” within an NTCIP field device that cause the device to initiate a transmission to the management station (typically a transportation management center (TMC)) and report configured data without the need to continuously poll the device.

EBR is built on the concept of “events,” which are configured within the NTCIP field device, and the concept of “trap channels,” which identify where the event information is to be transmitted. The structure is very flexible, allowing a single event to trigger transmissions to multiple management stations. The trap mechanism also includes the concept of aggregation, or accumulating data from a collection of events to be transmitted periodically or at a specific time. The NTCIP trap mechanism includes parameters to identify queues for each trap channel, acknowledgement requirements, retry counters, anti streaming parameters, and trigger requirements. The NTCIP trap mechanism also supports the configuration of block objects, called watch blocks, which can be monitored for changes and block objects (called report blocks), which contain the data to be sent to the management station when the event (trigger) is detected.

NOTE—Each event includes the reported data and a time stamp showing the time of the event (within 1 second).

Traps can report changes of state within an NTCIP field device without the need to constantly poll that device. In a wireless environment, traps can reduce bandwidth requirements by almost 90% without sacrificing data fidelity.

Previously, systems polled an NTCIP field device once per second to accurately track device operation, but traps can be configured to report changes of state without polling. An example of a two phase intersection, operating with a 90 second cycle length is illustrated in Figure 18. To track its second-by-second operation, a management station would have to transmit 90 GET commands and receive 90 Status messages for a total of 180 messages. However, there are really only four changes of state (macroscopic monitoring): phase 2 goes green (Main Street Green); phase 2 green ends; phase 4 goes green; and phase 4 green ends. Even with microscopic monitoring, there are only nine state changes.

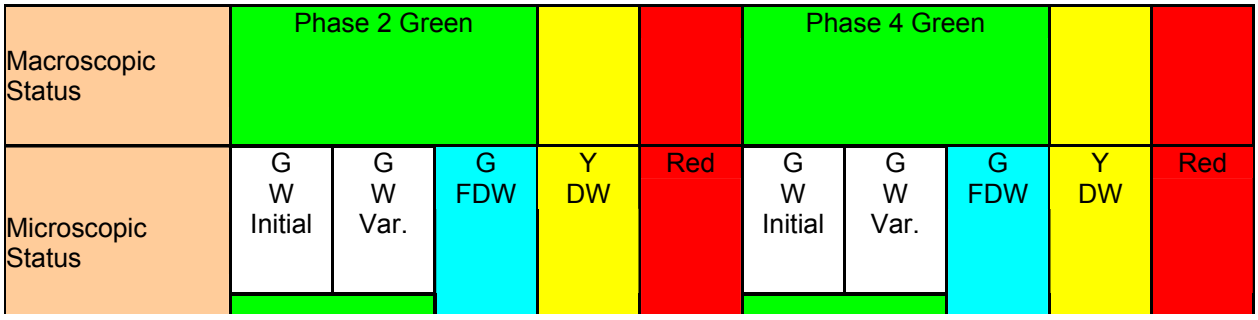

Figure 18 Example—Trap-Based Communications

For macroscopic monitoring to one second accuracy, there are only four messages, while for microscopic monitoring to one second accuracy, there are nine messages—representing a 95% to 97% reduction in the number of messages, although individual messages are larger.

Other uses for traps include:

a) Permit NTCIP field devices to report anomalies and alarms (e.g. cabinet door open, controller in flash, over temperature) to center systems.
b) Support peer-to-peer, event-driven communications, where one NTCIP field device can register with another NTCIP field device to receive advance warning of platoons, or priority or preemption conditions so that downstream intersections can react.

Annex D
EXAMPLE—NTCIP IMPLEMENTATIONS

D.1 INTRODUCTION

Several potential NTCIP Implementations are presented as examples of how various Information, Application, Transport, Subnetwork and Plant Levels may be combined. A user may create an NTCIP *stack* by selecting the appropriate NTCIP standards at each Level applicable to a unique application and system design. This approach enables the user to better specify choices specific to the system being deployed.

D.2 CENTER TO FIELD (C2F)

Three examples are provided for C2F communications.

D.2.1 Example—Center to Field (C2F) Implementation Without Routing

The example in Figure 19 shows one possible implementation of NTCIP C2F communications where routing through an intermediate device is not needed. In this example, the Transport Level is T2/ NULL because there is no need for a routing protocol.

Figure 19 highlights one implementation subset of the NTCIP Framework, and shows the standard(s) implemented at each NTCIP Framework Level. The implementation of both STMP and SNMP are shown at the Application Level and T2/NULL at the Transport Level. Together, these standards provide services for an NTCIP system, such as a traffic signal system, that does not involve routing through intermediate devices. This example shows the selection of both STMP and SNMP at the Application Level within the NTCIP Stack, since this is a common implementation in many systems, such as traffic signal systems, that use dynamic objects.

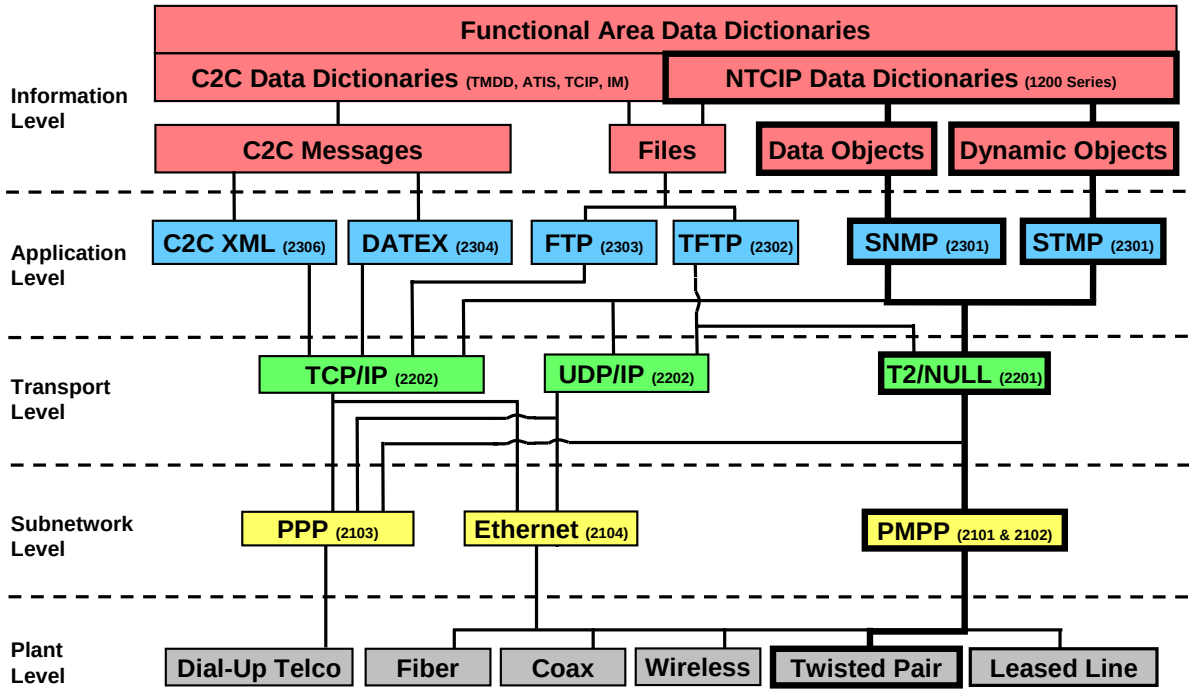

Figure 19 Example—Center to Field (C2F) Implementation Without Routing

The Subnetwork Level standard selected in this example is Point-to-MultiPoint, FSK modems. The Plant Level in this example is shown as agency-owned twisted-pair wire, but any suitable media can be used.

D.2.2 Example—Center to Field (C2F) Implementation With Routing

The example in Figure 20 shows one possible implementation of NTCIP C2F communications where routing through one or more intermediate devices is needed. The routing can take either the form of connectionless or connection-oriented transport delivery services, depending on the selection at the Transport Level. For connectionless transport delivery services, the selection of UDP/IP should be made at the Transport Level. For connection-oriented transport delivery services, TCP/IP should be selected as the appropriate Transport Level. In other words, TCP establishes a direct connection between the two devices through a handshake arrangement and then proceeds to transmit data with assurance that all messages are received –otherwise the messages are re-transmitted. UDP, on the other hand, uses more of a broadcast approach with no assurance that the message was actually received.

Figure 20 shows the standard(s) implemented at each NTCIP Framework Level, for the example subset. The implementation of both STMP and SNMP at the Application Level and TCP, UDP/IP at the Transport Level are also shown. Together, these standards provide services for an NTCIP system, such as a traffic signal system, with intermediate routing.

In this example, the Subnetwork Level standard selected is Point-to-Point, V-Series modems, and the Plant Level standard selected is leased line Telco-provided communications.

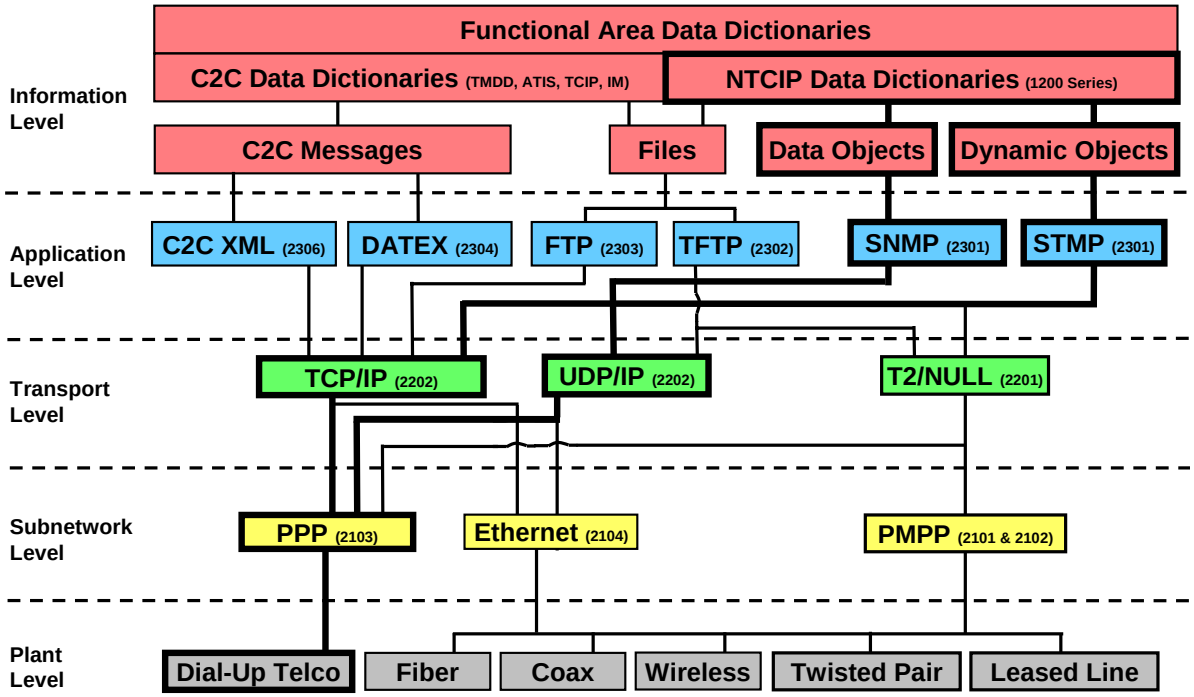

Figure 20 Example—Center to Field (C2F) Implementation With Routing

D.2.3 Example—Center to Field (C2F) Implementation—Routable & Non-Routable

The example in Figure 21 shows an implementation of NTCIP C2F communications where both routable and non-routable links are used, such as in a closed-loop traffic signal system—the classic case where the management station (central) dials up the field master and the field master then talks to subordinate

controllers using agency owned twisted wire. The routing can take either the form of connectionless or connection-oriented transport delivery services depending on the selection at the Transport Level.

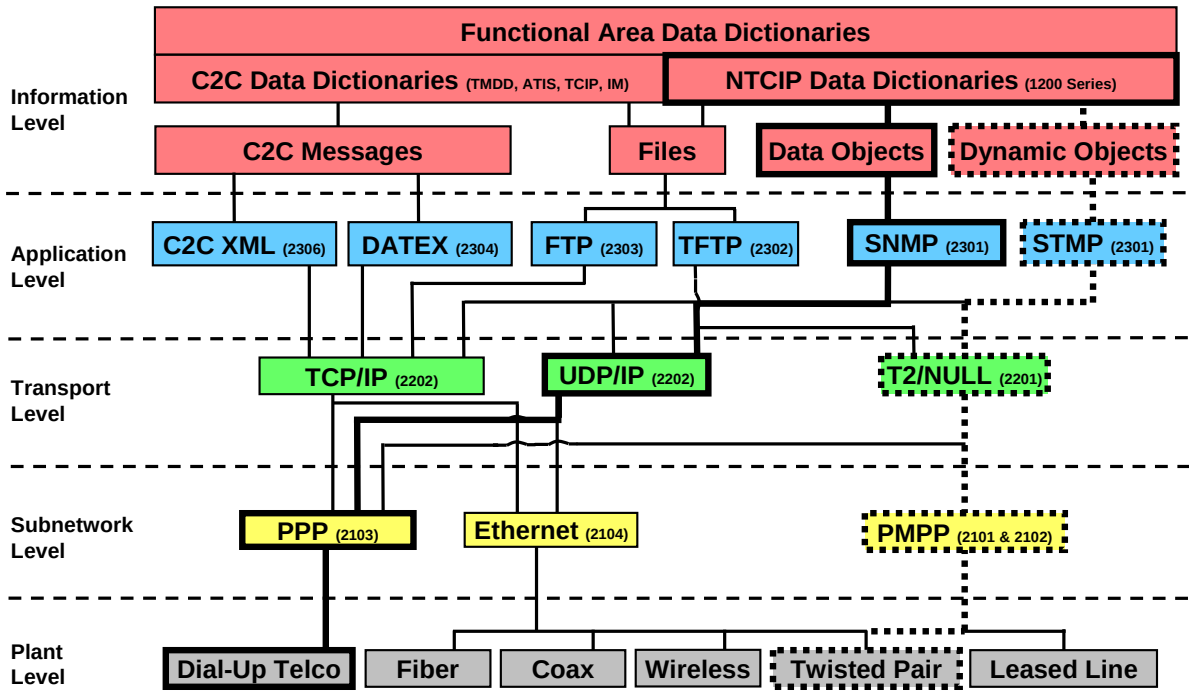

Figure 21 Example—Center to Field (C2F) Implementation With Routable and Non-Routable Links

For connectionless transport delivery services, the selection of UDP/IP should be made at the Transport Level. For connection-oriented transport delivery services, TCP/IP should be selected as the appropriate Transport Level. In other words, TCP establishes a direct connection between the two devices through a handshake arrangement and then proceeds to transmit data with assurance that all messages are received—otherwise the messages are re-transmitted. UDP, on the other hand, uses more of a broadcast approach with no assurance that the message was actually received.

Most closed-loop traffic signal systems use UDP/IP for the dial-up connection from the management station to the field master. Once the dial-up connection is made, the field master then communicates with subordinate local controllers using a traditional Point-to-MultiPoint and FSK Modem approach.

The example NTCIP implementation illustrated in Figure 21 highlights an implementation of the NTCIP Framework for a typical closed-loop traffic signal system. The bold lines denote the routable dial-up portion of the communication route. The dashed lines denote the non-routable fixed communications connection. Figure 21 shows the standard(s) implemented at each NTCIP Framework Level.

Figure 21 shows the implementation of both STMP and SNMP at the Application Level. SNMP is used for the dial-up portion (shown using solid bold lines and box outlines), while STMP is used when dynamic objects are needed for local controllers subordinate to the field master (shown using dashed lines and box outlines). TCP/IP or UDP/IP is selected for use at the Transport Level for the dial-up connection between the management station and the field master. Together, these standards provide services for an NTCIP system, such as a traffic signal system, that involves intermediate routing. T2/Null is used as the Transport Profile for the non-routable communications portion of the connection between the field master and the subordinate local controllers.

The Subnetwork Level standards selected in this example also depend on whether the routable or non-routable portions are being considered. Point-to-Point and V-Series modems are used when the Plant Level infrastructure is Telco-provided communications, for the dial-up portion of the closed-loop traffic signal system example. Meanwhile, Point-to-MultiPoint and FSK modems are used along with a twisted wire Plant Level for communications between the field master and the local controller.

D.3 CENTER TO CENTER (C2C)

Two examples are provided for C2C communications.

D.3.1 Example—Center to Center (C2C) Implementation Using DATEX

The example in Figure 22 shows one possible implementation of NTCIP C2C communications using DATEX.

Figure 22 depicts an example C2C NTCIP implementation, and is one variation of an approach using DATEX., which is intended to provide connection-oriented transport delivery services between transportation management centers supporting subordinate field devices.

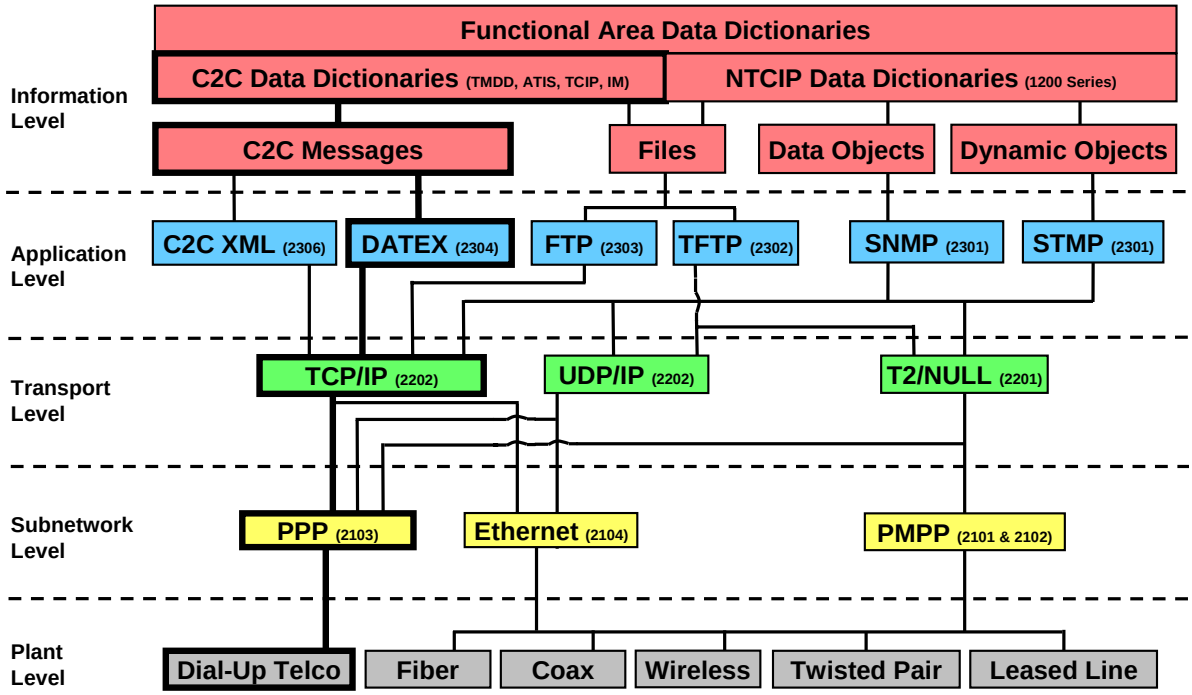

Figure 22 Example—Center to Center (C2C) Implementation With DATEX

The example in Figure 22 highlights one implementation subset of the NTCIP Framework for C2C communications, and shows the standard(s) implemented at each NTCIP Framework Level for using DATEX at the Application Level within the NTCIP Framework.

For C2C communications, the choices that are offered at the Application Level include DATEX and C2C XML. The choices that are defined for the Transport Level are UDP/IP for connectionless transport services and TCP/IP for connection-oriented transport delivery services. The Subnetwork Level options include a variety of high bandwidth options, such as Ethernet and PPP. In this case, one example might be to use Frame Relay with a PPP at the Subnetwork Level. The Plant Level can include a variety of options such as telco lines, as in this example, or fiber.

D.3.2 Example—Center to Center (C2C) Implementation Using C2C XML

The example shown in Figure 23 is one possible implementation of NTCIP C2C communications using C2C XML, and is intended to provide connection-oriented transport delivery services between transportation management centers supporting subordinate field devices.

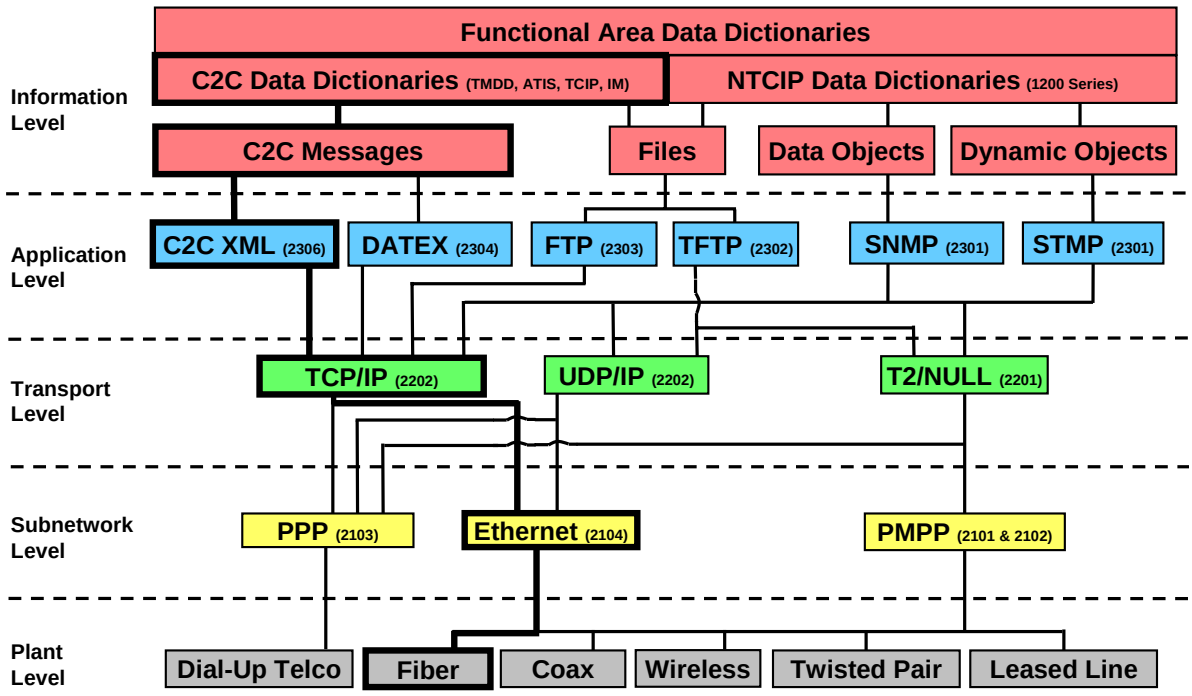

Figure 23 Example—Center to Center (C2C) Implementation With C2C XML

The example in Figure 23 highlights one implementation subset of the NTCIP Framework for C2C communications, and shows the standard(s) implemented at each NTCIP Framework Level for using C2C XML at the Application Level.

For C2C communications, the choices that are offered at the Application Level include DATEX and C2C XML. The choices that are defined for the Transport Level are UDP/IP for connectionless transport services, and TCP/IP for connection-oriented transport delivery services. The Subnetwork Level options include a variety of high bandwidth options, such as Ethernet and PPP. In this case, an example might be to use ATM at the Subnetwork Level. The Plant Level can be implemented using fiber.

Annex E
EXAMPLE—NTCIP COMMUNICATIONS BANDWIDTH CALCULATIONS

E.1 COMMUNICATIONS BANDWIDTH ANALYSIS

NTCIP standards do not address issues related to specifying bandwidth requirements or how bandwidth is allocated. Bandwidth is a measure of how much information can be sent through a connection. It is usually expressed as bps. As shown in Figure 24, the total bandwidth is accumulated by calculating information content plus a determination of overhead for SNMP and IP communications.

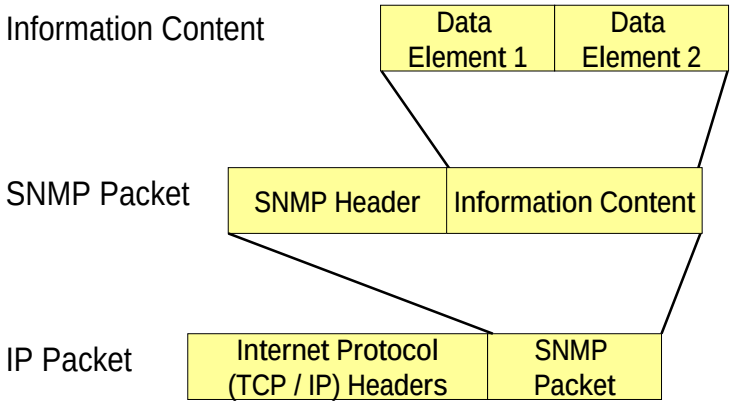

Figure 24 Communications Bandwidth Analysis

Planners and implementers should understand that specific media may limit how much information can be sent. Annex E provides several examples of how to estimate what can be sent, the overhead associated with sending it, and the organization of the physical media to support the information exchanges.

E.2 CENTER TO FIELD (C2F) BANDWIDTH ANALYSIS

In C2F communications, bandwidth considerations are of great concern to planners and implementers. Unlike an office environment, C2F communications links are generally less than 56 Kbps. Hundreds of existing systems use multi-drop, 1200 bps modems. Compounding the issue, these links tend to be dedicated to the transportation application and are the full responsibility of transportation personnel to design, implement and maintain. These bandwidth analyses should help to understand the factors and thinking that go into understanding bandwidth requirements and calculating an appropriate communications data rate.

NOTE—In general, some devices, such as dynamic message signs, advisory radio transmitters, ramp meters, traffic detector stations, and weather stations are much less sensitive to timing and latency issues, because communication between center and device is much less frequent. Communications to these devices can typically be operated satisfactorily using SNMP, even at 1200 bps.

The following two analyses are used to discuss a few of the many alternative techniques that can be used in a C2F multi-drop communications system using NTCIP. Examples (by no means a complete list) of options available include:

a) Use SNMP for all or some messages
b) Use STMP for all or some messages
c) Use only standard data elements, or also use some vendor-specific data elements

d) Use a bit rate of 1200, 2400, 4800, 9600, 19200, or other bit rate
e) Use half duplex or full duplex communications
f) For full duplex, overlap or do not overlap outgoing with incoming messages
g) Use twisted pair, fiber, radio, leased lines, or other media
h) For twisted pair and leased lines, use one or two pairs per channel
i) Use modems that are fast or slow to reach the ready state when communication is to be established
j) Limit the maximum number of devices on a channel to 2, 4, 6, 8, 10, or other number
k) Gather detector data on a clock time basis, for example every minute, or signal cycle basis
l) Request each type of data every second, every minute, or every hour.
m) Use a fixed or variable polling cycle duration
n) Use a fixed or variable device sequence in the polling cycle
o) Use a fixed or variable message sequence for each device
p) Wait for response in same poll or get it in next poll
q) Interleave upload/download messages or suspend status and get it over with ASAP
r) Use the same status message all the time, or different status messages
s) Insert occasional non-status requests in place of or in addition to status request
t) Allow spare time in a polling cycle for additional non-status messages, or allow the cycle to expand when non-status messages added

With the exception of vendor-specific data elements, any controller that meets all mandatory, optional and recommended requirements of the relevant NTCIP C2F standards for that field device type supports all of the relevant functions and operations suggested above and many more, without any software change.

For purposes of discussion, NTCIP C2F communications in an imaginary traffic control system are assumed. The system consists of a central management application that is used to set up, monitor and control a network of intersection traffic signal controllers. The primary communications requirements of the imaginary system are:

a) Synchronize the time and date in all field devices.
b) Provide a map display of the status of all intersections.
c) Control the overall coordination timing pattern to be put into effect.
d) Control the operation of two lane-closed signs.
e) Monitor all intersections for any abnormal conditions.
f) Accumulate volume and occupancy data for 16 detectors to perform off-line optimization.
g) Provide full upload and download of the complete database or programming data in each field device.
h) Support 24 signalized intersections in the system.

This example:

a) Discusses what data element definitions support this functionality;
b) Characterizes the overhead of sending the information via various protocols;
c) Compares and contrasts modems;
d) Defines the number of drops on a communications channel; and
e) Calculates appropriate modem speed to accommodate the information and timing characteristics.

A slightly modified set of requirements describes a polling sequence approach to define when message exchanges take place.

E.2.1 Estimate—Message Exchanges and Frequency

The first step in this analysis is to define what information exchanges are identified to meet the required functionality and how often they occur.

To synchronize the time and date in all field devices, the following data element from NTCIP 1201:2005 is used: globalTime—Section 2.4.1

This data element can be used in a message to set or retrieve the current date and time in a remote device. Typical usage is to send the command to all intersections at least once a day. The time in each individual intersection is checked (read) at least once a day, as well.

To provide a map display of an intersection the following data elements defined in NTCIP 1202:2005 are used:

a) phaseStatusGroupGreens—Section 2.2.4.4
b) phaseStatusGroupYellows—Section 2.2.4.3
c) phaseStatusGroupWalks—Section 2.2.4.7
d) phaseStatusGroupPedClears—Section 2.2.4.6
e) phaseStatusGroupVehCalls—Section 2.2.4.8
f) phaseStatusGroupPedCalls—Section 2.2.4.9
g) overlapStatusGroupGreens—Section 2.10.4.4
h) overlapStatusGroupYellows—Section 2.10.4.3
i) coordPatternStatus—Section 2.5.10
j) shortAlarmStatus—Section 2.4.9

These data elements provide green and yellow indications for up to 8 vehicle phases and 8 overlaps, walk and pedestrian clearance indications for up to 8 pedestrian movements, the current coordination pattern (cycle, split and offset) in effect, and an indication of preemption, problems with the coordination pattern, any detector fault, or some other type of fault condition. This information is intended to provide a real-time display and is typically read from each intersection controller on a once-per-second basis.

To control the timing pattern to put into effect and turn on and off the lane closed signs, the following data elements from NTCIP 1202:2005 are used:

a) systemPatternControl—Section 2.5.14
b) specialFunctionOutputControl – Section 2.4.14.3

This information is intended to be sent to all intersections about once per minute. To retrieve volume and occupancy data from two volume/occupancy detectors at a time, the following data elements from NTCIP 1202:2005 are used:

a) volumeOccupancySequence—Section 2.3.5.1
b) detectorVolume (1st detector)—Section 2.3.5.4.1
c) detectorOccupancy (1st detector)—Section 2.3.5.4.2
d) detectorVolume (2nd detector)—Section 2.3.5.4.1
e) detectorOccupancy (2nd detector)—Section 2.3.5.4.2

The volume and occupancy data would be read approximately once-per-minute. It is typical to have “count stations” spread out over several intersections. This type of information would be asked for from the intersections that have one or more count stations.

To provide additional information about the status of an intersection, the following data elements are used:

a) unitAlarmStatus1—Section 2.4.8
b) localFreeStatus—Section 2.5.11

These data elements are to read only when the shortAlarmStatus indicates some type of fault condition. They provide more detail about any potential fault condition. Typically, this occurs no more than once-per-hour.

To provide complete upload and download of a controller's database, the following block objects as defined in NTCIP 1202:2005 are used:

a) Data ID & Data Type—Section 3.1
b) Phase Data—Section 3.2
c) Vehicle Detector Data—Section 3.3
d) Pedestrian Detector Data—Section 3.4
e) Pattern Data—Section 3.5
f) Split Data—Section 3.6
g) Timebase Control Data—Section 3.7
h) Preempt Data—Section 3.8
i) Sequence Data—Section 3.9
j) Channel Data—Section 3.10
k) Overlap Data—Section 3.11
l) Port 1 Data—Section 3.12
m) Schedule Data—Section 3.13
n) Day Plan Data—Section 3.14
o) Event Configuration Data—Section 3.15
p) Dynamic Object Configuration Data—Section 3.17
q) Dynamic Object Owner Data—Section 3.18
r) Dynamic Object Status Data—Section 3.19
s) Miscellaneous Data—Section 3.20

These block objects are defined as OCTET STRING [an ASN.1 data type that is used to specify octets (eight-bit bytes) of binary or textual information] objects consisting of anywhere between 10 and 128 discrete objects]. NTCIP 1202:2005 defines block objects by grouping previously defined data element definitions into larger blocks (hence, the term ‘block objects’) enabling a faster transmission.

Additionally, each vendor most likely defines additional data elements and proprietary block objects to enable special features that set them apart from other vendors. The block objects could define the entire “database” of a device. The block objects would only be sent and retrieved on an as-needed basis and would, at most, occur no more than once-per-day. The block objects could represent the records in file upload or download. While these vendor-specific data elements are not currently defined in an NTCIP standard, they may respond to agency specification requirements.

In the course of fine-tuning an intersection, numerous programming entries for phase timing and coordination might be sent once or twice a day. The following data elements are typical:

a) phaseWalk—Section 2.2.2.2
b) phaseMinimumGreen—Section 2.2.2.4
c) phasePassage—Section 2.2.2.5
d) patternCycleTime—Section 2.5.7.2
e) patternOffsetTime—Section 2.5.7.3
f) splitTime—Section 2.5.9.3

The typical intersection is set up for five-phase operation and has only six timing patterns defined. It is assumed that only one phase or pattern would be adjusted at any one time. Therefore, the number of data elements associated with this type of operation is assumed to be five.

Table 14 summarizes the messages and their frequency.

Table 14 Frequency of Messages

Message Exchange	Frequency
Date and Time	1 per day—all
Intersection Map Data	1 per second X 24 intersections
Pattern Command	1 per minute—all
Detector Data	1 per minute X 8 intersections
Detailed Status	1 per hour X 8 intersections
Upload Download	1 per day X 24 intersections
Tuning	2 per day X 24 intersections

E.2.2 Estimate—Application Message Size

The following two rules of thumb can be used to estimate SNMP and STMP messages (not including Block Objects):

a) SNMP Message Size = 26 bytes of header + 23 bytes per data element
b) STMP Message Size = 1 byte of header + 1 byte per data element

These rules are approximations and do not include lower layer protocol overhead. The rules are based upon the assumption that most exchanges deal with status and control data elements that can be expressed in one byte. The majority of set up data elements can also be expressed in one byte. The rules of thumb would not apply to exchanges involving OCTET STRINGs or OBJECT IDENTIFIERs. If you accept the rules as such, you can skip the next two sections. Technical details follow, providing an in depth explanation on how exact sizes of messages can be derived.

E.2.2.1 SNMP Application Message Bits and Bytes

The actual bits and bytes of an SNMP message are defined using the Tag-Length-Value representation method defined in ISO 8825. All data elements can be expressed as Tag (or Type) of either SEQUENCE, INTEGER, OCTET STRING, or OBJECT IDENTIFIER. The Tag indicates how to think of the Value component. It indicates that it may be number, string (or text), or the identifier of something. It can also indicate that what follows is a series of data that is expressed as a Tag-Length-Value of something. There are several derived types that represent subsets of SEQUENCE, INTEGER, OCTET STRING, or OBJECT IDENTIFIER but any derived type resolves to one of the aforementioned ones.

The second component of a data element is its Length. For example, the Length of the INTEGER “1” when represented in computer terminology is one. It represents how many bytes it takes to store “1” in memory. The OCTET STRING “public” has a Length of 6 because it is expressed in 6 bytes.

The third component of a data element is its Value. The Value of INTEGER “112” is expressed as 0x70 in computer terminology [decimal 112 = 70 hexadecimal = 0111 0000 binary]. The OCTET STRING “p” is also expressed as Value 0x70. The reason a computer can differentiate the 0x70 as either “112” or “p” is because of the Tag.

An SNMP message is defined as a SEQUENCE and Length of two predefined fields that describe the protocol, plus a field that defines the data carried by the protocol. The predefined fields consist of version and community name. Both the version and community name are expressed in the Tag-Length-Value form. A data field that follows it describes the operation that is to be performed. The SetRequest PDU field at the end of row is expanded in the row below. This is illustrated in the SNMP Message Fields row of Table 14.

NOTE—The expanded **SetRequest PDU** field starts with the *Tag* of the operation, is followed by the *Length* of the data that follows, and consists of *Value*s of three *Tag-Length-Value* fields. The last field of **SetRequest PDU** consists of the **Variable Bindings** field. It is expanded on the next row.

As before, it begins as a *Tag-Length-Value* of a SEQUENCE and Length of one or more **Bindings**. The last row is closer to defining the actual data is, but we have to go through another *Tag-Length-Value* sequence to describe the *identity* and *value* of a single data element or **Bindings**.

From the communications perspective, each of the data elements defined in one of the Object Definitions Standards such as NTCIP 1201 or NTCIP 1202 has two components: an *identity* and a *value*. The *identity* part of the globalTime data element is its OBJECT IDENTIFIER (OID). The full OID of globalTime is:

a) <iso.org.dod.internet.private.enterprises.nema.transportation.devices.global.globalTimeManagement.1> ; or
b) <1.3.6.1.4.1.1206.4.2.6.3.1.0>

The *value* component of globalTime is an INTEGER with a value such as 925997608. This particular value represents May 6, 1999 at 2:33 PM UCT expressed as the seconds since Midnight January 1, 1970. For a more detailed discussion on OID, please refer to Section 6.

The actual number of bytes that are used to encode the *identity* and *value* of the globalTime data element varies with protocols used. For SNMP, each component of the data element is expressed in the form of *Tag-Length-Value*. For globalTime, the *identity Tag* is OID (0x06). The '0x' prefix indicates that the number presented is base 16, or hexadecimal. The *identify Length* value is 13 (0x0D). The *identity Value* is 1.3.6.1.4.1.1206.4.2.6.3.1.0 (0x2B06010401893604020603 0100). For the *value* component, the *value Tag* is INTEGER (0x02). The *value Length* is 4 (0x04). The *value* Value is 925997608 (0x37319A28).

Going through this exercise with various data elements, some general characteristics about the data elements used in this analysis can be derived. Most data elements are organized into tables and the OIDs of these data elements are 15 to 16 bytes long. Unlike globalTime, most data elements are defined as INTEGERs that, in the traffic signal controller application, have a value between 0 and 255. Most values are therefore expressed in one or two bytes. Compared to the example of globalTime, a typical identity would 2-3 bytes longer and the value would be 2-3 bytes shorter. The average binding for an individual data element is therefore 23 bytes, as shown in Figure 25. The fixed overhead of SNMP messages that are used to get or set one or more data elements is 26 bytes, as shown.

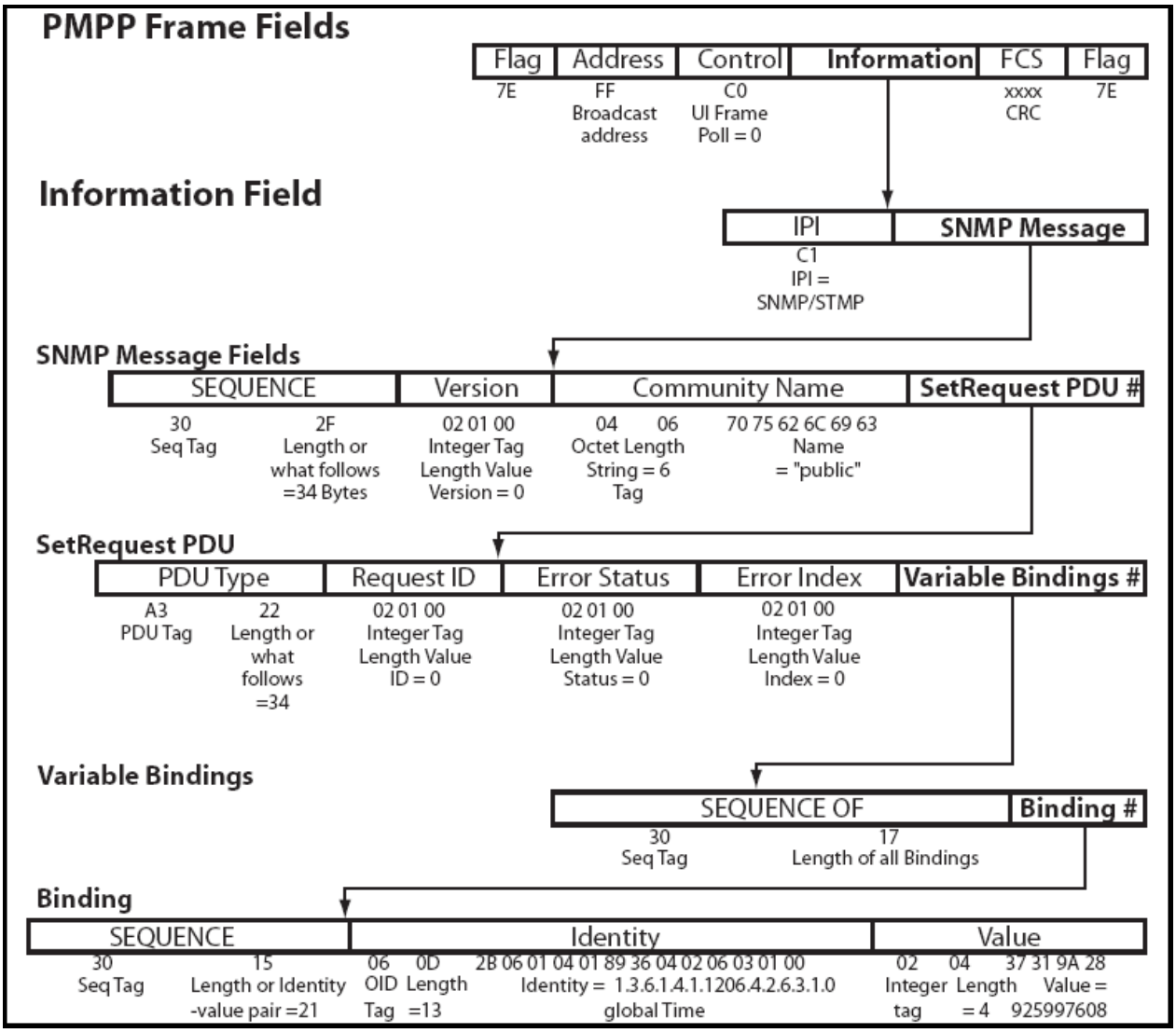

Figure 25 Set Time Operation Using SNMP Over PMPP

NOTE—Possible error in figure's SNMP Length, reported that Length should be 47 bytes, not 34 bytes.

E.2.3 ASN.1 Data Element Format and OID Decomposition

NTCIP data dictionaries follow a consistent structure, known as an OBJECT- TYPE Macro developed by the Internet Community using Abstract Syntax Notation One (ASN.1). This structure defines data elements using a variety of descriptive fields. The macro is an existing well-accepted standard for describing data, and the NTCIP effort adopted it as the descriptive language of choice.

```
dmsNumPermanentMsg  OBJECT-TYPE
  SYNTAX  INTEGER (0..65535)
  ACCESS  read-only
  STATUS  mandatory
  DESCRIPTION
  "<Definition> Indicates the current number of Messages
  stored in non-volatile, non-changeable memory (e.g.,
  EPROM). For CMS and BOS, this is the number of
  different messages that can be assembled.
```



```
  See the Specifications in association with
  Requirement 3.5.6.1 to determine the messages that
  must be supported.
  <Unit>message
  <Object Identifier> 1.3.6.1.4.1.1206.4.2.3.5.1"
::= { dmsMessage 1 }
```

The example data element is typical of such elements found in the NTCIP standards. This example identifies the number of messages held in non-volatile memory. The name of the data element is **dmsNumPermanentMsg**. The macro structure also describes the syntax of the data element. In this case, **dmsNumPermanentMsg** is a two-byte integer with a designated range between 0 and 65,535.

Information as to how this data element is to be accessed is also provided. For this example, read-only access indicates that the management station is not allowed to write to the data element. A data element with read-write access would indicate that the data element could be used to either read values from a database or write values to a database, while other options include not-accessible. The macro status field indicates whether the data element is mandatory or optional. Individual NTCIP standards, and their PRLs, should be reviewed to determine appropriate conformance requirements. The description field provides a clear and unambiguous definition of the intended use of the data element. In this case, the **dmsNumPermanentMsg** data element "Indicates the current number of Messages stored in non-volatile, non-changeable memory…" The last macro field indicates both the "parent" group and the number assigned to this "child" (data element) of the parent. The parent–child numbering scheme follows a tree structure for uniquely identifying data elements. See Figure 26.

Object Identifier 1.3.6.1.4.1.1206.4.2.3.5.1 is for the dmsNumPermanentMsg Data Element											
The above OID decomposes as follows:											
1	3	6	1	4	1	1206	4	2	3	5	1
iso	org	dod	internet	private	enterprise	nema	transportation	devices	dms	dmsMessages	dmsNum Permanent Msg

Figure 26 Example—Object Identifier (OID)

Decomposition of the OID shows exactly where the **dmsNumPermanentMsg** data element can be found on the ISO "tree". All NTCIP data elements are under the NEMA node on the ISO "tree". NEMA has identified four nodes under its control, described as:

a) **mgmt(1)—**The mgmt(1) subtree is used to identify data elements which are defined in NEMA-approved documents.
b) **experimental(2)—**The experimental(2) subtree is used to identify data elements used in NEMA experiments. This is where new MIBs are placed prior to being assigned to the transportation node.
c) **private(3)—**The private(3) subtree is used to identify data elements defined unilaterally. Enterprise specific data is defined under the private node.
d) **transportation(4)—**The transportation(4) subtree is used by NEMA specifically for different classes of transportation equipment.

NTCIP standards are represented under the transportation node. Under the Transportation node there are **protocol(1)**, **devices(2), tcip(3), tmdd(4),** and **adus(5)** subtree structures. The devices group has additional subtrees for each of the supported device data dictionaries: actuated signal controllers, ramp meters, dynamic message signs, closed circuit television, environmental sensor stations, globals, etc. New branches are added to the "tree" structure when new devices are included in the NTCIP family of standards.

E.2.3.1 STMP Application Message Bits and Bytes

The actual bits and bytes of an STMP message are defined using OER, as described in NTCIP 1102:2004. Because the content of an STMP message is defined within both the sending device and the receiving device prior to being sent, it is possible to eliminate a number of fields and reduce overhead significantly. OER starts out with the *Tag-Length-Value* representation method used by SNMP. However, if the *Tag*, *Length*, or *Value* is known, the component is eliminated. If a data element is always an INTEGER, the fact that it is an INTEGER is not sent (*tag*). If a data element is always 2 bytes long, the fact that it is 2 bytes long is not sent (*length*). Since all data are expressed as a SEQUENCE, all SEQUENCE *Tags* and SEQUENCE *Lengths* are eliminated as well. This all boils down to the fact that only the *Value* of a data element is sent. However, if the SYNTAX of a data element indicates that the value can be of variable length such as INTEGER (as opposed to INTEGER (0…255)), then the length could be either 1 byte, 2 bytes, 3 bytes or 4 bytes requiring the sending device to indicate the number of bytes used to transmit the value.

In Figure 25, the **Binding** for globalTime consisted of Sequence, Length and Value of the Identity and Value pair where the Identity and Value pair were each encoded as *Tag- Length-Value*. The **Binding** in STMP would be the Value-*Value* or simply 0x37 31 9A 28 as shown in the last row of Figure 27.

NOTE—The same principle of eliminating any known *Tag*, *Length*, or *Value* was applied to the **SNMP Message Fields, Set Request PDU, and Variable Bindings** fields in Figure 26. This resulted in the one-byte **STMP Message Fields and Set Request PDU** field shown in Figure 27.

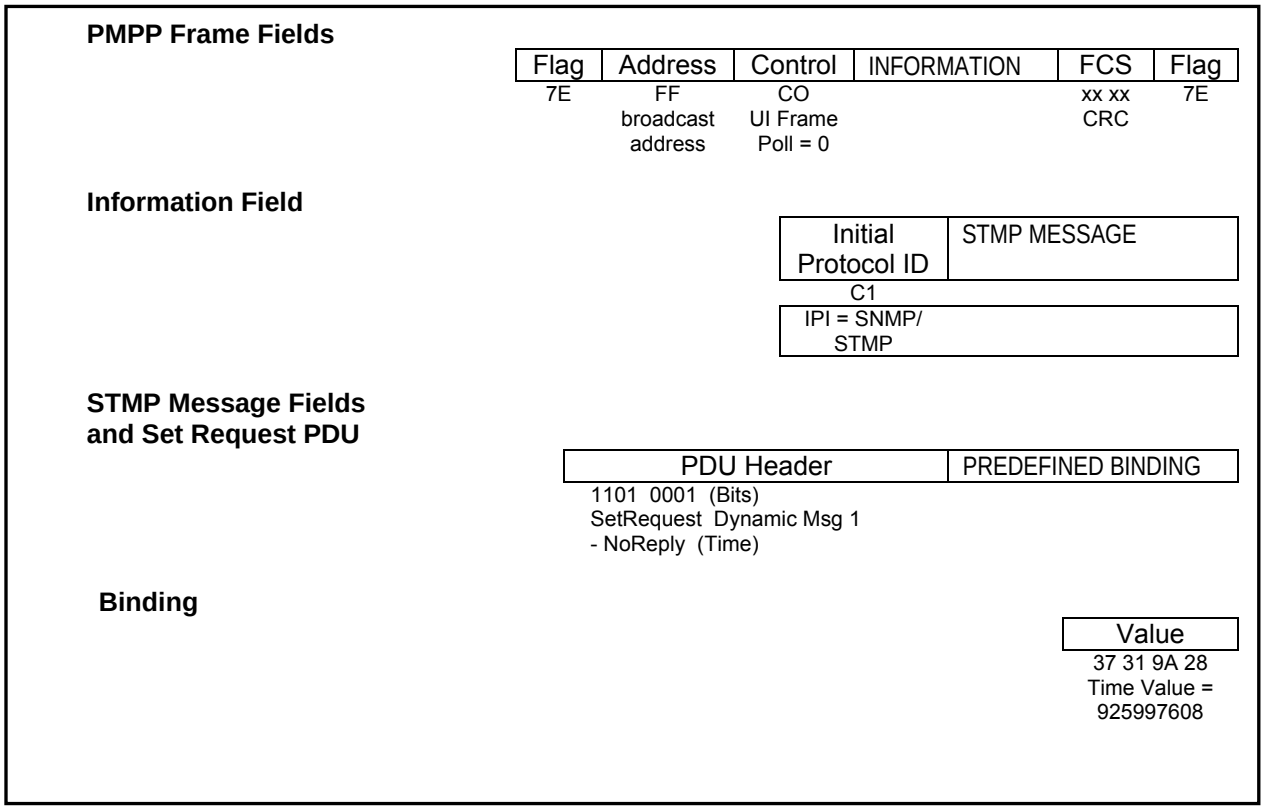

Figure 27 Set Time Operation Using STMP Over PMPP

E.2.4 Estimate—Application Message Exchanges

In any exchange of messages, one has to consider the size of both the command and response. In SNMP, the size of the command and response are approximately the same. In STMP, the size of the command and response are very different. Details follow.

E.2.4.1 SNMP Application Message Exchange Sizes

Using the SNMP rule of thumb, one can estimate the size of an SNMP message and exchanges by the number of data elements that are contained in them. Table 15 summarizes the messages in the example, how many data elements are in the message, the command size in bytes and the size of an exchange.

Table 15 Example—Message Size

SNMP Message Overhead	Data Elements	Command/Response Size	Exchange Size
Date and Time	1		98
Intersection Map Data	10		512
Pattern Command	3		190
Detector Data	5		282
Detailed Status	2		144
Upload Download	1 + bytes	59 – 177	118 - 254
Tuning	5		282

In SNMP, typical commands and the responses have about the same number of bytes. A getRequest command contains placeholders for values that would be contained in a response to it. In a setRequest, the values of data elements that are to be set are contained in the set command. In the corresponding setResponse, the same values would also be included to indicate what the data elements were actually set to. Therefore, in an SNMP exchange, the number of bytes is equal to two times the message size.

E.2.4.2 STMP Application Message Exchange Sizes

The rule of thumb for estimating STMP message size is 1 byte + 1 byte per data element. To estimate message exchanges, however, one has to understand that only a command or response contains the value(s) of any associated data element(s). To eliminate as much overhead as possible, a management application can also send a command where no reply is necessary. An STMP getRequest, getNext and setResponse do not contain any data element values. An STMP setRquestNoReply does not return any response. Table 16 summarizes the typical command and responses.

Table 16 Typical Command Responses

Command	Response
getRequest	GetResponse + value
getNext	GetResponse + value
setRequest + value	SetResponse
setRequestNoReply + value	[no response]

Table 17 shows what commands are used to set or get the data elements, and lists the size for each command and response. By summing the size of the command and response, the size of the message exchange can be derived.

Table 17 Derivation of STMP Message Exchange Sizes

Dynamic Message	Command	# of Data Elements	Size		
			Command	Response	Exchange
Date and Time	setRequestNoReply	1	2	-	2*
Date and Time	getRequest	1	1	2	3*
Intersection Map Data	getRequest	10	1	11	12
Pattern Command	setRequestNoReply	3	4	-	4
Detector Data	getRequest	9	1	10	10
Detailed Status	getRequest	2	1	3	3
Upload Download	N/A	-	-	-	-
Tuning	N/A	-	-	-	-
* NOTE—As with any rule of thumb, it does not always apply. The actual sizes are 5 and 6, respectively.					

Setting Date and Time and Pattern Command is handled with the setRequestNoReply commands. Retrieving of Date, and Time is handled with a getRequest command and getResponse reply. The Intersection Map Data, Detector Data and Detailed Status are handled with the getRequest command and getResponse response. Several Upload Download messages could, in theory, be defined as Dynamic Objects. However, the limited number of definable Dynamic Objects (13) tends to preclude this. Some implementations may specifically prohibit this, as well. The Tuning message, while easily defined in SNMP, cannot be predefined in STMP because various phases and patterns would have to be indexed.

E.2.5 Estimate—Transport and Subnetwork Protocol Size

The PMPP and PPP share a common header structure that has six fields associated with it. These fields consist of starting flag, address, control, information, checksum and a closing flag. The address field in PMPP is typically one byte, but could be extended. The address field in PPP is always 0xFF and is one byte. The fields are illustrated in Figure 27, in the **PMPP Frame Fields** row.

The first field in the **Information Field** indicates the next higher-level protocol to process the information. This field is referred to as the Initial Protocol Identifier (IPI). For non-networked communications, a “null” or no transport or network protocol is used. The IPI in this case is 0xC1 and indicates that the information should be passed directly to SNMP or STMP.

One particular facet of PMPP that may come into play, but is not factored into the rules of thumb, is byte stuffing. Byte stuffing ensures that the opening and closing flags are unique in any exchange. Any value of 125 (0x7D) or 126 (0x7E) occurring between the two flags is padded with an additional byte. In this way, reception of a Flag (0x7E) uniquely identifies the beginning or ending of an HDLC frame. PPP also uses the byte stuffing technique, but extends it to cover any value between 0x00 and 0x1F. On average, byte stuffing adds 1% overhead or 1 byte for every one hundred transmitted. It is very likely that, in the future, field devices are expected to support truly networked communications. Messages and exchanges could be routed from workstations on a local area network through a communications server or field processor to a device.

In this scenario, the Internet UDP/IP Protocols would be used. Figure 28 illustrates all the typical fields and values used in sending a set globalTime message over a typical office environment, network communications stack. The message is sent via the SNMP Application Profile over a UDP/IP Transport Profile over an Ethernet Subnetwork Profile. The use of UDP/IP has an overhead of 28 bytes. A typical Ethernet Frame has an overhead of 24 bytes. It is also possible to send STMP over UDP/IP over Ethernet. Figure 29 illustrates the same globalTime message sent via the STMP Application Profile over the same transport and subnetwork.

NOTE—The only difference in transport and subnetwork layers is the value of the Destination Port in the UDP Header. SNMP uses the value 161 (0x00 A1) and STMP uses the value 501 (0x01 F5).

The term UDP/IP may be unfamiliar to transportation personnel. However, if a computer supports TCP/IP or the Internet Protocol Suite, it supports UDP/IP, as well. What this means, for example, is that a message that is meant to set the time-of-day in a variable message sign can be generated by and routed through the computers involved in C2C communications. The use of UDP/IP over Ethernet also typifies a real implementation. A traffic signal controller and dynamic message sign system in Toronto, Canada uses SNMP over UDP/IP over a mix of Subnetwork technologies. The current Advanced Transportation Controller, Model 2070-type field controller may use a 10 Mbps Fiber Optic Ethernet Subnetwork, as an example.

Ethernet Frame Fields

Preamble	Destination Address	Source Address	Type or Len	Data	FCS	IFG
8 Bytes	08 00 90 03 4C F1 Bay Net 034CF1	08 00 20 09 00 C8 3Com 0900C8	08 00 IPI = IP		4 Bytes CRC	96 Bytes

Information Field

IP Header	UDP Header	SNMP Message

IP Header

Ver	IHL	Type of Service	Total Length		45 00	00 7C
Identifier			Flags	Fragment Offset	4E 57	00 00
Time to Live		Protocol	Header Checksum		3C 11	A5 C5
Source Address					84 A3	80 04
Destination Address					84 A3	01 0A

UDP Header

Source Port	Destination Port	0C A2	00 A1
Length	Checksum	00 39	00 00

SNMP Message Fields

SEQUENCE		Version	Community Name	SetRequest PDU
30 Seq Tag	2F Length = 47 (# bytes that follow)	02 01 00 Int Len Value Tag = 1 = 0	04 06 70 75 62 6c 69 63 Octet String Tag Length = 6 Name = "public"	

SetRequest PDU

PDU Type		Request ID	Error Status	Error Index	Variable Bindings
A3 PDU Tag	22 Length = 34 (# bytes that follow)	02 01 00 Int Len Value Tag = 1 = 0	02 01 00 same as Request ID	02 01 00 same as Request ID	

Variable Bindings

SEQUENCE OF		**VarBinds**
30 Seq Tag	17 Length = 23 (# bytes in all following Bindings)	

Variable Binding

SEQUENCE		Identity	Value
30 Seq Tag	15 Length = 21 (# bytes in this Binding)	06 0D 2B 06 01 04 01 89 36 04 02 06 03 01 00 OID Length Identity = 1.3.6.1.4.1.1206.4.2.6.3.1.0 Tag = 13 globalTime	02 04 37 31 9A 28 Integer Length Value = Tag = 4 925997608

Figure 28 Set Time Operation Using SNMP Over UDP/IP/Ethernet

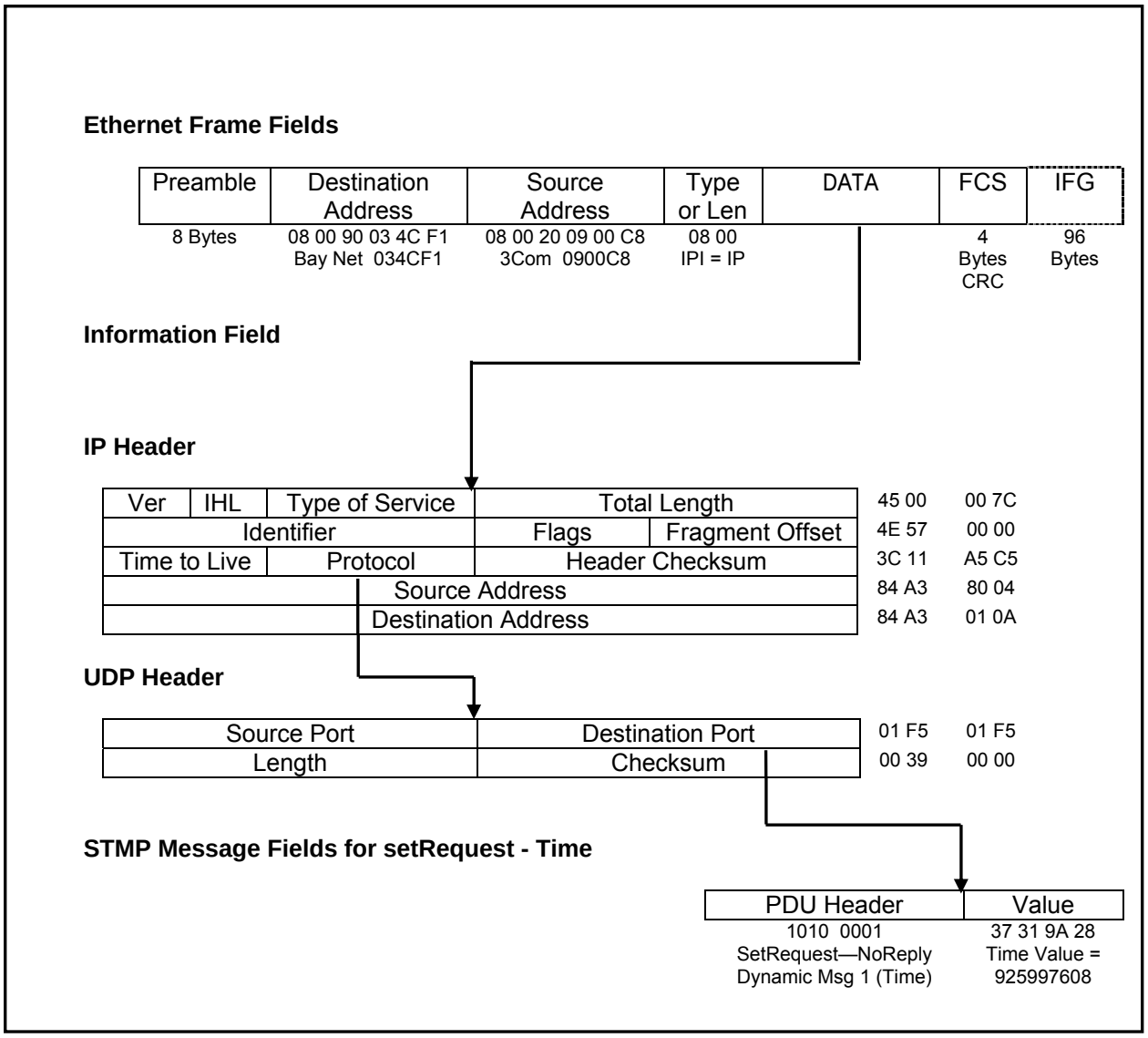

Figure 29 Set Time Operation Using STMP Over UDP/IP/Ethernet

Table 18 summarizes the overhead for the various transport and subnetwork protocols.

Table 18 Overhead Estimates

Transport and Subnetwork Protocol	Overhead per Message	Overhead per Exchange
Null over PMPP	7	14
Null over PPP*	7	14
UDP/IP over PMPP	35	70
UDP/IP over Ethernet	54	108
* NOTE—After the dial-up session has been established. Otherwise, these values would be higher.		

E.2.6 Estimate—Timing Factors

In performing a bandwidth analysis, numerous timing factors come into play. Processing delays, modem response, and duplexing mode may need to be considered. One can intuitively understand that the response to a command asking for 100 data elements takes longer to process than one that only asks for 1 data element. Once a message is received, the device needs to parse it to understand what is being asked for or what is being sent. Once it understands what, the device then either gathers data or store away data. It is very important to understand that processing delays vary according to message content and implementation. For the sake of this analysis, a processing delay value of 50 ms is used. Figure 30 shows a graphical representation of the various timing factors to be considered.

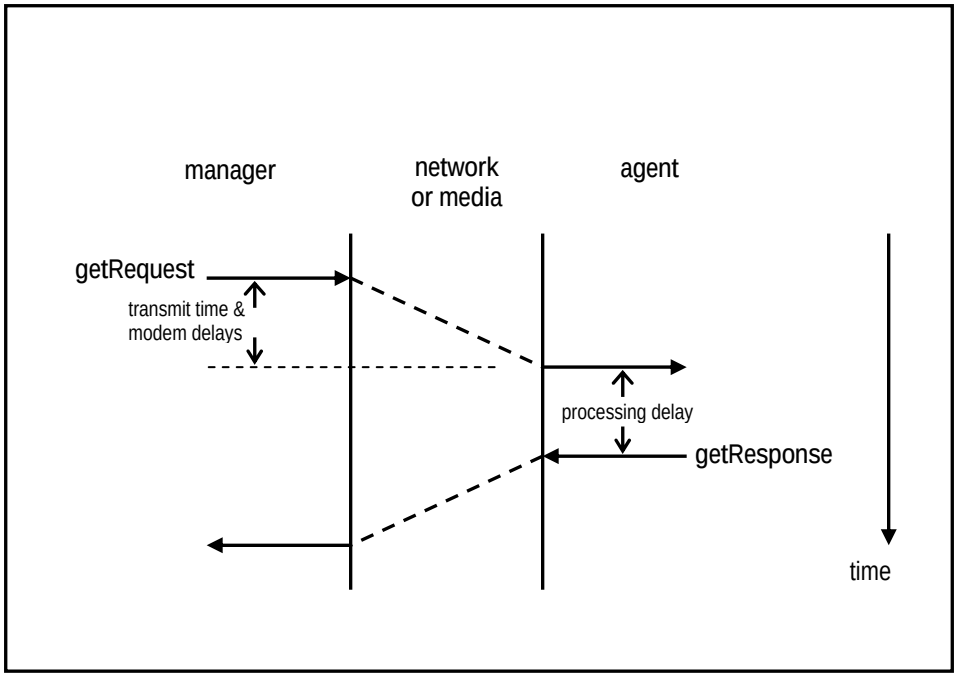

Figure 30 Timing Factors

Modem detect and turnoff delays can be as high as 2 or 3 seconds. The typical 56 Kbps modem that one might use to log into the Internet takes several seconds to “train” or adjust to the line characteristics. These may be suitable for Point-to-Point operation, but in a multi-drop environment, a “fast” turn-on/turn-off type modem is the only practical choice. For the sake of this analysis, it is assumed that a “fast” modem is be used and that the turn-on and turn-off delays are on the order of 10 ms each.

The duplexing mode of operation can have a significant impact on timing. In full-duplex mode, commands and responses can overlap, as shown in Figure 31. A second command can be sent while the response to a previous one is being received. In half-duplex mode, a second command cannot be sent until the response to the first is received. Full-duplexing can effectively cut the data rate requirements significantly. Table 19 summarizes the delays that are used in subsequent discussion.

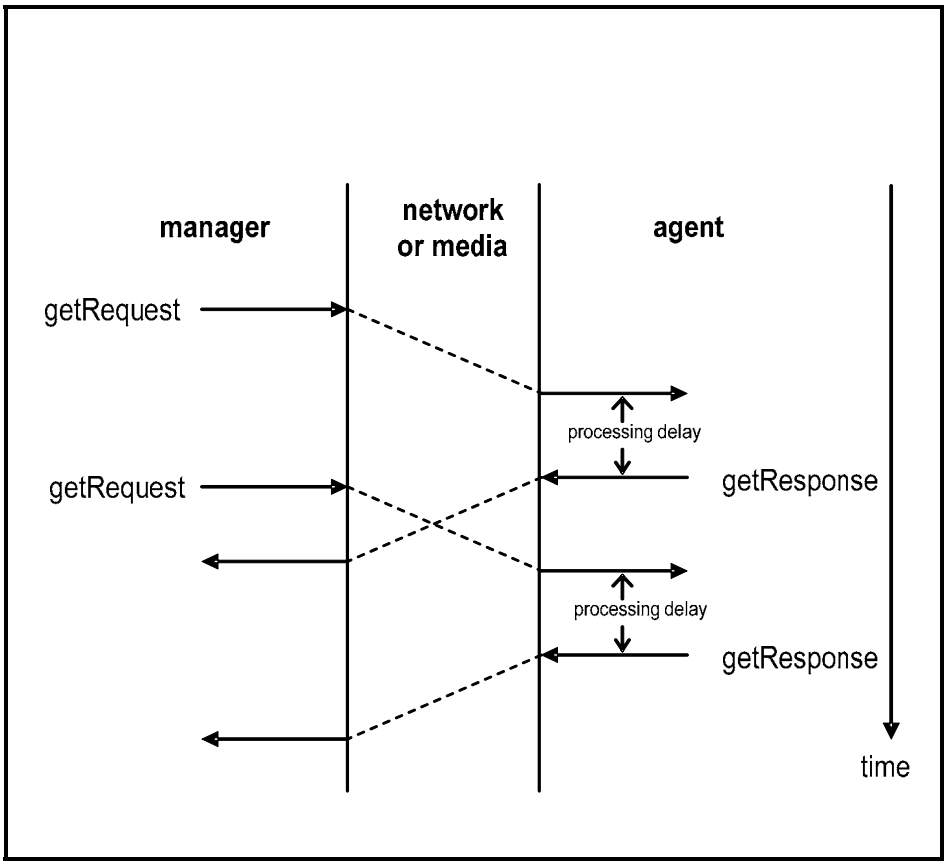

Figure 31 Full Duplexing

Table 19 Delay Estimate

Delays	Time
Modem Carrier Turn-on	10 ms
Modem Carrier Turn-off	10 ms
Processing Delay	10 ms
Total	30 ms

E.2.7 Modems

C2F communications have traditionally used 1200 bps FSK modems for wireline communications. These modems come in different versions for different applications, such as half- or full-duplex, leased telephone lines or agency-owned twisted pair cable, internal or external to the device, support TIA-232-F flow control or not. In analyzing bandwidth requirements for copper communications plant and attempting to increase modem bit rate, it is important to remember that not all modems and bit rates may be practical for a given implementation environment. In particular, the following issues should be considered:

a) Consumer modems used for general-purpose computer communications, for example, V.90 56 kbps, cannot be used in multi-drop field implementations because they are too slow to reach ready state prior to each transmission, requiring “training” time.
b) Consumer modems and modems designed for indoor use may not operate reliably in the temperature and humidity extremes encountered in field applications.

c) The fastest modems currently available for agency-owned twisted pair multidrop applications operate at up to 28,800 bps.
d) Currently, modems suitable for multi-drop operation over leased telephone lines cannot support 9600 bps unless “a metallic circuit” is provided.
e) There is a limit to the distance that a modem can operate on agency-owned twisted pair cable. The maximum distance is reduced as the bit rate increases, and as the number of devices on the channel increases.
f) There is no distance limit on leased telephone lines.
g) Modems from different manufacturers can vary greatly in their features and operational characteristics. A thorough test of the actual modem planned for use (in a real-world long distance multi-drop environment) should be made before committing to its use. Many vendors provide a list of modems that have been tested to work with their equipment, but this does not ensure that different modems can be mixed on the same channel.
h) Some field devices do not yet have a 9600 bps internal modem option
i) Modems that are external to the field device (connected by a serial cable) require a dedicated suitable TIA-232-F port on the field device, in addition to any serial port(s) used for other purposes, for example, laptop computer connection.
j) Asynchronous modems add a start and at least one stop bit for every byte (8 bits) transmitted; synchronous modems do not. This equates to a 25% increase in overhead.
k) Some processing devices have a limited capability to process the data handed it from the modem. Some legacy equipment isn’t powerful enough to process at 9600 bps.

There are similar but different lists of constraints and considerations for modems or transceivers for other types of plant such as fiber and radio.

E.2.7.1 Electrical Limitations

The maximum number of modems on a wireline channel due to electrical limitations is an issue of modem sensitivity, the desired signal-to-noise ratio for a given bit-error-rate, and the characteristics of the channel and interface. The *Communications Handbook for Traffic Control Systems* provides an example of the characteristics and calculations for a 1200 bps FSK multi-drop system. The handbook goes into more detail than what is presented here and covers other technologies such as wireless and fiber optic, and in that example, the calculations for any modem technology that uses a wireline (twisted pair) medium would apply. The formula for determining maximum number of drops is:

Number of Drops = (Sensitivity - Cable Loss - S/N Ratio)/Insertion Loss per Drop

Table 20 shows hypothetical characteristics of a 9600 bps modem and wiring that is to be used in a multi-drop configuration.

Table 20 Modem Parameters

Parameter	Value
Modulation Technique	Some type of Phase and Amplitude Modulation
Operation Mode / Line	Full Duplex / 4 wire (metallic or user owned)
Modem Frequencies	Center Frequency ~ 9600 Hz
Receiver Sensitivity	0 dBm to -39 dBm
Signal-to-Noise Ratio	15 dB for a Bit-Error-Rate of 1×10^{-5}
Cable Loss	3.3 dB/mile at 9600 Hz for 19 AWG
Insertion Loss	0.5 dB per drop
Distance	6 miles

The modulation technique used by a modem may not always be FSK. Phase Shift Keying (PSK) and Quadrature Amplitude Modulation (QAM) are typically used to increase throughput without necessarily increasing signaling frequency. The operating mode of the 9600 bps modem is assumed to be full duplex using two wire pairs (four wires). This configuration minimizes distortion from line reflections and the turnaround times associated with switching from transmit to receive. The signaling frequencies are assumed to be 9600 Hz. The signaling frequencies vary with modulation techniques but are quantified because they determine cable losses. Receiver sensitivity is an indication of how well a modem is at picking up weak signals and signal-to-noise ratio is the ability of a modem to pick out a signal with back ground noise (static). Insertion loss comes from a vendor's data sheet. Some type of loss is always associated with the connection to the channel. Usually this is due to slight impedance mismatches and physically routing the signal through a connector to the modem electronics. For a given signaling frequency, the size of the interconnect wire defines how much signal is lost over some distance. The cable loss is derived. The distance from the primary to the farthest secondary is assumed to be 6 miles.

Using the preceding formula, the calculations are:

Number of Drops = (Sensitivity - Cable Loss - S/N Ratio)/Insertion Loss per Drop
Number of Drops = (39 dBm - (8 miles x 3.3 dB per mile) - 15 dB)/.5 dB
Number of Drops = (39 dBm - 19.8 dB - 15 dB)/.5 dB
Number of Drops = 8.2

Rounding down the value, the maximum number of drops for this example is 8, from a purely electrical point of view.

E.2.7.2 Communications Limitations

The preceding discussion shows how to calculate the electrical limitations using modem and wireline techniques. It does not address the logical aspects of organizing a system into communications channels, defining what information is sent on each channel, or ensuring each channel can carry the desired information. Procedures and calculations to define the number of channels and drops per channel based on the message exchange requirements follow.

For this discussion, size of the message exchanges and how often they occur are shown in Table 21.

Table 21 Message Frequency and Size

Message Exchange	Frequency	SNMP Exchange (bytes)	STMP Exchange (bytes)
Date and Time	1 per day—all	98	2
Intersection Map Data	1 per second X 24 intersections	512	12
Pattern Command	1 per minute—all	190	4
Detector Data	1 per minute X 8 intersections	282	7
Detailed Status	1 per hour X 8 intersections	144	4
Upload Download	1 per day X 24 intersections	118—254	N/A
Tuning	2 per day X 24 intersections	282	N/A

Transport and Subnetwork Protocol Overhead estimates are shown in Table 22.

Table 22 Protocol Overhead Estimates

Transport and Subnetwork Protocol	Exchange Overhead (bytes)
Null over PMPP	14
Null over PPP	14
UDP/IP over PMPP	70

Processing and Modem Delay estimates are shown in Table 23.

Table 23 Delay Estimates

Delays	Time
Modem Carrier Turn-on	10 ms
Modem Carrier Turn-off	10 ms
Processing Delay	10 ms
Total	30 ms

The processing delay of traffic signal controllers is highly variable, with actual response times varying from 10 ms to 50 ms. To achieve response times at the lower end of this range, as shown in Table 23, techniques such as asynchronous messaging, whereby the response to a request is placed in a buffer for immediate transmission at the next poll, may be needed.

E.2.8 SNMP Timing

At this point, specific protocols to analyze, and then normalize, the data exchanges and delays to some common time interval should be chosen. For SNMP over Null over PMPP we have the values shown in Table 24.

Table 24 Normalized Data Using SNMP Over NULL Over PMPP for 24 Drop per Channel

Message Exchange	Frequency	Message Exchange Size Bytes	Messages per day	Bytes per day
Date and Time	1 per day—all	112	24	2688
Intersection Map Data	1 per second X 24 intersections	526	2073600	1090713600
Pattern Command	1 per minute—all	204	1440	293760
Detector Data	1 per minute X 8 intersections	296	11520	3409920
Detailed Status	1 per hour X 8 intersections	158	192	30336
Upload Download	1 per day X 24 intersections	268	24	6432
Tuning	2 per day X 24 intersections	296	48	14208
Totals per day			2,086,848	1,094,470,944

Normalizing the bytes per day to bps, we calculate the total system bandwidth required as:

1,094,470,944 bytes per day x 10 bits per byte / 86,400 seconds per day = 126,675 bits in one second

The value 126,675 is the average number of bits that are required to transmit all message exchanges to and from the 24 intersection controllers in one second.

Given a specific modem speed, we can calculate a first order approximation of the number of drops per channel and channels. For a 9600 bps modem, this is calculated as:

126,675 bits overall / 9600 desired bits per channel = 13.19 channels

and

24 intersections / 13.19 channels = 1.82 drops per channel

Rounding both of these figures downward, the use of 9600 bps modems would only work if there was a single modem and drop dedicated to each intersection. There are some systems that use a dedicated modem and cabling arrangement. If this is feasible, a second order approximation should be performed. A second order approximation accounts for the changes in the number of drops and the need to send broadcast messages on each channel. The impact of any delays should also be considered. Since the steps are same for any protocol combination, only the reiterations for STMP over NULL over PMPP using 1200 bps modems are illustrated. Additionally, it is assumed that no errors result in a need for retransmission.

Clearly, SNMP is not a feasible protocol choice unless each signal has a dedicated channel. In practice, such signal systems use STMP.

E.2.9 STMP Timing

The STMP and PMPP Protocols were designed to be open to address diverse communications needs, yet be very efficient to meet the limited bandwidth capability of current systems. A typical traffic control system of today that controls 24 intersections uses a proprietary communications scheme, is usually configured as two groups of 12 intersections or three groups of eight intersections, and uses internal 1200 bps FSK modems. Conversion of these systems to NTCIP may impact the existing configuration and require the use of higher speed modems. The following calculations for STMP over Null over PMPP should help understand the potential impact.

For this NTCIP Stack, Table 25 summarizes the messages, frequency and sizes of the message exchanges. As before, it is assumed that the system has 24 intersections on a single drop, and normalize the total exchanges to a per day basis.

Table 25 Normalized Data Using STMP Over NULL Over PMPP for 24 Drops per Channel

Message Exchange	Frequency	Message Exchange Size Bytes	Messages per day	Bytes per day
Date and Time (set)	1 per day—all	16	1	16
Date and Time (get)	1 per day X 24 intersections	17	24	408
Intersection Map Data	1 per second X 24 intersections	26	2,073,600	53,913,600
Pattern Command	1 per minute—all	18	1,440	25,920
Detector Data	1 per minute X 8 intersections	21	11,520	241,920
Detailed Status	1 per hour X 8 intersections	18	192	3,456
Upload Download	1 per day X 24 intersections	N/A	N/A	N/A
Tuning	2 per day X 24 intersections	N/A	N/A	N/A
Totals per day			2,086,777	54,185,320

Normalizing the bytes per day to bps:

54,185,320 bytes per day x 10 bits per byte / 86,400 seconds per day = 6271.45 bps

The value, 6271.45 bps, is the average data rate required to transmit all message exchanges to and from the 24 intersection controllers.

For this protocol configuration, let's try a 1200 bps modem. As before, calculate a first order approximation of the number of drops per channel and channels by dividing the number of averaged data rate value by the modem speed. For a 1200 bps modem, this would work out to drops per channel using channels.

These values seem reasonable, so perform a second order approximation. For the iteration shown in Table 26, a configuration of 6 channels of 4 drops per channels is used. The frequency of some of the messages is adjusted accordingly.

Table 26 Normalized Data Using STMP Over NULL Over PMPP for 4 Drops per Channel

Message Exchange	Frequency	Message Exchange Size Bytes	Messages per day	Bytes per day
Date and Time	1 per day—all	16	1	16
Date and Time	1 per day X 4 intersections	17	4	68
Intersection Map Data	1 per second X 4 intersections	26	345,600	8,985,600
Pattern Command	1 per minute—all	18	1,440	25,920
Detector Data	1 per minute X 2 intersections	21	2,880	60,480
Detailed Status	1 per hour X 1 intersections	18	24	432
Upload Download	1 per day X 1intersections	N/A	N/A	N/A
Tuning	2 per day X 24 intersections	N/A	N/A	N/A
Totals per day			349,949	9,072,516

Normalizing the bytes per day to bps:

9,072,516 bytes per day x 10 bits per byte / 86,400 seconds per day = 1050.06 bps

The value, 1050.06 bps, is the average data rate required to transmit all message exchanges to and from the 4 intersection controllers.

This appears to be a reasonable value, but delays should be taken into account. Any delays take away from the time that is available to actually transmit data. Total delay per second is calculated as:

349949 messages per day X .03 seconds delay per message/86400 seconds per day = 0.122 sec.

The value, 0.122 seconds, is the average delay per second. The modem speed to transmit 1050 bits in the remaining time is calculated as:

1050 bits / (1 sec. - 0.122 delay time) = 1195.90 bps

Therefore, 4 drops per channel at 200 bps is a suitable configuration. If more drops per channel are desired, the data rate can be increased to possibly 4800 or 9600 bps, by using faster modems.

E.3 CENTER TO FIELD (C2F) BANDWIDTH ALTERNATIVE ANALYSIS

In the previous scenario, the emphasis was on acquiring once-per-second data from all intersections. This may not necessarily apply to all situations. In applications other than traffic signal control, this would certainly not be the case. The following scenario addresses the same requirements as before except for map display. In this case, the map display is only for one intersection at a time. In general, this scenario might apply to cases real-time (once-per-second) data from a single device is needed, but system data can be exchanged on a once-per-minute or longer basis. The communications requirements are summarized as:

a) Synchronize the time and date in all field devices.
b) Provide a map display of the status of **one** intersection.
c) Control the overall timing pattern to be put into effect.
d) Control the operation of 2 lane-closed signs.
e) Monitor all intersections for any abnormal conditions.
f) Accumulate volume and occupancy data for 16 detectors to perform off-line optimization.
g) Provide full upload and download of the complete database or programming data in each field device.
h) Support 24 intersections.

E.3.1 Estimate—Message Exchanges and Frequency

In this scenario, all the previous messages are used but a new one is added. Since the map display information is only gathered from one intersection, the status of the other intersections is monitored by an "Intersection Status" set of data elements.

To provide indications of what coordination timing pattern is in effect and any abnormal condition at an intersection, the following data elements defined in NTCIP 1202:2005 are used:

a) systemPatternControl—Section 2.5.14
b) shortAlarmStatus—Section 2.4.9

These are the same data elements used in the map display. However, the intersection map display is to be gathered from only one intersection at a time. These data elements are used to monitor the other intersections in the system. For monitoring, these would be read from each intersection approximately once-per-minute.

Table 27 summarizes the new set of messages and how often they occur.

Table 27 Message Frequency Alternate Scenario

Message Exchange	Frequency
Date and Time	1 per day – all
Intersection Map Data	1 per second X 1 intersections
Intersection Status	1 per second X 23 intersections
Pattern Command	1 per minute—all
Detector Data	1 per minute X 8 intersections
Detailed Status	1 per hour X 8 intersections
Upload Download	1 per day X 24 intersections
Tuning	2 per day X 24 intersections

E.3.1.1 SNMP Application Message Exchange Sizes

Table 28 summarizes the messages in the new scenario.

Table 28 SNMP Message Sizes Alternate Scenario

SNMP Message Overhead	Data Elements	Command/Response Size	Exchange Size
Date and Time	1	49	98
Intersection Map Data	10	256	512
Intersection Status	2	72	144
Pattern Command	3	95	190
Detector Data	5	141	282
Detailed Status	2	72	144
Upload Download	1 + bytes	59—177	118—254
Tuning	5	141	282

The only addition is the intersection status message.

E.3.1.2 STMP Application Message Exchange Sizes

Table 29 is an update to Table 28. The only difference is the addition of the Intersection Status exchange.

Table 29 Derivation of STMP Message Sizes Alternate Scenario

Dynamic Message	Command	# of Data Elements	Size		
			Command	Response	Exchange
Date and Time	setRequestNoReply	1	2	-	2
Date and Time	getRequest	1	1	2	3
Intersection Map Data	getRequest	10	1	11	12
Intersection Status	getRequest	2	1	3	4
Pattern Command	setRequestNoReply	3	4	-	4
Detector Data	getRequest	9	1	10	10
Detailed Status	getRequest	2	1	3	3
Upload Download	N/A	-	-	-	-
Tuning	N/A	-	-	-	-

E.3.2 Other Estimates

In this example, the estimates for transport and subnetwork protocols remain the same. The timing factors and delays apply, as well.

E.3.3 Number and Size of Slots per Channel

The only new topic to be considered in the alternate example is the concept of communications slots. The number of communications slots per channel can best be thought of as the number of opportunities to communicate in any time period. It is not necessarily equal to the number of drops per channel. For example, assume that there are 8 drops per channel. If one needed to communicate with each drop once every minute, there could be 8 slots. The width of each slot in this case would be 7.5 seconds. An

arrangement of 60 slots, each 1 second wide, would be just as suitable if all exchanges took less than 1 second. If this were the case, 52 slots would be available for other uses.

In the alternate example, only one message exchange needs to take place on a once-per-second basis. All other exchanges take place on a once-per-minute, once-per-hour, or one-per-day basis.

If the once-per-second exchange could be completed in less than one-half second, and all of the other exchanges could each be completed in less than one-half second, the concept slotting arrangement could be used. There could be two slots one-half second wide. The first slot would be reserved for the once per second exchange. The second slot would be used to perform all the other exchanges but on a rotating basis.

E.3.4 Communications Drops (Drops per Channel)

At this point, we are ready to perform the timing analysis. As before, it is necessary to pick a specific NTCIP Communications Stack. In the following examples, the transport and subnetwork protocols are assumed to be T2/Null and PMPP. Table 30 summarizes the frequency and all overhead associated with message exchanges.

Table 30 Message Frequency and Size

Message Exchange	Frequency	SNMP Message Exchange Bytes	STMP Exchange Bytes
Date and Time	1 per day—all	112	15
Intersection Map Data	1 per second X 1 intersections	526	26
Intersection Status	1 per second X 23 intersections	158	17
Pattern Command	1 per minute—all	204	18
Detector Data	1 per minute X 8 intersections	296	21
Detailed Status	1 per hour X 8 intersections	158	18
Upload Download	1 per day X 24 intersections	268	N/A
Tuning	2 per day X 24 intersections	526	N/A

Since “system requirements” include communications to 24 intersection controllers, the use of the PMPP and “fast” modems would allow multiple secondary devices to share a communication link. However, the characteristics of wire and the distances between intersections may place an upper limit on how many devices can share a communications link. For the sake of this analysis, the maximum number of drops is assumed to be 9. This allows a management application and 8 intersections to share a channel. An assumption is made that the network of 24 intersections is to be organized into 3 drops of 8. The revised message frequency and size per channel is shown in Table 31.

Table 31 Message Frequency and Size

Message Exchange	Frequency	SNMP Message Exchange Bytes	STMP Exchange Bytes
Date and Time	1 per day—all	112	15
Intersection Map Data	1 per second X 1 intersections	526	26
Intersection Status	1 per second X 7 intersections	158	17
Pattern Command	1 per minute—all	204	18
Detector Data	1 per minute X 2 intersections	296	21
Detailed Status	1 per hour X 1 intersections	158	18

Message Exchange	Frequency	SNMP Message Exchange Bytes	STMP Exchange Bytes
Upload Download	1 per day X 8 intersections	268	N/A
Tuning	2 per day X 8 intersections	526	N/A

E.3.5 SNMP Timing

Next, exchanges are analyzed to gauge the impact of multiple secondary devices and the frequency of the exchanges. In this example, the assumption is that the user wants to see the Intersection Map Data with some accuracy. Two back-to-back samples of the signal display would not be the same as two spaced exactly one second apart.

Considering this, the Intersection Map Data exchange should take place once every second on the second. Since all other exchanges take place on a minute, hour, or day basis, the Intersection Map Data could be requested every second and all other exchanges requested on a rotating basis. To do this, the Intersection Map Data and the largest other exchange needs to take place within one second. The largest exchange other than Intersection Map Data is the Tuning exchange. After summing up the size of the exchanges and processing delays, the required data rate is computed as:

*Data Rate = bytes * 10 / (1 - delay)*

[The numbers of bytes is multiplied by 10 because in asynchronous communications, a start and stop bit is added to each byte.]

Table 32 summarizes the overhead and delays associated with the Intersection Map Data and the Tuning message exchanges.

Table 32 SNMP Overhead and Delay Estimate Example

Slot	Exchange	Size (bytes)	Delays and Processing (ms)
1	Intersection Map Data	526	70
2	Tuning	526	70
Totals			140

This results in a requirement to transmit 1052 bytes in one second with 140 ms of delays. The required data rate is computed as:

*Data Rate = (1024 +28) *10 / (1 - 0.140) = 1233 bps*

This number is too high for readily available “fast” multi-drop modems, so a new approach is considered. Since Tuning exchanges are meant to change the timing characteristics of the system and need to be performed manually, we can assume that it does not have to run concurrently with the Intersection Map Data exchange. Considering the next largest exchange, Detector Data, provides the values in Table 33.

Table 33 SNMP Overhead and Delay Estimate Second Example

Slot	Exchange	Exchange Overhead (bytes)	Delays and Processing (ms)
1	Intersection Map Data	526	70
2	Detector Data	296	70
Totals			140

*Data Rate = (794+28) *10 / (1 - 0.180) = 10024 bps*

This value is still too high to consider 9600 bps modems. It is always a good engineering rule of thumb to have some margin for error. To gain some margin, one could consider the use of fiber optic modems. The carrier turn-on and turn-off delays could be significantly less than 20 ms. If an intersection did not have pedestrian movements, dropping them from the Intersection Map Data Exchange could reduce the data rate to 8341 bps. Minimizing carrier turn-on and turn-off delays could also be accomplished by use of a simple full-duplex operation. The primary's carrier is always on and any secondary turns on its carrier as soon as it recognizes that it needs to send a response. Table 34 summarizes the overhead and delays for a simple full-duplex operation arrangement.

Table 34 SNMP Overhead and Delay Estimate Example

Slot	Exchange	Exchange Overhead (bytes)	Delays and Processing (ms)
1	Intersection Map Data	526	60
2	Detector Data	296	60
Totals			120

*Data Rate = (526+296) *10 / (1 - 0.120) = 9341 bps*

Looking at the other exchanges, they could be mapped in to the second slot on a rotating basis as shown in Table 35.

Table 35 SNMP Command and Response Mapping to Spare Slots

Interval	Command and Response
1	Date and Time
2	Detector Data—Pair 1
3	Detector Data—Pair 2
4	Detector Data—Pair 3
5	Detector Data—Pair 4
6	Detector Data—Pair 5
7	Detector Data—Pair 6
8	Detector Data—Pair 7
9	Detector Data—Pair 8
10	Status—Intersection 1
11	Status—Intersection 2
12	Status—Intersection 3
13	Status—Intersection 4
14	Status—Intersection 5
15	Status—Intersection 6
16	Status—Intersection 7
17	Status—Intersection 8
18	Pattern Command—Intersection 1
19	Pattern Command—Intersection 2
20	Pattern Command—Intersection 3
21	Pattern Command—Intersection 4
22	Pattern Command—Intersection 5
23	Pattern Command—Intersection 6
24	Pattern Command—Intersection 7
25	Pattern Command—Intersection 8
26	Spare
…	…
30	Spare

Assuming a total of 30 slots, each of these exchanges would have a resolution of once every 30 seconds. The only concern in this arrangement is that a new pattern command might not be transmitted until 29 seconds after it was selected. Since there are spare time slots, one approach could be to send any new Pattern Command as it occurs. In this case, however, it would be sent to all intersections using a group address.

E.3.6 STMP Timing

The preceding example shows that while SNMP could be used for some real-time applications, it may require higher data rates than traditional 1200 bps, FSK modems support. For these applications, STMP

is more suited. The following illustrates how to calculate the bandwidth requirements for an STMP over Null over PMPP stack.

In this example, the messages are defined as dynamic objects. The OIDs of the data elements that comprise the messages are downloaded to Dynamic Objects 1 through 6 as follows:

Dynamic Object 1 = Time and Date
Dynamic Object 2 = Intersection Map Data
Dynamic Object 3 = Intersection Status
Dynamic Object 4 = Pattern Command
Dynamic Object 5 = Detector Data
Dynamic Object 6 = Detailed Status

Following the same strategy as in the alternate SNMP timing example, the Intersection Map Data and one of the other exchanges could be sent every second. Assuming Detector Data is the largest exchange to be handled in one second, we have the values shown in Table 36.

Table 36 STMP Overhead and Delay Estimate Example

Slot	Exchange	Exchange Overhead (bytes)	Delays and Processing (ms)
1	Intersection Map Data	26	60
2	Detector Data	21	60
Total			120

*Data Rate = 47 *10 / (1 - 0.120) = 534 bps*

Since this is well under the 1200 bps data rate that is typically available, we might want to consider sending a Pattern Command every second, as well. This would result in overhead and delays as shown in Table 37.

Table 37 STMP Overhead and Delay Estimate Second Example

Slot	Exchange	Exchange Overhead (bytes)	Delays and Processing (ms)
1	Intersection Map Data	26	70
2	Pattern Command	18	20
3	Detector Data	21	70
Total			160

*Data Rate = 65 *10 / (1 - 0.160) = 774 bps*

NOTE—The Delay and Processing for a Pattern Command is only 20 ms. Since this is sent as a setRequestNoReply, there should not be any internal processing required.

Since 774 bps provides plenty of margin, one could increase the amount of information being brought back in the Intersection Map Data. For example, if the intersection had a preemption sequence, the following could be added from NTCIP 1202:2005: preemptState—Section 2.7.2.16.

This data element could be used to indicate the state or interval of a preemption sequence. Another data element that could be added from NTCIP 1202:2005 is: phaseStatusGroupPhaseNexts—Section 2.2.4.11.

This would indicate the next vehicle phase that is to be serviced at the end of any currently timing phase. As in the SNMP example, the other exchanges could be mapped in to the third slot on a rotating basis as shown in Table 35.

E.4 CENTER TO CENTER (C2C) BANDWIDTH ANALYSIS

C2C communications typically involve communications networks connecting many computers in a peer-to-peer arrangement. These networks typically involve both local area networks, for example, within a building or adjacent buildings) and wide area networks, for example, across town or across the nation. Bandwidth requirements vary for each link in each network, depending on the amount of C2C messaging traffic using that link, and whether or not the network is shared with other applications. Multiplexers, routers, switches, hubs, and other devices are commonly used to manage, segment and optimize computer networks.

The typical subnetwork consists of a local area network adapter that operates at 10 Mbps. In an office environment, even 100 Mbps is readily available (most newly installed office networks use 100 Mbps). Point-to-Point dialup and dedicated external links run at a maximum of 56 Kbits-per-second (Kbps). Most important to a planner or implementer is that there are plenty of information resources available. In today's business environment, there is usually a person with strong computer skills or network administrator that can help understand and quantify bandwidth and allocations issues.

In a C2C environment, the computers that run the transportation applications are typically just users of the “network” or communication links. Other applications, such as e-mail, database management, graphics design, and word processing, may also be users of the network. This has a big advantage when it comes to design and implementation. A network specialist usually handles its design and implementation. However, they expect the transportation system designer or implementer to be able to quantify what demands are likely to be placed on the network.

NTCIP has adopted two application level protocols for C2C communications: DATEX and C2C XML. Both approaches provide the same basic functionality, but they differ in the method of implementation, and each has some unique features. The Internet Protocol (IP) and both UDP and TCP are used at the transport level for both of these C2C communication solutions. Regardless of the application level protocol, C2C communications requires participating systems to exchange standard messages at the information level.

NTCIP C2C protocols are used for two basic types of message exchange. The first type involves a human operator at a center requesting information on a one-time basis from another center. Since a human is in the loop, the volume of such messages is small, and they are unlikely to be critical in any network design. The other type of messaging occurs when an operator at a center sets up a permanent subscription for data to be sent from another center automatically. There is no human in the loop and messaging is often repeated. Such subscriptions may request the data to be sent every x seconds, or only when it changes. In most cases, network traffic is minimized if subscriptions specify change-based triggers rather than time based. However, the network designer should consider peak loading conditions of the network. Thus, even if change-based triggers are used, the designer should consider network loading when most or all of the triggers are activated near-simultaneously.

It may be difficult for a system designer to anticipate all the different types of data that may be subscribed for between each pair of centers. It is recommended that designers gather actual operating experience from existing C2C networks to help make such estimates. One important consideration is the frequency of change in status or other data at each center, since changes are what other centers are interested in monitoring. For example, a center that manages only incidents and related information is not likely to generate as much message traffic on the network as one that manages 200 traffic signals, each of which changes status every few seconds.

It is possible to perform a worst-case analysis by considering the frequency, or quantity per second, of useful information generation at each center, estimate that other centers have an interest in receiving that information, and assigning the message loads to network links accordingly.

For DATEX, each IP packet containing data (a publication message—the most common type) is likely to contain at least 70 bytes of overhead information (including the IP header), plus the actual encoded data. Most messages contain only a relatively small quantity of actual data—say 20 to 100 bytes. An average DATEX-generated IP packet might contain 150 bytes. At this rate, a full duplex 56 Kbps wide area network link could support in the order of 30 messages per second in each direction. This is sufficient for some centers, such as the incident management center, but may not be sufficient for others, such as the large traffic signal system, or a center that wants to obtain a lot of data from other centers.

Annex F
EXAMPLE—CYCLICAL REDUNDANCY CHECK (CRC) ALGORITHM AND CALCULATIONS

CRC is an error-detection technique consisting of a cyclic algorithm performed on each “block” of data at the sending and receiving end of the transmission. As each block is received, the CRC value is checked against the CRC value sent along with the block. The CRC Algorithm applies to NTCIP 2001, NTCIP 2101:2001, NTCIP 2102:2003, and NTCIP 2103:2005. (It is also used in IEEE 1570.)

The following C Source Code is an implementation of a table lookup algorithm for calculating and verifying the frame check sequence (FCS) value used in the HDLC and PPP Protocols (NTCIP 2103:2005):

```
#include <stdio.h>
typedef unsigned short int u16;

u16 fcstab[256] = {
0x0000, 0x1189, 0x2312, 0x329b, 0x4624, 0x57ad, 0x6536, 0x74bf,
0x8c48, 0x9dc1, 0xaf5a, 0xbed3, 0xca6c, 0xdbe5, 0xe97e, 0xf8f7,
0x1081, 0x0108, 0x3393, 0x221a, 0x56a5, 0x472c, 0x75b7, 0x643e,
0x9cc9, 0x8d40, 0xbfdb, 0xae52, 0xdaed, 0xcb64, 0xf9ff, 0xe876,
0x2102, 0x308b, 0x0210, 0x1399, 0x6726, 0x76af, 0x4434, 0x55bd,
0xad4a, 0xbcc3, 0x8e58, 0x9fd1, 0xeb6e, 0xfae7, 0xc87c, 0xd9f5,
0x3183, 0x200a, 0x1291, 0x0318, 0x77a7, 0x662e, 0x54b5, 0x453c,
0xbdcb, 0xac42, 0x9ed9, 0x8f50, 0xfbef, 0xea66, 0xd8fd, 0xc974,
0x4204, 0x538d, 0x6116, 0x709f, 0x0420, 0x15a9, 0x2732, 0x36bb,
0xce4c, 0xdfc5, 0xed5e, 0xfcd7, 0x8868, 0x99e1, 0xab7a, 0xbaf3,
0x5285, 0x430c, 0x7197, 0x601e, 0x14a1, 0x0528, 0x37b3, 0x263a,
0xdecd, 0xcf44, 0xfddf, 0xec56, 0x98e9, 0x8960, 0xbbfb, 0xaa72,
0x6306, 0x728f, 0x4014, 0x519d, 0x2522, 0x34ab, 0x0630, 0x17b9,
0xef4e, 0xfec7, 0xcc5c, 0xddd5, 0xa96a, 0xb8e3, 0x8a78, 0x9bf1,
0x7387, 0x620e, 0x5095, 0x411c, 0x35a3, 0x242a, 0x16b1, 0x0738,
0xffcf, 0xee46, 0xdcdd, 0xcd54, 0xb9eb, 0xa862, 0x9af9, 0x8b70,
0x8408, 0x9581, 0xa71a, 0xb693, 0xc22c, 0xd3a5, 0xe13e, 0xf0b7,
0x0840, 0x19c9, 0x2b52, 0x3adb, 0x4e64, 0x5fed, 0x6d76, 0x7cff,
0x9489, 0x8500, 0xb79b, 0xa612, 0xd2ad, 0xc324, 0xf1bf, 0xe036,
0x18c1, 0x0948, 0x3bd3, 0x2a5a, 0x5ee5, 0x4f6c, 0x7df7, 0x6c7e,
0xa50a, 0xb483, 0x8618, 0x9791, 0xe32e, 0xf2a7, 0xc03c, 0xd1b5,
0x2942, 0x38cb, 0x0a50, 0x1bd9, 0x6f66, 0x7eef, 0x4c74, 0x5dfd,
0xb58b, 0xa402, 0x9699, 0x8710, 0xf3af, 0xe226, 0xd0bd, 0xc134,
0x39c3, 0x284a, 0x1ad1, 0x0b58, 0x7fe7, 0x6e6e, 0x5cf5, 0x4d7c,
0xc60c, 0xd785, 0xe51e, 0xf497, 0x8028, 0x91a1, 0xa33a, 0xb2b3,
0x4a44, 0x5bcd, 0x6956, 0x78df, 0x0c60, 0x1de9, 0x2f72, 0x3efb,
0xd68d, 0xc704, 0xf59f, 0xe416, 0x90a9, 0x8120, 0xb3bb, 0xa232,
0x5ac5, 0x4b4c, 0x79d7, 0x685e, 0x1ce1, 0x0d68, 0x3ff3, 0x2e7a,
0xe70e, 0xf687, 0xc41c, 0xd595, 0xa12a, 0xb0a3, 0x8238, 0x93b1,
0x6b46, 0x7acf, 0x4854, 0x59dd, 0x2d62, 0x3ceb, 0x0e70, 0x1ff9,
0xf78f, 0xe606, 0xd49d, 0xc514, 0xb1ab, 0xa022, 0x92b9, 0x8330,
0x7bc7, 0x6a4e, 0x58d5, 0x495c, 0x3de3, 0x2c6a, 0x1ef1, 0x0f78
};
```

```
u16 compute_fcs(unsigned char *data, int length)
{ u16 fcs;

fcs = 0xffff;
while (length--)
{ fcs = (fcs >> 8) ^ fcstab[(fcs ^ ((u16)*data)) & 0xff];
     data++;
}
return (fcs);
}
unsigned char pattern[8] =
     { 0x03, 0x3f, 0x5b, 0xec, 0x00, 0x00, 0x00, 0x00 };
int main(int argc, char *argv[])
{ int i, j, k;
     u16 fcs;
fcs = compute_fcs(pattern, 2); /* generate CRC for
transmission
*/

fcs = fcs^0xffff;
printf("%02x %02x %02x %02x\n", pattern[0], pattern[1],
fcs&0xff,
(fcs>>8)&0xff);

fcs = compute_fcs(pattern, 4); /* check CR on reception */
printf("%02x %02x %02x %02x %04x\n", pattern[0], pattern[1],
pattern[2],
pattern[3], fcs);
if (fcs != 0xf0b8)
     printf("Bad CRC on reception!\n");

#ifdef MSDOS_TEST
i = *(u16 far *)(0x0040006c); /* get a random number */
pattern[0] = (i&0xff);
pattern[1] = (i>>8)&0xff;

fcs = compute_fcs(pattern, 2);
fcs = fcs^0xffff;
printf("%02x %02x %02x %02x\n", pattern[0], pattern[1],
fcs&0xff,
(fcs>>8)&0xff);
     pattern[2] = fcs&0xff;
     pattern[3] = (fcs>>8)&0xff;

fcs = compute_fcs(pattern, 4);
printf("%02x %02x %02x %02x %04x\n", pattern[0], pattern[1],
     pattern[2],
     pattern[3], fcs);
     #endif
}
```

Table 38 is an example that shows the proper FCS value for a two-byte frame consisting of 0x03 and 0x3F for the address and control fields.

Table 38 Example—Frame Check Sequence (FCS) Value for Two-Byte Frame for Address and Control Fields

\/--first bit transmitted				last bit transmitted--\/
0111 1110	1100 0000	1111 1100	1101 1010 0011 0111	0111 1110
flag	address	control	FCS	flag

Annex G
DEVELOPMENT RESOURCES

A variety of NTCIP-related resources is available. Annex G identifies some resources used in NTCIP development and early NTCIP implementations, as well as developed materials.

G.1 WEB SITES

A wide range of official NTCIP documentation is available at www.ntcip.org including:

a) NTCIP Document Links
b) NTCIP MIB acquisition information
c) NTCIP Case Studies and white papers written during the development of initial NTCIP standards
d) Links to NTCIP Exerciser and NTCIP Field Devices Simulator

Other web sites of interest are shown in Table 39.

Table 39 NTCIP Related Websites

Web Site	Address	Description
NTCIP	www.ntcip.org	The official NTCIP website provides NTCIP standards and other NTCIP documents, and related status information and news.
IANA	www.iana.org/protocols/	The Internet Assigned Numbers Authority web site.
IEEE ITS Page	http://standards.ieee.org/	Links to all of the IEEE standards efforts, including ATIS, Incident Management, Data Dictionaries and Data Registries.
ISO	www.iso.ch/	The ISO web site.
ISO TC204	www.tiaonline.org/standards/secretariats_tags/iso_tc204/	The home page for ISO Technical Committee 204 (i.e., the committee for ITS standards).
ITE	www.ite.org/	The ITE web site.
ITS America	www.itsa.org/	The ITS America web site.
NEMA Standards	www.nema.org/stds/ts2.cfm www.nema.org/stds/ts4.cfm	Sites for ordering NEMA standards NEMA TS 2-2003 or NEMA TS 4-2005.
RFC Index	www.ietf.org/rfc.html	A search engine for all of the Internet RFCs.
SNMP	www.ietf.org/rfc/rfc1157.txt	IETF developed SNMP.
TCIP	www.aptastandards.com/TCIPProgram/tabid/113/Default.aspx	The home page for the Transit Communications Interface Profiles.

G.2 SOURCES OF PUBLIC DOMAIN SOFTWARE

There are two basic prototype implementations of NTCIP software. Neither of these packages was designed to operate a real system; rather, they were designed to provide tools to the industry to test equipment submitted as being conformant to a specific protocol. Unfortunately, there is no ongoing program to maintain these packages.

G.3 NTCIP EXERCISER

This NTCIP Exerciser is able to read in any properly-formatted MIB from a floppy disk and support the exchange of fully conformant NTCIP messages under the direction of the operator. The Exerciser package supports the creation of simple macros to enable the user to perform a number of operations sequentially and to record the results. The 2000-era, build 3.3 version of the Exerciser supports the simulation of either a management station or an agent. The Exerciser version 3 supports SNMP only, Null Transport Profile, and both the PMPP-232 Subnetwork Profile and the PPP Subnetwork Profile. The Exerciser version 3 was designed for Windows NT, has not been updated, but is still available at www.ntcip.org/library/software/ .

G.4 FIELD DEVICE SIMULATOR

The U.S. DOT also sponsored the development of a DOS-based program to emulate a field device that supports the data elements contained in the Global Object Definitions. This program supports SNMP only, the Null Transport Profile and the PMPP-232 Subnetwork Profile. The Beta 2 of this software is also available at www.ntcip.org/library/software/ .

Annex H
SECTION REVIEW QUESTIONS

H.1 SECTION 1 REVIEW

1. The type of communications that involves a computer at a management center communicating with various devices at the roadside is referred to as _____ communications.

a) Center to Field (C2F)
b) Center to Center (C2C)
c) Field to Center (F2C)
d) All of the above

2. The type of communications that involves messages sent between two or more management systems is referred to as _____ communications.

a) Building to Building (B2B)
b) System to System (S2S)
c) Center to Center (C2C)
d) None of the above

3. What is not part of NTCIP?

a) Center to Center (C2C) specifications
b) Family of communications protocols
c) Incident Management data
d) Dynamic Message Sign (DMS) data dictionary

4. What is the primary goal of NTCIP?

a) Interoperability
b) Interference
c) Interchangeability
d) Intermodalism

5. Name the website where general information on the subject of NTCIP can be found.

a) www.nasa.org
b) www.ieee.org
c) www.tmdd.org
d) www.ntcip.org

H.2 SECTION 1 REVIEW ANSWERS

1. a) Center to Field (C2F)
2. c) Center to Center (C2C)
3. c) Incident Management data
4. a) Interoperability
5. d) www.ntcip.org

H.3 SECTION 2 REVIEW

1. Referring to the National ITS Architecture, NTCIP Center to Field (C2F) protocols link which Subsystems.

a) Center and Field
b) Field and Vehicle
c) Center and Remote Access
d) Center and Vehicle

2. A dynamic message sign is an example of a device that can take advantage of Center to _____ protocols.

a) Center
b) Field
c) Vehicle
d) None of the above

3. A transit management center communicating with a traffic management center can take advantage of Center to _____ protocols.

a) Center
b) Field
c) Vehicle
d) None of the above

4. Given the five levels that make up the NTCIP Framework, Plant, Information, Subnetwork, Application, and Transport. Arrange the NTCIP Framework levels in order.

a) Plant-Subnetwork-Transport-Application-Information
b) Information-Application-Subnetwork-Transport-Plant
c) All of the above
d) None of the above

5. Which of the following are not Application Level protocols currently available and commonly used for Center to Field (C2F) communications?

a) Data Exchange Protocol (DATEX)
b) Simple Network Management Protocol (SNMP)
c) Simple Transportation Management Protocol (STMP)
d) Both (b) and (c)

6. The Simple Transportation Management Protocol (STMP) uses _____ to improve bandwidth efficiency.

a) Data elements
b) Objects
c) Dynamic Data
d) Dynamic Objects

7. Which of the following are Application Level protocol choices currently available for Center to Center communications?

a) Data Exchange Protocol (DATEX) and Simple Network Management Protocol (SNMP)
b) Common Object Request Broker Architecture (CORBA) and Simple Transportation Management Protocol (STMP)
c) Data Exchange Protocol (DATEX) and Center to Center XML (C2C XML)
d) XML and Internet Protocol (IP)

8. If message routing through intermediate communications hub or field master is required, what two Transport Level protocol options are available for use with the Internet Protocol (IP)?

a) Data Exchange Protocol (DATEX) and Simple Network Management Protocol (SNMP)
b) Transmission Control Protocol (TCP) and User Datagram Protocol (UDP)
c) Transportation Transport Protocol (T2, formerly known as NULL protocol) and User Datagram Protocol (UDP)
d) Transmission Control Protocol (TCP) and Transportation Transport Protocol (T2, formerly know as NULL protocol)

9. NTCIP and non-NTCIP devices can be mixed on the same communications channel.

a) Always
b) None of the time
c) Routinely

10. True or False. One approach to the introduction of NTCIP in a center to field (C2F) system is to operate two separate systems—one NTCIP and one non-NTCIP—during the transition period.

a) True
b) False

11. What Transport Level protocol selection is used with non-routable protocols (no routing of messages through an intermediate hub or field master)?

a) Simple Network Management Protocol (SNMP)
b) Transmission Control Protocol (TCP)
c) User Datagram Protocol (UDP)
d) Transportation Transport Protocol (T2, formerly known as NULL protocol)

12. The selection of Information Level standards and data elements is based upon the desired_____ of the system being implemented.

a) Manufacturer
b) Functionality
c) Communications media
d) Procurement method

13. Can Center to Center (C2C) communications occur within the same building?

a) Yes
b) No
c) Always
d) Never

H.4 SECTION 2 REVIEW ANSWERS

1. a) Center and Field
2. b) Field
3. a) Center
4. a) Plant-Subnetwork-Transport-Application-Information
5. a) Data Exchange Protocol (DATEX)
6. d) Dynamic objects
7. c) Data Exchange Protocol (DATEX) and Center to Center XML (C2C XML)
8. b) Transmission Control Protocol (TCP) and User Datagram Protocol (UDP)
9. c) Routinely
10. a) True, operating two separate systems is one strategy for migrating from nonstandard
11. d) Transportation Transport Protocol (T2, formerly known as NULL protocol)
12. b) Functionality
13. a) Yes, Center to Center (C2C) communications take place between computer systems, and those computers or systems may be within the same building or in separate buildings.

H.5 SECTION 3 REVIEW

1. True or False. An agency does not need to have a solid understanding of their project requirements before entering into contract negotiations.

a) True
b) False

2. When manufacturers/developers define data elements for specific functions that are not covered by the NTCIP device standard and add them to the MIB, these additional data elements are called: _____.

a) MIB extensions
b) Other functions
c) Overhead
d) None of the above

H.6 SECTION 3 REVIEW ANSWERS

1. (b) False, an agency needs to have a solid understanding of their requirements before entering into negotiations, and preferably before developing procurement documents.
2. (a) MIB extensions

H.7 SECTION 4 REVIEW

1. As the NTCIP data dictionary standards are revised, the document organization has moved toward including seven main topic areas that support systems engineering. What are those seven main topic areas?

a) Foreword, General, Object Definitions, Conformance Statement, Framework, and Annex
b) Executive Summary, Introduction, Dialogs, Data Elements, Conformance Statement, and Profile Implementation Statement
c) Concept of Operations, Requirements (focusing on Functional), Dialogs (and Sequences), Data Dictionary (including Object Definitions and MIBs), Requirements Traceability Matrices, Test Procedures, and PRL Conformance Statement
d) None of the above

2. NTCIP standards have added a_____ as a means of providing a more user-friendly presentation of conformance.

a) Data Element Summary
b) Concept of Operations
c) Profile Requirements List (PRL)
d) Glossary

3. A Profile Requirements List (PRL) that has been filled out to indicate project-specific needs and requirements is called a _____.

a) Functional Requirements List (FRL)
b) Concept of Operations (ConOps)
c) Project Management Plan (PMP)
d) Profile Implementation Conformance Specification (PICS)

H.8 SECTION 4 REVIEW ANSWERS

1. c) Concept of Operations, Functional, Dialogues, Data Dictionary, RTM, PRL, and Test Procedures
2. c) Profile Requirements List (PRL)
3. d) Profile Implementation Conformance Specification (PICS)

H.9 SECTION 7 REVIEW

1. True or False. Before any testing begins, a clear statement and understanding of the requirements to be fulfilled and minimum acceptable performance levels is needed.

a) True
b) False

2. Name one software tool that is available for use in NTCIP testing and is freely available for download from the NTCIP website.

a) Field Profile Test Suite
b) NTCIP Exerciser
c) Field Simulation Suite
d) No software tools exist for testing

H.10 SECTION 7 REVIEW ANSWERS

1. a) True
2. b) NTCIP Exerciser

§