A Joint Standard of AASHTO, ITE, and NEMA

NTCIP 2201:2003 v01.15

National Transportation Communications for ITS Protocol

Transportation Transport Profile

September 2005

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

NOTICES

Copyright Notice

PDF File License Agreement

To the extent that these materials are distributed by AASHTO / ITE / NEMA in the form of an Adobe® Portable Document Format (PDF) electronic data file (the "PDF File"), AASHTO / ITE / NEMA authorizes each registered PDF File user to view, download, copy, or print the PDF File available from the authorized Web site, subject to the terms and conditions of this license agreement:

(a) you may download one copy of each PDF File for personal, noncommercial, and intraorganizational use only;
(b) ownership of the PDF File is not transferred to you; you are licensed to use the PDF File;
(c) you may make one more electronic copy of the PDF File, such as to a second hard drive or burn to a CD;
(d) you agree not to copy, distribute, or transfer the PDF File from that media to any other electronic media or device;
(e) you may print one paper copy of the PDF File;
(f) you may make one paper reproduction of the printed copy;
(g) any permitted copies of the PDF File must retain the copyright notice, and any other proprietary notices contained in the file;
(h) the PDF File license does not include (1) resale of the PDF File or copies, (2) republishing the content in compendiums or anthologies, (3) publishing excerpts in commercial publications or works for hire, (4) editing or modification of the PDF File except those portions as permitted, (5) posting on network servers or distribution by electronic mail or from electronic storage devices, and (6) translation to other languages or conversion to other electronic formats;
(i) other use of the PDF File and printed copy requires express, prior written consent.

Data Dictionary and MIB Distribution Permission

To the extent that these materials are distributed by AASHTO / ITE / NEMA in the form of a Data Dictionary ("DD") or Management Information Base ("MIB"), AASHTO / ITE / NEMA extend the following permission:

You may make and/or distribute unlimited copies, including derivative works, of the DD or MIB, including copies for commercial distribution, provided that:
(i) each copy you make and/or distribute contains the citation "Derived from NTCIP 0000 [insert the document number]. Used by permission of AASHTO / ITE / NEMA.";
(ii) the copies or derivative works are not made part of the standards publications or works offered by other standards developing organizations or publishers or as works-for-hire not associated with commercial hardware or software products intended for field implementation;

PRL and RTM Distribution Permission

A PICS is a Profile Requirements List which is completed to indicate the features that are supported in an implementation. Visit www.ntcip.org for information on electronic copies of the MIBs, PRLs, and RTMs.

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.

ACKNOWLEDGEMENTS

This publication was prepared by the NTCIP Profiles Working Group, and was reviewed and recommended by the Joint Committee on the NTCIP. The Joint Committee is organized under a Memorandum of Understanding among the American Association of State Highway and Transportation Officials (AASHTO), the Institute of Transportation Engineers (ITE), and the National Electrical Manufacturers Association (NEMA). The Joint Committee on the NTCIP consists of six representatives from each of the standards organizations, and provides guidance for NTCIP development.

At the time that this document was prepared, the following individuals were members of the NTCIP Profiles Working Group:

- Jas Bhullar
- Doug Crawford
- Robert De Roche (Chair)
- Robert Force
- Willard "Bud" Kent
- Paul Lim-Hing
- Bob Martindale
- Amit Misra
- Alex Mousadi
- Kenneth Vaughn

Other individuals providing input to the document include:

- Joey Baumgartner
- Albert Bonificio
- Blake Christie
- Arthur Felgate
- Michael Forbis
- Joseph Herr
- Earl Hoekman
- David Kingery
- Al Kosik
- Doug Lowe
- Gary Meredith
- Don Ninke
- Brian Paulsmeyer
- Jeff Racz
- Michael Robinson
- James Rose
- Joerg "Nu" Rosenbohm
- Bancroft Scott
- Bo Strickland
- Sonja Sun
- Paul Thorpe
- Keith Vennel
- Hoi Wong

In addition to the many volunteer efforts, recognition is also given to those organizations who supported the efforts of the working groups by providing comments and funding for the standard, including:

- ARINC
- Caltrans
- Eagle Traffic Control Systems
- Econolite Control Products
- Federal Highway Administration
- Iteris
- JPL
- Mastec
- Ministry of Transportation, Ontario
- Minnesota Department of Transportation
- Mitretek
- Naztec
- PB Farradyne, a Division of PBQD
- Peek Traffic
- Skyline Products
- Southwest Research Institute
- Trevilon
- VHB
- Virginia Department of Transportation

FOREWORD

This document uses only metric units.

The context of the NTCIP is one part of the Intelligent Transportation Systems standardization activities covering base standards, profiles, and registration mechanisms.

- Base Standards define procedures and rules for providing the fundamental operations associated with communications and information that is exchanged over communications links.
- Profiles define subsets or combinations of base standards used to provide specific functions or services. Profiles prescribe particular subsets or options available in base standards necessary for accomplishing a particular function or service. This provides a basis for the development of uniform, nationally recognized conformance.
- Registration Mechanisms provide a means to specify and uniquely identify detailed parameters within the framework of base standards and/or profiles.

This publication provides the definition of an NTCIP Transport Profile. It defines requirements for the transport and network layers of a communications stack based upon the ISO-OSI Reference Model. It also contains optional and conditional clauses that are applicable to specific environments for which they are intended.

For more information about NTCIP standards, visit the NTCIP Web site at http://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 will 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 will be 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
fax: (703) 841-3331
e-mail: ntcip@nema.org

A User Comment should be submitted in the following form:

Standard Publication number and version:
Page:
Paragraph or Clause:
Comment:

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

Approvals

This standards publication was separately balloted and approved by AASHTO, ITE, and NEMA after recommendation by the Joint Committee on the NTCIP. Each organization has approved this standard as the following standard type, as of the date:

AASHTO – Standard Specification; March 2003
ITE – Software Standard; December 2002
NEMA – Standard; October 2002

History

The technical specifications of NTCIP 2201 are identical to the former version, except as noted in the development history below:

NTCIP 2201 version 01.11. May 2001 -- Distributed for user comment.

NTCIP 2201 version 01.12. Revised per user comments. September 2001 – Accepted as a Recommended Standard by the Joint Committee on the NTCIP.

NTCIP 2201 version 01.14. February 2002 – General format revisions. NTCIP Standards Bulletin B0074 sent v01.14 for ballot and approval. Jointly Approved in March 2003. Approved by AASHTO in March 2003, approved by ITE in December 2002, and approved by NEMA in October 2002.

NTCIP 2201:2003 v01.15. September 2005 – Prepared document for publication with revised front matter and format editing.

Compatibility of Versions

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.

Anyone using this document should seek information about the version number that is of interest to them in any given circumstance. The MIB, the PRL, and the PICS should all reference the version number of the standards publication that was the source of the excerpted material.

Compliant systems based on later, or higher, version numbers MAY NOT be compatible with compliant systems based on earlier, or lower, version numbers. Anyone using this document should also consult NTCIP 8004 for specific guidelines on compatibility.

INTRODUCTION

This publication defines a transport profile that is a combination of standards intended to meet specific requirements for transport services in transportation devices and management centers in a non-networked environment. Its scope covers the transport and network layers of the OSI Reference Model. It contains mandatory requirement statements that are applicable to all devices claiming conformance to this standard. It also contains optional and conditional requirements that may be applicable to a specific environment in which a device is used.

Annex A is normative and contains a Profile Requirements List in the form of PICS proforma. Annex B is informative and contains a set of service primitives that can be used to characterize the services provided by the Transportation Transport (T2) Profile.

The following keywords apply to this document: AASHTO, ITE, NEMA, NTCIP, Profile, Transport, NULL, and T2.

In 1992, the NEMA 3-TS Transportation Management Systems and Associated Control Devices Section began the effort to develop the NTCIP. Under the guidance of the Federal Highway Administration's NTCIP Steering Group, the NEMA effort was expanded to include the development of communications standards for all transportation field devices that could be used in an Intelligent Transportation Systems (ITS) network.

In September 1996, an agreement was executed among AASHTO, ITE, and NEMA to jointly develop, approve, and maintain the NTCIP standards.

After research into how national and international standards organizations combine protocols and standards to address all seven layers of the ISO-OSI Reference Model, the committee adopted the approach defined in the *NTCIP Profile Framework.* Following that approach, a complete protocol stack can be specified by application, transport, and subnetwork profiles. An application profile addresses the application, presentation, and session layers. A transport profile addresses the transport and network layers. A subnetwork profile addresses the data link and physical layers. The *NTCIP - Transportation Transport Profile* (TP - T2) is a transport profile for use in center-to-field and field-to-field communications.

Within the Joint AASHTO/ITE/NEMA Committee on the NTCIP, the Profiles Working Group is concerned with the methodology of defining profiles, and their documentation in Standards Publications. This standard defines a transport profile and the requirements for linking an application profile to a subnetwork profile. In the case of non-networked communications, there is little functionality associated with the network and transport layers. In this case, the purpose of transport service is to transform the information at the application - transport interface to a form compatible with the transport - subnetwork interface. In addition to this, the transport service provides a multiplexing function to route incoming frames to multiple applications layers. The objective is to facilitate the specification of ITS characterized by a high degree of interoperability and interchangeability of its components.

< This page is intentionally left blank. >

CONTENTS

< This page is intentionally left blank. >

Section 1
GENERAL

1.1 SCOPE

This standard is applicable to transportation devices and management systems that must operate in Intelligent Transportation Systems. It specifies a set of procedures applicable to the transport and network layers of the ISO - OSI Reference Model. The set of procedures provides a linking mechanism between the application and subnetwork profiles in non-networked environments. In this environment, no transport and network layer services are needed other than interfacing the profiles and upper layer protocol multiplexing. It is intended to provide a standard interface technique to ensure interoperability, especially in cases where multiple communications stacks co-exist.

1.2 PROFILE-PROTOCOL-LAYER RELATIONSHIP

This transport profile specifies the provision for non-networked, connectionless transport and network services between an end system connected to a subnetwork and another compatible end system on the same subnetwork. The end systems must use mutually agreed upon access methods at the application, transport, and subnetwork levels in order to communicate. An end system is compatible only if the suboptions (e.g., TCP) are compatible. A complete transport profile requires knowledge of the subnetwork type, access method, circuit type, and service type. The layers, base standards and profile taxonomy that make up this profile are shown in Figure 1-1.

ISO Layers	Base Standards	Profile
TRANSPORT LAYER	Part of IAB STD 6 (UDP)	NTCIP 2201 Transportation Transport (T2) Profile
NETWORK LAYER	Part of IAB STD 17 (MIB-II)	

Figure 1-1
Transportation - Transport Profile Relationship

As implied by the figure, only parts of existing base standards are referenced. In the OSI Reference Model, the transport layer is responsible for reliable data transfer between two end-systems, using flow control and error recovery, and may provide multiplexing. The network layer is responsible for data transfers across the network, independent of both the media comprising the underlying subnetworks and the topology of those subnetworks. The intended environment of this profile is where flow control and error recovery are the responsibility of the subnetwork and where information exchanges occur in a non-network system, where there is only a single subnetwork. The only remaining function that is applicable to the T2 Profile is application multiplexing. The relationship of the T2 Profile to the other layers of the OSI Model is that it serves as a link in transforming the interface information at the Application - Transport boundary to the interface information at the Transport-Subnetwork boundary.

1.3 REFERENCES

For approved revisions, contact:

NTCIP Coordinator
National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, VA 22209-3806
fax: (703) 841-3331
e-mail: ntcip@nema.org

For draft revisions of this document, which are under discussion by the relevant NTCIP Working Group, and recommended revisions of the NTCIP Joint Committee, visit the World Wide Web at http://www.ntcip.org.

1.3.1 Normative References

The following standards (normative references) contain provisions, which, through reference in this text, constitute provisions of this Standard. Other documents and standards (other references) are referenced in these documents, which might provide a complete understanding of the structure and use of profiles. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Internet Architecture Board
http://www.rfc-editor.org/
http://www.ietf.org/rfc.html
http://www.rfc-editor.org/repositories.html

IAB STD 2 *Assigned Numbers (RFC 1700),* J. Reynolds, J. Postel, October 1994.

IAB STD 6 *User Datagram Protocol (RFC 768),* J. Postel, August 1980.

IAB STD 17 *Management Information Base (RFC 1213),* K. McCloghrie, M. Rose. March 1991.

NTCIP 7001 Assigned Numbers - Part 1.

1.3.2 Other References

NTCIP 2001:1996 *National Transportation Communications for ITS Protocol Class B Profile*

Guide to Open System Specifications, European Workshop for Open Systems, http://www.ewos.be/goss/top.htm, June 9, 1997

Internetworking with TCP/IP Volume II: Design, Implementation, and Internals, 2nd Edition, 1998, D. Comer and D. Stevens, Prentice Hall, ISBN 0-13-973843-6

1.4 TERMS

For the purpose of this Standard, the following definitions apply:

Application Layer	That portion of the OSI Basic Reference Model (Layer 7) that provides access to the communications services.
data	Information before it is interpreted.
Data Link Layer	That portion of the OSI Basic Reference Model (Layer 2) responsible for flow control, framing, synchronization, and error control over a communications link.
datagram	A self-contained unit of data transmitted independently of other datagrams.
end system	The source or destination of an information exchange.
host	(Internet usage) The physical and/or logical part of the end-system's application. A computer attached to one or more networks that supports users and runs application programs.
Intelligent Transportation Systems	A major national initiative to apply information, communication and control technologies in order to improve the efficiency of surface transportation.
intermediate system	A system that participates in an information exchange but is not the source or destination of the exchange.
internet	Any collection of connected networks where information can be passed from one network to another.
Internet	A large collection of connected networks running the Internet suite of protocols.
Internet protocol	The network protocol offering a connectionless mode network service in the Internet suite of protocols.
Internet Protocol Suite	A collection of computer-communication protocols originally developed under Defense Advanced Research Projects Agency (DARPA) sponsorship.
internetwork	The ability of devices to communicate across multiple networks.
network	A collection of subnetworks connected by intermediate systems and populated by end systems.
Network Layer	That portion of an OSI Basic Reference Model (Layer 3) 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 devices that contain information about setup, control, and status of the layers in a communications stack. The term refers to all devices, both intermediate and end systems, that are present on the network or internetwork.
Open Systems Interconnection	An international effort to facilitate communications among computers of different manufacture and technology.
OSI Reference Model	A widely accepted structuring technique that provides an abstract representation of the communication process that is divided into seven basic, functional layers.
Physical Layer	That portion of an OSI Basic Reference Model (Layer 1) responsible for

	the electrical and mechanical interface between communicating systems.
Presentation Layer	That portion of an OSI Basic Reference Model (Layer 6) responsible for converting and organizing data from one format to another.
proforma	A guide provided in advance to prescribe form or describe items.
Session Layer	That portion of an OSI Basic Reference Model (Layer 5) which manages a series of data exchanges between end-system applications.
subnetwork	A physical network within a network. All devices on a subnetwork share a common physical medium.
taxonomy	A classification scheme for referencing profiles or sets of profiles unambiguously.
TCP/IP Reference Model	An alternate to the OSI Basic Reference Model that organizes the communications process into 4 layers. It consists of host-to-network, internet, transport, and application layers.
Transport Layer	That portion of an OSI Basic Reference Model (Layer 4) which attempts to guarantee reliable data transfer between two end-systems, using flow control and error recovery, and may provide multiplexing.

1.5 ABBREVIATIONS AND ACRONYMS

The abbreviations used in this Standard Publication are defined as follows:

AASHTO	American Association of State Highway and Transportation Officials
AID	Application Identifier
FTP	File Transfer Protocol
IAB STD	Internet Architecture Board Standard
IP	Internet Protocol
IPI	Initial Protocol Identifier
ISO	International Organization for Standardization
ITE	Institute of Transportation Engineers
ITS	Intelligent Transportation Systems
MIB	Management Information Base
NEMA	National Electrical Manufacturers Association
NTCIP	National Transportation Communications for ITS Protocol
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PRL	Profile Requirements List
RFC	(Internet) Request for Comments
SNMP	Simple Network Management Protocol
STMP	Simple Transportation Management Protocol
T2	Transportation Transport (Profile and Protocol)
TCP	Transmission Control Protocol
TP	Transport profile
UDP	User Datagram Protocol

Section 2
CONFORMANCE

2.1 GENERAL REQUIREMENTS

Implementations claiming conformance to the Transportation Transport Profile shall support the following elements as stated.

a. All requirements in the remainder of Section 2 of this profile.
b. All of the constraints specified in Annex A (normative) of this profile.
c. All mandatory requirements of the standards referenced by this profile.

2.2 TRANSPORTATION TRANSPORT PROTOCOL (T2)

The T2 Profile defines the specifics of a protocol to provide transport and network layer services. As such, T2 refers to both a profile and a protocol. The protocol provides connectionless transport and network layer services for implementations that communicate in a non-networked environment. Information transfers can only occur between hosts that reside on the same subnetwork. It provides a means to multiplex multiple application layers and supports multiple client-server interactions.

2.2.1 Major Capabilities

A conforming implementation of T2 shall support the following capabilities in accordance with the indicated base standards:

a. upper layer interfacing as specified in Subclause 2.2.2
b. data transfer as specified in Subclauses 2.2.3 and 2.2.4
c. port number formatting per Subclause 2.2.5
d. lower layer interfacing as specified in Subclause 2.2.6

2.2.2 Upper Layer Interfacing

The upper layer interface shall allow:

a. Receive operations on the receive ports that return the data octets and an indication of source port.
b. Send operations that allow an Upper Layer PDU to be sent, specifying the data, source and destination ports.
c. Provide an indication of the logical address of the source and destination of the PDU.
d. Creation and assignment of unused Port Numbers.

2.2.3 Data Send Procedures

The application layer shall provide an indication of the encapsulation method for how the Upper Layer PDU is to be encapsulated by the transport header. For compatibility with prior standards, there are three possible methods. The header format associated each method is summarized in the following table.

Table 2-1
PDU Encapsulation Summary

Encapsulation Method	Header Format
1	No header added - PDU is SNMP GetRequest, GetNextRequest, GetResponse, or SetRequest, or STMP message
2	T2 header with AID and Source and Destination ports added
3	AID added - PDU is SNMP Trap

In the first encapsulation method, the Upper Layer PDU (UL-PDU) shall be passed directly to the lower layer(s) as the Transportation Transport PDU (T2-PDU); no source or destination port information shall be added. This method is meant to provide backwards compatibility with the original NTCIP Class B Communications Profile (NTCIP 2001) and SNMP messages.

In the second encapsulation, a header consisting of an Application Identifier (AID) with a value of 0x41 followed by the source and destination port fields (Subclause 2.2.5) shall be prefixed to the Upper Layer PDU. This shall constitute the Transportation Transport PDU and be passed to the lower layer(s). Any application protocol, including SNMP and STMP may use this encapsulation method.

In the third encapsulation method, a value of 0x31 shall be prefixed to the Upper Layer PDU. This shall be passed to the lower layer(s) as the Transportation Transport PDU; no source or destination port information shall be added. This method is meant to provide a mechanism for transporting SNMP Traps which was not fully defined in the original NTCIP Class B Communications Profile.

Figure 2-1 illustrates the information that is added to the Upper Layer PDU in each encapsulation method.

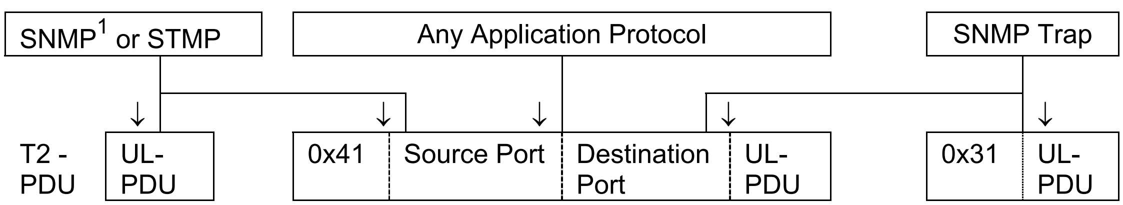

Figure 2-1
PDU Encapsulation Methods

SNMP GetRequest, GetNextRequest, GetResponse, SetRequest, or STMP messages can be sent with either the first or second encapsulation method. SNMP Traps can be sent with either the second or third encapsulation method. Any other connectionless-oriented application protocol shall only use the second encapsulation method.

NOTE – When the T2 PDU is passed to the Data Link Layer, an Initial Protocol Identifier or similar type of identifier is added so that the receiving a Data Link Layer can distinguish which transport service is to receive the incoming frame. Annex C of this standard illustrates how an IPI is added to the PMPP protocol.

2.2.3.1 Logical to Physical Address Translation

An application layer may identify a destination by means of a global address, such as an IP Address. An implementation may also support multiple subnetworks. In this case, the send procedures shall include a

[1] Only SNMP GetRequest, GetNextRequest, GetResponse, or SetRequest

mechanism to convert a logical address to appropriate subnetwork and a physical address that is native to that particular subnetwork.

As an example, consider a management application that runs a personal computer. The application manages three distinct subnetworks (systems) that are connected to it via a fiberoptic cable, dialup modem, and an FSK modem respectively. The fiberoptic interface uses the Ethernet Protocol, the dialup modem uses PPP, and the FSK Modem used PMPP. At the application layer, the first device in each subnetwork is assigned an IP address as follows: 192.168.1.1, 192.168.2.1, and 192.168.3.1. For the Ethernet Protocol, the address 192.168.1.1 is converted to a 48-bit Ethernet address such as 0x080090034CF1. For PPP, the address 192.168.2.1 is converted to a 7-digit phone number such as 555-1212. For PMPP, the address 192.168.3.1 is convert to a 16-bit HDLC address such as 0x0005. The application layer uses the same addressing scheme for all devices irrespective of the physical means by which they are connected. In this example, the address translation table referenced in Clause 2.3 would be appropriate.

2.2.4 Data Receive Procedures

Before passing a frame to the Upper Layer interface, the first octet of the frame (first octet after the IPI) shall be the Application Identifier (AID). (The IPI is not used and discarded.) The actions associated with AID values and corresponding parsing method are summarized in the following table.

Table 2-2
PDU Parsing Summary

AID Value	Parsing Method	Action
0x30	1	Pass PDU directly to SNMP Handler (Port 161)
0x31	2	Strip AID and pass remaining PDU To SNMP Trap Handler (Port 162)
0x81 - 0xFD	3	Pass PDU directly to STMP Handler (Port 501)
0x41	4	Strip AID, decode destination port number, and pass remaining data to appropriate application handler port number

A value of 0x30 shall indicate that the AID is part of the Upper Layer PDU and the PDU shall be passed directly up to any bound SNMP Application Layer for processing as a get, set, or get next operation. In this case, the source and destination port fields values are implicitly set to 161.

A value of 0x31 shall indicate that the AID is NOT part of the Upper Layer PDU and the PDU shall be passed up to any bound SNMP Application Layer for processing as a trap. In this case, the source and destination port field values are implicitly set to 162.

A value of 0x81 - 0xFD shall indicate that the AID is part of the Upper Layer PDU and the PDU shall be passed up to any bound STMP Application Layer. In this case, the source and destination port values are set implicitly to 501.

A value of 0x41 shall indicate that the AID is followed by the source and destination ports fields of the Transportation Transport Protocol header. The Upper Layer PDU begins with the first octet after the source and destination port fields. It is passed to any bound application layer indicated by the destination port field.

Figure 2-2 illustrates the information extracted from the data contained in the T2-PDU.

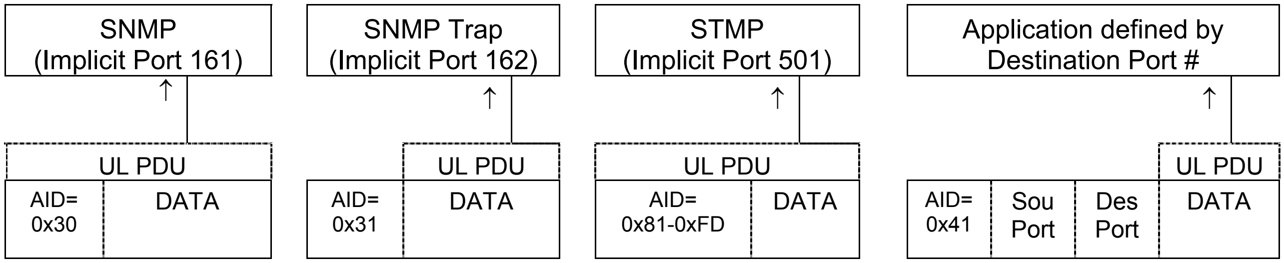

Figure 2-2
PDU Parsing Methods

All other values of the AID shall be indicative of some erroneous condition and the frame shall be discarded.

2.2.4.1 Physical to Logical Address Translation

As in the send procedures, an application layer may identify a destination by means of a global address, such as an IP Address. Multiple subnetworks may also be involved. In this case, the receive procedures shall include a mechanism to convert a physical address that is native to the subnetwork to the logical address that is used by the application.

2.2.5 Port Numbers

When Encapsulation Method 2 or Parsing Method 4 is used, a conforming implementation shall support the following fields:

a. source port
b. destination port

The format of the source and destination ports is two 16-bit fields that conform to RFC 768 and are illustrated in the following figure.

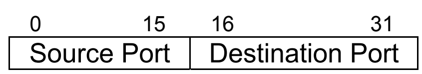

Figure 2-3
T2 Header Format

The source port indicates the port of the sending process. The destination port indicates the port of the intended recipient. See Figure C-3 of this standard for a coding example.

In the absence of other information or defining procedures, the source port can be assumed to be the port to which a reply should be addressed. It should be noted that some application layer protocols initiate communications on a "well known" port number and then switch to another un-assigned port number to perform transfers. The point at which the transition takes place is protocol dependent.

The allocation of unused port numbers is considered a function of the transport protocol. The requirement for allocation of unused port number is based upon the needs of an application protocol and, therefore, optional.

NTCIP 7001 Assigned Numbers—Part 1 defines the list of "well known" port numbers that are assigned to NTCIP specific applications. If T2 is the only transport profile implemented, then any port number not contained in that list shall be assumed unallocated. If T2 operates in parallel with the NTCIP 2202 Internet Transport Profiles, then IAB STD 1700 should be consulted in addition to NTCIP 7001. A particular operating system may impose additional constraints on the range of unallocated port numbers.

2.2.6 Lower Layer Interfacing

The lower layer interface shall allow:

a. Send operations that allow a transport PDU to be sent, specifying the data and the source and destination addresses that are native to the subnetwork being used.
b. Receive operations on one or more subnetworks that return the data octets and an indication of the physical address of the source and destination of the data.

2.3 MIB-II SUPPORT

If SNMP is supported and the logical to physical address translation functionality as expressed by the NetToMedia Table in MIB-II is supported, then the NetToMedia Table in MIB-II as defined in RFC 1213, Sections 3.7 and 6.6 shall be supported.

NOTE – In the Internet Suite of Protocols, a logical address is used at the Application Layer to isolate that layer from any Data Link or Physical Layer dependencies. For example, the URL of a webpage would remain the same irrespective of whether the actual connection was through a local area network (LAN) or a dialup modem. The NetToMedia Table can be used to map a logical address to an address appropriate to the Data Link and/or Physical Layers. In the case of a LAN, the NetToMedia Table could provide a translation from an IP Address to an Ethernet Address. In the case of a dialup modem, the Group could provide a translation from an IP Address to the telephone number. The use of logical address at the Application Layer is not mandatory and, therefore, support for the NetToMedia Table is optional.

< This page is intentionally left blank. >

Annex A
TRANSPORTATION TRANSPORT PROFILE REQUIREMENTS LIST
(Normative)

A.1 INTRODUCTION

This annex provides the Profile Requirements List (PRL) for implementations of the Transportation – Transport Profile in the form of proforma. A Profile Implementation Conformance Specification (PICS) for an implementation is generated by an implementer or supplier by indicating the appropriate level of support provided by an implementation.

To claim conformance with this profile, an implementation shall satisfy the mandatory conformance requirements of this profile.

An implementation's completed PRL is called the PICS. The PICS states which capabilities and options of the protocol have been implemented. The following can use the PICS:

a. The protocol implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight.

b. The supplier and user, as a detailed indication of the capabilities of the implementation.

c. The user, as a basis for initially checking the possibility of interworking with another implementation (note that, while interworking can never be guaranteed, failure to do so can often be predicted from incompatible PICSs).

d. A user, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.1.1 Notation

The following notations and symbols are used to indicate status and conditional status in the PRL and PICS within all NTCIP standards. Not all of these notations and symbols may be used within this standard.

A.1.1.1 Status Symbols

The following symbols are used to indicate base standard and profile status:

m	mandatory
m.<n>	support of every item of the group labeled by the same numeral <n> required, but only one is active at time.
o	optional
o.<n>	optional, but support of at least one of the group of options labeled by the same numeral <n> is required
c	conditional
n/a	non-applicable (i.e., logically impossible in the scope of the profile)
x	excluded or prohibited

The o.<n> notation is used to show a set of selectable options (i.e., one or more of the set must be implemented) with the same identifier <n>. Two character combinations are used for dynamic conformance requirements. In this case, the first character refers to the static (implementation) status, and the second refers to the dynamic (use); thus "mo" means "mandatory to be implemented, optional to be used." Base standard requirements are shown using the equivalent notations in upper case (e.g., M, O, X).

The classification of the requirements and options in Internet RFCs does not correspond to the convention described in above, and shall be mapped into the profile as follows:

RFC	Profile
MUST	Mandatory[2]
SHOULD	Mandatory[2]
MAY	Optional
SHOULD NOT	Prohibited
MUST NOT	Prohibited

A.1.1.2 Conditional Status Notation

The following predicate notations are used:

<predicate>:	This notation introduces a single item that is conditional on the <predicate>.
<predicate>::	This notation introduces a table or a group of tables, all of which are conditional on the <predicate>.

The <predicate>: notation means that the status following it applies only when the PRL or PICS states that the feature or features identified by the predicate are supported. In the simplest case, <predicate> is the identifying tag of a single PICS item. The <predicate>:: notation may precede a table or group of tables in a clause or subclause. When the group predicate is true then the associated clause shall be completed. The symbol <predicate> also may be a Boolean expression composed of several indices. "AND," "OR," and "NOT" shall be used to indicate the Boolean logical operations.

A.1.1.3 Support Column Symbols

This profile is in the form of a PICS and, therefore, includes a support column. An implementer claims support of an item by circling the appropriate answer (Yes, No, or N/A) in the support column:

Yes	Supported by the implementation.
No	Not supported by the implementation.
N/A	Not applicable.

A.1.1.4 Footnotes

Footnotes to the proforma are indicated by superscript numerals. The footnote appears on the page of the first occurrence of the numeral. Subsequent occurrences of a numeral refer to the footnote of the first occurrence.

A.1.1.5 Instructions for Completing the PRL

A Profile implementer shows the extent of compliance to a Profile by completing the PRL. The implementer indicates whether mandatory requirements are complied with, and whether optional functions are supported. The resulting completed PRL is called a PICS. Where this profile refines the features of the base standards, the requirements expressed in this PRL shall be applied (as indicated in PRL items with no "Profile Support" column) to constrain the allowable responses in the base standard

[2] In the course of adapting communications industry standards to the transportation industry, there may be exceptions where specific mandatory requirements are not applicable to the new environment. Where these exceptions are made, a justification shall be provided.

PICS proforma. When this profile makes additional requirements, the "Support" column for such PRLs shall be completed. In this column, each response shall be selected either from the indicated set of responses, or it shall comprise one or more parameter values as requested. If a conditional requirement is inapplicable, use the Not Applicable (NA) choice. If a mandatory requirement is not satisfied, exception information must be supplied by entering a reference Xi, where i is a unique identifier, to an accompanying rationale for the noncompliance. When the profile requirement is expressed as a two-character combination (as defined in A.1.1 above), the response shall address each element of the requirement; e.g., for the requirement "mo," the possible compliant responses are "yy" or "yn."

A.2 STANDARDS REFERENCED

This profile specifies the provisions for connectionless-oriented transport and network services in a non-networked environment. This type of service may apply to end systems that support multiple application layer protocols and operate in a many-to-one client server mode of operation. It references parts of the following standards:

IAB STD 2 *(RFC 1700: 1992, Assigned Numbers)*
IAB STD 6 *(RFC 768: 1980, User Datagram Protocol)*
IAB STD 17 *(RFC 1213: 1991, Management Information Base)*
NTCIP 7001 *Assigned Numbers - Part 1*

A.3 PICS REQUIREMENTS LISTS

A.3.1 Implementation Identification

Ref	Question	Response
1	Supplier	
2	Contact point for queries about the profile	
3	Implementation Name(s) and Version(s)	
4	Date of statement	
5	Other Information: Machine Name, Operating Systems, System Name	
6	Amendments or revisions to the base standards or profiles that are applicable.	

A.3.2 T2 Global Statement of Conformance

Are all mandatory requirements met for:

Ref	Standard	Response
1	Transportation Transport Protocol	

A.4 BASIC REQUIREMENTS

The following table lists the major requirements for a Transportation Transport Profile implementation, and asks if the listed protocols and object definition groups have been implemented:

Index	Protocol	Clause of Profile	Profile Status	Support
T2	Is Transportation Transport Protocol supported?	2.2	m	Yes
snmp	Does the implementation support SNMP?	2.3	o	Yes No
ipAddr	Does the implementation implement an IPv4 to Physical Address translation?		o	Yes No
NetTo Media	Are IAB STD 17 (RFC 1213), MIB-II, Sections 3.7 and 6.6, IP NetToMedia Table objects supported?		snmp AND ipAddr:m	Yes No

A.5 T2 GENERAL/MAJOR CAPABILITIES

Item	Protocol Feature	Base Standard		Profile		Support
		Reference	Status	Clause	Status	
ul-inter	Provide upper layer interfacing	RFC 768	M	2.2.2	m	Yes
pdt	Perform data transfers	RFC 768	M	2.2.5, 2.2.4	m	Yes
ll-inter	Perform lower layer interfacing	RFC 768	M	2.2.6	m	Yes
addr-addr	Perform address translation	--	--	2.2.3.1, 2.2.4.1	o	Yes No
apn	Allocate port numbers			2.2.5	o	Yes No

A.5.1 T2 Interfaces

Item	Protocol Feature	Base Standard		Profile		Support
		Reference	Status	Clause	Status	
App-T2	Application/Transport Interface					
T2 -Rec	Receive operations that return the data octets and source port and source address	RFC 768	M	2.2.2	m	Yes
T2-Send	Send operations that specify data, source and destination ports				m	Yes
Log-Addr	Does Application use logical addressing	RFC 768	M	2.2.2	o	Yes
Log-Xlate	Is function for translation from logical to physical addresses implemented			2.2.3.1	Log-Addr:m	Yes
Phy-Xlate	Is function for translation from physical to logical addresses implemented			2.2.4.1	Log-Addr:m	Yes
Alloc	Does Application require allocation of unused port numbers	RFC 768	M	2.2.2	o	Yes No
S-Alloc	Is function for allocation of unused port numbers implemented				Alloc:m	Yes
T2-DL	Transport/Data Link Interface					

Item	Protocol Feature	Base Standard		Profile		Support
		Reference	Status	Clause	Status	
T2-Req	Send operations that specify data and physical destination address	RFC 768	M	2.2.6	m	Yes
T2-Ind	Receive operations that return the data octets and physical source address				m	Yes

A.5.2 T2 Procedures

Item	Protocol Feature	Base Standard		Profile		Support
		Reference	Status	Clause	Status	
EM	Encapsulation Methods					
EM1	Method 1 (SNMP [except trap] or STMP)	--	--	2.2.3	m	Yes
EM2	Method 2 (Any Application)				m	Yes
EM3	Method 3 (SNMP Trap)				m	Yes
Emi	IPI added to T2 PDU to form Data Link Information Field				m	Yes
EMb	Proper format of Source and Destination Port Number Fields	RFC 768	M	2.2.5	m	Yes
PM	Parsing Methods					
Pmi	IPI stripped from Data Link Information Field to form T2 PDU	--	--	2.2.4	m	Yes
PM1	Method 1 (SNMP)				m	Yes
PM2	Method 2 (SNMP Trap)				m	Yes
PM3	Method 3 (STMP)				m	Yes
PM4	Method 4 (Any Application)				m	Yes
UM3a	Discard data for unknown ports				m	Yes
UM3b	Proper format of Source and Destination Port Number Fields	RFC 768	M	2.2.5	m	Yes
Alloc	Does Application require allocation of unused port numbers				o	Yes No
S-Alloc	Is allocation of unused port numbers implemented				Alloc:m	Yes

A.6 MIB-II NET TO MEDIA TABLE SUPPORT

snmp AND ipAddr::

Item	Object Definitions			Base Standard		Profile		Support
	Object	Syntax	Access	Reference	Status	Clause	Status	
ntm.1	ipNetToMedia Table	SEQUENCE OF IpNetTo MediaEntry	not-accessible	RFC1213, Section 6.6	M	2.3	m	Yes
ntm.1.1	ipNetTo MediaEntry	IpNetTo MediaEntry	not-accessible	RFC1213, Section 6.6	M		m	Yes
ntm.1.1.1	ipNetTo MediaIndex	INTEGER	read-write	RFC1213, Section 6.6	M		m	Yes
ntm.1.1.2	ipNetTo MediaEntry PhysAddress	Phys Address	read-write	RFC1213, Section 6.6	M		m	Yes
ntm.1.1.3	IpNetTo MediaNetAddress	IP Address	read-write	RFC1213, Section 6.6	M		m	Yes
ntm.1.1.4	IpNetTo MediaNetType	INTEGER	read-write	RFC1213, Section 6.6	M		m	Yes

Annex B
TRANSPORTATION TRANSPORT INTERFACES
(Informative)

B.1 INTRODUCTION

Most computers provide communications services as part of the operating system. The interface to these services is operating system dependent but is usually well defined in public domain documentation. In Unix based systems, access to transport services is through what is called the Berkley Sockets Interface. In Windows based systems, access is through Winsock. Specifications for these interfaces are available on the Internet.

Transportation field devices are only now adopting an open architecture approach where an Application Programming Interface to operating system services is defined and accessible. Traditionally, field devices have used an embedded approach. In this case, the application program and transport services have tightly integrated and the exact definition of the interface is proprietary.

This standard does not require or prescribe a specific interface between the application and transport profiles. It does not require or prescribe one for the interface between the transport and subnetwork profiles, either. However, for the sake of understanding how the Transportation Transport services can be used, a set of service primitives, associated parameters, and a description of the T2 functions is presented.

B.2 UPPER LAYER INTERFACING

The primitives and parameters of the services at the interface between the lower layers and the transportation transport protocol can be described as follows:

```
T2_Send        (
               PDU,
               EncapsulationMethod,
               SourcePort,
               DestinationPort,
               MediaIndependent_SourceAddress,
               MediaIndependent_DestinationAddress,
               AddressFamily
               )

T2_Receive     (
               PDU,
               ParsingMethod,
               SourcePort,
               DestinationPort,
               MediaIndependent_SourceAddress,
               MediaIndependent_DestinationAddress,
               AddressFamily
               )

T2_Allocate    (
               Port,
               ReceiveProcessPointer
               )
```

The T2_Send primitive can be passed from the upper layer to the transport/network layer to request that a PDU be sent to remote application defined by a port number and address. The T2_Receive primitive can be passed from the transport / network layer to the upper layer to indicate the arrival of a PDU from a remote application defined by a port number and address. The T2_Allocate primitive can be passed from the upper layer to the transport/network layer to request that a port number be allocated to or de-allocated from the indicated receiving process. The parameters of the primitives are defined to cover all encapsulation and parsing methods and an address translation mechanism. They may not be applicable to all implementations.

The PDU parameter in the T2_Send primitive is the information field to be encapsulated. The PDU parameter in the T2_Receive primitive is the information field that was un-encapsulated.

The EncapsulationMethod shall be indicative of how the Transportation Transport Protocol is to encapsulate the Upper Layer PDU before passing it down to the subnetwork. As defined in Section 2, there are four possible methods.

The ParsingMethod shall be indicative of how the Transportation Transport Protocol is to un-encapsulate the Lower Layer PDU into the various components before passing it up to the application profile. As defined in Section 2, there are four possible methods. An application layer would use it to determine how any response should be encapsulated.

The SourcePort, if present, is the port number assigned to the entity that is sending the PDU.

The DestinationPort, if present, is the port number assigned to the entity that is to receive the PDU.

The MediaIndependent_SourceAddress is indicative of the logical address of the entity sending the PDU. The exact format is determined by the Address Family.

The MediaIndependent_DestinationAddress is indicative of the logical address of the entity (or entities) to receive the PDU. The exact format is determined by the Address Family.

The AddressFamily is indicative of the form of logical address used to express the Source and Destination Address parameters. For example, IPv4 uses a 4-byte addressing scheme whereas IPv6 uses a 16-byte scheme.

In the T2_Allocate primitive, the Port parameter, when set equal to zero, indicates that an unused port number should be assigned or allocated to the ReceiveProcessPointer. When the Port parameter is not equal to zero, it indicates that any port number for the associated ReceiveProcessPointer should be de-allocated.

B.3 LOWER LAYER INTERFACING

The primitives and parameters of the services at the interface between transportation transport protocol/profile and the lower layers are described as follows:

```
DL_Request    (
              MediaDependent_SourceAddress,
              MediaDependent_DestinationAddress,
              IPI,
              PDU
              )
```


DL_Indication (
MediaDependent_SourceAddress,
MediaDependent_DestinationAddress,
PDU
)

DL_Request primitive is passed from the transport/network layer to the data link layer to request that a PDU be sent to a remote media dependent addresses. The Initial protocol Identifier (IPI) and the T2 Protocol Data Unit (PDU) form the information field of the lower layer.

The DL_Indication primitive is passed from the data link to the transport/network layer to indicate the arrival of a PDU from the specified media dependent remote entity. The first byte of Data Link Layer Information Field is the IPI indicating the process to call (T2) and is discarded. The remaining bytes of the Information Field are passed as the PDU.

The MediaDependent_SourceAddress is indicative of the physical address of the entity that sent the PDU. This may or may not be known. The exact format is subnetwork dependent.

The MediaDependent_DestinationAddress is indicative of the physical address of the entity (or entities) to receive the PDU. The exact format is subnetwork dependent.

The exact process of how a media independent source or destination address is converted to a media dependent address is address family and subnetwork dependent and not prescribed. In the case where the address family indicates an IP Address as the media independent address type, for example, the technique that the Internet Protocols could be used. The IP Address is the next hop address and is passed to a NetToMedia Table. The IP Address is the `ipNetToMediaNetAddress` index that points to the `ipNetToMediaIfIndex` or I/O port and driver to send over and the `ipNetToMediaPhysAddress` is the Physical Address that is passed to the Data Link Layers as the media dependent destination address. An additional column could be implemented to provide media independent source address conversion.

< This page is intentionally left blank. >

Annex C
ENCAPSULATION EXAMPLES
(Informative)

The following figures provide examples of the three T2 encapsulation methods.

The first, two examples illustrate Encapsulation Method 1. This method is only used with SNMP GetRequest, GetNextRequest, GetResponse, or SetRequest PDU Types or STMP. Figure C-1 shows the encapsulation of an get, set, or getnext into a T2 PDU and then a PMPP Frame. The SNMP PDU is passed to the T2 layer and becomes the T2 PDU. No additional fields are added in this method because the first byte of the SNMP PDU (Sequence Tag) serves as the Application Identifier (AID). The appropriate T2 Initial Protocol Identifier (IPI) is added to the T2 PDU when it is passed down is to the Data Link Layer. For the PMPP Protocol the IPI is 0xC1. The T2 Frame is then embedded in the PMPP Frame as shown. This method provides compatibility with the original Class B Communications Profile.

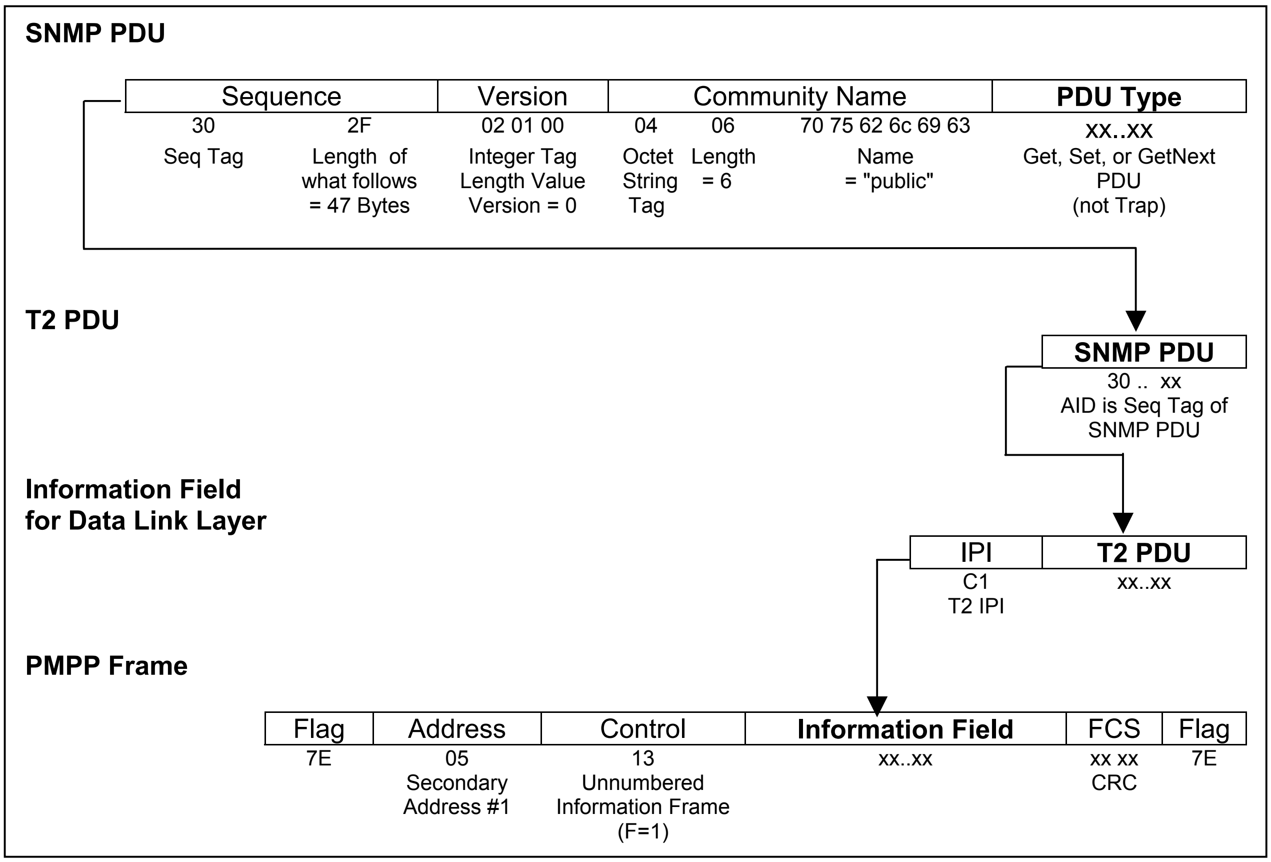

Figure C-1
SNMP GetRequest, GetNextRequest, GetResponse, or SetRequest Encapsulation

Parsing Method 1 corresponds to Encapsulation Method 1. If the first byte of the T2 PDU (after the IPI Byte) is 0x30, the entire T2 PDU is to be sent to SNMP Message processing entity. The implied port numbers in this case would be 161 (0xA1).

Encapsulation Method 1 also applies to STMP. Figure C-2 shows the encapsulation for an STMP SetRequest-NoReply for Dynamic Message 1 (assumed to be GlobalTime). The STMP PDU is passed to the T2 layer and becomes the T2 PDU. No additional fields are added in this method because the first byte of the STMP PDU (PDU Header) serves as the Application Identifier (AID). The appropriate T2 Initial Protocol Identifier (IPI) is added to the T2 PDU when it is passed down is to the Data Link Layer. For the PMPP Protocol the IPI is 0xC1. The T2 Frame is then embedded in the PMPP Frame as shown. This method provides compatibility with the original Class B Communications Profile.

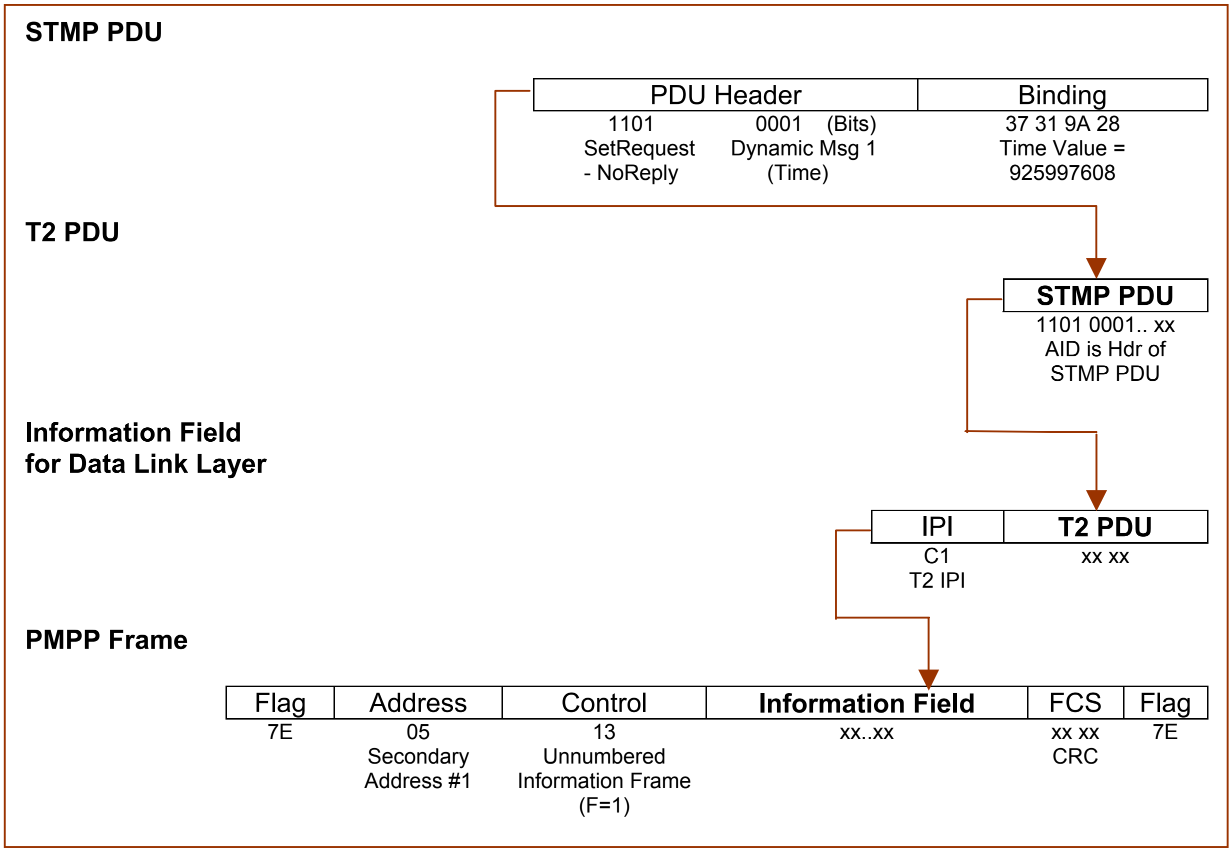

Figure C-2
STMP Encapsulation

Parsing Method 3 also corresponds to Encapsulation Method 1. If the first byte of the T2 PDU (after the IPI Byte) is the range of 0x0x81-0xFD, the entire T2 PDU is to be sent to the STMP Message processing entity. In this case, the implied port numbers would be 501 (0x1F5).

The third example illustrates Encapsulation Method 2 into a PMPP Frame. This method can be used with any application that uses a connectionless-oriented transport service. Even though a TFTP Read Request PDU is shown, the same encapsulation method would apply to any SNMP message type (including Traps), STMP, or other similar protocol. For SNMP Get, Set, and GetNext, the port number values would be 161 (0xA1). For SNMP Traps, the port number values would be 162 (0xA2). For TFTP, the port number values would be 69 (0x45).

The Upper Layer PDU along with the appropriate port number information is passed to the T2 layer . The Application Identifier (0x41) is added to indicate that the specific application protocol is defined by the source and destination port numbers that follow. This information defines the T2 PDU. The appropriate T2 Initial Protocol Identifier is added to the T2 PDU when it is passed down is to the Data Link Layer. For the PMPP Protocol the IPI is 0xC1. The T2 Frame is then embedded in the PMPP Frame as shown.

The flexibility of this method is that any Upper Layer PDU can be routed or multiplexed over the T2 Protocol if the application layer has an assigned or well-know port number associated with it. The operation is effectively the same as the one used in the Internet's User Datagram Protocol (UDP). Although it not mandatory to use this encapsulation method, it is the preferred method because it is the same across for all applications.

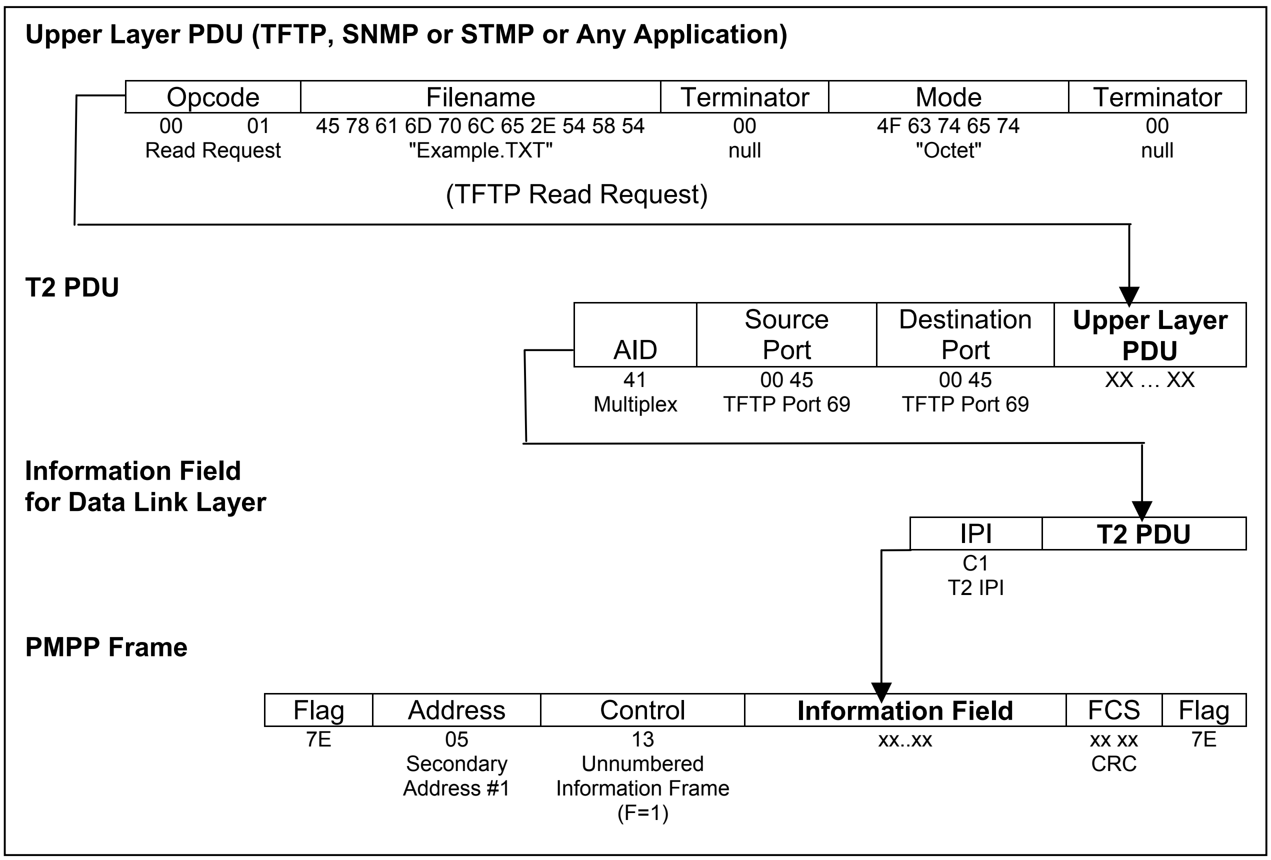

Figure C-3
Any Upper Layer Protocol Encapsulation

Parsing Method 4 corresponds to Encapsulation Method 2. If the first byte of the T2 PDU (after the IPI Byte) is 0x41, then the Upper Layer PDU begins after the destination port field and is sent to the application protocol indicated by the destination port number.

The fourth example illustrates Encapsulation Method 3 into a PMPP Frame. This method is only used with SNMP Traps. The SNMP Trap PDU is passed to the T2 Layer where the Application Identifier (0x31) is added. The appropriate T2 Initial Protocol Identifier is added to the T2 PDU when it is passed down to the Data Link Layer. For the PMPP Protocol the IPI is 0xC1. The T2 Frame is then embedded in the PMPP Frame as shown. This method is only used with STMP.

The original Class B Communications Profile did not provide a scheme for handling SNMP Traps. This method provides a means for specifically designating that an Upper Layer PDU is an SNMP Trap and should be routed accordingly.

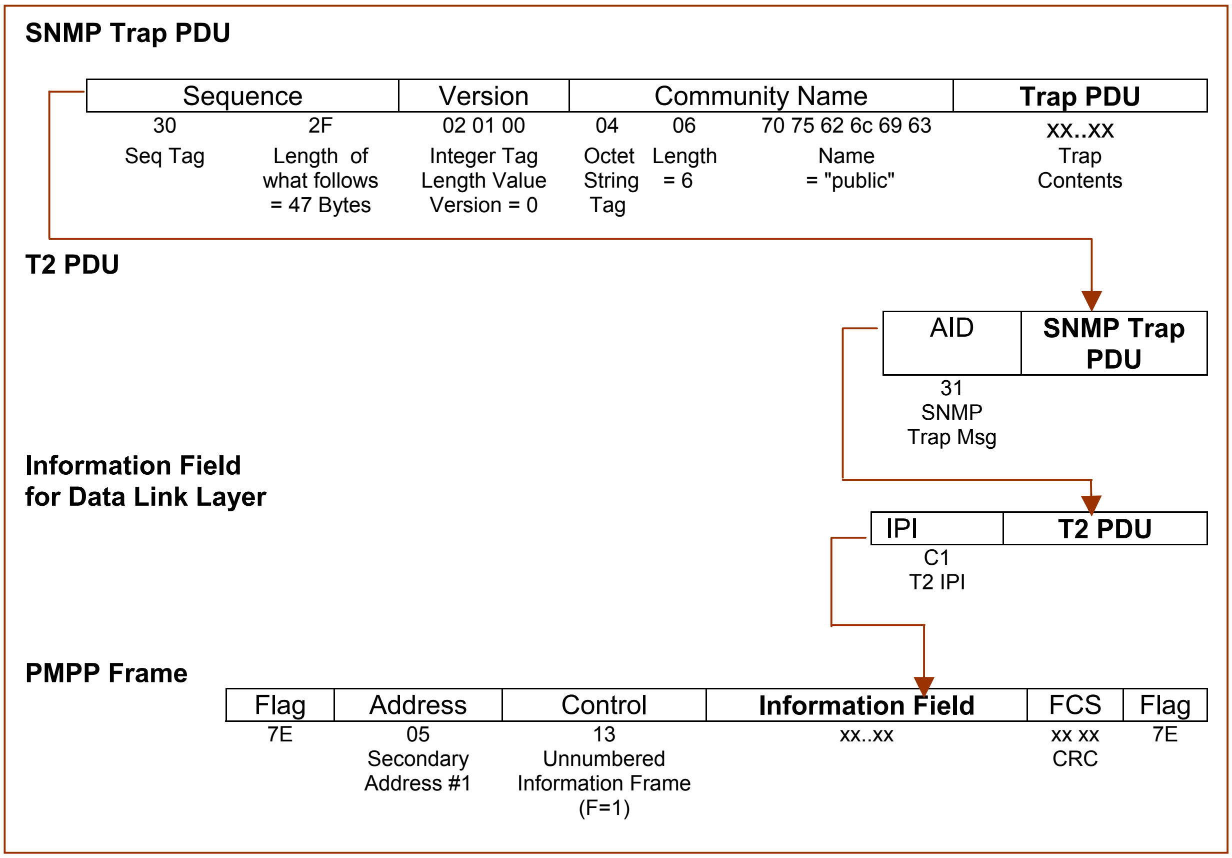

Figure C-4
SNMP Trap PDU Encapsulation

Parsing Method 2 corresponds to Encapsulation Method 3. If the first byte of the T2 PDU (after the IPI Byte) is 0x31, then the Upper Layer PDU begins with the next byte and is sent to the SNMP Trap processing entity. The implied port numbers in this case would be 162 (0xA1).

§

