CLASS C

PROFILE

Prepared by:
NTCIP Steering Group

May 1996

Table of Contents

FOREWORD....................i

Section 1: GENERAL 1-1

1.1 SCOPE 1-1
1.1.1 Background....................1-1
1.1.2 Purpose of Document 1-1

1.2 REFERENCES....................1-2
1.2.1 Normative References 1-2
1.2.2 Other References....................1-2
1.2.3 Contact Information 1-3

1.3 DEFINITIONS 1-4

1.4 ABBREVIATIONS AND ACRONYMS....................1-5

Section 2: CLASS C PROFILE OVERVIEW 2-1

2.1 GENERAL ASPECTS 2-1
2.1.1 Optional Features 2-1
2.1.2 Stack Definition....................2-1
2.1.3 Co-residence 2-1

2.2 THE TRANSPORT LAYER 2-1

2.3 THE APPLICATION LAYER....................2-2
2.3.1 SNMP....................2-2
2.3.1 The FTP Model....................2-2
2.3.3 Data Transfer Functions....................2-3

Section 3: CLASS C PROFILE REQUIREMENTS....................3-1

3.1 GENERAL....................3-1
3.2 PROTOCOL STACK....................3-1
3.2.1 Physical Layer Definition....................3-1
3.2.2 Data Link Layer Definition....................3-1
3.2.3 Network Layer Definition....................3-1
3.2.4 Transport Layer Definition....................3-1
3.2.5 Session Layer Definition 3-2
3.2.6 Presentation Layer Definition 3-2
3.2.7 Application Layer Definition 3-2

Table of Figures

FIGURE 2-1. MODEL FOR FTP USE....................2-2
FIGURE 2-2. REMOTE CONTROL OF FILE TRANSFER....................2-3

FOREWORD

The purpose of this document is to provide a definition of Class C, one of the component standards of the National Transportation Communications for ITS Protocol (NTCIP). It is intended as a communications protocol standard for interconnecting transportation and traffic control equipment. NTCIP represents a suite of protocols and information management standards that apply to the entire industry. The Class C Profile addresses the need for a specific protocol that is geared to the communications requirements of field devices such as highway advisory radios, traffic controllers, variable message signs, and other similar devices that may use file transfer. It is, by design, implementable in current, state-of-the-art field devices that are part of traffic control or traffic management systems.

The original effort in the development of NTCIP began with the NEMA traffic control equipment manufacturers' desire to extend the TS-2 Standard for traffic control hardware to cover the more complex issues of systems interoperability and communications standards. The initial goal was to develop a common protocol and define a minimum set of control and status messages so that end users could intermix not only different manufacturers' controllers and 170-type controllers, but also all other field-related devices. Currently, it is being extended to cover intra- and inter- traffic management and control center communications. The ultimate goal is to define standard protocols for all aspects of communications for transportation and traffic control networks that are needed for ITS.

The figure below depicts the current NTCIP protocol suite family; other protocol suites will be added on an as needed basis. The shaded block indicates the focus of this document.

		PROFILES			
		Class B	Class A	Class C	Class E
PROTOCOLS	Application	STMF	STMF	Telnet FTP SNMP	Telnet FTP SNMP
	Presentation	Null	Null	Null	Null
	Session	Null	Null	Null	Null
	Transport	Null	UDP	TCP	TCP
	Network	Null	IP	IP	IP
	Data Link	PMPP	PMPP	PMPP	PPP
	Physical	EIA 232E FSK	EIA 232E FSK	EIA 232E FSK	EIA 232E

In preparation of this Standards Publication, input of users and other interested parties has been sought and evaluated. Inquiries, comments, and proposed or recommended revisions should be submitted to:

Attn: Alberto J. Santiago
NTCIP Information /HSR-11
Turner-Fairbank Highway Research Ctr.
6300 Georgetown Pike
McLean, VA 22101
fax: (703) 285-2264

Section 1
GENERAL

1.1 SCOPE

The National Transportation Communications for ITS Protocol (NTCIP) establishes a common method of interconnecting Intelligent Transportation Systems (ITS) and traffic control equipment. It establishes the protocol and procedures for establishing communications between ITS-related components, defines procedures to develop and register a common set of messages related to controlling and managing an ITS, and defines a set of messages specifically related to current traffic control systems. The Class C protocol stack described in this document is appropriate for the reliable exchange of data between field devices and controllers that are on separate subnetworks; the primary feature of this profile is reliable data transfer and support for routing.

1.1.1 Background

As transportation control equipment becomes more sophisticated, planners, users, and equipment manufacturers have recognized the need to establish system inter-operability and integration standards. Different agencies have defined protocols to work with their specific hardware but a problem arises when attempting to integrate additional or other existing hardware into a system. There is no common protocol with which the equipment can communicate. The various systems utilize different synchronizing techniques, data formats, and error-recovery schemes. If a national ITS program is to be implemented then communications standards and protocols must be established.

The problem is exemplified by considering just the integration issues related to traffic signal controllers. Currently there is no standardized method to communicate with either National Electrical Manufacturers Association (NEMA) -type controllers or 170-type controllers. In systems that use a remote communications unit to interface with the input/output backpanel signals of the controllers, severe limitations are imposed on the capabilities of the system. Alternately, when systems from different manufacturers are integrated into the same central control system, the communications protocol to each system is different and manufacturer specific. Local controller units operating within the system normally have to be from the same manufacturer of the system.

The integration problem is compounded when existing systems are tied into transportation management systems. Not only do these systems have to deal with traffic controllers, but they must also handle such devices as surveillance cameras, variable message signs, and other ITS-related hardware. Ideally, all of this equipment should be able to share the same communications channel. Industry-wide communications standards are needed not just for connectivity and interoperability reasons, but also to accommodate future technology growth. As the needs of ITS become clearer, the ability to transfer data throughout the system will become crucial. A communications standard must accommodate the presently recognized needs as well as the yet undefined needs of the future.

1.1.2 Purpose of Document

This document provides the information necessary to build a functional Class C protocol stack on ITS devices. This document outlines the protocols used at each layer of the NTCIP Class C stack, defines the parameters for each of these protocols, and provides settings for each of the parameters. The document does not completely describe the requirements of each of the selected protocols; implementors should obtain the appropriate normative references to ensure that any implementation is compliant with the specification requirements and with NTCIP in general.

1.2 REFERENCES

1.2.1 Normative References

The following standards contain provisions that, through reference in this text, constitute provisions of this Standard. While end users of NTCIP do not need to obtain these documents, they do provide a complete understanding of the protocol. 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. Contact information regarding the referenced standards is given at the end of this section.

NTCIP Steering Group Draft — *Point to Multi-Point Protocol (PMPP)*

NTCIP Steering Group Draft — *Class B Profile*

EIA/TIA-232-C, *Interface Between Data Terminal Equipment and Data Communications Equipment Employing Serial Binary Data Interchange*, 1969

RFC: 765, *File Transfer Protocol (FTP),* October 1985

RFC 791, *Internet Protocol (IP),* September 1981

RFC 792, *Internet Control Message Protocol (ICMP),* September 1981

RFC: 793, *Transmission Control Protocol (TCP),* September 1981

RFC: 854, *Telnet Protocol Specification,* May 1983

RFC 1155, *Management Information Base for Network of TCP/IP-based Internets*

RFC 1157, *A Simple Network Management Protocol (SNMP)*

1.2.2 Other References

The following list of documents may also be helpful to the reader in gaining a full understanding of the NTCIP.

ITU-T Recommendation X.210 | ISO/IEC 10731, *Information technology ¾ Open Systems Interconnection ¾ Conventions for the definition of OSI services.*

CCITT Recommendation X.213 (1992) | ISO/IEC 8348:1992, *Information technology ¾ Network service definition for Open Systems Interconnection.*

CCITT Recommendation X.200 (1988), *Reference model of Open Systems Interconnection for CCITT Applications.*

ISO 7498:1984, *Information processing systems ¾ Open Systems Interconnection ¾ Basic Reference Model.*
(The above pair of documents are basically the same in content and text. They will be superseded by a new edition to be published by end of the Summer 1994. At that time, there will only be one version in "identical text" format and the reference will then appear in 1.2.1 above.)

CCITT Recommendation X.212 (1988), *Data Link service definition for Open Systems Interconnection for CCITT applications.*

ISO/IEC 8886:1992, *Information technology ¾ Telecommunications and information exchange between systems ¾ Data link service definition for Open Systems Interconnection.*

International Technical Support Center Raleigh N.C., *TCP/IP Tutorial and Technical Overview*, Document Number GG24-3376-01, IBM Corp, Armonk, NY, June 5, 1990, 2nd Edition.

1.2.3 Contact Information

1.2.3.1 ISO/IEC Standards

Members of the ISO maintain registers of currently valid ISO/IEC International Standards. For the United States of America, the member of ISO is the American National Standards Institute (ANSI), which may be contacted as follows:

ANSI
11 West 42nd Street, 13th Floor
New York, New York 10036
(212) 642-4900

1.2.3.2 ITU-T Recommendations

The ITU-T Secretariat Bureau (ITU-TSB), formerly CCITT, maintains a list of the currently valid ITU-T Recommendations. The ITU-TSB may be contacted as follows:

International Telecommunications Union
Place des Nations
CH-1211
Geneva 20
Switzerland
+41-22-730-5285

1.2.3.3 RFC Documents

Pertinent RFC's may be obtained by writing to:

DDN Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA 22021

Electronic copies of RFC documents may be obtained using "anonymous FTP" to the host nic.ddn.mil or ds.internic.net.

1.3 DEFINITIONS

connection: A logical communication path identified by a pair of sockets.

data connection: A simplex connection over which data is transferred, in a specified mode and type. The data transferred may be a part of a file, an entire file or a number of files. The path may be between a server-DTP and a user-DTP, or between two server-DTPs.

data port: The passive data transfer process "listens" on the data port for a connection from the active transfer process in order to open the data connection.

file: An ordered set of computer data (including programs), of arbitrary length, uniquely identified by a pathname.

FTP commands: A set of commands that comprise the control information flowing from the user-FTP to the server-FTP process.

host: A computer that is a source or destination of messages from the point of view of the communications network.

Identification: An Internet Protocol field. This identifying value assigned by the sender aids in assembling the fragments of a datagram.

IP: Internet Protocol.

mode: The manner in which data is to be transferred via the data connection. The mode defines the data format during transfer including EOR and EOF. The transfer modes defined in FTP are described in Section 0.

NVT: The Network Virtual Terminal as defined in the TELNET Protocol.

port: The portion of a socket that specifies which logical input or output channel of a process is associated with the data.

process: A program in execution. A source or destination of data from the point of view of the TCP or other host-to-host protocol.

PUSH: A control bit occupying no sequence space, indicating that this segment contains data that must be pushed through to the receiving user.

segment: A logical unit of data, in particular a TCP segment is the unit of data transferred between a pair of TCP modules.

server-DTP: The data transfer process, in its normal "active" state, establishes the data connection with the "listening" data port, sets up parameters for transfer and storage, and transfers data on command from its PI. The DTP can be placed in a "passive" state to listen for, rather than initiate a, connection on the data port.

server-FTP process: A process or set of processes that performs the function of file transfer in cooperation with a user-FTP process and, possibly, another server. The functions consist of a protocol interpreter (PI) and a data transfer process (DTP).

server-PI: The portion of a server-FTP that "listens" on Port L for a connection from a user-PI and establishes a TELNET communication connection. It receives standard FTP commands from the user-PI, sends replies, and governs the server-DTP.

socket: An address which specifically includes a port identifier, that is, the concatenation of an Internet Address with a TCP port.

TELNET connections: The full-duplex communication path between a user-PI and a server-PI, operating according to the TELNET Protocol.

user-DTP: The portion of a user-FTP that "listens" on the data port for a connection from a server-FTP process. If two servers are transferring data between them, the user-DTP is inactive.

user-FTP process: A set of functions including a protocol interpreter, a data transfer process and a user interface, which performs the function of file transfer in cooperation with one or more server-FTP processes. The user interface allows a local language to be used in the command-reply dialogue with the user.

user-PI: The portion of the user-FTP that initiates the TELNET connection from its port U to the server-FTP process, initiates FTP commands, and governs the user-DTP if that process is part of the file transfer.

user: A person or a process on behalf of a person wishing to obtain file transfer service. The person may interact directly with a server-FTP process, but use of a user-FTP process is preferred because the protocol design is weighted towards automata.

1.4 ABBREVIATIONS AND ACRONYMS

ANSI	American National Standards Institute
API	Application Programmer's Interface
BPS	Bits per Second
CCITT	International Telegraph and Telephone Consultative Committee
DARPA	Defense Advanced Research Projects Agency
DLSDU	Data Link Service Data Unit
EIA	Electronic Industries Association
EOF	End of File
EOR	End of Record
FCS	Frame Check Sequence
FHWA	Federal Highway Administration
FSK	Frequency Shift Keying
HDLC	High-level Data Link Control
IANA	Internet Assigned Number Authority
ICMP	Internet Control Message Protocol
IEC	International Electro-technical Commission
IETF	International Engineering Task Force
IP	Internet Protocol
IPI	Initial Protocol Identifier
ISO	International Organization for Standardization
ITS	Intelligent Transportation Systems
ITU-T	International Telecommunications Union, Telecommunications Sector
NEMA	National Electrical Manufacturers Association
NSF	National Science Foundation
NTCIP	National Transportation Communications for ITS Protocol
OSI	Open Systems Interconnection
PI	Protocol Interpreter
PMPP	Point to Mulit-Point Protocol
PPP	Point-to-Point Protocol
SDU	Service Data Unit
SNMP	Simple Network Management Protocol
STMF	Simple Transportation Management Framework
STMP	Simple Transportation Management Protocol
TCP	Transmission Control Protocol
TIA	Telecommunications Industries Association
UCC	Unbalanced Connectionless Class
UDP	User Datagram Protocol
UI	Unnumbered Information
UP	Unnumbered Poll

Section 2
CLASS C PROFILE OVERVIEW
(Authorized Engineering Information)

2.1 GENERAL ASPECTS

The Class C profile is established to facilitate the connection and control of field devices with sufficient processing power and communications capability to support a higher overhead protocol. This protocol can support reliable data transfer and routing between network nodes. There is no specific required architecture, but the underlying assumption is that there is a Primary/Secondary relationship between devices and controllers. While the protocol will function on low speed communications links, the amount of overhead necessary to support the routing functions limit its practicality on these links.

A more complete discussion of the applications of this class profile can be found in NEMA TS 3-1.

2.1.1 Optional Features

A particular implementation may add to this standard, but it must also provide the capability stated herein. For example, a product may implement fiber in its Physical Layer, but to meet standard, it must also support either EIA/TIA-232-E or FSK.

2.1.2 Stack Definition

The Class B Profile is the protocol stack defined for transportation management applications. It consists of the seven (7) ISO layers as defined in 3.2.1 through 3.2.7.

2.1.3 Co-residence

Multiple profiles may reside in the same physical unit. If a physical unit can meet the protocols for Classes A, B, and C, then it conforms to each of the three profiles.

2.2 THE TRANSPORT LAYER

The Class C Profile uses the Transmission Control Protocol (TCP, RFC 793) to provide the transport layer. To provide a reliable connection between pairs of processes is the primary purpose of TCP.

The protocol provides the following services for its using applications:

a. Stream Data Transfer: It transfers a contiguous stream of bytes from the originator to the destination. It provides the necessary segmenting and reassembly and has a PUSH function to ensure that the last data sent is received.
b. Reliability: It provides an acknowledgment and retransmission facility to ensure that all data sent is received. The data is sequenced so that the data can be reassembled at the receiving end and missing data can be detected and recovered.
c. Flow Control: The receiving host returns a value that indicates how many more bytes of data it can receive without overflowing.
d. Full Duplex: TCP can send and receive concurrent data at the same time.

2.3 THE APPLICATION LAYER

The application layer provides the interface to the user and it in turn uses the layers below to communicate with an application on another device. The type of application will drive which set of protocols will be used. The Class C Application provides the set of application program interfaces (APIs) for the user's software. It is composed of Simple Network Management Protocol (SNMP), the *File Transfer Protocol (FTP),* and the *Telnet Protocol Specification.*

2.3.1 SNMP

The concept of SNMP is defined in RFC 1157.

2.3.2 The FTP Model

The objectives of FTP in this profile are:

a. to permit sharing of files (computer programs and/or data),
b. to shield a user from variations in file storage systems among hosts, and
c. to transfer data reliably and efficiently.

FTP, though usable directly by a user at a terminal, is designed primarily for use by programs.

In the model described in Figure 2-1, the user-protocol interpreter initiates the TELNET connection. At the initiation of the user, standard FTP commands are generated by the user-PI and transmitted to the server process via the TELNET connection. Standard replies are sent from the server-PI to the user-PI over the TELNET connection in response to the commands.

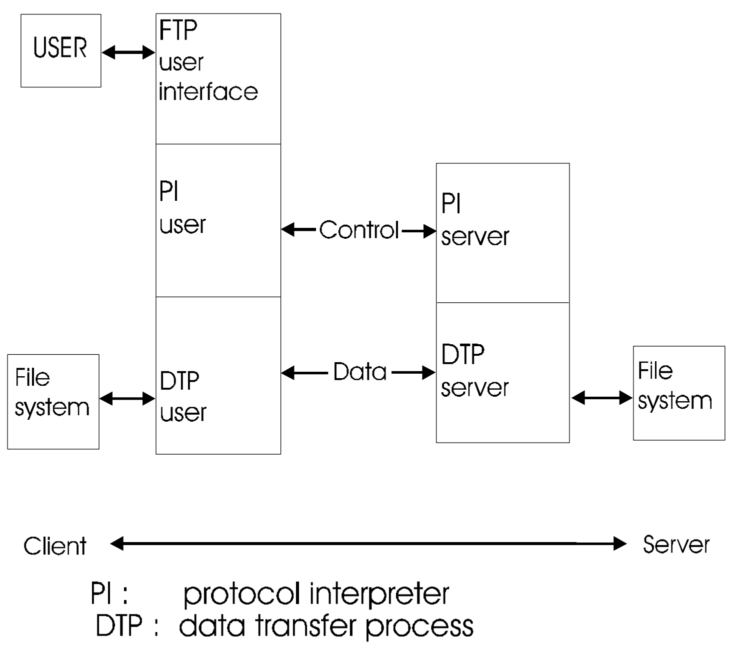

Figure 2-1. Model For FTP Use

The FTP commands specify the parameters for the data connection (data port, transfer mode, representation type, and structure) and the nature of file system operation (store, retrieve, append, delete, etc.). The user-DTP or its designate should "listen" on the specified data port, and the server should initiate the data connection and data transfer in accordance with the specified parameters. It should be noted that the data port need not be in the same host that initiates the FTP commands via the TELNET connection, but the user or the user-FTP process must ensure a "listen" on the specified data port. It should also be noted that the data connection may be used for simultaneous sending and receiving.

In another situation, a user may wish to transfer files between two hosts, neither of which is the user’s local host. TELNET connections are set up to the two servers and a data connection is established between them. In this manner control information is passed to the user-PI but data is transferred between the server data transfer processes. TRANSFER is a model of this server-server interaction.

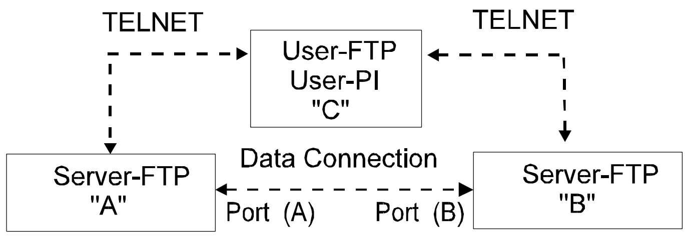

Figure 2-2. Remote Control of File Transfer

The protocol requires that the TELNET connections be open while data transfer is in progress. It is the responsibility of the user to request the closing of the TELNET connections when finished using the FTP service, while it is the server who takes the action. The server may abort data transfer if the TELNET connections are closed without command.

2.3.3 Data Transfer Functions

Files are transferred only via the data connection. The TELNET connection is used for the transfer of commands, which describe the functions to be performed, and the replies to these commands (see the Section on FTP Replies in RFC 765). Several commands are concerned with the transfer of data between hosts. These data transfer commands include the MODE command, which specifies how the bits of the data are to be transmitted, and the STRUcture and TYPE commands, which are used to define the way in which the data is to be represented. The transmission and representation are basically independent but "Stream" transmission mode is dependent on the file structure attribute. Also, if "Compressed" transmission mode is used, the nature of the filler byte depends on the representation type.

Section 3
CLASS C PROFILE REQUIREMENTS

3.1 GENERAL

In referenced RFC documents, all instances of the word “will” in clauses or sections defining required behavior shall be interpreted as the mandatory “shall”; all instances of the word “may” shall be interpreted as the mandatory “must”. Any exceptions to this will be explicitly defined in the profile/specification requirements clauses.

For the purposes of this profile, implementors must assume that all optional features documented in the referenced specifications or RFC’s are NOT present; however, this should not be construed to restrict implementation of any optional feature.

3.2 PROTOCOL STACK

The protocol stack shall be as defined in 3.2.1 through 3.2.7.

3.2.1 Physical Layer Definition

The Physical Layer shall conform to at least one of the interfaces defined in NTCIP Steering Group - Class B Profile, 3.2.1.

3.2.2 Data Link Layer Definition

The Data Link Layer shall conform to the NTCIP Steering Group Draft - Point to Multi-Point Protocol (PMPP).

The IPI shall specify: “IP”, i.e., the IPI value shall be 0x’21’.

3.2.3 Network Layer Definition

The Network Layer shall conform to the Internet Protocol as defined in RFC 791.

The protocol field shall specify TCP, 0x’06’.

3.2.4 Transport Layer Definition

The Transport Layer shall conform to the Transmission Control Protocol (TCP) as specified in RFC 793.

The well-known TCP ports that shall be supported are:

Content	Port
FTP Data	20
FTP Control	21
SNMP	161
SNMP Trap	162
STMP	????[1]
STMP Trap	????

3.2.5 Session Layer Definition

The Session Layer shall be a NULL layer.

3.2.6 Presentation Layer Definition

The Presentation Layer shall be a NULL layer.

3.2.7 Application Layer Definition

The Application Layer shall conform to the SNMP Protocol as specified in RFC 1157 and the File Transfer Protocol as specified in RFC 765. The File Transfer Protocol in turn specifies the use of the Telnet Protocol (RFC 854).

SNMP requests and responses will use SNMP Port 161; SNMP Traps will use SNMP Port 162. The File Transfer protocol will use TCP Port 21 for control; and TCP Port 20 for data.

[1] The STMF Port numbers are in the process of being assigned.