A Joint Standard of AASHTO, ITE, and NEMA

NTCIP 1204 version v03

National Transportation Communications for ITS Protocol

Environmental Sensor Station (ESS) Interface Protocol

published in October 2009

This Adobe® PDF copy of an NTCIP standard 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 22209-3801

file version 1204 v03.08r2 rita

TRF Distribution Permission

A Testing Requirements Form (“TRF”) may be a Testing Requirements Traceability Table and/or Test Procedures. To the extent that these materials are distributed by AASHTO / ITE / NEMA in the form of a TRF, AASHTO / ITE / NEMA extend the following permission:

a) you may make and/or distribute unlimited electronic or hard copies, including derivative works of the TRF, provided that each copy you make and/or distribute contains the citation “Based on NTCIP 0000 [insert the standard number] TRF. Used by permission. Original text © AASHTO / ITE / NEMA.”;
b) you may not modify the logical flow of any test procedure, without clearly noting and marking any such modification; and
c) if the TRF excerpt is made from an unapproved draft, add to the citation “TRF excerpted from a draft standard containing preliminary information that is subject to change.”

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

< This page is intentionally left blank. >

ACKNOWLEDGEMENTS

NTCIP 1204 v03 was prepared by the NTCIP Environmental Sensor Station Working Group (ESS WG), which is a subdivision of 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.

When NTCIP 1204 v03 was prepared, the following individuals were members of the NTCIP ESS WG:

- Steve Howe
- Liang Hsia
- Phil LaBanca
- Gene Martin (Chair)
- Curt Pape
- Ken Vaughn
- John Whited

Other individuals providing input to NTCIP 1204 v03 include:

- Russ Brookshire
- Blake Christie
- Ron Meyer
- Paul Pisano
- Dan Roosevelt
- John Stenz

In addition to the many volunteer efforts, recognition is also given to those organizations that supported the effort by providing comments and funding, including:

- U.S. Department of Transportation / Research and Innovative Technology Administration
- Aurora
- Campbell Scientific
- Castle Rock Consulting
- Colorado DOT
- Environment Canada
- Florida DOT
- Force America
- Ice and Snow Technologies
- Intelligent Devices, Inc.
- Iowa DOT
- Ministry of Ontario
- Minnesota DOT
- Noblis
- PBS&J
- Pennsylvania DOT
- Qualimetrics
- Surface Systems, Incorporated
- Trevilon Corporation
- Vaisala
- Virginia DOT
- Washington State DOT

FOREWORD

NTCIP 1204 v03, an NTCIP standards publication, identifies and defines how a management station may wish to interface with a field device to control and monitor pavement sensors, weather stations, air quality monitors, and other equipment related to the monitoring of and response to environmental conditions in an NTCIP-conformant fashion. NTCIP 1204 v03 uses only metric units.

NTCIP 1204 v03 is titled *Environmental Sensor Station (ESS) Interface Protocol* to express the multiple sections and annexes that are included in this standards publication. This NTCIP 1200-series standards publication has grown beyond the “object definitions” that were reflected in the title for the NTCIP 1204 versions v01 and v02.

NTCIP 1204 v03 contains four normative and four informative annexes:

a) Annex A is normative and contains a Requirements Traceability Matrix (RTM) that traces requirements to the dialogs and data elements used to fulfill it.
b) Annex B is informative and provides a graphical representation of the major nodes of the ISO tree as defined by NTCIP 1204 v03.
c) Annex C is normative and defines the test procedures for NTCIP 1204 v03.
d) Annex D is informative and identifies the significant revisions in NTCIP 1204 v03 that have been made since previous versions of NTCIP 1204 v03.
e) Annex E is informative and responds to user requests by providing an explanation as to how certain complex features can be supported by NTCIP 1204 v03 and why certain other features are not supported by NTCIP 1204 v03.
f) Annex F is normative and serves as a placeholder to define certain details that are likely to be moved to other standards at a future date.
g) Annex G is informative and provides sample encodings of block objects.
h) Annex H is normative and defines the controller configuration objects that affect the value of globalSetIDParameter whenever they are changed.

NTCIP 1204 v03 is also an NTCIP Data Dictionary standard. Data Dictionary standards provide definitions of data elements for use within NTCIP systems; they are approved by AASHTO, ITE, and NEMA through a ballot process, after a recommendation by the NTCIP Joint Committee. For more information about NTCIP standards, visit 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 standards publication’s content. A “Request for Interpretation” 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 22209-3801
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.

Approvals

NTCIP 1204 v03 was separately balloted and approved by AASHTO, ITE, and NEMA after recommendation by the Joint Committee on the NTCIP. Each organization has approved NTCIP 1204 v03 as the following standard type, as of the date:

AASHTO—Standard Specification; April 2008
ITE—Software Standard; October 2008
NEMA—Standard; October 2008

History

In 1992, the NEMA 3-TS Transportation Management Systems and Associated Control Devices Section began development of the NTCIP. The Transportation Section's purpose was in response to user needs to include standardized systems communication in the NEMA TS 2 standard, *Traffic Controller Assemblies*. 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. Under the guidance of a Joint AASHTO/ITE/NEMA Committee on the NTCIP, a Working Group was created to develop the object definitions for Environmental Sensor Stations. The first meeting of this working group was in November 1996, and the 1204 version 01 was produced in 1998. In 2001, efforts began to update and enhance the standard, which resulted in version 02, produced in 2005. In 2006, efforts began to update and enhance the standard with test procedures, which resulted in NTCIP 1204 v03.

NTCIP 1204 version 01 (v01) was published as NTCIP 1204:1998 and was also formerly known as NEMA TS 3.7. NTCIP 1204 v02 was developed to reflect lessons learned, to update to new documentation formats, and to add new features such as the control of automated de-icing equipment. NTCIP 1204 v03 was developed to add test procedures (Annex C) and to correct issues that had been identified with NTCIP 1204 v02.

NTCIP 1204 v03.03—In May 2007, the Joint Committee on the NTCIP accepted v03.03 as a User Comment Draft. Standards Bulletin B0121 sent v03.03c out for comment.

NTCIP 1204 v03.08—In January 2008, the Joint Committee on the NTCIP accepted v03.07 as a Recommended Standard. Standards Bulletin B01217 sent v03.07b out for ballot and approval. In January to May 2008, the ESS WG disposed of seven comments as editorial, and one comment as technical. The version v03.07d was revised to include the disposition of the seven comments, and with explanatory notes on the technical changes. To further prepare this version v03.08 for publication, portions were edited for purposes of clarity and consistency, and for grammar and format.

Compatibility of Versions

To distinguish NTCIP 1204 v03 (as published) from previous drafts, NTCIP 1204 v03 also includes NTCIP 1204 v03.08 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 1204 v03 is designated, and should be cited as, NTCIP 1204 v03. Anyone using NTCIP 1204 v03 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 NTCIP 1204 v03 should also consult NTCIP 8004 v02 for specific guidelines on compatibility.

INTRODUCTION

NTCIP 1204 v03 provides definitions of data elements for environmental sensor data, including weather data, pavement condition data, water level data, and air quality data. The data is defined using the Simple Network Management Protocol (SNMP) object-type format as defined in RFC 1212 and NTCIP format defined in NTCIP 8004 v02. This data would typically be exchanged between a management station and a field device using one of the NTCIP 1103 v01 recognized Application Layers (e.g., SNMP). The data may also be exchanged among management stations using other protocols.

NTCIP 1204 v03 defines requirements that are applicable to all NTCIP environments and it also contains optional and conditional sections that are applicable to specific environments for which they are intended.

The following keywords apply to NTCIP 1204 v03: AASHTO, ITE, NEMA, NTCIP, ESS, data, data dictionary, object, environmental sensor.

< This page is intentionally left blank. >

CONTENTS

FIGURES

TABLES

Section 1
GENERAL
[INFORMATIVE]

1.1 SCOPE

Environmental sensors include a wide array of sensors, including those that monitor weather, roadway surface, water level, and air quality conditions. These sensors are typically connected to a nearby microprocessor termed a remote processor unit (RPU). An environmental sensor station (ESS) consists of the RPU plus its suite of sensors.

Typically, this equipment is permanently located at a site along a travel corridor. In some cases, the "stations" may be portable, or even mobile. For NTCIP 1204 v03 purposes, all three types of stations are called ESS. In the transportation community, these devices are frequently used to improve roadway maintenance and traffic operations.

Environmental sensors are also frequently co-located with pavement treatment systems (PTS) and, in fact, may use the same controller. Thus, for NTCIP 1204 v03 purposes, the term ESS may also include a PTS.

NOTE—The PTS portion of NTCIP 1204 v03 may be placed in a separate standard in the future.

Unfortunately, there have not been standards defining how these devices communicate with management systems. As a result, each manufacturer has developed its own protocol to meet its own particular needs. This approach has resulted in systems that are not interchangeable or interoperable. If an agency wishes to use either a central management system or additional ESS from a different vendor, the agency encounters significant systems integration challenges, requiring additional resources to address. These additional resource requirements inhibit information sharing within and between various potential users of the data and prevent vendor independence. Without manufacturer independence, resource requirements further increase because of a lack of a competitive market.

These problems have not been limited to weather and environmental monitoring. Many other devices also need to exchange information. In surface transportation, examples include traffic signal controllers, dynamic message signs, bus priority sensors, etc. To address these problems, NTCIP is developing a family of open standards for communications between field devices and central management systems.

NTCIP 1204 v03 is part of that larger family and is designed to define an interoperable and interchangeable interface between a transportation management system and an ESS, while still allowing for extensions beyond NTCIP 1204 v03 to allow for new functions as needed. This approach is expected to support the deployment of ESS from one or more vendors in a consistent and resource-efficient way.

NTCIP 1204 v03 only addresses a subset of the requirements needed for procurement. It does not address requirements related to the performance of the sensors (e.g., accuracy, the supported detection range, the time it takes to detect conditions, etc.), hardware components, mounting details, etc.

NTCIP 1204 v03 standardizes the communications interface by identifying the various operational needs of the users (Section 2) and subsequently identifying the necessary requirements (Section 3) that support each need. NTCIP 1204 v03 then defines the NTCIP standardized communications interface used to fulfill these requirements by identifying the dialogs (Section 4) and related data concepts (Section 5) that support each requirement.

Traceability among the various sections is defined by the Protocol Requirements List (Section 3.3) and the Requirements Traceability Matrix (Annex A). Conformance requirements for NTCIP 1204 v03 are provided in Section 3.3.

An implementation of NTCIP 1204 v03 requires lower level services to structure, encode, and exchange the data concepts defined by NTCIP 1204 v03. NTCIP 1204 v03 assumes that the data concepts are exchanged by one of the protocols defined in NTCIP 2301 v02.

1.2 REFERENCES

Normative references contain provisions that, through reference in this text, constitute provisions of NTCIP 1204 v03. Other references in NTCIP 1204 v03 might provide a complete understanding of the entire protocol and the relations between all parts 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 standard listed.

1.2.1 Normative References

Glossary of Meteorology	*Glossary of Meteorology, Second Edition; American Meteorological Society, 2000*
AASHTO / ITE / NEMA NTCIP 1103 v01	*Transportation Management Protocols* published January 2009
AASHTO / ITE / NEMA NTCIP 1201:2005	*Global Object (GO) Definitions—version 02* published October 2005 NOTE—NTCIP 1201:2005 is referenced as “NTCIP 1201 v02.”
AASHTO / ITE / NEMA NTCIP 2301 v02	*Simple Transportation Management Framework (STMF) Application Profile (AP) (AP-STMF)* *publication anticipated*
WMO No. 306:1995	*Technical Regulations; Manual on Codes, International Codes, Volume 1.2, Annex II, FM 94-X Ext. BUFR—Binary Universal Form for the Representation of Meteorological Data*

1.2.2 Other References

IAB STD 16	(RFC 1155) *Structure and Identification of Management Information for TCP/IP-based Internets,* M. Rose, K. McCloghrie, May 1990, (RFC 1212) *Concise MIB Definitions*, M. Rose and K. McCloghrie, March 1991
National ITS Architecture, Version 5.0	*National ITS Architecture, FHWA, 2003*
AASHTO / ITE / NEMA NTCIP 2201:2003	*Transportation Transport Profile (T2)* published Sepptember 2005
AASHTO / ITE / NEMA NTCIP 2202:2001	*Transport Profile for Internet (TCP/IP and UDP/IP)* published December 2001
AASHTO / ITE / NEMA NTCIP 8004 v02	*Structure and Identification of Management Information (SMI)* *publication anticipated*

AASHTO / ITE / NEMA NTCIP 8007 v01	*Testing and Conformity Assessment Documentation within NTCIP Standards Publications* published May 2008
OMG Unified Modeling Language Specification, Version 1.5	*OMG Unified Modeling Language Specification,* Object Management Group, 2003
Publication No. FHWA-SA-98-008	*Assessing the Results of the Strategic Highway Research Program, FHWA, 1998* www.workzonesafety.org/research/record/82
Weather-Responsive Traffic Management	*Weather-Responsive Traffic Management – Concept of Operations. FHWA, 2003* ops.fhwa.dot.gov/weather/best_practices/WeatherConOps0103.pdf

1.2.3 Contact Information

1.2.3.1 American Meteorological Society & World Meteorological Organization (WMO) Documents

Prepayment is required prior to shipment of these documents. Printed copies are available from:

American Meteorological Society
45 Beacon Street
Boston, MA 02108
(617) 227-2425
www.ametsoc.org

1.2.3.2 Internet Documents

Obtain Request for Comment (RFC) electronic documents from several repositories on the World Wide Web, or by "anonymous" File Transfer Protocol (FTP) with several hosts. Browse or FTP to:

www.rfc-editor.org
www.rfc-editor.org/repositories.html
for FTP sites, read ftp://ftp.isi.edu/in-notes/rfc-retrieval.txt

1.2.3.3 National ITS Architecture

The National ITS Architecture may be viewed on-line at:

www.its.dot.gov/arch/index.htm

1.2.3.4 NTCIP Standards

Copies of NTCIP standards may be obtained from:

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

Draft amendments, which are under discussion by the relevant NTCIP Working Group, and amendments recommended by the NTCIP Joint Committee are available.

1.2.3.5 Object Management Group Documents

Copies of OMG standards may be obtained electronically from the Object Management Group at:

www.omg.org

1.2.3.6 Office of the Federal Coordinator for Metrology (OFCM) Documents

Copies of OFCM documents may be obtained from:

Office of the Federal Coordinator for Meteorology
8455 Colesville Rd., Suite 1500
Silver Spring, MD 20910
(301) 427-2002
www.ofcm.gov

1.3 GENERAL STATEMENTS

There are no general statements in NTCIP 1204 v03.

1.4 TERMS AND ACRONYMS

For the purposes of NTCIP 1204 v03, the following terms, definitions, acronyms, and abbreviations apply. Meteorological terms not defined in this section are in accordance with their definitions in the *Glossary of Meteorology*. Electrical and electronic terms not defined here are used in accordance with their definitions in IEEE Std 100-2000. English words not defined here or in IEEE Std 100-2000 are used in accordance with their definitions in *Webster's New Collegiate Dictionary*.

TERM	DEFINITION
Binary Universal Form for the Representation of Meteorological Data (BUFR)	The name of the WMO standard binary code for the exchange and storage of non-gridded meteorological data.
Compatible	The ability of two or more systems or components to exchange information. NOTE—See IEEE 610.12-1990.
Consistent	The ability of two or more systems or components to exchange information and use the supported information that has been exchanged and gracefully reject any unsupported information according to defined rules.
Current	Reflecting the conditions at the present time (or at the time at which the data is time stamped) as determined by the Controller.
Deprecated	In the context of a MIB, “deprecated” is an object STATUS value that indicates the object is valid in limited circumstances, but has been replaced by another. NOTE—This definition is modified from “Understanding SNMP MIBS.”To maintain multi-version interoperability (backward compatibility) for legacy implementations, objects with a STATUS value of “deprecated” may require support. When necessary to support legacy implementations, required support for objects with a STATUS value of “deprecated” is indicated using the PICS or Protocol Requirements List (PRL). See NTCIP 8004 v02.

TERM	DEFINITION
Environmental Monitoring Equipment Package	The component within a management subsystem which performs advanced processing of the collected environmental data. This would include the analysis, forecasting and packaging of weather and road condition information for resource management.
Environmental Sensor Station (ESS)	A location that includes a remote processor unit (RPU) connected to one or more sensors for the collection of environmental or meteorological data. It may also include a pavement treatment system (PTS). NOTE—The acronym ESS may also be used as a plural.
Feature	A behavior of the device.
Interchangeable	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. NOTE—See National Telecommunications and Information Administration, U.S. Department of Commerce
Interoperable	The ability of two or more systems or components to exchange information and use the information that has been exchanged. NOTE—See NTCIP 8004 v02.
Management Information Base (MIB)	Management information of object definitions so that devices on a network can be remotely monitored, configured and controlled. The information is provided in a format called Abstract Syntax Notation.1 (ASN.1), which is an international standard for defining objects.
Management Station	The computer system with which the device communicates. Typically, the management station commands and monitors the device.
Obsolete	In the context of a MIB, an object STATUS value that indicates the definition is no longer valid, was found to be flawed, was redundant or not useful. NOTE—In the next (or some future) edition of a standard, the object or group with a STATUS value of “obsolete” may be removed. This definition is modified from “Understanding SNMP MIBS.” See NTCIP 8004 v02.
Operator	The person who interfaces with the management station software, typically located at a control center.
Protocol	A specific set of rules, procedures, and conventions defining the format and timing of data transmissions between devices that are accepted and used to understand each other. NOTE—See NTCIP 8004 v02.

TERM	DEFINITION
Remote Processor Unit (RPU)	A field processor that collects data from sensors and can communicate the collected data to other computers; the processor may also process the collected data and/or control equipment.
Requirement	A requirement describes a condition or capability to which a system shall conform; either derived directly from user needs, or stated in a contract, standard, specification, or other normative document. A desired feature, property, or behavior of a system.
Requirements Traceability	The ability to follow or study the logical progression among the needs, requirements, and design details in a step-by-step fashion.
Return	When discussing device requirements for providing data when an external system requests it, the term “return” is understood to mean that the data is sent to the requester.
Road Weather Data Collection Market Package	A set of components that perform all operations related to sensing, collecting, processing, and exchanging environmental related information, including the exchange of data among the dispersedly located equipment.
Road/Weather Information System	The collection of RPUs and sensors connected to a central system for analysis and use by maintenance personnel and transportation system managers.
Sensor	A device that is capable of detecting a condition and reporting the result to an RPU.
Simple Network Management Protocol (SNMP)	A communications protocol developed by the IETF, used for configuration and monitoring of network devices.
Simple Transportation Management Framework (STMF)	Describes the organization of information within devices and methods of retrieving or modifying any information within the device. NOTE—STMF also explains how to generate and use computer-readable information organization descriptions. See NTCIP 8004 v02.
Specification	A document that references a standard either to define the capabilities of a component (e.g., a specification sheet) or to define the required capabilities of a component being procured (e.g., a procurement specification).
Sub-Feature	A specialization of a more generic feature.
Upload	To transfer information from the referenced device to the central computer or an attached portable computer.
User	A person using the system that is developed.
User Need	The business or operational problem (opportunity) that is to be fulfilled to justify procurement or use. NOTE—While this is termed a “user need” within the NTCIP community, it reflects needs of all stakeholders.

1.5 OTHER ABBREVIATIONS AND ACRONYMS

The abbreviations and acronyms used in NTCIP 1204 v03 are defined as follows:

ASN.1	Abstract Syntax Notation One
IANA	Internet Assigned Number Authority
IP	Internet Protocol
OID	OBJECT IDENTIFIER
PMPP	Point to Multi-Point Protocol
PRL	Profile Requirements List
PTS	Pavement Treatment System (or Systems)
T2	Transportation Transport Profile
TCP	Transmission Control Protocol
TMC	Traffic Management Center
TMP	Transportation Management Protocol
UDP	User Datagram Protocol
WG	Working Group

Section 2
CONCEPT OF OPERATIONS
[NORMATIVE]

Section 2 defines the user needs that subsequent sections within NTCIP 1204 v03 address. Accepted system engineering processes detail that requirements should only be developed to fulfill well-defined user needs. The first stage in this process is to identify the ways in which the system is intended to be used. In the case of NTCIP 1204 v03, this entails identifying the various ways in which transportation operations personnel may use ESS information to fulfill their duties.

This concept of operations provides the reader with:

a) a detailed description of the scope of this standard;
b) an explanation of how an ESS is expected to fit into the larger context of an ITS network;
c) a starting point in the procurement process; and
d) an understanding of the perspective of the designers of NTCIP 1204 v03.

Section 2 is intended for all readers of NTCIP 1204 v03, including:

a) transportation operations managers
b) transportation operations personnel
c) transportation engineers
d) system integrators
e) device manufacturers

For the first three categories of readers, Section 2 is useful to understand how ESS equipment can be used in their system. For this audience, Section 2 serves as the starting point in the procurement process, and enables these readers to become familiar with each feature covered by NTCIP 1204 v03 and determine whether that feature is appropriate for their implementation. If it is, then the procurement specification needs to require support for the feature and all of the mandatory requirements related to that feature.

For the last two categories of readers, Section 2 provides a more thorough understanding as to why the more detailed requirements (as specified later in NTCIP 1204 v03 exist.

2.1 TUTORIAL [INFORMATIVE]

A concept of operations describes a proposed system from the users' perspective. Typically, a concept of operations is used on a project to ensure that system developers understand users' needs. Within the context of NTCIP standards, a concept of operations documents the intent of each feature for which NTCIP 1204 v03 supports a communications interface. It also serves as the starting point for users to select which features may be appropriate for their project.

The concept of operations starts with a discussion of the current situation and problems that have led to the need to deploy systems covered by the scope of NTCIP 1204 v03 and to the development of NTCIP 1204 v03 itself. This discussion is presented in layman's terms such that both the potential users of the system and the system developers can understand and appreciate the situation.

The concept of operations then documents key aspects about the proposed system, including:

a) Reference physical architecture—The reference physical architecture defines the overall context of the proposed system and defines which specific interface is addressed by NTCIP 1204 v03. The reference physical architecture may be supplemented with one or more samples that describe how the reference physical architecture may be realized in an actual deployment.

b) Architectural Needs—The architectural needs section discusses the issues and needs relative to the system architecture that have a direct impact on NTCIP 1204 v03.
c) Features—The features identify and describe the various functions that users may want the device to perform. These features are derived from the high level user needs identified in the problem statement but are refined and organized into a more manageable structure that forms the basis of the traceability tables contained in Section 3 and Annex A.

The architectural needs and features are collectively called *user needs*. Section 3 uses these user needs in the analysis of the system to define the various functional requirements of an ESS. Each user need shall be traced to one or more functional requirements, and each functional requirement shall be derived from at least one user need. This traceability is shown in the Protocol Requirements List (PRL) as provided in Section 3.3.

While NTCIP 1204 v03 is intended to standardize communications across a wide range of deployments, it is not intended to mandate support for every feature for every deployment. Therefore, the PRL also defines each user need and requirement as mandatory, optional, or conditional. The only items marked mandatory are those that relate to the most basic functionality of the device. To obtain a device that meets specific needs, the user first identifies which optional needs are necessary for the specific project.

Each requirement identified is then presented in the Requirements Traceability Matrix (RTM) in Annex A, which defines how the requirement is fulfilled through standardized dialogs and data element definitions provided in Section 4 and Section 5.

A conformant device may support other user needs, as long as they are conformant with the requirements of NTCIP 1204 v03 and its normative references (see Section 1.2.1). For example, a device may support data that has not been defined by NTCIP 1204 v03; however, when exchanged via one of the NTCIP 2301 v02 protocols, the data shall be properly registered with a valid OBJECT IDENTIFIER under the Global ISO Naming Tree.

NOTE—Off-the-shelf interoperability and interchangeability can only be obtained by using well-documented user needs, along with their corresponding requirements and design, that are broadly supported by the industry as a whole. Designing a system that uses environments or features not defined in a standard or not typically deployed in combination with one another inhibits the goals of interoperability and interchangeability, especially if the documentation of these user needs is not available for distribution to system integrators. Standards allow implementations to support additional user needs to support innovation, which is constantly needed within the industry, but users should be aware of the risks involved with using such environments or features.

The concept of operations concludes by describing the degree to which security issues have been addressed by the NTCIP 1204 v03 and by providing a description of how NTCIP 1204 v03 relates to the National ITS Architecture.

2.2 CURRENT SITUATION AND PROBLEM STATEMENT [INFORMATIVE]

Transportation system managers use ESS in a variety of ways to improve transportation system operations. The primary uses of ESS data support the following:

a) sharing data with the broader weather community contributes to better weather forecasts
b) improved highway maintenance operations through supporting timely, accurate, and relevant weather forecasting and knowledge of existing road weather conditions
c) more accurate traveler information, which can result in better route planning by travelers and more effective, safer transportation system use

d) improved management of facilities maintenance resources, leading to more timely facilities clearance and improved traveler safety
e) more effective use of advisory and regulatory mechanisms to ensure public safety
f) enhanced monitoring of potential hazardous conditions, to improve transportation system security and traveler safety

Additional information about how this data can be used is provided in *Weather-Responsive Traffic Management—Concept of Operations.* One of the most common ESS deployed by transportation system managers is the road/weather ESS. These ESS are used to collect information about road and weather conditions, such as precipitation and air and surface temperatures. With the data returned by these ESS, transportation system managers can determine when there are incipient hazardous travel conditions because of precipitation, fog, high winds, snow, ice, and/or flooding. When travel is becoming hazardous because of snow and/or ice, transportation system managers can dispatch road maintenance crews to treat the roads and remove snow and ice if possible. Transportation system managers can also use ESS in conjunction with other Intelligent Transportation System (ITS) devices, such as Dynamic Message Signs (DMS), to advise travelers of poor travel conditions or to notify travelers of travel policy changes because of bad weather. For example, foggy conditions could trigger a DMS to display a lowered speed limit in a high-speed area. Snow and ice conditions could trigger a DMS to display a requirement for travelers to use chains on their tires. Icy conditions on bridges or roadways can also lead to the triggering of a spraying device that sprays anti-icing or de-icing chemicals on bridge or roadway surfaces to improve driving conditions. High water or high wind conditions could trigger a DMS to display a message either recommending that travelers choose a different route or that they reduce their speed to protect themselves against the potential hazard.

Although the normal use of ESS is by transportation system managers, the data from these ESS are sometimes used by emergency management personnel. For example, when flood conditions occur, regardless of their extent, emergency management personnel use data on the depth of water in areas covered by ESS to determine how and when to respond to flooding. Emergency management personnel re-route travelers from flooded areas, in some cases by deploying (in conjunction with transportation system personnel) signs indicating that sections of road are closed because of flooding.

A transportation system manager may also be interested in using an ESS to measure air quality. This data can be used to monitor concentrations of certain chemicals to ensure that they do not exceed toxic levels. For example, tunnel systems frequently use sensors to ensure that carbon monoxide levels stay within safe levels. The data can also provide a valuable resource to air quality management systems to determine the accuracy of predictions. Finally, some research has suggested that air quality hot spots could be monitored to encourage traffic to avoid these areas during problematic periods.

ESS are typically deployed along the roadside as part of a network of sensors that report their findings to a central management system. The ESS data received at the central system is processed to provide the transportation system manager with intelligence about road weather conditions that can trigger operator action. For example, high wind conditions might trigger a warning to travelers; if the high wind conditions are severe or in an area where they constitute a high risk, they might trigger the closing of a bridge or a section of roadway. Likewise, a network of ESS may also be used to provide the transit system operator information about conditions that affect the health or safety of transit riders. The processing logic could be rather simple (e.g., monitoring high winds) or very complex (e.g., predictions of weather conditions on or near the road). In the latter case, the ESS data would likely serve as one of many inputs; others might include data from the national weather service and other sources.

However, ESS can also be deployed on a vehicle. Usually these ESS are atmospheric sensors or pavement sensors, gathering information about snow and ice conditions, pavement conditions, and similar data designed to provide the transportation system manager with information about conditions along a particular section of roadway. The data from mobile ESS are used to complement those from stationary ESS also deployed along the transportation network.

2.3 REFERENCE PHYSICAL ARCHITECTURE [INFORMATIVE]

NTCIP 1204 v03 addresses the communications interface between a management station and a controller. The relationship between these and other logical components is depicted in Figure 1. However, the actual physical arrangement of these components may vary from deployment to deployment; Sections 2.3.1 through 2.3.3 provide sample physical architectures that are supported by this reference architecture.

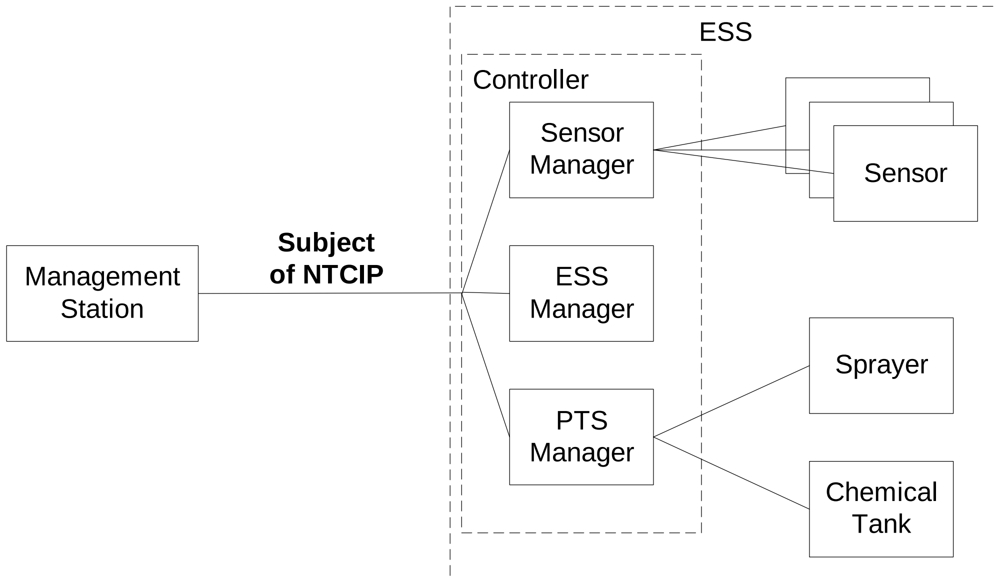

Figure 1 Reference Architecture

The major components of the system are:

a) **Management Station—**One or more host computing platforms that manage one or more NTCIP field devices, such as an ESS. Management stations are typically located in some type of management center (e.g., a Traffic Management Center) and may be a considerable distance from the ESS. Other types of management stations include maintenance laptops that a field technician may use on a trip to visit the device or a field processor that may monitor the data reported from the ESS and automatically activate signs or other equipment under certain conditions. There may be multiple management stations for a given ESS. Within the ESS community, a management station is sometimes called a central processing unit or "CPU". The management station is responsible for configuring, monitoring, and controlling the ESS.
b) **ESS—**A Controller and its connected equipment, such as environmental sensors and/or pavement treatment equipment, including sprayer(s) and a chemical tank. Each of its sub-components is defined further.
c) **Controller—**A host computing platform that is used to manage the collection and reporting of sensor data and/or to manage the treatment of pavement for icing conditions. It includes an ESS Manager and may include a Sensor Manager and/or a PTS Manager. Within the ESS community, a controller is sometimes called a remote processing unit or "RPU". The controller is responsible for continually monitoring conditions. When a controller receives a request from a management station, it shall immediately respond with its most recent reading for that data. A system operator should be aware that the nature of some information may require significant time to collect (e.g., average wind speed), or may be dated (e.g., information stored in a log); thus, the information contained in the response may have been collected some time prior to the request being sent.
d) **Sensor Manager—**The portion of the controller that manages the collection and reporting of sensor data.

e) **ESS Manager—**The portion of the controller that deals with general functionality that applies to both sensor management and PTS management.
f) **PTS Manager—**The portion of the controller that manages the treatment of pavement for icing conditions.
g) **Sensor**—A device that responds to a physical stimulus and transmits a resulting impulse.
h) **Sprayer—**A device that dispenses the chemical from the storage tank as a fine mist.
i) **Chemical Tank—**A vessel used to store a chemical mixture for later application to the roadway.

Descriptions of sample physical architectures supported by this reference architecture follow.

2.3.1 Remote Weather Station

Figure 2 depicts a wind sensor on a bridge that is connected by a low speed wireless radio link because of its remote location.

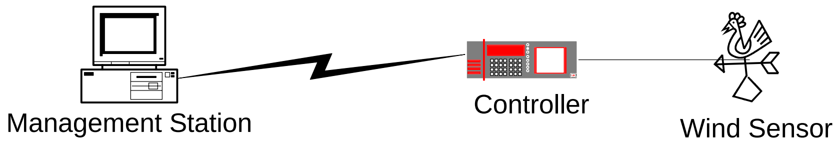

Figure 2 Remote Weather Station

2.3.2 Sprayer Combined with a Pavement Sensor

Figure 3 depicts an ESS consisting of a controller, a pavement sensor, and a sprayer. The controller is connected to the management station through a separate connection, perhaps a dial-up link.

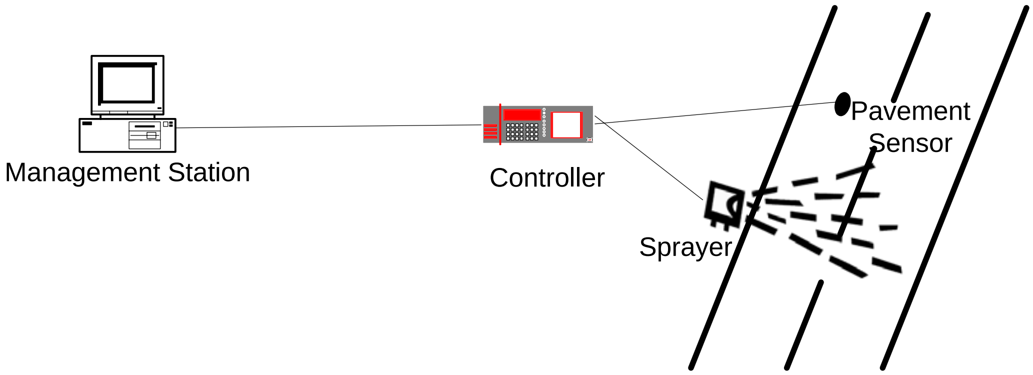

Figure 3 Sprayer Combined with a Pavement Sensor

2.3.3 Pavement Treatment Station

Figure 4 depicts an ESS that only consists of sprayers for pavement treatment.

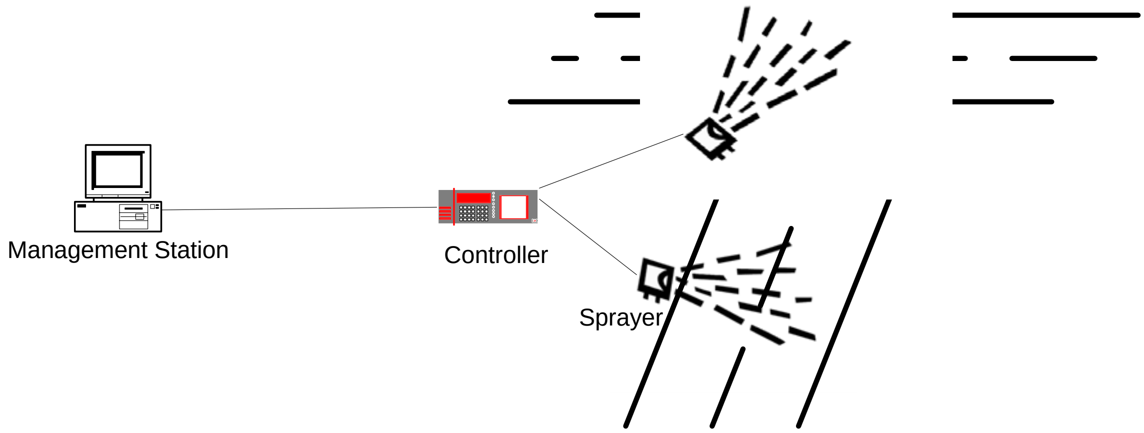

Figure 4 Pavement Treatment Station

2.4 ARCHITECTURAL NEEDS

An ESS is expected to operate in the communications environment defined as follows.

2.4.1 Generic Architectural Needs

The features defined in Annex F.1.1 shall be incorporated into this standard by reference.

NOTE—Some user needs apply to a wide range of different types of NTCIP devices. It is expected that These user needs are expected to eventually be documented in a separate standard. However, at this time, the separate standard does not exist. Instead, these user needs are defined in Annex F as an interim step to the creation of a separate standard.

2.5 FEATURES

Section 2.5 identifies and describes the various features that may be offered by the ESS. It is divided into the following:

a) ESS Manager Features
b) Sensor Manager Features
c) Pavement Treatment System Manager Features

2.5.1 ESS Manager Features

Section 2.5.1 identifies and describes the various features that may be offered by the ESS Manager, which is the part of the controller that performs the functionality that may apply to both a Sensor Manager and a PTS Manager. It consists of the following features:

a) Generic Features
b) Monitor Door Status
c) Monitor Power
d) Monitor Mobile Station Data

2.5.1.1 Generic Features

The features defined in Annex F.1.2 shall be incorporated into this standard by reference.

NOTE—Some user needs apply to a wide range of different types of NTCIP devices. These user needs are expected to eventually be documented in a separate standard. However, at this time, the separate standard does not exist. Instead these user needs are defined in Annex F as an interim step to the creation of a separate standard.

2.5.1.2 Monitor Door Status

A transportation system operator may wish to determine if any doors on the ESS equipment are open; this may assist the operator in determining whether maintenance crews have properly secured the controller after maintenance and/or may act as an indication to the system to treat any data as suspect.

2.5.1.3 Monitor Power

A transportation system operator may wish to monitor the power for the ESS to ensure proper operation.

2.5.1.4 Monitor Mobile Station Data

A transportation system operator may wish to monitor the movements of a mobile ESS and, if it is part of a mobile pavement treatment system, monitor the chemicals being dispersed.

2.5.2 Sensor Manager Features

Section 2.5.2 identifies and describes the various features that may be offered by the Sensor Manager:

a) monitor weather conditions
b) monitor pavement
c) monitor subsurface conditions
d) monitor human readings
e) monitor water levels
f) monitor air quality and biohazards
g) monitor mobile weather profile

2.5.2.1 Monitor Weather Conditions

This feature focuses on weather conditions that can directly or indirectly affect the transportation system. It includes the following sub-features.

2.5.2.1.1 Monitor Atmospheric Pressure

A transportation system operator may need to monitor the atmospheric pressure in the vicinity of the ESS.

2.5.2.1.2 Monitor Winds

A transportation system operator may need to monitor the current wind conditions in the vicinity of the ESS.

2.5.2.1.3 Monitor Temperature

A transportation system operator may need to monitor the temperature at the ESS location.

2.5.2.1.4 Monitor Humidity

A transportation system operator may need to monitor the humidity at the ESS location.

2.5.2.1.5 Monitor Precipitation

A transportation system operator may need to monitor the amount, intensity, and type of precipitation in the vicinity of the ESS.

2.5.2.1.6 Monitor Solar Radiation

A transportation system operator may need to monitor the amount of solar radiation in the vicinity of the ESS.

2.5.2.1.7 Monitor Visibility

A transportation system operator may need to monitor the visibility in the vicinity of the ESS.

2.5.2.1.8 View Weather Image

A transportation system operator may need to visually inspect weather conditions and/or verify the reported weather conditions.

2.5.2.2 Monitor Pavement

This feature focuses on monitoring road conditions that may adversely affect transportation operations immediately or in the near future. It supports the transportation system operator's ability to dispatch equipment to address the condition or to provide appropriate warnings. It includes the following sub-features.

2.5.2.2.1 Monitor Pavement Surface Condition

A transportation system operator may need to monitor the pavement surface temperature and moisture condition (e.g., dry, wet, snowy, icy, chemical concentration, etc.).

2.5.2.2.2 Monitor Icing Conditions

A transportation system operator may need to monitor whether pavement conditions are likely for ice formation on the pavement. This includes the ability to monitor pavement temperature (i.e., as opposed to surface temperature), the depth of any water film on the surface, and the predicted freeze point of the surface. Further, if passive sensor technologies are used, the operator needs to configure and monitor the parameters defining the current treatments being applied to validate the configuration.

2.5.2.2.3 View Pavement Image

A transportation system operator may need to visually inspect pavement conditions and/or verify the reported pavement conditions.

2.5.2.3 Monitor Subsurface Conditions

A transportation system operator may need to retrieve the conditions below the road surface, such that the operator may monitor conditions that could damage roads and/or affect the onset of icing conditions.

2.5.2.4 Monitor Human Readings

A transportation system operator may need to retrieve data that was manually observed and entered by field personnel.

2.5.2.5 Monitor Water Level

A transportation system operator may need to monitor the depth of water at one or more locations (e.g., over a roadway, in a stream, of a reservoir, etc.).

2.5.2.6 Monitor Air Quality and Biohazards

A transportation system operator may need to monitor the current air quality in the vicinity of the ESS and determine whether there are airborne biohazards in the vicinity of the ESS.

2.5.2.7 Monitor Mobile Weather Profile

A transportation system operator may need to monitor information that is specific to a mobile station such as speed, direction of travel, miles traveled, and detected state of the roadway, which includes friction.

2.5.3 Pavement Treatment System Manager Features

The following identify and describe the various features that may be offered by a Pavement Treatment System Manager. It consists of the following features:

a) manage stationary spray system
b) manage mobile spray system

2.5.3.1 Manage Stationary Spray System

A transportation system operator may need to manage the application of anti-icing or de-icing chemicals through the use of a sprayer connected to the ESS (e.g., a bridge sprayer). The management of this device includes the configuration, monitoring, and activation of this device.

2.5.3.2 Manage Mobile Spray System

A transportation system operator may need to manage the application of anti-icing or de-icing chemicals from a mobile pavement treatment system (e.g., a salt truck).

2.6 BACKWARD COMPATIBILITY NEEDS

2.6.1 Backward Compatible with NTCIP 1204 v01

A newer transportation system component may need to communicate with other components that conform to NTCIP 1204 v01.

2.6.2 Backward Compatible with NTCIP 1204 v02

A newer transportation system component may need to communicate with other components that conform to NTCIP 1204 v02.

2.7 SECURITY

NTCIP 1204 v03 does not address any security issues. Any security pertaining to protecting the communications with an ESS should be implemented either physically by protecting the communications access points, or logically by enabling security features associated with the underlying communications protocols.

2.8 RELATIONSHIP OF USER NEEDS TO NATIONAL ITS ARCHITECTURE FLOWS [INFORMATIVE]

There are seven National ITS Architecture Flows associated with the operation of an ESS. These are:

a) Environmental Sensors Control
b) Environmental Conditions Data
c) Environmental Probe Data
d) Emissions Sensor Control
e) Area Pollution Data
f) Roadway Treatment System Control
g) Roadway Treatment System Data

The main user need groups (features), as identified in Section 2, are related to the National ITS Architecture Flows as shown in Table 1.

Table 1 User Need Groups and National ITS Architecture Flows

User Need Group	Source	Architecture Flow	Destination
Manage ESS	MCMS	environmental sensors control	RS
	TMS	environmental sensors control	RS
	STWS	environmental sensors control	RS
	WS	environmental sensors control	RS
	EMMS	emissions sensor control	RS
Monitor weather conditions	RS	environmental conditions data	MCMS
	RS	environmental conditions data	TMS
	RS	environmental conditions data	STWS
	RS	environmental conditions data	WS
	RS	environmental probe data	TMS
Monitor pavement	RS	environmental conditions data	MCMS
	RS	environmental conditions data	TMS
	RS	environmental conditions data	STWS
	RS	environmental conditions data	WS
	RS	environmental probe data	TMS
Monitor subsurface conditions	RS	environmental conditions data	MCMS
	RS	environmental conditions data	TMS
	RS	environmental conditions data	STWS
	RS	environmental conditions data	WS
	RS	environmental probe data	TMS
Monitor human readings	RS	environmental conditions data	MCMS
	RS	environmental conditions data	TMS
	RS	environmental conditions data	STWS
	RS	environmental conditions data	WS
	RS	environmental probe data	TMS
Monitor flood levels	RS	environmental conditions data	MCMS
	RS	environmental conditions data	TMS
	RS	environmental conditions data	STWS
	RS	environmental conditions data	WS
	RS	environmental probe data	TMS
Monitor air quality	RS	area pollution data	EMMS
Control Pavement Treatment	MCMS	roadway treatment system control	RS
	RS	roadway treatment system status	MCMS

Section 3
FUNCTIONAL REQUIREMENTS
[NORMATIVE]

Section 3 defines the Functional Requirements based on the user needs identified in Section 2, and provides the reader with:

a) A tutorial
b) Architectural Requirements—These requirements relate to the architectural needs defined in Section 2.4.
c) Data Exchange Requirements—These requirements relate to the features identified in Section 2.5 that can be realized through a data exchange. For example, this includes the requirement to be able to retrieve weather information.
d) Supplemental Requirements—These are additional requirements derived from Section 2 that do not fall into categories b) or c). For example, they include requirements related to the number of pavement sensor inputs that an ESS may require, which may be a supplemental requirement to providing pavement sensor data.

Section 3 is intended for all readers, including:

a) transportation operations managers
b) transportation operations personnel
c) transportation engineers
d) system integrators
e) device manufacturers

For the first three categories of readers, Section 3 provides a useful understanding of the details of what NTCIP 1204 v03 requires of an ESS. Section 3.3 may be particularly useful to these readers in preparing procurement specifications and mapping various rows of this table to the more detailed text contained elsewhere.

For the last two categories of readers, Section 3 provides a useful understanding of what is required of equipment conforming to NTCIP 1204 v03. For these readers, Section 3.3 provides a table to document the capabilities of their implementations.

3.1 TUTORIAL [INFORMATIVE]

Section 3 defines the requirements that are intended to fulfill the user needs identified in Section 2. This is achieved through the development of a Protocol Requirements List (PRL) that traces each user need to one or more requirements. The details of each requirement are then presented following the PRL. The functional requirements are presented in three broad categories as follows:

a) Architectural Requirements—These requirements define the required behavior of the system in exchanging data across the communications interface, including any restrictions to general architectural requirements, based upon the architectural needs identified in Section 2.
b) Data Exchange Requirements—These requirements define the required behavior of the system in exchanging data across the communications interface based on the features identified in Section 2.
c) Supplemental Requirements—These requirements define additional requirements of the system that are derived from the architectural and/or data exchange requirements, but are not themselves architectural or data exchange requirements. A given supplemental requirement may relate to multiple architectural and/or data exchange requirements. Supplemental requirements frequently include range capabilities of the equipment.

3.2 SCOPE OF THE INTERFACE [INFORMATIVE]

Section 3.2 does not apply in the context of NTCIP 1204 v03.

3.3 PROTOCOL REQUIREMENTS LIST

The PRL, in Sections 3.3.7and 3.3.8, maps the user needs defined in Section 2 to the requirements defined in Section 3. The PRL can be used by:

a) a user or specification writer to indicate which requirements are to be implemented in a project-specific implementation
b) the protocol implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight
c) the supplier and user, as a detailed indication of the capabilities of the implementation
d) the user, as a basis for initially checking the potential interoperability with another implementation

3.3.1 User Needs Column

The user needs are defined in Section 2, and the PRL is based on the user needs within Section 2. The section identifier and section name are indicated in these columns.

3.3.2 Requirements Column

The requirements are defined in Section 3, and the PRL references the traces from user needs to these requirements. The section identifier and section name are indicated in these columns.

3.3.3 Conformance Column

The following notations and symbols are used to indicate status and conditional status in the PRL. Not all of these notations and symbols may be used in NTCIP 1204 v03.

3.3.3.1 Status Symbols

The symbols in Table 2 are used to indicate status.

Table 2 Status Symbols

M	Mandatory
M.#	Support of every item of the group labeled by the same numeral # required, but only one is active at time
O	Optional
O.# (range)	Part of an option group. Support of the number of items indicated by the ‘(range)’ is required from all options labeled with the same numeral #
C	Conditional
N/A	Not applicable (i.e., logically impossible in the NTCIP 1204 v03 scope)
X	Excluded or prohibited

The O.# (range) notation is used to show a set of selectable options (e.g., O.2 (1..*) would indicate that one or more of the option group 2 options is required to be implemented). Two character combinations are used for dynamic 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."

3.3.3.2 Conditional Status Notation

The predicate notations in Table 3 may be used.

Table 3 Predicate Notations

<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 states that the feature or features identified by the predicate are supported. In the simplest case, <predicate> is the identifying tag of a single PRL item. The <predicate>:: notation may precede a table or group of tables in a section. When the group predicate is true then the associated section 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.

The predicates used in NTCIP 1204 v03 map to sections as indicated in Table 4:

Table 4 Predicate to Section Mapping

Predicate	Section
Active	3.6.9
Air	2.5.2.6
CO	3.5.2.3.6.1
CO2	3.5.2.3.6.2
Compressed	F.1.1.2
ESS	F.1.1.1
Icing	2.5.2.2.2
Mobile	The device is able to operate while in motion.
N2O	3.5.2.3.6.3
NO2	3.5.2.3.6.4
O3	3.5.2.3.6.6
PM10	3.5.2.3.6.7
Passive	3.6.10
Pavement	2.5.2.2
Pressure	2.5.2.1.1
SO2	3.5.2.3.6.5
Subsurface	2.5.2.3
Temperature	2.5.2.1.3
Weather	2.5.2.1
Wind	2.5.2.1.2

3.3.4 Project Requirement Column

The Project Requirement (support) column can be used in a procurement specification to identify the required features for the given procurement or by an implementer to identify which features have been implemented. In either case, the user circles the appropriate answer (Yes, No, or N/A) in the Project Requirement column. See Table 5.

Table 5 Project Requirement Column Options

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

3.3.5 Additional Project Requirements Column

The Additional Project Requirements column may used by a procurement specification to provide additional notes and requirements for the device to be procured or may be used by an implementer to provide any additional details about the implementation. In some cases, default text already exists in this field, which the user should complete to fully specify the equipment. However, additional text can be added to this field as needed to fully specify a feature.

3.3.6 Instructions for Completing the PRL

In the Project Requirement column, each response shall be selected either from the indicated set of responses (for example: Yes / No / N/A), or it shall reference additional items that are to be attached.

If a conditional requirement is inapplicable, use the Not Applicable (N/A) choice. If a mandatory requirement is not satisfied, exception information shall be supplied by entering a reference Xi, where i is a unique identifier, to an accompanying rationale for the non-conformance. When the status is expressed as a two-character combination (as defined in Annex A.3.1), the response shall address each element of the requirement; e.g., for the requirement "mo," the possible compliant responses are "yy" or "yn."

To claim conformance with this standard, an implementation shall satisfy the mandatory and selected optional requirements as identified in the PRL.

NOTE—An agency specification can allow for flexibility in a deliverable by leaving the selection in the Project Requirement column blank for a given row. For example, an agency specification could allow for either passive or active icing detectors by selecting ‘Yes’ on line 2.5.2.2.2, and leaving lines 3.6.9 and 3.6.10 blank.

3.3.7 Protocol Requirements List (PRL) Table

† Designates that this requirement is composed of several more detailed requirements as defined in the second half of the PRL contained in Section 3.3.8.

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
2.4	Architectural Needs			M	Yes	
2.4.1	Generic Architectural Needs			M	Yes	(See F.1.1)
2.5	Features			M	Yes	
2.5.1	ESS Manager Features			M	Yes	
2.5.1.1	Generic Features			M	Yes	(See F.1.2)
2.5.1.2	Monitor Door Status			O	Yes / No	
		3.5.1.2.1	Retrieve ESS Door Status	M	Yes / NA	
2.5.1.3	Monitor Power			O	Yes / No	
		3.5.1.2.2	Retrieve Battery Status	O.6 (1..*)	Yes / No / NA	
		3.5.1.2.3	Retrieve Line Volts	O.6 (1..*)	Yes / No / NA	
2.5.1.4	Monitor Mobile Station Data			Mobile:M	Yes / NA	
		3.5.1.3.1	Retrieve Mobile ESS Movement	M	Yes / NA	
		3.5.1.3.3	Retrieve Compressed Mobile Station Data	M	Yes / NA	
2.5.2	Sensor Manager Features			O.1 (1..*)	Yes / No	
2.5.2.1 (Weather)	Monitor Weather Conditions			O.2 (1..*)	Yes / No / NA	
2.5.2.1.1 (Pressure)	Monitor Atmospheric Pressure			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.1	Retrieve Atmospheric Pressure	M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.6.1	Required Number of Atmospheric Pressure Sensors	M	Yes / NA	The ESS shall support at least ____ atmospheric pressure sensors.
2.5.2.1.2 (Wind)	Monitor Winds			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.2	Retrieve Wind Data	M	Yes / NA	
		3.6.2	Required Number of Wind Sensors	M	Yes / NA	The ESS shall support at least ____ wind sensors.
2.5.2.1.3 (Temperature)	Monitor Temperature			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.3	Retrieve Temperature	M	Yes / NA	
		3.5.2.3.2.4	Retrieve Daily Minimum and Maximum Temperature	M	Yes / NA	
		3.6.3	Required Number of Temperature Sensors	M	Yes / NA	The ESS shall support at least ____ temperature sensors (1..255).
2.5.2.1.4	Monitor Humidity			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.5	Retrieve Humidity	M	Yes / NA	
		3.6.4	Required Number of Humidity Sensors	M	Yes / NA	The ESS shall support at least ____ humidity sensors.
2.5.2.1.5	Monitor Precipitation			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.6.1	Retrieve Precipitation Presence	M	Yes / NA	
		3.5.2.3.2.6.2	Retrieve Precipitation Rates	O	Yes / No / NA	
		3.5.2.3.2.6.3	Retrieve Precipitation Totals	O	Yes / No / NA	
		3.5.2.3.5.2	Retrieve Precipitation Situation	M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.6.5	Required Number of Precipitation Sensors	M	Yes / NA	The ESS shall support at least ____ precipitation sensors.
2.5.2.1.6	Monitor Solar Radiation			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.7	Retrieve Solar Radiation	M	Yes / NA	
		3.6.6	Required Number of Solar Radiation Sensors	M	Yes / NA	The ESS shall support at least ____ solar radiation sensors.
2.5.2.1.7	Monitor Visibility			O.3 (1..*)	Yes / No / NA	
		3.5.2.3.2.8	Retrieve Visibility	M	Yes / NA	
		3.6.7	Required Number of Visibility Sensors	M	Yes / NA	The ESS shall support at least ____ visibility sensors.
2.5.2.1.8	View Weather Image			O	Yes / No / NA	
		3.5.2.1.9	Configure Snapshot Camera	M	Yes / NA	
		3.5.2.3.8	Retrieve Snapshot	M	Yes / NA	Upon ESS delivery the FTP username shall be ____________. Upon ESS delivery, the FTP password shall be __________.
		3.5.2.3.9	Retrieve Snapshot Camera Configuration	M	Yes / NA	
		3.5.2.4.1	Capture Snapshot Image	M	Yes / NA	
		3.5.2.4.2	Delete Snapshot	M	Yes / NA	
		3.5.2.4.3	Copy Snapshot	M	Yes / NA	
		3.6.20	Required Number of Snapshot Cameras	M	Yes / NA	The ESS shall support at least ____ snapshot cameras (1..255).
		3.6.23	Support Camera Number in Filename	O	Yes / No / NA	
		3.6.24	Support Sequence Number in Filename	O	Yes / No / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.6.25	Support Date in Filename	O	Yes / No / NA	
		3.6.26	Support Time in Filename	O	Yes / No / NA	
		3.6.27	Support Long Filenames	O	Yes / No / NA	
2.5.2.2 (Pavement)	Monitor Pavement			O.2 (1..*)	Yes / No / NA	
2.5.2.2.1	Monitor Pavement Surface Condition			M	Yes / NA	
		3.5.2.1.6	Configure Pavement Sensor	M	Yes / NA	
		3.5.2.3.3.1	Retrieve Pavement Surface Condition	M	Yes / NA	
		3.5.2.3.3.4	Retrieve Adjacent Snow Depth	O	Yes / No / NA	
		3.5.2.3.3.5	Retrieve Roadway Snow Depth	O	Yes / No / NA	
		3.5.2.3.3.6	Retrieve Roadway Ice Thickness	O	Yes / No / NA	
		3.6.8	Required Number of Pavement Sensors	M	Yes / NA	The ESS shall support at least ____ pavement sensors (1..255).
2.5.2.2.2 (Icing)	Monitor Icing Conditions			O	Yes / No / NA	
		3.5.2.1.8	Configure Passive Ice Detection Logic	Passive:M	Yes / NA	
		3.5.2.3.3.2	Retrieve Icing Conditions—Active	Active:M	Yes / NA	
		3.5.2.3.3.3	Retrieve Icing Conditions—Passive	Passive:M	Yes / NA	
		3.6.9 (Active)	Active Pavement Treatment Sensors	Icing:O.5 (1..2)	Yes / No / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.6.10 (Passive)	Passive Pavement Treatment Sensors	Icing:O.5 (1..2)	Yes / No / NA	
2.5.2.2.3	View Pavement Image			O	Yes / No / NA	
		3.5.2.1.9	Configure Snapshot Camera	M	Yes / NA	
		3.5.2.3.8	Retrieve Snapshot	M	Yes / NA	Upon ESS delivery the FTP username shall be ____________. Upon ESS delivery, the FTP password shall be __________.
		3.5.2.3.9	Retrieve Snapshot Camera Configuration	M	Yes / NA	
		3.5.2.4.1	Capture Snapshot Image	M	Yes / NA	
		3.5.2.4.2	Delete Snapshot	M	Yes / NA	
		3.5.2.4.3	Copy Snapshot	M	Yes / NA	
		3.6.20	Required Number of Snapshot Cameras	M	Yes / NA	The ESS shall support at least ____ snapshot cameras (1..255).
		3.6.23	Support Camera Number in Filename	O	Yes / No / NA	
		3.6.24	Support Sequence Number in Filename	O	Yes / No / NA	
		3.6.25	Support Date in Filename	O	Yes / No / NA	
		3.6.26	Support Time in Filename	O	Yes / No / NA	
		3.6.27	Support Long Filenames	O	Yes / No / NA	
2.5.2.3 (Subsurface)	Monitor Subsurface Conditions			O.2 (1..*)	Yes / No / NA	
		3.5.2.1.7	Configure Subsurface Sensor	Subsurface:M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.5.2.3.4.1	Retrieve Basic Subsurface Conditions	M	Yes / NA	
		3.5.2.3.4.2	Retrieve Subsurface Moisture	O	Yes / No / NA	
		3.6.11	Required Number of Subsurface Sensors	M	Yes / NA	The ESS shall support at least ____ subsurface sensors (1..255).
2.5.2.4	Monitor Human Readings			O.2 (1..*)	Yes / No / NA	
		3.5.2.3.5.1	Retrieve Wind Situation	M	Yes / NA	
		3.5.2.3.5.2	Retrieve Precipitation Situation	M	Yes / NA	
		3.5.2.3.5.3	Retrieve Cloud Situation	M	Yes / NA	
		3.5.2.3.5.4	Retrieve Visibility Situation	M	Yes / NA	
		3.5.2.3.5.5	Retrieve Ground State	O	Yes / No / NA	
		3.5.2.3.5.6	Retrieve Pavement State	O	Yes / No / NA	
2.5.2.5	Monitor Water Level			O.2 (1..*)	Yes / No / NA	
		3.5.2.3.7	Retrieve Water Level	M	Yes / NA	
		3.6.22	Required Number of Water Level Sensors	M	Yes / NA	The ESS shall support at least ____ water level sensors (1..255).
2.5.2.6 (Air)	Monitor Air Quality and Biohazards			O.2 (1..*)	Yes / No / NA	
		3.5.2.3.6.1 (CO)	Retrieve Carbon Monoxide Reading	O.4 (1..*)	Yes / No / NA	
		3.5.2.3.6.2 (CO2)	Retrieve Carbon Dioxide Reading	O.4 (1..*)	Yes / No / NA	
		3.5.2.3.6.3 (N2O)	Retrieve Nitrous Oxide Reading	O.4 (1..*)	Yes / No / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.5.2.3.6.4 (NO2)	Retrieve Nitrogen Dioxide Reading	O.4 (1..*)	Yes / No / NA	
		3.5.2.3.6.5 (SO2)	Retrieve Sulfur Dioxide Reading	O.4 (1..*)	Yes / No / NA	
		3.5.2.3.6.6 (O3)	Retrieve Ozone Reading	O.4 (1..*)	Yes / No / NA	
		3.5.2.3.6.7 (PM10)	Retrieve Small Particulate Matter Reading	O.4 (1..*)	Yes / No / NA	
		3.6.13	Required Number of Carbon Monoxide Sensors	CO:M	Yes / NA	The ESS shall support at least ____ carbon monoxide sensors.
		3.6.14	Required Number of Carbon Dioxide Sensors	CO2:M	Yes / NA	The ESS shall support at least ____ carbon dioxide sensors.
		3.6.15	Required Number of Nitrous Oxide Sensors	N2O:M	Yes / NA	The ESS shall support at least ____ nitrous oxide sensors.
		3.6.16	Required Number of Nitrogen Dioxide Sensors	NO2:M	Yes / NA	The ESS shall support at least ____ nitrogen dioxide sensors.
		3.6.17	Required Number of Sulfur Dioxide Sensors	SO2:M	Yes / NA	The ESS shall support at least ____ sulfur dioxide sensors.
		3.6.18	Required Number of Ozone Sensors	O3:M	Yes / NA	The ESS shall support at least ____ ozone sensors.
		3.6.19	Required Number of Small Particulate Matter Sensors	PM10:M	Yes / NA	The ESS shall support at least ____ small particulate matter sensors.
2.5.2.7	Monitor Mobile Weather Profile			O	Yes / No / NA	
		3.5.2.3.1	Retrieve Weather Profile with Mobile Sources	M	Yes / NA	
		F.2.1.2.1	Retrieve Current Configuration of Logging Service	M	Yes / NA	
		F.2.1.2.2	Configure Logging Service	M	Yes / NA	
		F.2.1.2.3	Retrieve Logged Data	M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		F.2.1.2.4	Clear Log	M	Yes / NA	
		F.2.1.2.5	Retrieve Capabilities of Event Logging Service	M	Yes / NA	
		F.2.1.2.6	Retrieve Total Number of Logged Events	M	Yes / NA	
2.5.3	Pavement Treatment System Manager Features			O.1 (1..*)	Yes / No / NA	
2.5.3.1	Manage Stationary Spray System			Mobile:X; M	Yes / No / NA	
		3.5.3.1.1	Retrieve Stationary Pavement Treatment Configuration	M	Yes / NA	
		3.5.3.1.2	Configure Stationary Pavement Treatment System	M	Yes / NA	
		3.5.3.2.1	Retrieve Pavement Treatment Status	M	Yes / NA	
		3.5.3.4.1	Set PTS Operational Mode	M	Yes / NA	
		3.5.3.4.2	Manually Activate PTS Sprayer	M	Yes / NA	
		3.6.12	Required Number of Pavement Treatment Products	M	Yes / NA	The ESS shall support at least ____ pavement treatment products (1..255).
2.5.3.2	Manage Mobile Spray System			Mobile: M	Yes / No / NA	
		3.5.3.1.3	Retrieve Mobile Pavement Treatment Configuration	M	Yes / NA	
		3.5.3.1.4	Configure Mobile Pavement Treatment System	O	Yes / No / NA	
2.6	Backwards Compatibility Needs			O	Yes / No	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
2.6.1	Backwards Compatibility with Version 1			O	Yes / No / NA	
		3.5.4.1	Version 1 Wind Sensor Meta Data	O	Yes / No / NA	
		3.5.4.2	Version 1 Average Wind Sensor Data	O	Yes / No / NA	
		3.5.4.3	Version 1 Spot Wind Sensor Data	O	Yes / No / NA	
		3.5.4.4	Version 1 Wind Gust Data	O	Yes / No / NA	
		3.5.4.5	Version 1 Wind Situation	O	Yes / No / NA	
		3.5.4.6	Version 1 Water Depth	O	Yes / No / NA	
		3.5.4.7	Version 1 Solar Radiation	O	Yes / No / NA	
		3.5.4.8	Version 1 Surface Water Depth	O	Yes / No / NA	
		3.5.4.9	Version 1 Surface Conductivity	O	Yes / No / NA	
2.6.2	Backwards Compatibility with Version 2			O	Yes / No	
		3.5.4.10	Version 2 Station Meta Data Block	O	Yes / NA	
		3.5.4.11	Version 2 Weather Block	O	Yes / No / NA	
		3.5.4.12	Version 2 Pavement Block	O	Yes / No / NA	
F.1.1	Generic Architectural Needs			M	Yes	
F.1.1.1 (ESS)	Provide Live Data			M	Yes	
		F.2.1.1.1	Retrieve Data	M	Yes	
		F.2.1.1.2	Deliver Data	M	Yes	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		F.2.1.1.3	Explore Data	M	Yes	
		3.6.21	Maximum Response Time for Requests	M	Yes	The Maximum Response Time for all requests shall be ___ milliseconds.
F.1.1.2 (Compressed)	Provide Compressed Data			Mobile:M; O	Yes / No	
		3.5.1.1.2	Retrieve Compressed Station Metadata	M	Yes	
		3.5.2.3.2.9	Retrieve Compressed Weather Data	Weather:M	Yes / NA	
		3.5.2.3.3.7	Retrieve Compressed Pavement Condition Data	Pavement:M	Yes / NA	
		3.5.2.3.4.3	Retrieve Compressed Subsurface Condition Data	Subsurface:M	Yes / NA	
		3.5.2.3.6.8	Retrieve Compressed Air Quality Data	Air:M	Yes / NA	
		3.6.21	Maximum Response Time for Requests	M	Yes	The Maximum Response Time for all requests shall be ___ milliseconds.
F.1.1.3	Provide Off-line Log Data			Mobile:M ; O	Yes / No	
		F.2.1.2.1	Retrieve Current Configuration of Logging Service	M	Yes / NA	
		F.2.1.2.2	Configure Logging Service	M	Yes / NA	
		F.2.1.2.3	Retrieve Logged Data	M	Yes / NA	
		F.2.1.2.4	Clear Log	M	Yes / NA	
		F.2.1.2.5	Retrieve Capabilities of Event Logging Service	M	Yes / NA	
		F.2.1.2.6	Retrieve Total Number of Logged Events	M	Yes / NA	
		F.2.2.1.5.1	Set Time	M	Yes / NA	
		F.2.2.1.5.2	Retrieve Current Time	M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		3.6.21	Maximum Response Time for Requests	M	Yes	The Maximum Response Time for all requests shall be ___ milliseconds.
		F.2.3.1†	Supplemental Requirements for Event Monitoring	M	Yes / NA	
F.1.2	Generic Features			M	Yes	
F.1.2.1	Retrieve the Device Identity			M	Yes	
		3.5.1.1.1	Retrieve ESS Characteristics	M	Yes	
		3.5.1.1.3	Configure ESS Manager	M	Yes	
		3.5.2.1.1	Retrieve Atmospheric Pressure Height	Pressure:M	Yes / NA	
		3.5.2.1.2	Retrieve Meta-Data for Each Wind Sensor	Wind:M	Yes / NA	
		3.5.2.1.3	Retrieve Temperature Sensor Meta-Data	Temperature:M	Yes / NA	
		3.5.2.1.4	Retrieve Pavement Sensor Meta-Data	Pavement:M	Yes / NA	
		3.5.2.1.5	Retrieve Subsurface Sensor Meta-Data	Subsurface:M	Yes / NA	
		F.2.2.1.1	Retrieve Device Component Information	M	Yes	
		F.2.2.1.2	Retrieve Device Configuration Identifier	M	Yes	
		F.2.2.1.3	Retrieve Supported Standards	M	Yes	
		F.2.2.1.4	Retrieve System Name	M	Yes	
F.1.2.2	Control External Devices			O	Yes / No	
		F.2.2.1.6	Retrieve External Port Information	M	Yes / NA	
		F.2.2.1.7	Configure Port Information	M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		F.2.2.2.1	Monitor Status of External Device	M	Yes / NA	
		F.2.2.4.1	Control External Device	M	Yes / NA	
		F.2.3.2	Required Number of Auxiliary Ports	M	Yes / NA	The ESS shall support at least ____ binary analog output ports (1..255).

3.3.8 Protocol Requirements List—Supplemental Table

Supplemental Requirement ID	Supplemental Requirement	Conformance	Project Requirement	Additional Project Requirements
F.2.3	Generic Supplemental Requirements			
F.2.3.1	Supplemental Requirements for Event Monitoring	M	Yes / NA	
F.2.3.1.1	Record and Timestamp Events	M	Yes / NA	
F.2.3.1.2	Support a Number of Event Classes	M	Yes / NA	The ESS shall support at least ____ event classes (1..255).
F.2.3.1.3	Support a Number of Event Types to Monitor	M	Yes / NA	The ESS shall support at least ____ event types (1..255).
F.2.3.1.4	Support Monitoring of Event Types	M	Yes / NA	
F.2.3.1.4.1	Support On-Change Events	M	Yes / NA	
F.2.3.1.4.2	Support Greater Than Events	M	Yes / NA	
F.2.3.1.4.3	Support Less Than Events	M	Yes / NA	
F.2.3.1.4.4	Support Hysteresis Events	M	Yes / NA	
F.2.3.1.4.5	Support Periodic Events	M	Yes / NA	
F.2.3.1.4.6	Support Bit-flag Events	M	Yes / NA	
F.2.3.1.5	Support Event Monitoring on Any Data	M	Yes / NA	
F.2.3.1.6	Support a Number of Events to Store in Log	M	Yes / NA	The ESS shall support storing at least ____ events in the log (1..65535).

3.4 ARCHITECTURAL REQUIREMENTS

There are no unique architectural requirements defined in NTCIP 1204 v03. The architectural needs are fully met through the generic architectural requirements defined in Annex F.

3.5 DATA EXCHANGE REQUIREMENTS

Data exchange requirements for ESS follow.

3.5.1 ESS Manager Requirements

Requirements for managing an ESS Manager follow.

3.5.1.1 ESS Configuration Requirements

Requirements for configuring an ESS Manager follow.

3.5.1.1.1 Retrieve ESS Characteristics

Upon request, the ESS shall return information related to the station type, category, and location.

3.5.1.1.2 Retrieve Compressed Station Metadata

Upon request, the ESS shall return the following information about the station:

a) station category
b) type of station
c) location of ESS
d) location of sensors
e) pavement treatment information

3.5.1.1.3 Configure ESS Manager

Upon request, the ESS shall store the textual description of the ESS location, as provided within the request.

3.5.1.2 ESS Status Monitoring Requirements

Requirements for monitoring the status of an ESS Manager follow.

3.5.1.2.1 Retrieve ESS Door Status

Upon request, the ESS shall return an indication as to whether any doors related to the ESS (e.g., cabinet doors, housing doors, etc.) are open.

3.5.1.2.2 Retrieve Battery Status

Upon request, the ESS shall return the charge status of the battery.

3.5.1.2.3 Retrieve Line Volts

Upon request, the ESS shall return the voltage on the incoming A/C power line.

3.5.1.3 ESS Data Retrieval Requirements

Requirements for retrieving data from an ESS Manager follow.

3.5.1.3.1 Retrieve Mobile ESS Movement

Upon request, the ESS shall return the speed, location, and direction of the mobile platform.

3.5.1.3.2 Retrieve Mobile Treatment Information [Deprecated in v03]

~~Upon request, the ESS shall return the pavement treatment that the mobile platform is dispersing.~~

NOTE—This requirement is effectively deleted from NTCIP 1204 v03. It was determined to be redundant with requirement 3.5.3.1.3 and is more appropriately defined there (i.e., according to the terms used in NTCIP 1204 v03, the ESS Manager does not include pavement treatment information; that information would be stored in the PTS Manager.)

3.5.1.3.3 Retrieve Compressed Mobile Station Data

Upon request, the ESS shall return the following information about the station in a compressed form:

a) location of ESS
b) speed of ESS
c) pavement treatment information

3.5.1.4 ESS Control Requirements

There are no control requirements for the ESS Manager.

3.5.2 Sensor Manager Requirements

Requirements for managing a Sensor Manager follow.

3.5.2.1 Sensor Configuration Requirements

Requirements for configuring a Sensor Manager follow.

NOTE—A Sensor Manager may also require a user to configure proprietary data during initial set-up.

3.5.2.1.1 Retrieve Atmospheric Pressure Height

Upon request, the ESS shall return the relative height of the atmospheric pressure sensor.

3.5.2.1.2 Retrieve Metadata for Each Wind Sensor

Upon request, the ESS shall return the location and relative height of each wind sensor connected to the ESS.

3.5.2.1.3 Retrieve Temperature Sensor Metadata

Upon request, the ESS shall return the number of temperature sensors and the relative height of each.

3.5.2.1.4 Retrieve Pavement Sensor Metadata

Upon request, the ESS shall return the number of pavement sensors and the following information for each sensor:

a) a textual description of the location that the sensor is monitoring
b) the type of pavement the sensor is monitoring
c) the relative height of the pavement with respect to the station height
d) an indication of the amount of sunlight to which the monitored pavement is subjected
e) an indication of the sensor technology used.

3.5.2.1.5 Retrieve Subsurface Sensor Metadata

Upon request, the ESS shall return the number of subsurface sensors and the following information for each sensor:

a) a textual description of the location that the sensor is monitoring
b) the type of subsurface the sensor is monitoring
c) the depth of the sensor location

3.5.2.1.6 Configure Pavement Sensor

Upon request, the ESS shall store configuration information for a specified pavement sensor.

3.5.2.1.7 Configure Subsurface Sensor

Upon request, the ESS shall store configuration information for a specified subsurface sensor.

3.5.2.1.8 Configure Passive Ice Detection Logic

Upon request, the ESS shall store information regarding the pavement treatments being applied so that the ESS may more accurately estimate icing conditions using passive logic.

3.5.2.1.9 Configure Snapshot Camera

Upon request, the ESS shall store a textual description of the location to which the camera points and the filename to be used when storing new snapshots.

NOTE—This requirement was modified in NTCIP 1204 v03 to more accurately reflect the design.

3.5.2.2 Sensor Status Monitoring Requirements

There are no status monitoring requirements for the Sensor Manager.

3.5.2.3 Sensor Data Retrieval Requirements

Requirements for retrieving data from a Sensor Manager follow.

3.5.2.3.1 Retrieve Weather Profile with Mobile Sources

Upon request, the ESS shall return a list of records recorded by the ESS over a period of time, with each record containing the following information about the mobile station:

a) location
b) speed
c) bearing
d) odometer reading
e) roadway friction
f) local weather observation
g) time of reading

3.5.2.3.2 Monitor Weather Condition

Requirements for monitoring weather conditions follow.

3.5.2.3.2.1 Retrieve Atmospheric Pressure

Upon request, the ESS shall return the current atmospheric pressure.

3.5.2.3.2.2 Retrieve Wind Data

Upon request, the ESS shall return the current wind speed and direction for each wind sensor connected to the ESS.

3.5.2.3.2.3 Retrieve Temperature

Upon request, the ESS shall return the current ambient air temperature.

3.5.2.3.2.4 Retrieve Daily Minimum and Maximum Temperature

Upon request, the ESS shall return the minimum and maximum ambient air temperatures that have been recorded within the previous 24 hours.

3.5.2.3.2.5 Retrieve Humidity

Upon request, the ESS shall return the current humidity, dew point, and wet bulb temperature.

3.5.2.3.2.6 Monitor Precipitation

Requirements for monitoring precipitation follow.

3.5.2.3.2.6.1 Retrieve Precipitation Presence

Upon request, the ESS shall return an indication of whether precipitation is currently detected and an indication of the make and model of the sensor so that the management station is able to be aware of the likely accuracy of the reading.

3.5.2.3.2.6.2 Retrieve Precipitation Rates

Upon request, the ESS shall return the rate at which precipitation is currently falling and the start and stop time of the latest recorded precipitation.

3.5.2.3.2.6.3 Retrieve Precipitation Totals

Upon request, the ESS shall return the total amounts of precipitation recorded over the last one hour, three hours, six hours, twelve hours, and 24 hours.

3.5.2.3.2.7 Retrieve Solar Radiation

Upon request, the ESS shall return the solar radiation data. The types of measured solar radiation data that the ESS shall provide are:

a) total minutes of sun over a 24 hour period
b) instantaneous infrared
c) instantaneous ultraviolet
d) visible
e) near-infrared
f) total radiation over a user set period

3.5.2.3.2.8 Retrieve Visibility

Upon request, the ESS shall return the current visibility distance.

3.5.2.3.2.9 Retrieve Compressed Weather Data

Upon request, the ESS shall return, in a compressed form, all current weather information, as defined in Sections 3.5.2.3.2.1 through 3.5.2.3.2.6.2, that is supported by the device.

3.5.2.3.3 Monitor Pavement Condition

Requirements for monitoring pavement conditions follow.

3.5.2.3.3.1 Retrieve Pavement Surface Condition

Upon request, the ESS shall return the current temperature of the pavement surface and shall indicate any presence of moisture on the surface along with an indication of whether any of this data might be in error. The ESS shall also indicate the make and model of the sensor so that the management station is able to properly interpret the accuracy of the data and precise meanings of code lists.

3.5.2.3.3.2 Retrieve Icing Conditions—Active

Upon request, the ESS shall return:

a) the current pavement temperature
b) the depth at which the pavement temperature is measured
c) the depth of any water/solution film covering the roadway

d) the predicted freeze point of the roadway surface.
e) the current surface temperature

3.5.2.3.3.3 Retrieve Icing Conditions—Passive

Upon request, the ESS shall return:

a) the current pavement temperature
b) the depth at which the pavement temperature is measured
c) the depth of any water/solution film covering the roadway
d) the predicted freeze point of the roadway surface
e) the current surface temperature
f) the conductivity of the roadway
g) the chemical(s) used for pavement treatment

3.5.2.3.3.4 Retrieve Adjacent Snow Depth

Upon request, the ESS shall return the current depth of snow adjacent to the traveled way (e.g., roadway, rail line, etc.).

3.5.2.3.3.5 Retrieve Roadway Snow Depth

Upon request, the ESS shall return the current depth of snow and packed snow on the traveled way (i.e., roadway, rail line, etc.).

3.5.2.3.3.6 Retrieve Roadway Ice Thickness

Upon request, the ESS shall return the current thickness of ice on the traveled way.

3.5.2.3.3.7 Retrieve Compressed Pavement Condition Data

Upon request, the ESS shall return, in compressed form, all current pavement condition information, as defined in Sections 3.5.2.3.3.1 through 3.5.2.3.3.6, that is supported by the device.

3.5.2.3.4 Monitor Subsurface Conditions

Requirements for monitoring subsurface conditions follow.

3.5.2.3.4.1 Retrieve Basic Subsurface Conditions

Upon request, the ESS shall return the current subsurface information. Subsurface information shall define the types of subsurface material, and environmental conditions by depth.

3.5.2.3.4.2 Retrieve Subsurface Moisture

Upon request, the ESS shall return the amount of moisture currently present in the subsurface of the roadway.

3.5.2.3.4.3 Retrieve Compressed Subsurface Condition Data

Upon request, the ESS shall return, in a compressed form, all current subsurface condition information, as defined in Section 3.5.2.3.4.1through 3.5.2.3.4.2, supported by the device.

3.5.2.3.5 Monitor Situation Assessments

Requirements for monitoring situation assessments follow.

3.5.2.3.5.1 Retrieve Wind Situation

Upon request, the ESS shall return the assessment of the wind situation (e.g., calm, light breeze, gale, gusty, etc.). The assessment may be made through automated processes, or if it is a staffed station, manually.

3.5.2.3.5.2 Retrieve Precipitation Situation

Upon request, the ESS shall return the assessment of the type and intensity of the current precipitation situation (e.g., no precipitation, moderate snow, heavy rain, etc.). The assessment may be made through automated processes, or if it is a staffed station, manually.

3.5.2.3.5.3 Retrieve Cloud Situation

Upon request, the ESS shall return the assessment of the cloud situation (e.g., clear, partly cloudy, cloudy, etc.). It is assumed that the assessment was manually entered by an authorized observer at the ESS site.

3.5.2.3.5.4 Retrieve Visibility Situation

Upon request, the ESS shall return the assessment of the visibility situation (e.g., clear, smoke, sun glare, etc.). The assessment may be made through automated processes, or if it is a staffed station, manually.

3.5.2.3.5.5 Retrieve Ground State

Upon request, the ESS shall return the assessment of the ground state next to the roadway (e.g., dry, wet, flooded, icy, drifting snow, etc.). The assessment may be made through automated processes, or if it is a staffed station, manually.

3.5.2.3.5.6 Retrieve Pavement State

Upon request, the ESS shall return the assessment of the pavement state (e.g., dry, wet, standing water, flowing water, packed snow, etc.). The assessment may be made through automated processes, or if it is a staffed station, manually.

3.5.2.3.6 Monitor Air Quality and Biohazard Conditions

Requirements for monitoring air quality and biohazard conditions follow.

3.5.2.3.6.1 Retrieve Carbon Monoxide Reading

Upon request, the ESS shall return the current carbon monoxide reading.

3.5.2.3.6.2 Retrieve Carbon Dioxide Reading

Upon request, the ESS shall return the current carbon dioxide reading.

3.5.2.3.6.3 Retrieve Nitrous Oxide Reading

Upon request, the ESS shall return the current nitrous oxide reading.

3.5.2.3.6.4 Retrieve Nitrogen Dioxide Reading

Upon request, the ESS shall return the current nitrogen dioxide reading.

3.5.2.3.6.5 Retrieve Sulfur Dioxide Reading

Upon request, the ESS shall return the current sulfur dioxide reading.

3.5.2.3.6.6 Retrieve Ozone Reading

Upon request, the ESS shall return the current ozone reading.

3.5.2.3.6.7 Retrieve Small Particulate Matter Reading

Upon request, the ESS shall return the current small particulate matter reading.

3.5.2.3.6.8 Retrieve Compressed Air Quality Data

Upon request, the ESS shall return all current air quality condition information supported by the device in a compressed form.

3.5.2.3.7 Retrieve Water Level

Upon request, the ESS shall return the current depth of water at defined locations (e.g., over a roadway, in a stream, of a reservoir, etc.).

3.5.2.3.8 Retrieve Snapshot

Upon request, the ESS shall return a copy of the specified snapshot image.

3.5.2.3.9 Retrieve Snapshot Camera Configuration

Upon request, the ESS shall return the location in which new snapshots are stored, the textual description of the location to which the camera points and the filename to be used when storing new snapshots.

NOTE—This requirement was revised in NTCIP 1204 v03 to more accurately reflect the design.

3.5.2.4 Sensor Control Requirements

Requirements for controlling a Sensor Manager follow.

3.5.2.4.1 Capture Snapshot Image

Upon request, the ESS shall capture, and store to a temporary location, the current image (snapshot) from the specified attached camera.

3.5.2.4.2 Delete Snapshot

Upon request, the ESS shall delete the specified snapshot image.

3.5.2.4.3 Copy Snapshot—[Deprecated]

~~Upon request, the ESS shall copy the specified snapshot image to a new file with the specified filename.~~

NOTE—This requirement was Deprecated from NTCIP 1204 v03. Requirement 3.5.2.4.3 was determined to be redundant with requirement 3.5.3.1.3 and is more appropriately defined with PTS requirements (i.e., according to the terms used in NTCIP 1204 v03, the ESS Manager does not include pavement treatment information; that information would be stored in the PTS Manager.)

NOTE—This requirement was added in NTCIP 1204 v02; however, it was later determined that FTP does not support a copy operation. This requirement had been replaced with the ability to define a sequence number in the filename. See Section 3.6.24.

3.5.3 PTS Manager Requirements

Requirements for managing a PTS Manager follow.

3.5.3.1 PTS Configuration Requirements

Requirements for configuring a PTS Manager follow.

3.5.3.1.1 Retrieve Stationary Pavement Treatment Configuration

Upon request, the PTS shall return:

a) the sensors that the PTS monitors to determine when to trigger the sprayers;
b) the duration required for a signal to activate the sprayer; and
c) the mix of chemicals to use when spraying.

3.5.3.1.2 Configure Stationary Pavement Treatment System

Upon request, the PTS shall change the configuration of the following parameters per the values contained in the request:

a) the sensors that the PTS monitors to determine when to trigger the sprayers;
b) the duration required for a signal to activate the sprayer; and
c) the mix of chemicals to use when spraying.

3.5.3.1.3 Retrieve Mobile Pavement Treatment Configuration

Upon request, the PTS shall return the configuration data identifying the mix of chemicals to be used when spraying.

3.5.3.1.4 Configure Mobile Pavement Treatment System

Upon request, the PTS shall change the configuration of the following parameters per the values contained in the request:

a) the spray amount and width; and
b) the mix of chemicals to use when spraying.

3.5.3.2 PTS Status Monitoring Requirements

Requirements for monitoring the status of a PTS Manager follow.

3.5.3.2.1 Retrieve Pavement Treatment Status

Upon request, the PTS shall return the current status of the sprayer and the number of spray events that have occured.

3.5.3.3 PTS Data Retrieval Requirements

There are no data retrieval requirements for a PTS Manager.

3.5.3.4 PTS Control Requirements

Requirements for controlling a PTS Manager follow.

3.5.3.4.1 Set PTS Operational Mode

Upon request, the PTS shall change its operational mode to that requested. Possible operational modes shall include:

a) Off, which shall prevent any operation of the sprayer
b) Manual, which shall allow manual activation of the sprayer
c) Automatic, which shall allow either manual activation or activation based on internal logic per the configuration parameters

3.5.3.4.2 Manually Activate PTS Sprayer

Upon request, the PTS shall trigger the sprayer to spray its pavement treatment solution. The trigger shall be activated for the configured duration period.

3.5.4 Backward Compatibility Requirements

The following define the requirements for backward compatibility.

3.5.4.1 NTCIP 1204 v01 Wind Sensor Meta Data

Upon request, the ESS shall return the relative height of the first wind sensor connected to the ESS in the NTCIP v01 format.

3.5.4.2 NTCIP 1204 v01 Average Wind Sensor Data

Upon request, the ESS shall return the average wind speed and direction for the first wind sensor in the NTCIP 1204 v01 format.

3.5.4.3 NTCIP 1204 v01 Spot Wind Sensor Data

Upon request, the ESS shall return the spot wind speed and direction for the first wind sensor in the NTCIP 1204 v01 format.

3.5.4.4 NTCIP 1204 v01 Wind Gust Data

Upon request, the ESS shall return the wind gust speed and direction for the first sensor in the NTCIP 1204 v01 format.

3.5.4.5 NTCIP 1204 v01 Wind Situation

Upon request, the ESS shall return the assessment of the wind situation (e.g., calm, light breeze, gale, gusty, etc.) in the NTCIP 1204 v01 format.

3.5.4.6 NTCIP 1204 v01 Water Depth

Upon request, the ESS shall return the water depth as measured by the first sensor in the NTCIP 1204 v01 format.

3.5.4.7 NTCIP 1204 v01 Solar Radiation

Upon request, the ESS shall return the solar radiation in the NTCIP 1204 v01 format.

3.5.4.8 NTCIP 1204 v01 Surface Water Depth

Upon request, the ESS shall return the surface water depth in the NTCIP 1204 v01 format.

3.5.4.9 NTCIP 1204 v01 Surface Conductivity

Upon request, the ESS shall return the surface conductivity in the NTCIP 1204 v01 format.

NOTE—It should be noted that the NTCIP 1204 v01 definition was ambiguous and resulted in interoperability problems.

3.5.4.10 NTCIP 1204 v02 Station Meta Data Block

Upon request, the ESS shall return the Station Meta Data Block in the NTCIP 1204 v02 format.

3.5.4.11 NTCIP 1204 v02 Weather Block

Upon request, the ESS shall return the Weather Block in the NTCIP 1204 v02 format.

3.5.4.12 NTCIP 1204 v02 Pavement Block

Upon request, the ESS shall return the Pavement Block in the NTCIP 1204 v02 format.

3.6 SUPPLEMENTAL REQUIREMENTS

Supplemental requirements for ESS follow. These requirements do not directly involve communications between the management station and the ESS, but, if the supplemental requirement is selected in the PRL, the ESS shall fulfill the stated requirement to claim conformance to NTCIP 1204 v03.

3.6.1 Required Number of Atmospheric Pressure Sensors

The communications interface only allows the ESS to return a single value for the atmospheric pressure; however, this value may be derived from multiple sensors. The ESS shall support the number of atmospheric pressure sensors as defined by the agency specification. If the agency specification does not define the number of atmospheric pressure sensors, the ESS shall support at least one atmospheric pressure sensor.

3.6.2 Required Number of Wind Sensors

The ESS shall support the number of wind sensors as defined by the agency specification. If the agency specification does not define the number of wind sensors, the ESS shall support at least one wind sensor.

3.6.3 Required Number of Temperature Sensors

The ESS shall support the number of temperature sensors as defined by the agency specification. If the agency specification does not define the number of temperature sensors, the ESS shall support at least one temperature sensor.

3.6.4 Required Number of Humidity Sensors

The communications interface only allows the ESS to return a single value for the humidity; however, this value may be derived from multiple sensors. The ESS shall support the number of humidity sensors as defined by the agency specification. If the agency specification does not define the number of humidity sensors, the ESS shall support at least one humidity sensor.

3.6.5 Required Number of Precipitation Sensors

The communications interface only allows the ESS to return a single set of values for precipitation data; however, this value may be derived from multiple sensors. The ESS shall support the number of precipitation sensors as defined by the agency specification. If the agency specification does not define the number of precipitation sensors, the ESS shall support at least one precipitation sensor.

3.6.6 Required Number of Solar Radiation Sensors

The communications interface only allows the ESS to return a single set of values for the solar radiation; however, these values may be derived from multiple sensors. The ESS shall support the number of solar radiation sensors as defined by the agency specification. If the agency specification does not define the number of solar radiation sensors, the ESS shall support at least one solar radiation sensor.

3.6.7 Required Number of Visibility Sensors

The communications interface only allows the ESS to return a single value for the visibility; however, this value may be derived from multiple sensors. The ESS shall support the number of visibility sensors as defined by the agency specification. If the agency specification does not define the number of visibility sensors, the ESS shall support at least one visibility sensor.

3.6.8 Required Number of Pavement Sensors

The ESS shall support the number of pavement sensors as defined by the agency specification. If the agency specification does not define the number of pavement sensors, the ESS shall support at least one pavement sensor.

3.6.9 Active Pavement Treatment Sensors

The ESS shall determine the predicted freeze-point of the pavement by actively freezing a portion of the mixture on the roadway surface.

3.6.10 Passive Pavement Treatment Sensors

The ESS shall determine the freeze point of the pavement through an algorithm that does not require the freezing of the chemical mixture on the roadway surface.

NOTE—Different makes and models of equipment may use different algorithms for a variety of reasons. To overcome problems that may result from this variation, NTCIP 1204 v03 links each pavement sensor with a row of the module table so that a system can identify the make and model of the specific pavement sensor.

3.6.11 Required Number of Subsurface Sensors

The ESS shall support the number of subsurface sensors as defined by the agency specification. If the agency specification does not define the number of subsurface sensors, the ESS shall support at least one subsurface sensor.

3.6.12 Required Number of Pavement Treatment Products

The ESS shall support the number of pavement treatment products as defined by the agency specification. If the agency specification does not define the number of pavement treatment products, the ESS shall support at least one pavement treatment product.

3.6.13 Required Number of Carbon Monoxide Sensors

The communications interface only allows the ESS to return a single value for carbon monoxide; however, this value may be derived from multiple sensors. The ESS shall support the number of carbon monoxide sensors as defined by the agency specification. If the agency specification does not define the number of carbon monoxide sensors, the ESS shall support at least one carbon monoxide sensor.

3.6.14 Required Number of Carbon Dioxide Sensors

The communications interface only allows the ESS to return a single value for carbon dioxide; however, this value may be derived from multiple sensors. The ESS shall support the number of carbon dioxide sensors as defined by the agency specification. If the agency specification does not define the number of carbon dioxide sensors, the ESS shall support at least one carbon dioxide sensor.

3.6.15 Required Number of Nitrous Oxide Sensors

The communications interface only allows the ESS to return a single value for nitrous oxide; however, this value may be derived from multiple sensors. The ESS shall support the number of nitrous oxide sensors as defined by the agency specification. If the agency specification does not define the number of nitrous oxide sensors, the ESS shall support at least one nitrous oxide sensor.

3.6.16 Required Number of Nitrogen Dioxide Sensors

The communications interface only allows the ESS to return a single value for nitrogen dioxide; however, this value may be derived from multiple sensors. The ESS shall support the number of nitrogen dioxide sensors as defined by the agency specification. If the agency specification does not define the number of nitrogen dioxide sensors, the ESS shall support at least one nitrogen dioxide sensor.

3.6.17 Required Number of Sulfur Dioxide Sensors

The communications interface only allows the ESS to return a single value for sulfur dioxide; however, this value may be derived from multiple sensors. The ESS shall support the number of sulfur dioxide sensors as defined by the agency specification. If the agency specification does not define the number of sulfur dioxide sensors, the ESS shall support at least one sulfur dioxide sensor.

3.6.18 Required Number of Ozone Sensors

The communications interface only allows the ESS to return a single value for ozone; however, this value may be derived from multiple sensors. The ESS shall support the number of ozone sensors as defined by the agency specification. If the agency specification does not define the number of ozone sensors, the ESS shall support at least one ozone sensor.

3.6.19 Required Number of Small Particulate Matter Sensors

The communications interface only allows the ESS to return a single value for small particulate matter; however, this value may be derived from multiple sensors. The ESS shall support the number of small particulate matter sensors as defined by the agency specification. If the agency specification does not define the number of small particulate matter sensors, the ESS shall support at least one small particulate matter sensor.

3.6.20 Required Number of Snapshot Cameras

The ESS shall support the number of snapshot cameras as defined by the agency specification. If the agency specification does not define the number of snapshot cameras, the ESS shall support at least one snapshot camera.

3.6.21 Maximum Response Time for Requests

The ESS shall process all requests in accordance with all of the rules of the relevant base standards (i.e., NTCIP 1103 v01 and NTCIP 2303:2001), including updating the value in the database and initiating the transmission of the appropriate response (assuming that the ESS has permission to transmit) within the Maximum Response Time. If the agency specification does not indicate the Maximum Response Time, the Maximum Response Time shall be 100 milliseconds. The Maximum Response Time is measured as the time between the receipt of the last byte of the request and the transmission of the first byte of the response.

3.6.22 Required Number of Water Level Sensors

The ESS shall support the number of water level sensors as defined by the agency specification. If the agency specification does not define the number of water level sensors, the ESS shall support at least one water level sensor.

3.6.23 Support Camera Number in Filename

The ESS shall support the ability to specify a field in the filename parameter that is replaced by the camera number when a snapshot is saved. This shall be the '<camera>' field as defined in Section 5.16.3.6.

3.6.24 Support Sequence Number in Filename

The ESS shall support the ability to specify a field in the filename parameter that is replaced by the current sequence number when a snapshot is saved. This shall be the '<sequence>' field as defined in Section 5.16.3.6.

3.6.25 Support Date in Filename

The ESS shall support the ability to specify a field in the filename parameter that is replaced by the current UTC date when a snapshot is saved. This shall be the '<date>' field as defined in Section 5.16.3.6.

3.6.26 Support Time in Filename

The ESS shall support the ability to specify a field in the filename parameter that is replaced by the current UTC time when a snapshot is saved. This shall be the '<time>' field as defined in Section 5.16.3.6.

3.6.27 Support Long Filenames

The ESS shall support the ability to specify filenames up to 255 characters in length, including any extension.

Section 4
DIALOGS
[NORMATIVE]

Section 4 defines the dialogs (i.e., sequence of data exchanges) that fulfill various Data Exchange requirements defined in Section 3.5. As SNMP communications are largely driven by the management station, most of the requirements define how the device shall respond to the various possible actions a management station might take.

The NTCIP standards effort is based on SNMP. This protocol offers a high degree of flexibility as to how the management station structures its requests. For example, with SNMP, the management station can do any of the following:

a) Send only those requests that are critical at the current time, whereas a standardized dialog typically sends requests relating to all associated data, regardless of whether it is critical for current purposes
b) Combine a number of requests in a single packet, whereas a standardized dialog dictates the exact contents of each packet
c) Separate a group of requests into multiple packets, whereas a standardized dialog dictates the exact contents of each packet
d) Interweave requests from multiple dialogs, whereas a standardized dialog dictates the exact ordering of messages, which are not interrupted with other messages

This flexibility can be a powerful tool allowing a management system to optimize the use of communication facilities, which is the primary reason that SNMP was chosen as the core NTCIP protocol. However, the flexibility also means that there are numerous allowable variations in the management process that a management station may choose to use and that an agent shall support to conform to NTCIP 1204 v03.

Unfortunately, this flexibility presents a challenge to ensuring interoperability. While a conformant ESS is required to support all operations defined within this standard, ensuring that a given ESS actually supports every possible combination would be impractical. Instead, most agencies only require that the device be tested to a standard set of procedures, which would use standardized dialogs (as defined in Section 4.2, Annex A, and Annex F.3). To improve communications efficiency, management stations may use non-standard dialogs (e.g., a combination of GET and/or SET requests that is not defined as a standardized dialog, but which a conformant device is required to support according to the ACCESS and SetConstraint rules defined in Section 4.3 and Section 5). Because these more efficient dialogs may not be known until the acquisition of the management station, which may be years after the acquisition of the device, there is a potential for an interoperability problem to arise.

To overcome this complication, Section 4 defines a lowest common denominator approach to communications between a management station and a device. It defines the standardized dialog for each Data Exchange Requirement. Management stations may support other dialogs to fulfill these same requirements, as long as these dialogs are consistent with the rules defined in NTCIP 1204 v03. Such a management station is termed a “consistent management station.” A consistent management station interoperates with any “conformant” device. However, since an agency cannot be certain that a device is 100% conformant to every possible scenario (given practical constraints), interoperability problems could still arise.

A “conformant management station” is required to offer a mode in which it only uses the standardized dialogs as defined in Section 4. With this limited definition, there is relatively little variability in what constitutes a conformant management station. Thus, fully testing a management station for conformance is a relatively straight forward process that can be done within the practical constraints faced by most procuring agencies. Thus, a conformant management station provides an agency with a much greater chance of achieving interoperability with off-the-shelf devices that have been tested against NTCIP 2104 v03, and the designation of such a system is intended to provide a guaranteed base level of interoperability.

The rules for the standardized dialogs follow:

a) The dialogs are defined by a sequence of GET or SET requests. These requests shall equate to the GET and SET operations defined in Annexes F.3.1and 0 and shall be transmitted as a single message.
b) The contents of each request are identified by an object name. Each object name consists of an object type and an instance identifier. Definitions of each object type are provided in Section 5 and NTCIP 1201 V02. The meaning of the instance identifier is provided by these same definitions coupled with standard SNMP rules (see RFC 1212).
c) Each message shall contain all of the objects as shown, unless otherwise indicated
d) A message shall not contain any other objects
e) The contents of each message sent by the management station may appear in any order
   NOTE—Ideally, the order of objects should match the order as shown in NTCIP 1204 v03 to provide the highest probability of interoperability. However, it is recognized that many implementations may use off-the-shelf software, which may prevent the designation of an exact ordering of objects and as a result, this ordering is not a requirement of NTCIP 1204 v03.
f) After sending a message, the management station shall not transmit any other data across the communications channel until the earlier of:
   1) The management station receiving a response from the device; or
   2) The expiration of the maximum response time.
g) If the response indicates an error occurred in the operation, the management station shall exit the process, unless specific error-handling rules are specified by the dialog.
h) Dialogs containing a sequence of only GET requests may request objects in any order.

However, since consistent management stations can alter the order of requests, this standard defines rules for when certain data exchanges are allowed. Unless otherwise indicated, a conformant device shall allow an object to be retrieved (through a GET request) or altered (through a SET request, if the object is writable) at any time. However, the access to some data is associated with a state machine, and Section 4.3 defines the various rules that apply to these state machines.

Finally, Section 4.4 presents an overview of all of the data defined by this standard, prior to presenting the complete definition for each piece of data in Section 5.

4.1 TUTORIAL [INFORMATIVE]

The Requirements Traceability Matrix (RTM) in Annex A identifies the standardized dialog that can be used to achieve each of the data exchange requirements defined in Section 3.5. Simple data exchange requirements reference one of the generic SNMP dialogs along with a list of data elements. These equate to a single message being sent (e.g., a GET request) containing the referenced data elements followed by the appropriate response per the generic dialog specification.

Section 4.1 defines the standardized dialogs for the more complicated data exchange requirements. Each of these dialogs is defined by a number of steps. Many of the steps reference data elements that are defined in Section 5. These data elements are also shown in the corresponding row of the RTM along with their precise section number.

The dialogs may also be accompanied by an informative figure that provides a graphical depiction of the normative text. The figures conform to the Unified Modeling Language and depict the management station as an outside actor sending a series of messages to the device and the device returning responses. If there is any conflict between the figure and the text, the text takes precedence.

4.2 SPECIFIED DIALOGS

4.2.1 Capture Snapshot Image

The standardized dialog for a management station to capture a snapshot image shall be:

a) The management station shall SET essSnapshotCameraCommand.x to the value of captureSnapsot (2).
b) The ESS shall take the picture with camera x.
c) The ESS shall store the captured picture to the directory essSnapshotCameraStoragePath.x and to the file essSnapshotCameraFilename.x.
d) The management station shall repeatedly GET essSnapshotCameraCommand.x until it equals ready (1).
e) The management station shall GET essSnapshotCameraError.x to verify the picture was successful.
f) The ESS shall respond with the indicated value.

This process is depicted in the UML diagram in Figure 5.

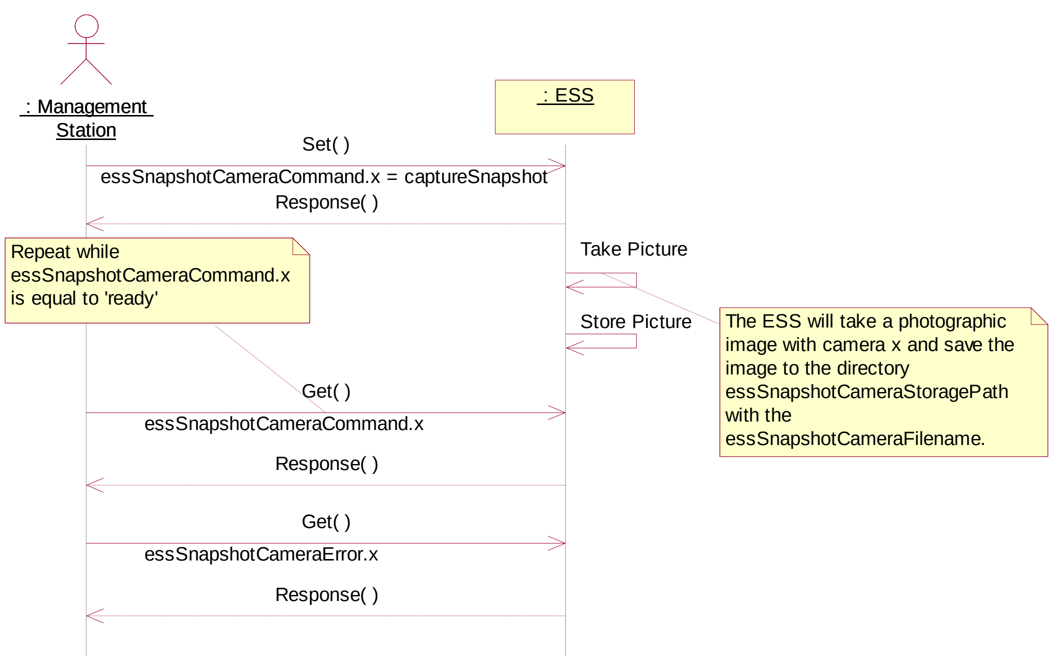

Figure 5 Dialog for Capture Snapshot Image

4.2.2 Retrieve Snapshot

The standardized dialog for a management station to retrieve a snapshot image shall conform to NTCIP 2303:2001.

The device shall adhere to the following rules:

a) The login directory is the root directory and a user cannot traverse to any parent directories.
b) Subdirectories may be used.
c) FTP access user name and password shall be defined in the agency specification.
d) Device shall use FTP port "21."
e) Zero or one FTP login session with the specification username shall exist at any given time.

4.2.3 Delete Snapshot

The standardized dialog for a management station to delete a snapshot image shall conform to NTCIP 2303:2001 and to the rules defined in Section 4.2.

4.2.4 Copy Snapshot—Deprecated in v03

~~The standardized dialog for a management station to copy a snapshot image shall conform to NTCIP 2303 (File Transfer Protocol) and to the rules defined in Clause 4.2.~~

NOTE—This dialog was added in NTCIP 1204 v02; however, it was later determined that FTP does not support a copy operation. It has been replaced with the ability to define a sequence number in the filename. See Section 3.6.24.

4.2.5 Retrieve Stationary Pavement Treatment Configuration

The standardized dialog for a management station to retrieve the pavement treatment configuration for a stationary ESS shall be as follows:

a) The management station shall GET numEssTreatments.0.
b) For each treatment from 1 to the number of treatments, the management station shall GET the following objects:
    1) essPaveTreatProductType.x
    2) essPaveTreatProductForm.x
    3) essPercentProductMix.x
c) The management station shall GET the following objects:
    1) ptsSignalDuration
    2) ptsMonitoringDetectors

Where,
    x = the index of the treatment

4.2.6 Retrieve Icing Conditions—Passive

The standardized dialog for a management station to retrieve the current and predicted icing conditions from a passive sensor shall be as follows:

a) (Precondition) The management station is aware of the sensor from which data is desired.
b) The management station shall GET the following objects:
    1) essSurfaceTemperature.x
    2) essPavementTemperature.x
    3) essSurfaceSalinity.x
    4) essSurfaceFreezePoint.x
    5) essSurfaceBlackIceSignal.x
    6) essPavementSensorError.x
c) The management station shall GET the following objects:
    1) essSurfaceIceOrWaterDepth.x
    2) essSurfaceConductivityV2.x
    3) pavementSensorTemperatureDepth.x
    NOTE—These are NTCIP 1204 v02 objects that may result in a noSuchName error.

d) The management station shall GET numEssTreatments.0.
e) For each treatment from 1 to the number of treatments, the management station shall GET the following objects:
    1) essPaveTreatProductType.y
    2) essPaveTreatProductForm.y
    3) essPercentProductMix.y

Where,
x = the sensor index,
y = the index of the treatment

4.2.7 Configure Stationary Pavement Treatment System

The standardized dialog for a management station to configure a stationary pavement treatment system shall be as follows:

a) The management station shall GET numEssTreatments.0.
b) For each treatment from 1 to the number of treatments, the management station shall SET the following objects to the desired values:
    1) essPaveTreatProductType.x
    2) essPaveTreatProductForm.x
c) The management station shall SET every instance of essPercentProductMix.x to the desired values such that the total of all instances shall equal 100.
d) The management station shall SET the following objects to their desired values:
    1) ptsSignalDuration.0
    2) ptsMonitoringDetectors.0

Where,
x = the index of the treatment

4.2.8 Configure Passive Ice Detection Logic

The standardized dialog for a management station to configure the passive ice detection logic shall be as follows:

a) The management station shall GET numEssTreatments.0.
b) For each treatment from 1 to the number of treatments, the management station shall SET the following objects to the desired values:
    1) essPaveTreatProductType.x
    2) essPaveTreatProductForm.x
c) The management station shall set every instance of essPercentProductMix.x to the desired value such that the total of all instances shall equal 100.

Where,
x = the index of the treatment

4.2.9 Configure Mobile Pavement Treatment System

The standardized dialog for a management station to configure a mobile pavement treatment system shall be as follows:

a) The management station shall GET numEssTreatments.0.
b) For each treatment from 1 to the number of treatments, the management station shall SET the following objects to the desired values:
    1) essPaveTreatProductType.x
    2) essPaveTreatProductForm.x

c) The management station shall SET every instance of essPercentProductMix.x to the desired values such that the total of all instances shall equal 100.
d) The management station shall SET the following objects to their desired values:
   1) essPaveTreatmentAmount.0
   2) essPaveTreatmentWidth.0

Where,
x = the index of the treatment

4.2.10 Configure Snapshot Camera

The standardized dialog for a management station to configure a snapshot camera shall be as follows:

a) (Precondition) The management station shall be aware of which row in the table is to be configured.
b) For the specified row, the management station shall SET the following object to the desired value:
   1) essSnapshotCameraDescription.x
a) For the specified row, the management station shall SET the following object to the desired value:
   1) essSnapshotCameraFilename.x

Where,
x = the index of the snapshot camera

NOTE—The standardized dialog is divided into two steps for backward compatibility; older devices may not support the essSnapshotCameraFilename object. These devices store the files using a filename convention that is manufacturer-specific.

4.2.11 Retrieve Snapshot Camera Configuration

The standardized dialog for a management station to retrieve the snapshot camera configuration shall be as follows:

a) The management station shall GET essSnapshotNumberOfCameras.0.
b) For each snapshot camera from 1 to the number of cameras, the management station shall GET the following objects:
   1) essSnapshotCameraDescription.x
   2) essSnapshotCameraStoragePath.x
c) For each snapshot camera from 1 to the number of cameras, the management station shall GET the following object:
   1) essSnapshotCameraFilename.x

Where,
x = the index of the treatment

NOTE—The standardized dialog is divided into two steps for backwards compatibility; older devices may not support the essSnapshotCameraFilename object. These devices store the files using a filename convention that is manufacturer-specific.

4.2.12 Retrieve Precipitation Presence

The standardized dialog for a management station to retrieve precipitation presence shall be as follows:

a) The management station shall GET essPrecipYesNo.0.
b) The management station shall GET precipitationSensorModelInformation.0.

   NOTE—This is an NTCIP 1204 v02 object that may result in a noSuchName. This level of detail was not provided in NTCIP 1204 v01.

NOTE—To understand the sensor model information, the management station either needs to be aware of the contents of the Module Table prior to performing this dialog or to retrieve the information after performing the dialog.

4.2.13 Retrieve Solar Radiation

The standardized dialog for a management station to retrieve solar radiation shall be as follows:

a) The management station shall GET essTotalSun.0.
b) The management station shall GET the following objects:
   1) essInstantaneousTerrestrialRadiation.0.
   2) essInstantaneousSolarRadiation.0.
   3) essTotalRadiation.0.
   4) essTotalRadiationPeriod.0.
   NOTE—These are NTCIP 1204 v02 objects that may result in a noSuchName error.
c) If the device responds with a noSuchName error, the management station shall GET essSolarRadiation.0

4.2.14 Retrieve Pavement Surface Condition

The standardized dialog for a management station to retrieve precipitation presence shall be as follows:

a) (Precondition) The management station shall be aware of which row of the table is to be retrieved.
b) The management station shall GET the following objects for the pavement sensor of interest:
   1) essSurfaceStatus.x
   2) essSurfaceTemperature.x
   3) essPavementSensorError.x
c) The management station shall GET pavementSensorModelInformation.x.
   NOTE—This is an NTCIP 1204 v02 object that may result in a noSuchName. This level of detail was not provided in NTCIP 1204 v01.

NOTE—To understand the sensor model information, the management station either needs to be aware of the contents of the Module Table prior to performing this dialog or to retrieve the information after performing the dialog.

4.2.15 Retrieve Icing Conditions—Active

The standardized dialog for a management station to retrieve active icing condition information shall be as follows:

a) (Precondition) The management station shall be aware of which row of the table is to be retrieved.
b) The management station shall GET the following objects for the pavement sensor of interest:
   1) essSurfaceTemperature.x
   2) essPavementTemperature.x
   3) essSurfaceFreezePoint.x
   4) essSurfaceBlackIceSignal.x
   5) essPavementSensorError.x
c) The management station shall GET the following objects for the pavement sensor of interest:
   1) essSurfaceIceOrWaterDepth.x
   2) pavementSensorTemperatureDepth.x

NOTE—These are NTCIP 1204 v02 objects that may result in a noSuchName error. This level of detail was not provided in NTCIP 1204 v01.

4.3 STATE TRANSITION DIAGRAMS

The following define the states for various object classes that may be supported by the device.

4.3.1 Pavement Treatment System State Transition Diagram

Figure 6 depicts the state transition diagram for the Pavement Treatment System class.

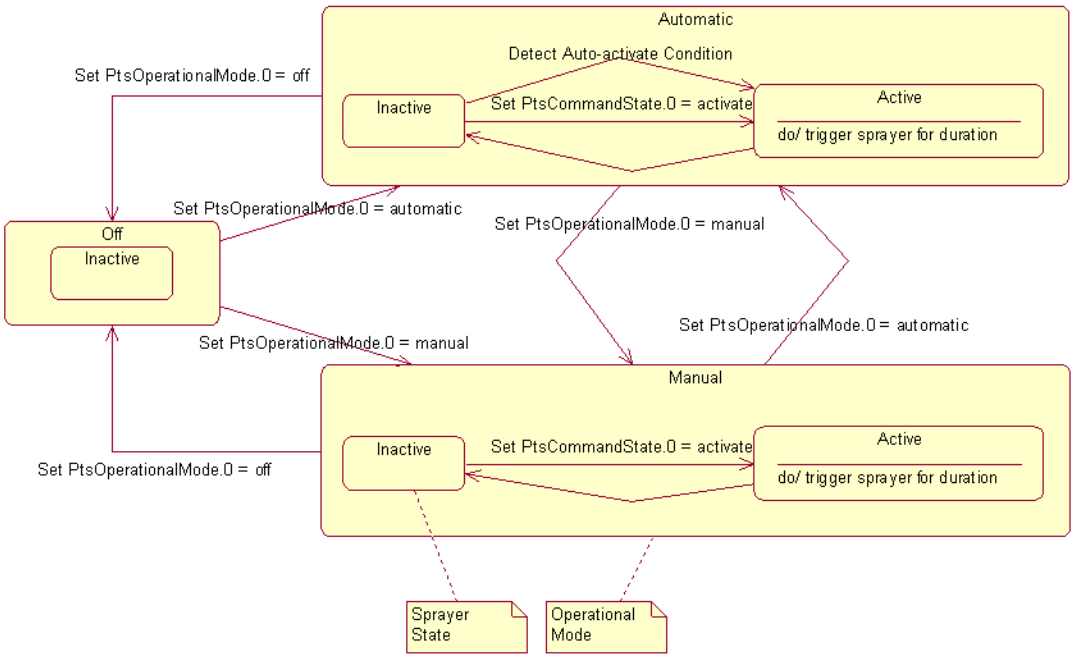

Figure 6 Pavement Treatment System State Machine Diagram

4.3.1.1 Off

When in the "off" state, the PTS shall not trigger the sprayer even if commanded to do so and shall always be inactive. The PTS shall transition to the requested operational mode, upon request.

4.3.1.2 Automatic

When in the "automatic" state, the PTS shall monitor conditions and trigger the sprayer based on a manufacturer specific algorithm. The algorithm shall only consider input from the detectors selected in the ptsMonitoringDetectors object. The PTS shall also trigger the sprayer if commanded to do so via the ptsCommandStateV3 object. The PTS shall transition to the requested operational mode, upon request.

4.3.1.3 Manual

When in the "manual" state, the PTS shall trigger the sprayer if commanded to do so via the ptsCommandStateV3 object. The PTS shall transition to the requested operational mode, upon request.

4.3.1.4 Inactive

When in the "inactive" state, the PTS shall not be spraying.

4.3.1.5 Active

Upon entering the "active" state, the PTS shall trigger the sprayer and spray the chemical for a duration as defined by the ptsSignalDuration object. Upon expiration of this duration, the PTS shall automatically transition back to the "inactive" state.

4.4 CLASS DIAGRAMS

The relationships between data elements are described through the use of UML class diagrams. Figure 7 provides a sample class diagram.

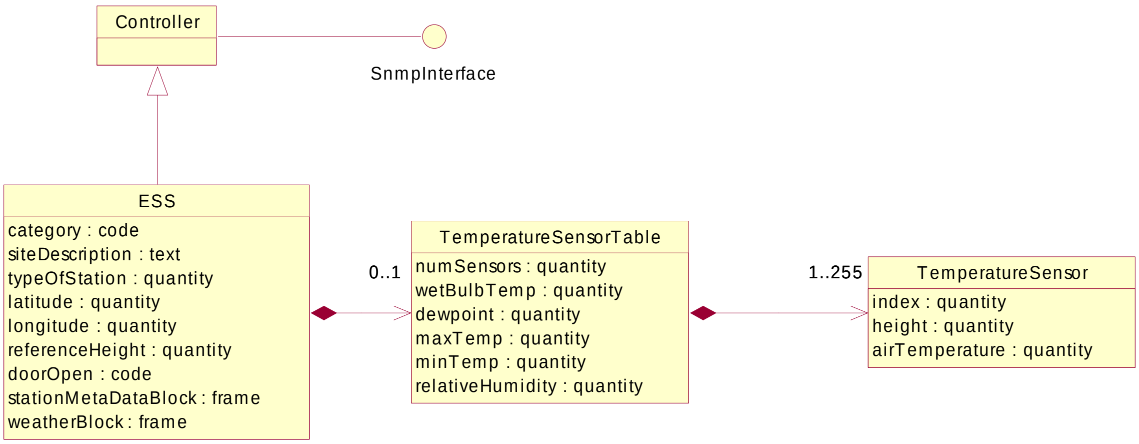

Figure 7 Sample Class Diagram for Temperature Sensors

Each box represents a logical class (grouping) of data. The box contains a name in the upper compartment and a list of any applicable attributes (i.e., individual pieces of information to describe the class) in the lower compartment. Lines between classes indicate that the classes have a relationship.

A diamond on the end of a line indicates aggregation. The class that has the diamond is the whole; the other class represents a part. If the diamond is filled in, it indicates composition, meaning that the part may only be owned by one whole and when the whole is deleted, all of its parts are automatically deleted. However, a part of an aggregate relationship that is not a composition can exist without the whole or may be part to several wholes. At a university, for example, a course would be considered to be an aggregation of students. A student may be enrolled in several courses at once and the student is not "deleted" if the course is cancelled.

A number at the end of a relationship line indicates the number of instances of the class that may exist in relation to one instance of the other class. An asterisk (*) indicates an infinite number. A range of values may be indicated in the format of a number followed by two periods followed by another number.

An open arrow indicates that the class from which the arrow originates is a type of the class to which the arrow points (i.e., an ESS is a type of controller).

A circle connected by a line indicates an interface for the class. An interface is one or more operations that may be performed. Within the context of NTCIP 1204 v03, there are two interfaces, the SNMP Interface (as shown in the sample diagram), and the FTP Interface (not shown here, but shown later in Section 4).

After the diagram, there is text describing the important rules depicted in the diagram

Each piece of data referenced is depicted in a class diagram and named according to ISO 14817 naming conventions. However, these naming conventions violate the rules for SNMP object names, as defined by RFC 1212. Thus, each class diagram is associated with a table that maps the descriptive names to the SNMP object names and the section number of the MIB where the data is defined.

4.4.1 ESS Characteristics

4.4.1.1 ESS Characteristics Class Diagram

An ESS can be described by a number of attributes as defined in the following sections and as depicted in the UML class diagram provided in Figure 8. The diagram indicates that an ESS is a type of a Controller. The data that may be supported by a Controller is defined by NTCIP 1201 V02. The Controller, and thus the ESS, shall support an SNMP Interface as defined in Annex F. A MobilePlatform is a special type of ESS that is able to collect information while in motion. While MobilePlatforms are relatively new to the industry, this standard provides a basic level of support for monitoring such devices.

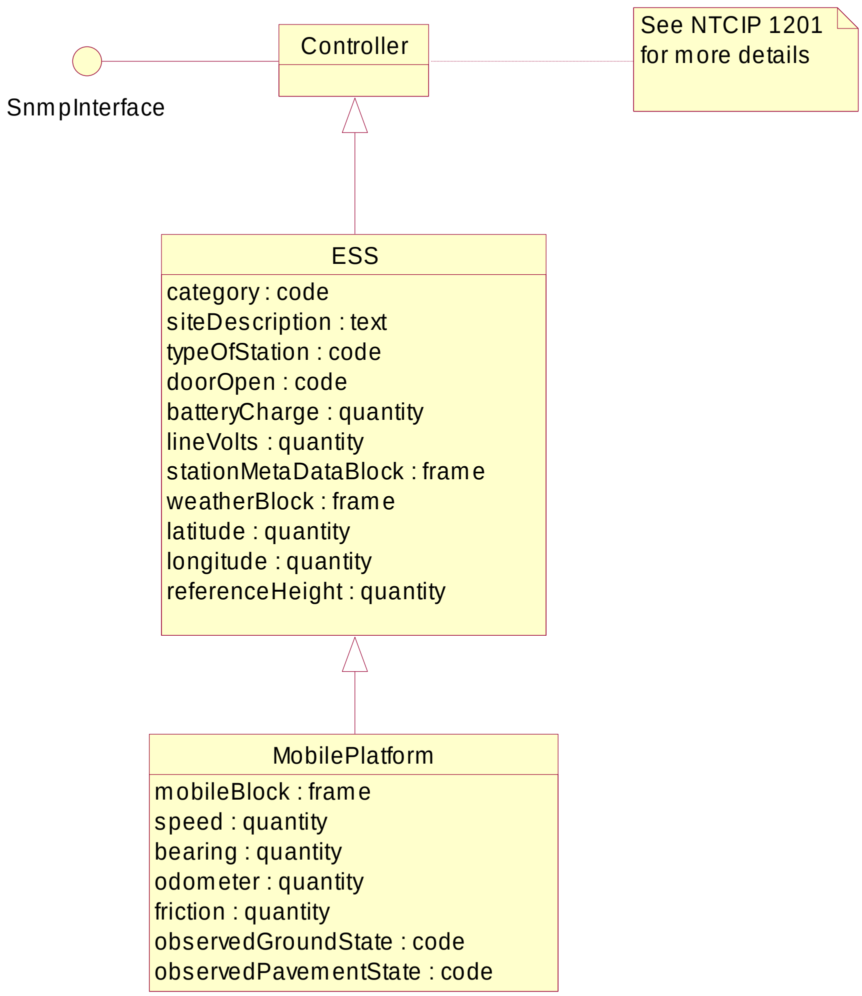

Figure 8 ESS Characteristics Class Diagram

4.4.1.2 ESS

An ESS is any type of device that is able to detect one or more environmental conditions and/or provide for pavement treatment. It can be described through a number of attributes as indicated Table 6.

Table 6 ESS Descriptive Name, Section, and Object Name

Descriptive Name	Section	Object Name
ESS.category:code	5.2.1	essNtcipCategory
ESS.siteDescription:text	5.2.2	essNtcipSiteDescription
ESS.typeOfStation:code	5.3.1	essTypeofStation
ESS.doorOpen:code	5.3.2	essDoorStatus
ESS.batteryCharge:quantity	5.3.3	essBatteryStatus
ESS.lineVolts:quantity	5.3.4	essLineVolts
ESS.stationMetaDataBlock:frame	5.3.5	essStationMetaDataBlock
ESS.weatherBlock:frame	5.3.6	essWeatherBlock
ESS.latitude:quantity	5.4.1	essLatitude
ESS.longitude:quantity	5.4.2	essLongitude
ESS.referenceHeight:quantity	5.5.1	essReferenceHeight

4.4.1.3 Mobile Platform

A mobile platform is a type of ESS that is able to operate while in motion. It can be described through a number of attributes as indicated Table 7below.

Table 7 Mobile Platform Attributes

Descriptive Name	Section	Object Name
MobilePlatform.mobileBlock:frame	5.3.7	essMobileBlock
MobilePlatform.speed:quantity	5.4.3	essVehicleSpeed
MobilePlatform.bearing:quantity	5.4.4	essVehicleBearing
MobilePlatform.odometer:quantity	5.4.5	essOdometer
MobilePlatform.friction:quantity	5.12.1	essMobileFriction
MobilePlatform.observedGroundState:code	5.12.2	essMobileObservationGroundState
MobilePlatform.observedPavementState:code	5.12.3	essMobileObservationPavement

4.4.2 Pressure Sensor

4.4.2.1 Pressure Sensor Class Diagram

The ESS shall support one logical atmospheric pressure sensor if required by the agency specification. This information is depicted in Figure 9.

NOTE—The logical sensor may represent a value derived from multiple physical sensors.

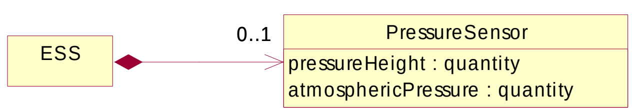

Figure 9 Pressure Sensor Class Diagram

4.4.2.2 Pressure Sensor

A pressure sensor is a sensor that detects the atmospheric pressure. It can be described through a number of attributes as indicated in Table 8.

Table 8 Pressure Sensor Attributes

Descriptive Name	Section	Object Name
PressureSensor.pressureHeight:quantity	5.5.2	essPressureHeight
PressureSensor.atmosphericPressure:quantity	5.5.4	essAtmosphericPressure

4.4.3 Wind Data

4.4.3.1 Wind Data Class Diagram

The ESS shall support one wind sensor table if required by the agency specification. The wind sensor table shall be associated with the number of wind sensors as defined in the agency specification.The information supported by the wind sensor is depicted in Figure 10.

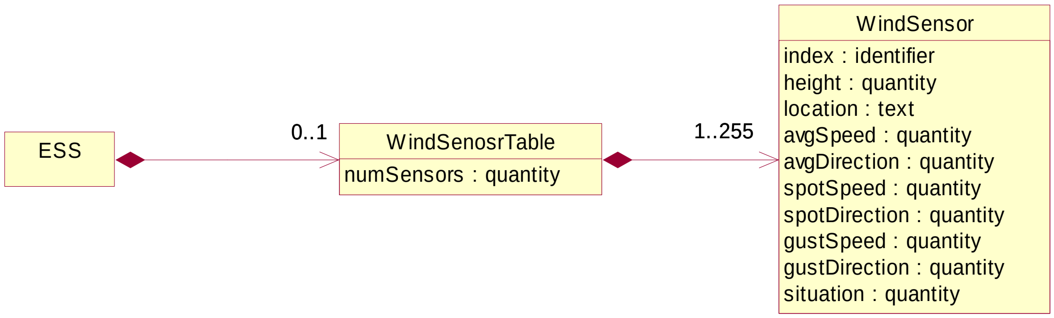

Figure 10 Wind Sensor Class Diagram

4.4.3.2 Wind Sensor

A wind sensor is a sensor that reports the wind speed and direction. It can be described through a number of attributes as indicated in Table 9.

Table 9 Wind Sensor Attributes

Descriptive Name	Section	Object Name
WindSensor.index:identifier	5.6.10.1	windSensorIndex
WindSensor.height:quantity	5.6.10.2	windSensorHeight
WindSensor.location:text	5.6.10.3	windSensorLocation
WindSensor.avgSpeed:quantity	5.6.10.4	windSensorAvgSpeed
WindSensor.avgDirection:quantity	5.6.10.5	windSensorAvgDirection
WindSensor.spotSpeed:quantity	5.6.10.6	windSensorSpotSpeed
WindSensor.spotDirection:quantity	5.6.10.7	windSensorSpotDirection
WindSensor.gustSpeed:quantity	5.6.10.8	windSensorGustSpeed
WindSensor.gustDirection:quantity	5.6.10.9	windSensorGustDirection
WindSensor.situation:code	5.6.10.10	windSensorSituation

4.4.3.3 Wind Sensor Table

The wind sensor table contains information about all of the wind sensors supported by the ESS. It can be described through a number of attributes as indicated in Table 10.

Table 10 Wind Sensor Table

Descriptive Name	Section	Object Name
WindSensorTable.numSensors:quantity	5.6.8	windSensorTableNumSensors

4.4.4 Temperature

4.4.4.1 Temperature Class Diagram

The ESS shall support one temperature sensor table if required by the agency specification. The temperature sensor table shall be associated with the number of temperature sensors as defined in the agency specification. The information supported by the temperature sensor is depicted in Figure 11.

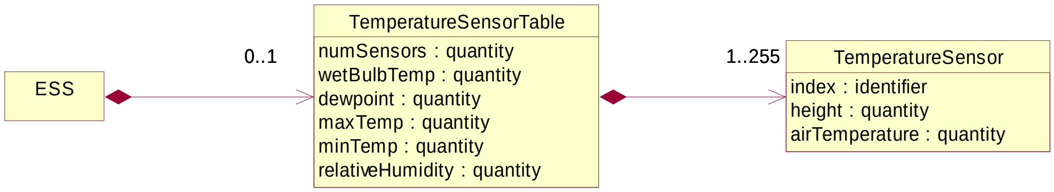

Figure 11 Temperature Sensor Class Diagram

4.4.4.2 Temperature Sensor Table

The temperature sensor table contains summary information about all of the temperature sensors supported by the ESS. It can be described through a number of attributes as indicated in Table 11.

Table 11 Temperature Sensor Table Attributes

Descriptive Name	Section	Object Name
TemperatureSensorTable.numSensors:quantity	5.7.1	essNumTemperatureSensors
TemperatureSensorTable.wetBulbTemp:quantity	5.7.4	essWetbulbTemp
TemperatureSensorTable.dewpoint:quantity	5.7.5	essDewpointTemp
TemperatureSensorTable.maxTemp:quantity	5.7.6	essMaxTemp
TemperatureSensorTable.minTemp:quantity	5.7.7	essMinTemp
TemperatureSensorTable.relativeHumidity:quantity	5.8.1	essRelativeHumidity

4.4.4.3 Temperature Sensor

A temperature sensor is a sensor that reports the current air temperature at a defined height. It can be described through a number of attributes as indicated in Table 12.

Table 12 Temperature Sensor Attributes

Descriptive Name	Section	Object Name
TemperatureSensor.index:identifier	5.7.3.1	essTemperatureSensorIndex
TemperatureSensor.height:quantity	5.7.3.2	essTemperatureSensorHeight
TemperatureSensor.airTemperature:quantity	5.7.3.3	essAirTemperature

4.4.5 Precipitation

4.4.5.1 Precipitation Class Diagram

The ESS shall support one logical precipitation sensor if required by the agency specification. The ESS shall support one water level sensor table if required by the agency specification. The water level sensor table shall be associated with the number of water level sensors as defined in the agency specification. The information supported by these sensors are depicted in Figure 12.

NOTE—The logical sensor may represent a value derived from multiple physical sensors.

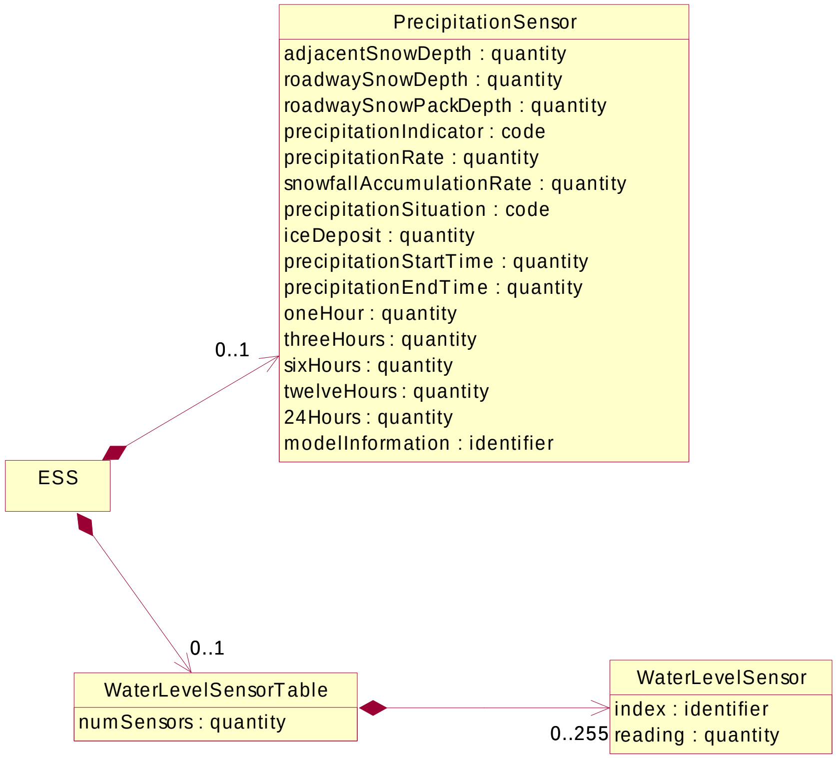

Figure 12 Precipitation Sensor Class Diagram

4.4.5.2 Precipitation Sensor

A precipitation sensor is a sensor that reports information about precipitation. It can be described through a number of attributes as indicated in Table 13.

Table 13 Precipitation Sensor Attributes

Descriptive Name	Section	Object Name
PrecipitationSensor.adjacentSnowDepth:quantity	5.8.3	essAdjacentSnowDepth
PrecipitationSensor.roadwaySnowDepth:quantity	5.8.4	essRoadwaySnowDepth
PrecipitationSensor.roadwaySnowPackDepth:quantity	5.8.5	essRoadwaySnowPackDepth
PrecipitationSensor.precipitationIndicator:code	5.8.6	essPrecipYesNo
PrecipitationSensor.precipitationRate:quantity	5.8.7	essPrecipRate
PrecipitationSensor.snowfallAccumulationRate:quantity	5.8.8	essSnowfallAccumRate
PrecipitationSensor.precipitationSituation:code	5.8.9	essPrecipSituation
PrecipitationSensor.iceDeposit:quantity	5.8.10	essIceThickness
PrecipitationSensor.precipitationStartTime:quantity	5.8.11	essPrecipitationStartTime
PrecipitationSensor.precipitationEndTime:quantity	5.8.12	essPrecipitationEndTime
PrecipitationSensor.oneHour:quantity	5.8.13	essPrecipitationOneHour
PrecipitationSensor.threeHours:quantity	5.8.14	essPrecipitationThreeHours
PrecipitationSensor.sixHours:quantity	5.8.15	essPrecipitationSixHours
PrecipitationSensor.twelveHours:quantity	5.8.16	essPrecipitationTwelveHours
PrecipitationSensor.24Hours:quantity	5.8.17	essPrecipitation24Hours
PrecipitationSensor.modelInformation:identifier	5.8.18	precipitationSensorModelInformation

4.4.5.3 Water Level Sensor Table

The water level sensor table contains information about all of the water level sensors supported by the ESS. It can be described through a number of attributes as indicated in Table 14.

Table 14 Water Level Sensor Table Attributes

Descriptive Name	Section	Object Name
WaterLevelSensorTable.numSensors:quantity	5.8.19	waterLevelSensorTableNumSensors

4.4.5.4 Water Level Sensor

A water level sensor is a sensor that reports the current level of water as measured from a defined point. It can be described through a number of attributes as indicated in Table 15.

Table 15 Water Level Sensor Attributes

Descriptive Name	Section	Object Name
WaterLevelSensor.index:identifier	5.8.21.1	waterLevelSensorIndex
WaterLevelSensor.reading:quantity	5.8.21.2	waterLevelSensorReading

4.4.6 Radiation

4.4.6.1 Radiation Class Diagram

The ESS shall support one logical radiation sensor if required by the agency specification. The information supported by the radiation sensor is depicted in Figure 13.

NOTE—The logical sensor may represent a value derived from multiple physical sensors.

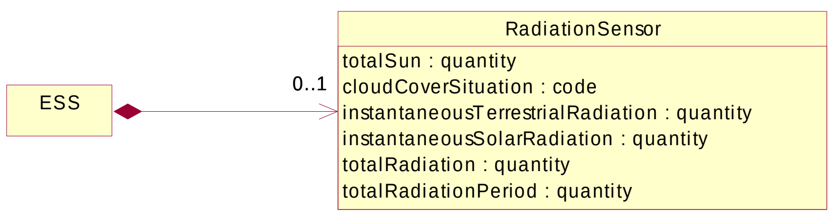

Figure 13 Radiation Sensor Class Diagram

4.4.6.2 Radiation Objects

A radiation sensor is a sensor that reports the amount of solar and terrestrial radiation to which the sensor is exposed. It can be described through a number of attributes as indicated Table 16.

Table 16 Radiation Sensor Attributes

Descriptive Name	Section	Object Name
RadiationSensor.totalSun:quantity	5.9.2	essTotalSun
RadiationSensor.cloudCoverSituation:code	5.9.3	essCloudSituation
RadiationSensor.instantaneousTerrestrialRadiation:quantity	5.9.4	essInstantaneousTerrestrialRadiation
RadiationSensor.instantaneousSolarRadiation:quantity	5.9.5	essInstantaneousSolarRadiation
RadiationSensor.totalRadiation:quantity	5.9.6	essTotalRadiation
RadiationSensor.totalRadiationPeriod:quantity	5.9.7	essTotalRadiationPeriod

4.4.7 Visibility

4.4.7.1 Visibility Class Diagram

The ESS shall support one logical visibility sensor if required by the agency specification. The information supported by the visibility sensor is depicted in Figure 14.

NOTE—The logical sensor may represent a value derived from multiple physical sensors.

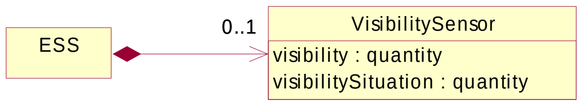

Figure 14 Visibility Sensor Class Diagram

4.4.7.2 Visibility Sensor

A visibility sensor is a sensor that reports the distance at which things are visible. It can be described through a number of attributes as indicated in Table 17.

Table 17 Visibility Sensor Attributes

Descriptive Name	Section	Object Name
VisibilitySensor.visibility:quantity	5.10.1	essVisibility
VisibilitySensor.visibilitySituation:code	5.10.2	essVisibilitySituation

4.4.8 Pavement Sensor Data

4.4.8.1 Pavement Sensor Data Class Diagram

The ESS shall support one pavement sensor table if required by the agency specification. The pavement sensor table shall be associated with the number of pavement sensors as defined in the agency specification. The information supported by these sensors is depicted in Figure 15.

If the ESS uses a passive pavement sensor to predict the temperature at which ice forms, the ESS shall also support the pavement treatment table. The pavement treatment table shall be associated with the number of pavement treatments as defined in the agency specification. See Section 4.4.12.

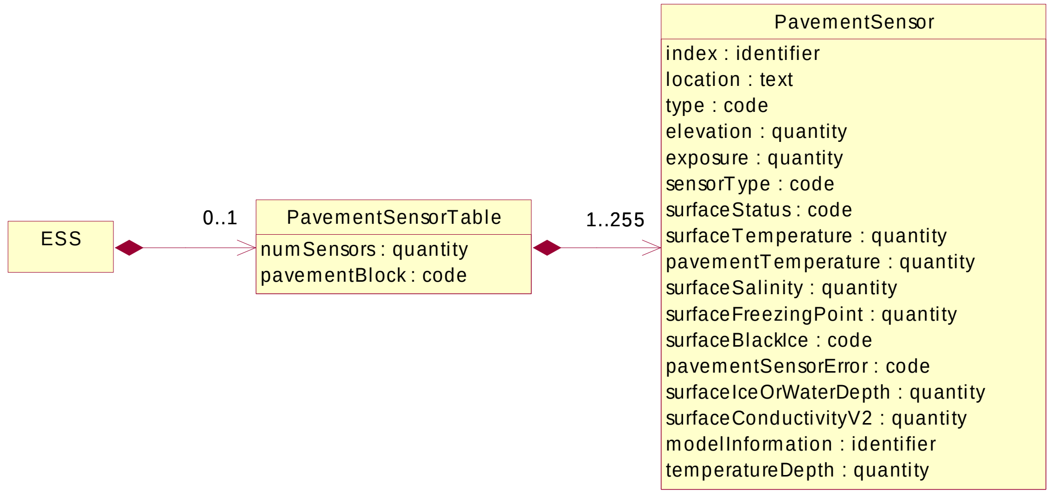

Figure 15 Pavement Sensor Class Diagram

4.4.8.2 Pavement Sensor Data

A pavement sensor provides information related to the state of the pavement. It can be described through a number of attributes as indicated in Table 18.

Table 18 Pavement Sensor Attributes

Descriptive Name	Section	Object Name
PavementSensor.index:identifier	5.11.3.1	essPavementSensorIndex
PavementSensor.location:text	5.11.3.2	essPavementSensorLocation
PavementSensor.type:code	5.11.3.3	essPavementType
PavementSensor.elevation:quantity	5.11.3.4	essPavementElevation
PavementSensor.exposure:quantity	5.11.3.5	essPavementExposure
PavementSensor.sensorType:code	5.11.3.6	essPavementSensorType
PavementSensor.surfaceStatus:code	5.11.3.7	essSurfaceStatus
PavementSensor.surfaceTemperature:quantity	5.11.3.8	essSurfaceTemperature
PavementSensor.pavementTemperature:quantity	5.11.3.9	essPavementTemperature
PavementSensor.surfaceSalinity:quantity	5.11.3.11	essSurfaceSalinity
PavementSensor.surfaceFreezingPoint:quantity	5.11.3.13	essSurfaceFreezePoint
PavementSensor.surfaceBlackIce:code	5.11.3.14	essSurfaceBlackIceSignal
PavementSensor.pavementSensorError:code	5.11.3.15	essPavementSensorError
PavementSensor.surfaceOrWaterDepth:quantity	5.11.3.16	essSurfaceIceOrWaterDepth
PavementSensor.surfaceConductivityV2:quantity	5.11.3.17	essSurfaceConductivityV2
PavementSensor.modelInformation:identifier	5.11.3.18	pavementSensorModelInformation
PavementSensor.temperatureDepth:quantity	5.11.3.19	pavementSensorTemperature Depth

4.4.8.3 Pavement Sensor Table

The pavement sensor table provides information related to the various pavement senosrs supported by the ESS. It can be described through a number of attributes as indicated in Table 19.

Table 19 Pavement Sensor Table Attributes

Descriptive Name	Section	Object Name
PavementSensorTable.numSensors:quantity	5.11.1	numEssPavementSensors
PavementSensorTable.pavementBlock:code	5.11.7	essPavementBlock

4.4.9 Subsurface Data

4.4.9.1 Subsurface Data Class Diagram

The ESS shall support one subsurface sensor table if required by the agency specification. The subsurface sensor table shall be associated with the number of subsurface sensors as defined in the agency specification. The information supported by these sensors are depicted in Figure 16.

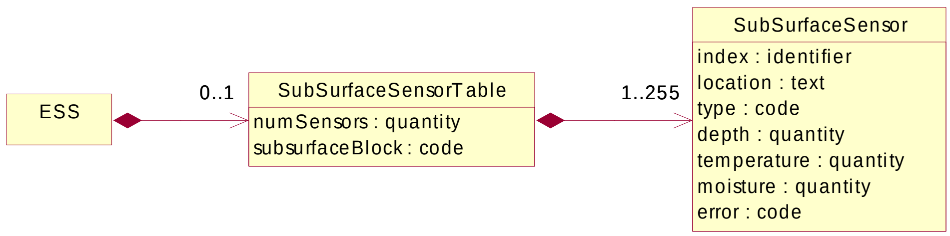

Figure 16 Subsurface Sensor Class Diagram

4.4.9.2 Subsurface Sensor Table

The subsurface sensor table provides summary information related to the subsurface sensors supported by the ESS. It can be described through a number of attributes as indicated in Table 20.

Table 20 Subsurface Sensor Table Attributes

Descriptive Name	Section	Object Name
SubSurfaceSensorTable.numSensors:quantity	5.11.4	numEssSubSurfaceSensors
SubSurfaceSensorTable.subsurfaceBlock:code	5.11.8	essSubsurfaceData

4.4.9.3 Subsurface Sensor

A subsurface sensor provides information related to the state of the pavement subsurface. It can be described through a number of attributes as indicated in Table 21

Table 21 Subsurface Sensor Attributes

Descriptive Name	Section	Object Name
SubSurfaceSensor.index:identifier	5.11.6.1	essSubSurfaceSensorIndex
SubSurfaceSensor.location:text	5.11.6.2	essSubSurfaceSensorLocation
SubSurfaceSensor.type:code	5.11.6.3	essSubSurfaceType
SubSurfaceSensor.depth:quantity	5.11.6.4	essSubSurfaceSensorDepth
SubSurfaceSensor.temperature:quantity	5.11.6.5	essSubSurfaceTemperature
SubSurfaceSensor.moisture:quantity	5.11.6.6	essSubSurfaceMoisture
SubSurfaceSensor.error:code	5.11.6.7	essSubSurfaceSensorError

4.4.10 Air Quality Data

4.4.10.1 Air Quality Data Class Diagram

The ESS shall support one logical air quality sensor if required by the agency specification. The information supported by this sensor is depicted in Figure 17.

NOTE—The logical sensor may represent a value derived from multiple physical sensors.

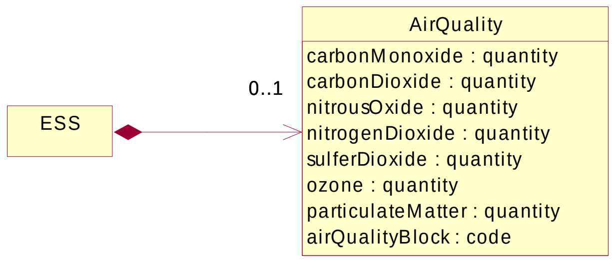

Figure 17 Air Quality Sensor Class Diagram

4.4.10.2 Air Quality Sensor

An air quality sensor is a sensor that reports the concentration of various chemicals in the air. It can be described through a number of attributes as indicated in Table 22.

Table 22 Air Quality Sensor Attributes

Descriptive Name	Section	Object Name
AirQuality.carbonMonoxide:quantity	5.14.1	essCO
AirQuality.carbonDioxide:quantity	5.14.2	essCO2
AirQuality.nitrousOxide:quantity	5.14.3	essNO
AirQuality.nitrogenDioxide:quantity	5.14.4	essNO2
AirQuality.sulferDioxide:quantity	5.14.5	essSO2
AirQuality.ozone:quantity	5.14.6	essO3
AirQuality.particulateMatter:quantity	5.14.7	essPM10
AirQuality.airQualityBlock:code	5.14.8	essAirQualityData

4.4.11 Snapshot Data

4.4.11.1 Snapshot Data Class Diagram

The ESS shall support one snapshot camera table if required by the agency specification. The snapshot camera table shall be associated with the number of snapshot cameras as defined in the agency specification. The information supported by these entities are depicted in Figure 18.

The ESS shall also support a dynamic number of snapshots managed through the FTP Interface.

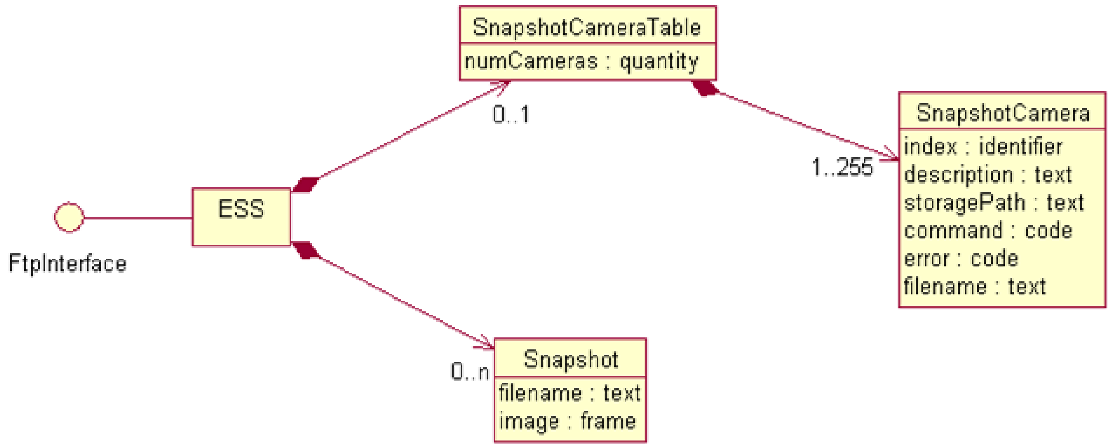

Figure 18 Snapshot Class Diagram

4.4.11.2 Snapshot Camera

A snapshot camera allows an ESS to capture and store a snapshot image. It can be described through a number of attributes as indicated in Table 23.

Table 23 Snapshot Camera Attributes

Descriptive Name	Section	Object Name
SnapshotCamera.index:identifier	5.16.3.1	essSnapshotCameraIndex
SnapshotCamera.description:text	5.16.3.2	essSnapshotCameraDescription
SnapshotCamera.storagePath:text	5.16.3.3	essSnapshotCameraStoragePath
SnapshotCamera.command:code	5.16.3.4	essSnapshotCameraCommand
SnapshotCamera.error:code	5.16.3.5	essSnapshotCameraError
SnapshotCamera.filename:text	5.16.3.6	essSnapshotCameraFilename

4.4.11.3 Snapshot

A snapshot is any image that has been captured by the snapshot camera. It can be described through a number of attributes as indicated in Table 24.

Table 24 Snapshot Attributes

Descriptive Name	Section	Object Name
Snapshot.filename:text	5.17.1	<not an SNMP object>
Snapshot.image:frame	5.17.2	<not an SNMP object>

4.4.11.4 Snapshot Camera Table

The snapshot camera table provides summary information related to the snapshot cameras supported by the ESS. It can be described through a number of attributes as indicated in Table 25.

Table 25 Snapshot Camera Table Attibutes

Descriptive Name	Section	Object Name
SnapshotCameraTable.numCameras:quantity	5.16.1	essSnapshotNumberOfCameras

4.4.12 Pavement Treatment System

4.4.12.1 Pavement Treatment System Class Diagram

The ESS shall support a pavement treatment system if required by the agency specification. The information supported by the PTS is depicted in Figure 19.

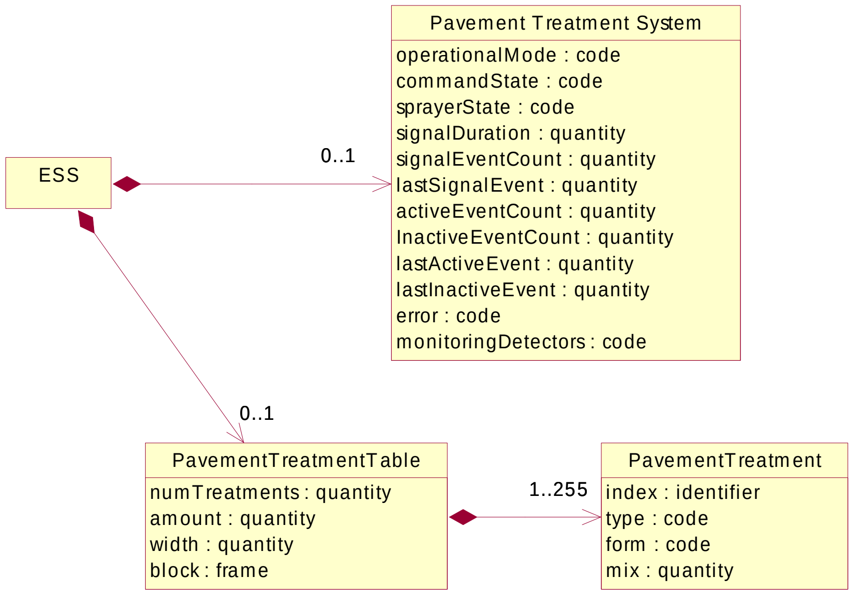

Figure 19 Pavement Treatment Class Diagram

4.4.12.2 Pavement Treatment System

A pavement treatment system is a system that controls the operation of a sprayer that disperses chemicals to prevent ice from forming on roadways. It can be described through a number of attributes as indicated in Table 26.

Table 26 Pavement Treatment System Attributes

Descriptive Name	Section	Object Name
PTS.operationalModeV3.code	5.13.19	ptsOperationalModeV3
PTS.commandStateV3:code	5.13.20	ptsCommandStateV3
PTS.sprayerState:code	5.13.9	ptsSprayerState
PTS.signalDuration:quantity	5.13.10	ptsSignalDuration
PTS.signalEventCount:quantity	5.13.11	ptsSignalEventCount
PTS.lastSignalEvent:quantity	5.13.12	ptsLastSignalEvent
PTS.activeEventCount:quantity	5.13.13	ptsActiveEventCount
PTS.inactiveEventCount:quantity	5.13.14	ptsInactiveEventCount
PTS.lastActiveEvent:quantity	5.13.15	ptsLastActiveEvent
PTS.lastInactiveEvent:quantity	5.13.16	ptsLastInactiveEvent
PTS.error:code	5.13.17	ptsError
PTS.monitoringDetectors:code	5.13.18	ptsMonitoringDetectors

4.4.12.3 Pavement Treatment Table

The pavement treatment table contains information about the various treatments that may be applied to the roadway surface. It can be described through a number of attributes as indicated in Table 27.

Table 27 Pavement Treatment Table Attributes

Descriptive Name	Section	Object Name
PavementTreatmentTable.numTreatments:quantity	5.13.1	numEssTreatments
PavementTreatmentTable.amount:quantity	5.13.4	essPaveTreatmentAmount
PavementTreatmentTable.width:quantity	5.13.5	essPaveTreatmentWidth
PavementTreatmentTable.block:frame	5.13.6	pavementTreatmentBlock

4.4.12.4 Pavement Treatment

A pavement treatment is a chemical that can be applied to a roadway surface to inhibit ice formation or promote ice melting. It can be described through a number of attributes as indicated in Table 28.

Table 28 Pavement Treatment Attributes

Descriptive Name	Section	Object Name
PavementTreatement.index:identifier	5.13.3.1	essPavementTreatmentIndex
PavementTreatement.type:code	5.13.3.2	essPaveTreatProductType
PavementTreatement.form:code	5.13.3.3	essPaveTreatProductForm
PavementTreatement.mix:quantity	5.13.3.4	essPercentProductMix

Section 5
ESS OBJECT DEFINITIONS
[NORMATIVE]

Section 5 defines those objects that are specifically used by Environmental Sensor Stations (ESS). The objects are defined using the OBJECT-TYPE macro as specified in RFC 1212 and NTCIP 8004 v02. The text provided from Section 5 through the end of Section 5.16.3.6 (except the section headings) constitutes the standard NTCIP1204-v03 MIB.

All of the objects defined in this NTCIP 1204 v03 reside under the “ess” node of the global naming tree. To aid in object management, the “ess” node has been subdivided into logical categories, each defined by a node under the “ess” node. The individual objects are then located under the appropriate node.

Conformance requirements for any object is determined by the use of the Requirements Traceability Matrix (RTM) in Annex A. To support any defined Requirement, an implementation shall support all objects to which the Requirement traces in the RTM. The value of the STATUS field for every object in the MIB is "mandatory," and indicates that it is mandatory if any associated Requirement is selected.

For all bitmapped objects, if a bit is zero (0), then the referenced function is disabled or not supported, and if a bit is one (1), then the referenced function is enabled or supported.

A computer readable format of this information, called a Management Information Base, is available from NEMA (ntcip@nema.org). The MIB has been verified using SMICng Version 2.2.07 (Book).

Previous versions of NTCIP 1204 v03 defined data elements that have been replaced to resolve ambiguities; however, central systems may need to interoperate with older equipment and support such data elements. Annex D documents the reason that the ESS WG decided to deprecate various objects.

5.0 MIB Comment Header

```
--*********************************************************************
-- Filename:    1204v0308.MIB
-- Description: This MIB defines the Environmental Sensor Station
--              Objects
--*********************************************************************
```

5.1 MIB HEADER

```
NTCIP1204-v03 DEFINITIONS ::= BEGIN
IMPORTS
  Counter
      FROM RFC1155-SMI
  DisplayString
      FROM RFC1213-MIB
  OBJECT-TYPE
      FROM RFC-1212
  ess, OerString
      FROM NTCIP8004v02;

-- For the purpose of this section, the following OBJECT IDENTIFIERS
-- are used:

essBufr OBJECT IDENTIFIER ::= {ess 1}
```



```
-- This node contains objects that describe BUFR information based on
-- the BUFR Standards.

essNtcip OBJECT IDENTIFIER ::= {ess 2}
-- This node contains objects that describe surface transportation
-- environmental information which deviate from the BUFR Standards.
```

5.2 IDENTIFICATION OBJECTS

```
-- These are objects used to describe the identification of the
-- environmental sensor station.
essNtcipIdentification  OBJECT IDENTIFIER ::= {essNtcip 1 }
```

5.2.1 Station Category

```
essNtcipCategory OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        permanent (2),
                        transportable (3),
                        mobile (4)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the category of station as it relates to
mobility.
<SetConstraint>read-only
<DescriptiveName>ESS.category:code
<Valid Value Rule>
value          description
other          of a design not listed in this standard.
permanent      not designed to be relocated.
transportable  able to be relocated, but does not take readings while moving.
mobile         capable of taking readings while moving.
<Data Concept Type>Data Element"
::=   { essNtcipIdentification 1 }
```

5.2.2 Site Description

```
essNtcipSiteDescription OBJECT-TYPE
SYNTAX      DisplayString (SIZE (0..255))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>A textual description of the station's location.
<SetConstraint>always
<Informative>Version 02 of the standard incorrectly defined a SetConstraint
of read-only, even though the object has always been defined with an ACCESS
of read-write. This was corrected in version 03.
<DescriptiveName>ESS.siteDescription:text
<Data Concept Type>Data Element"
::=   { essNtcipIdentification 2 }
```

5.3 DATA INSTRUMENTATION OBJECTS

```
-- Contains objects used to describe the type of data and the type of
-- instrumentation used to collect the data being received from the
-- ess.
essBufrInstrumentation  OBJECT IDENTIFIER ::= { essBufr 2 }
essNtcipInstrumentation OBJECT IDENTIFIER ::= { essNtcip 15 }
```

```
-- It is also recognized that there would be a great value of an object
-- to indicate the quality of data; however, this is a very complex
-- topic and thus we have not determined an appropriate
-- mechanism.
```

5.3.1 Type of Station

```
essTypeofStation OBJECT-TYPE
SYNTAX      INTEGER (0..3)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Integer value that indicates the type of station. If
the station is a hybrid station, it shall be defined as two stations, one
staffed and one automatic.
<Informative>The value of 2 has been explicitly defined as being reserved. To
be defined as two stations (e.g., for a hybrid), a station needs to have two
addresses and two instances of this MIB.
<SetConstraint>read-only
<DescriptiveName>ESS.typeOfStation:code
<Valid Value Rule>
value              description
0 - automatic      the data is collected electronically/mechanically
1 - staffed        the data is collected by humans
2 - reserved
3 - missingValue   the type of station is unknown.
<Data Concept Type>Data Element"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 02 001"
::=   { essBufrInstrumentation 1 }
```

5.3.2 Door Status

```
essDoorStatus OBJECT-TYPE
SYNTAX      INTEGER (0..1)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates whether any of the doors attached to the
station are open. If the value is one (1), at least one door is open; if the
value is zero (0), all doors associated with the ESS are closed.
<SetConstraint>read-only
<DescriptiveName>ESS.doorOpen:code
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 1 }
```

5.3.3 Battery Status

```
essBatteryStatus OBJECT-TYPE
SYNTAX      INTEGER (0..101)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the current charge stored in the battery.
<DescriptiveName>ESS.batteryCharge:quantity
<Valid Value Rule>
Values 0 to 100 indicate percent of full charge. The value 101 indicates an
error in determining the percent of charge.
<Data Concept Type>Data Element
<Unit>Percent"
::=   { essNtcipInstrumentation 2 }
```


5.3.4 Line Volts

```
essLineVolts OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the voltage measured on the incoming power
line for the controller. The value reported will indicate one-half of the
actual voltage; thus, this object will indicate a value of 55 when the
voltage is 110 Vrms. This object shall only be used to indicate A/C power
conditions. If the line power is DC, this object shall not apply (i.e., will
either not be supported or have a value of 255) and the essBatteryStatus
object shall indicate the status of the batteries.
<DescriptiveName>ESS.lineVolts:quantity
<Valid Value Rule>
Values 0 through 254 shall indicate valid values. The value 254 shall mean a
voltage of 508 Vrms or greater. The value of 255 shall indicate an error
condition or missing value.
<Data Concept Type>Data Element
<Unit>2 Volts Root Mean Squared (Vrms) (i.e., the value reported shall be
one-half the actual voltage)."
::=   { essNtcipInstrumentation 3 }
```

5.3.5 Station Meta Data Block

```
-- This object has been deprecated. See Annex D.2.8 for more
-- information.
essStationMetaDataBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>An OER encoded string of the EssStationMetaData
structure as defined below. This object is used for uploading configuration
data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssStationMetaData ::= SEQUENCE {
essNtcipCategory.0,                 OPTIONAL,   -- @NTCIP1204-v03
essTypeOfStation.0,                 OPTIONAL,   -- @NTCIP1204-v03
essLatitude.0,                      OPTIONAL,   -- @NTCIP1204-v03
essLongitude.0,                     OPTIONAL,   -- @NTCIP1204-v03
essReferenceHeight.0,               OPTIONAL,   -- @NTCIP1204-v03
essPressureHeight.0,                OPTIONAL,   -- @NTCIP1204-v03
essWindSensorHeight.0,              OPTIONAL,   -- @NTCIP1204-v03
temperatureMetaData     SEQUENCE OF TemperatureMetaData     OPTIONAL,
pavementMetaData        SEQUENCE OF PavementMetaData  OPTIONAL,
subSurfaceMetaData      SEQUENCE OF SubSurfaceMetaData      OPTIONAL,
treatmentMetaData       SEQUENCE OF TreatmentMetaData OPTIONAL
}

TemperatureMetaData ::= SEQUENCE {
essTemperatureSensorIndex.x         OPTIONAL,   -- @NTCIP1204-v03
essTemperatureSensorHeight.x        OPTIONAL    -- @NTCIP1204-v03
}
```

```
PavementMetaData ::= SEQUENCE {
essPavementSensorIndex.x,          OPTIONAL,   -- @NTCIP1204-v03
essPavementType.x,                 OPTIONAL,   -- @NTCIP1204-v03
essPavementElevation.x             OPTIONAL,   -- @NTCIP1204-v03
essPavementExposure.x              OPTIONAL,   -- @NTCIP1204-v03
essPavementSensorType.x            OPTIONAL    -- @NTCIP1204-v03
}

SubSurfaceMetaData ::= SEQUENCE {
essSubSurfaceSensorIndex.x         OPTIONAL,   -- @NTCIP1204-v03
essSubSurfaceType.x                OPTIONAL,   -- @NTCIP1204-v03
essSubSurfaceSensorDepth.x         OPTIONAL    -- @NTCIP1204-v03
}

TreatmentMetaData ::= SEQUENCE {
essPavementTreatmentIndex.x        OPTIONAL,   -- @NTCIP1204-v03
essPaveTreatProductType.x          OPTIONAL,   -- @NTCIP1204-v03
essPaveTreatProductForm.x          OPTIONAL,   -- @NTCIP1204-v03
essPercentProductMix.x             OPTIONAL    -- @NTCIP1204-v03
}
<Informative>This object has been replaced by essStationMetaDataV3Block (See
5.3.8).
<SetConstraint>read-only
<DescriptiveName>ESS.stationMetaDataBlock:frame
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 4 }
```

5.3.6 Weather Block

```
-- This object has been deprecated. See Annex D.2.8 for more
-- information.
essWeatherBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>An OER encoded string of the EssWeatherData
structure as defined in Section 4. This object is used for uploading current
weather data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssWeatherData ::= SEQUENCE {
essAtmosphericPressure.0           OPTIONAL,   -- @NTCIP1204-v03
essWindData                        EssWindData            OPTIONAL,
essTemperatureData                 EssTemperatureData     OPTIONAL,
essPrecipData                      EssPrecipData          OPTIONAL,
essVisibilityData                  EssVisibilityData      OPTIONAL
}

EssWindData ::= SEQUENCE {
essAvgWindDirection.0              OPTIONAL,   -- @NTCIP1204-v03
essAvgWindSpeed.0                  OPTIONAL,   -- @NTCIP1204-v03
essWindSituation.0                 OPTIONAL,   -- @NTCIP1204-v03
```



```
essMaxWindGustSpeed.0                OPTIONAL,   -- @NTCIP1204-v03
essMaxWindGustDir.0                  OPTIONAL,   -- @NTCIP1204-v03
essSpotWindDirection.0               OPTIONAL,   -- @NTCIP1204-v03
essSpotWindSpeed.0                   OPTIONAL    -- @NTCIP1204-v03
}

EssTemperatureData ::= SEQUENCE {
essWetBulbTemp.0                     OPTIONAL,   -- @NTCIP1204-v03
essDewpointTemp.0                    OPTIONAL,   -- @NTCIP1204-v03
essMaxTemp.0                         OPTIONAL,   -- @NTCIP1204-v03
essMinTemp.0                         OPTIONAL,   -- @NTCIP1204-v03
essRelativeHumidity.0                OPTIONAL,   -- @NTCIP1204-v03
      -- for (
--    x = 1;
--    x < essNumTemperatureSensors.0;
--    x++)
temperatureTable               SEQUENCE OF Temperature OPTIONAL
}

Temperature ::= SEQUENCE {
essTemperatureSensorIndex.x          OPTIONAL,   -- @NTCIP1204-v03
essAirTemperature.x                  OPTIONAL    -- @NTCIP1204-v03
}

EssPrecipData ::= SEQUENCE {
essWaterDepth.0                      OPTIONAL,   -- @NTCIP1204-v03
essAdjacentSnowDepth.0               OPTIONAL,   -- @NTCIP1204-v03
essRoadwaySnowDepth.0                OPTIONAL,   -- @NTCIP1204-v03
essRoadwaySnowPackDepth.0            OPTIONAL,   -- @NTCIP1204-v03
essPrecipYesNo.0                     OPTIONAL,   -- @NTCIP1204-v03
essPrecipRate.0                      OPTIONAL,   -- @NTCIP1204-v03
essSnowfallAccumRate.0               OPTIONAL,   -- @NTCIP1204-v03
essPrecipSituation.0                 OPTIONAL,   -- @NTCIP1204-v03
essIceThickness.0                    OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationStartTime.0          OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationEndTime.0            OPTIONAL    -- @NTCIP1204-v03
}
Editor's Note - I think we decided to remove the following yes?
essPrecipitationOneHour.0            OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationThreeHours.0         OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationSixHours.0           OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationTwelveHours.0        OPTIONAL,   -- @NTCIP1204-v03
essPrecipitation24Hours.0            OPTIONAL    -- @NTCIP1204-v03

EssVisibilityData ::= SEQUENCE {
essSolarRadiation.0                  OPTIONAL,   -- @NTCIP1204-v03
essTotalSun.0                        OPTIONAL,   -- @NTCIP1204-v03
essCloudSituation.0                  OPTIONAL,   -- @NTCIP1204-v03
essVisibility.0                      OPTIONAL,   -- @NTCIP1204-v03
essVisibilitySituation.0             OPTIONAL    -- @NTCIP1204-v03
}
<Informative>This object has been replaced by essWeatherV3Block (See 5.3.9).
<SetConstraint>read-only
<DescriptiveName>ESS.weatherBlock:frame
<Data Concept Type>Data Element"
```

```
::=   { essNtcipInstrumentation 5 }
```

5.3.7 Mobile Block

```
essMobileBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssMobileData structure
as defined below. This object is used for uploading current mobile station
data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssMobileData ::= SEQUENCE {
      essLatitude.0                       OPTIONAL, -- @NTCIP1204-v03
      essLongitude.0                      OPTIONAL, -- @NTCIP1204-v03
      essReferenceHeight.0                OPTIONAL, -- @NTCIP1204-v03
      essVehicleSpeed.0                   OPTIONAL, -- @NTCIP1204-v03
      essVehicleBearing.0                 OPTIONAL, -- @NTCIP1204-v03
      essVehicleOdemeter.0                OPTIONAL, -- @NTCIP1204-v03
      essMobileFriction.0                 OPTIONAL, -- @NTCIP1204-v03
      essMobileObservationGroundState.0   OPTIONAL, -- @NTCIP1204-v03
      essMobileObservationPavement.0      OPTIONAL, -- @NTCIP1204-v03
      essPaveTreatmentAmount.0            OPTIONAL, -- @NTCIP1204-v03
      essPaveTreatmentWidth.0             OPTIONAL  -- @NTCIP1204-v03
      }
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.mobileBlock:frame
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 6 }
```

5.3.8 Station Meta Data Block Version 3

```
essStationMetaDataV3Block OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssStationMetaDataV3
structure as defined below. This object is used for uploading configuration
data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssStationMetaDataV3 ::= SEQUENCE {
essNtcipCategory.0,                OPTIONAL,   -- @NTCIP1204-v03
essTypeOfStation.0,                OPTIONAL,   -- @NTCIP1204-v03
essLatitude.0,                     OPTIONAL,   -- @NTCIP1204-v03
essLongitude.0,                    OPTIONAL,   -- @NTCIP1204-v03
essReferenceHeight.0,              OPTIONAL,   -- @NTCIP1204-v03
essPressureHeight.0,               OPTIONAL,   -- @NTCIP1204-v03
precipitationSensorModelInformation.0     OPTIONAL -- @NTCIP1204-v03
windMetaData.0,         SEQUENCE OF WindMetaData            OPTIONAL,
```

```
temperatureMetaData     SEQUENCE OF TemperatureMetaData     OPTIONAL,
pavementMetaData        SEQUENCE OF PavementMetaData        OPTIONAL,
subSurfaceMetaData      SEQUENCE OF SubSurfaceMetaData      OPTIONAL,
treatmentMetaData       SEQUENCE OF TreatmentMetaData       OPTIONAL,
}

WindMetaData ::= SEQUENCE {
windSensorIndex.x                  OPTIONAL,   -- @NTCIP1204-v03
windSensorHeight.x                 OPTIONAL,   -- @NTCIP1204-v03
}

TemperatureMetaData ::= SEQUENCE {
essTemperatureSensorIndex.x        OPTIONAL,   -- @NTCIP1204-v03
essTemperatureSensorHeight.x       OPTIONAL,   -- @NTCIP1204-v03
}

PavementMetaData ::= SEQUENCE {
essPavementSensorIndex.x,          OPTIONAL,   -- @NTCIP1204-v03
essPavementType.x,                 OPTIONAL,   -- @NTCIP1204-v03
essPavementElevation.x             OPTIONAL,   -- @NTCIP1204-v03
essPavementExposure.x              OPTIONAL,   -- @NTCIP1204-v03
essPavementSensorType.x            OPTIONAL,   -- @NTCIP1204-v03
pavementSensorModelInformation.x   OPTIONAL,   -- @NTCIP1204-v03
pavementSensorTemperatureDepth.x   OPTIONAL    -- @NTCIP1204-v03
}

SubSurfaceMetaData ::= SEQUENCE {
essSubSurfaceSensorIndex.x         OPTIONAL,   -- @NTCIP1204-v03
essSubSurfaceType.x                OPTIONAL,   -- @NTCIP1204-v03
essSubSurfaceSensorDepth.x         OPTIONAL,   -- @NTCIP1204-v03
}

TreatmentMetaData ::= SEQUENCE {
essPavementTreatmentIndex.x        OPTIONAL,   -- @NTCIP1204-v03
essPaveTreatProductType.x          OPTIONAL,   -- @NTCIP1204-v03
essPaveTreatProductForm.x          OPTIONAL,   -- @NTCIP1204-v03
essPercentProductMix.x             OPTIONAL,   -- @NTCIP1204-v03
}

<SetConstraint>read-only
<DescriptiveName>ESS.stationMetaDataBlock:frame
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 7 }
```

5.3.9 Weather Block Version 3

```
essWeatherV3Block OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssWeatherDataV3
structure as defined below. This object is used for uploading current weather
data from the ESS in a bandwidth efficient manner.
```

The OPTIONAL fields shall be present if the data is supported by the implementation and is valid. The OPTIONAL fields shall be omitted for any data that is invalid or not supported by the implementation.

```
EssWeatherDataV3 ::= SEQUENCE {
essAtmosphericPressure.0            OPTIONAL,   -- @NTCIP1204-v03
essWindData                   SEQUENCE OF EssWindDataV3     OPTIONAL,
essTemperatureData                  EssTemperatureData      OPTIONAL,
essPrecipData                       EssPrecipDataV3         OPTIONAL,
essVisibilityData                   EssVisibilityDataV3     OPTIONAL,
essRadiationData                    EssRadiationData        OPTIONAL,
}

EssWindDataV3 ::= SEQUENCE {
windSensorIndex.x                   OPTIONAL,   -- @NTCIP1204-v03
windSensorAvgSpeed.x                OPTIONAL,   -- @NTCIP1204-v03
windSensorAvgDirection.x            OPTIONAL,   -- @NTCIP1204-v03
windSensorSpotSpeed.x               OPTIONAL,   -- @NTCIP1204-v03
windSensorSpotDirection.x           OPTIONAL,   -- @NTCIP1204-v03
windSensorGustSpeed.x               OPTIONAL,   -- @NTCIP1204-v03
windSensorGustDirection.x           OPTIONAL,   -- @NTCIP1204-v03
windSensorSituation.x               OPTIONAL,   -- @NTCIP1204-v03
}

EssTemperatureData ::= SEQUENCE {
essWetBulbTemp.0                    OPTIONAL,   -- @NTCIP1204-v03
essDewpointTemp.0                   OPTIONAL,   -- @NTCIP1204-v03
essMaxTemp.0                        OPTIONAL,   -- @NTCIP1204-v03
essMinTemp.0                        OPTIONAL,   -- @NTCIP1204-v03
essRelativeHumidity.0               OPTIONAL,   -- @NTCIP1204-v03
      -- for (
--    x = 1;
--    x < essNumTemperatureSensors.0;
--    x++)
temperatureTable              SEQUENCE OF Temperature OPTIONAL,
}

Temperature ::= SEQUENCE {
essTemperatureSensorIndex.x         OPTIONAL,   -- @NTCIP1204-v03
essAirTemperature.x                 OPTIONAL,   -- @NTCIP1204-v03
}

EssPrecipDataV3 ::= SEQUENCE {
waterLevelSensorTable               SEQUENCE OF WaterLevel  OPTIONAL,
essAdjacentSnowDepth.0              OPTIONAL,   -- @NTCIP1204-v03
essRoadwaySnowDepth.0               OPTIONAL,   -- @NTCIP1204-v03
essRoadwaySnowPackDepth.0           OPTIONAL,   -- @NTCIP1204-v03
essPrecipYesNo.0                    OPTIONAL,   -- @NTCIP1204-v03
essPrecipRate.0                     OPTIONAL,   -- @NTCIP1204-v03
essSnowfallAccumRate.0              OPTIONAL,   -- @NTCIP1204-v03
essPrecipSituation.0                OPTIONAL,   -- @NTCIP1204-v03
essIceThickness.0                   OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationStartTime.0         OPTIONAL,   -- @NTCIP1204-v03
essPrecipitationEndTime.0           OPTIONAL,   -- @NTCIP1204-v03
}
```



```
WaterLevel ::= SEQUENCE {
waterLevelSensorIndex.x            OPTIONAL,   -- @NTCIP1204-v03
waterLevelSensorReading.x          OPTIONAL,   -- @NTCIP1204-v03
}

EssVisibilityDataV3 ::= SEQUENCE {
essVisibility.0                    OPTIONAL,   -- @NTCIP1204-v03
essVisibilitySituation.0           OPTIONAL,   -- @NTCIP1204-v03
}

EssRadiationData ::= SEQUENCE {
essTotalSun.0                      OPTIONAL,   -- @NTCIP1204-v03
essInstantaneousTerrestrialRadiation.0    OPTIONAL, -- @NTCIP1204-v03
essInstantaneousSolarRadiation.0   OPTIONAL,   -- @NTCIP1204-v03
essTotalRadiation.0                OPTIONAL,   -- @NTCIP1204-v03
essTotalRadiationPeriod.0          OPTIONAL,   -- @NTCIP1204-v03
essCloudSituation.0                OPTIONAL,   -- @NTCIP1204-v03
}
<SetConstraint>read-only
<DescriptiveName>ESS.weatherBlock:frame
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 8 }
```

5.4 LOCATION OBJECTS

```
-- Contains objects used to describe the location of the ess that is
-- transmitting the collected data.
essNtcipLocation OBJECT IDENTIFIER ::= {essNtcip 2 }
```

5.4.1 Latitude

```
essLatitude OBJECT-TYPE
SYNTAX      INTEGER (-90000000..90000001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The latitude in 10^-6 degrees of the ESS station,
per WGS-84 datum.
<SetConstraint>read-only
<DescriptiveName>ESS.latitude:quantity
<Valid Value Rule>
The essLatitude at the North Pole is 90,000,000. The essLatitude at the South
Pole is -90,000,000. The value 90,000,001 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>latitude"
REFERENCE   "Resolution based on on-going location referencing activities;
the WMO Binary Code Form FM 94 BUFR Table B item 0 05 001 can be obtained by
dividing this value by 10."
::=   { essNtcipLocation 1 }
```

5.4.2 Longitude

```
essLongitude OBJECT-TYPE
SYNTAX      INTEGER (-180000000..180000001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The east longitude in 10^-6 degrees from the Prime
Meridian of the ESS location.
```

```
<SetConstraint>read-only
<DescriptiveName>ESS.longitude:quantity
<Valid Value Rule>
The essLongitude of 180 degrees West shall be -180,000,000. The essLongitude
of 180 degrees East shall be 180,000,000. The value 180,000,001 shall
indicate a missing value.
<Data Concept Type>Data Element
<Unit>longitude"
REFERENCE   "Resolution based on on-going location referencing activities;
the WMO Binary Code Form FM 94 BUFR Table B item 0 06 001 can be obtained by
dividing this value by 10."
::=   { essNtcipLocation 2 }
```

5.4.3 Vehicle Speed

```
essVehicleSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the current speed being reported by the
vehicle in kilometers per hour.
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.speed:quantity
<Valid Value Rule>
The value 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>kilometers per hour"
::=   { essNtcipLocation 3 }
```

5.4.4 Vehicle Bearing

```
essVehicleBearing OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the current bearing of the vehicle in
degrees, measured clockwise from True North.
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.bearing:quantity
<Valid Value Rule>
The value 0 shall indicate that the vehicle is stopped. The value 361 shall
indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>degrees"
::=   { essNtcipLocation 4 }
```

5.4.5 Odometer

```
essOdometer OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the current odometer reading of the
vehicle in meters.
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.odometer:quantity
<Data Concept Type>Data Element
<Unit>meters"
```



```
::=   { essNtcipLocation 5 }
```

5.5 STATION ELEVATION OBJECTS

```
-- Contains objects used to describe the elevation and atmospheric
-- pressure at the ess that is transmitting the collected data along
-- with the height of various sensors
essNtcipHeight OBJECT IDENTIFIER ::= {essNtcip 3 }
essBufrLocationVertical OBJECT IDENTIFIER ::= {essBufr 7 }
```

5.5.1 Reference Height

```
essReferenceHeight OBJECT-TYPE
SYNTAX      INTEGER (-400..8001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The reference elevation of the ESS in meters above
mean sea level. For a permanent station, this height shall be measured to the
base of the structure; for transportable stations, this height shall be
measured to the ground surface upon which the station resides; and for
mobile, this height shall be measured to the surface under the vehicle.
<SetConstraint>read-only
<DescriptiveName>ESS.referenceHeight:quantity
<Valid Value Rule>
The value of 8001 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>meters"
REFERENCE   "Resolution based on WMO Binary Code Form FM 94 BUFR Table B item
0 07 001."
::=   { essNtcipHeight 1 }
```

5.5.2 Pressure Height

```
essPressureHeight OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The height of the pressure sensor with respect to
the essReferenceHeight in meters.
<SetConstraint>read-only
<DescriptiveName>PressureSensor.pressureHeight:quantity
<Valid Value Rule>
The value of 1001 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>meters"
REFERENCE   "essReferenceHeight plus this value equals the WMO Binary Code
Form FM 94 BUFR Table B item 0 07 001."
::=   { essNtcipHeight 2 }
```

5.5.3 Wind Sensor Height

```
-- This object has been deprecated. See Annex D.1.3 for more information.
essWindSensorHeight OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The height of the primary wind sensor with respect
to the essReferenceHeight in meters.
```

```
<SetConstraint>read-only
<Valid Value Rule>
The value of 1001 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>meters"
::=   { essNtcipHeight 3 }
```

5.5.4 Atmospheric Pressure

```
essAtmosphericPressure OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The force per unit area exerted by the atmosphere in
1/10ths of millibars, a.k.a. tenths of hectoPascals.
<SetConstraint>read-only
<DescriptiveName>PressureSensor.atmosphericPressure:quantity
<Valid Value Rule>
A value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>Decapascal"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 07 004."
::=   { essBufrLocationVertical 4 }
```

5.6 WIND DATA SECTION

```
-- Contains objects used to describe the wind data that is collected at
-- the ess.
essBufrWind OBJECT IDENTIFIER ::= {essBufr 11 }
essNtcipWind OBJECT IDENTIFIER ::= {essNtcip 4 }
```

5.6.1 Average Wind Direction

-- This object has been deprecated. See Annex D.1.3 for more information.

```
essAvgWindDirection OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>A two minute average of the direction from which the
wind is blowing measured clockwise in degrees from true North and measured at
a height as indicated by essWindSensorHeight. A value of 361 shall indicate
an error condition or missing value.
<SetConstraint>read-only
<Data Concept Type>Data Element
<Unit>degrees"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 11 001."
::=   { essBufrWind 1 }
```

5.6.2 Average Wind Speed

-- This object has been deprecated. See Annex D.1.3 for more information.

```
essAvgWindSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>A two minute average of the wind speed in tenths of
meters per second as measured by the primary wind sensor.
<SetConstraint>read-only
```



```
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of meters per second"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 11 002."
::=   { essBufrWind 2 }
```

5.6.3 Spot Wind Direction

```
-- This object has been deprecated. See Annex D.1.3 for more information.
essSpotWindDirection OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The direction from which the wind is blowing
measured in degrees clockwise from true North and measured at a height as
indicated by essWindSensorHeight. A value of 361 shall indicate an error
condition or missing value. For mobile platforms, the wind direction shall be
corrected for vehicle movement.
<SetConstraint>read-only
<Data Concept Type>Data Element
<Unit>degrees"
::=   { essNtcipWind 1 }
```

5.6.4 Spot Wind Speed

```
-- This object has been deprecated. See Annex D.1.3 for more information.
essSpotWindSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The wind speed in tenths of meters per second
measured by the primary wind sensor. For mobile platforms, the wind speed
shall be corrected for vehicle movement.
<SetConstraint>read-only
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of meters per second"
::=   { essNtcipWind 2 }
```

5.6.5 Wind Situation

```
-- This object has been deprecated. See Annex D.1.3 for more information.
essWindSituation OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        unknown (2),
                        calm (3),
                        lightBreeze (4),
                        moderateBreeze (5),
                        strongBreeze (6),
                        gale (7),
                        moderateGale (8),
                        strongGale (9),
                        stormWinds (10),
                        hurricaneForceWinds (11),
                        gustyWinds (12)}
ACCESS      read-only
```

```
STATUS      deprecated
DESCRIPTION "<Definition>Describes the weather and travel situation in terms
of wind from staffed stations only. Specific ranges for these values are
defined in the Glossary of Meteorology.
<SetConstraint>read-only
<Valid Value Rule>
Range       Meaning
other       not defined within this standard, see manufacturers documentation
unknown     Unknown conditions
calm        Calm
lightBreeze Light breeze
moderateBreeze    Moderate breeze
strongBreeze      Strong breeze
gale        Gale
moderateGale      Moderate gale
strongGale  Strong gale
stormWinds  Storm winds
hurricaneForceWinds     Hurricane force winds
gustyWinds  defined by a peak and a lull of greater than 46.3 tenths of
            meters per second within a 2 minute period.
<Data Concept Type>Data Element"
::=   { essNtcipWind 3 }
```

5.6.6 Wind Gust Speed

```
-- This object has been deprecated. See Annex D.1.3 for more information.
essMaxWindGustSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The maximum wind gust recorded by the primary wind
sensor during the 10 minutes preceding the observation measured in tenths of
meters per second.
<SetConstraint>read-only
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of meters per second"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 11 041."
::=   { essBufrWind 41 }
```

5.6.7 Wind Gust Direction

```
-- This object has been deprecated. See Annex D.1.3 for more information.
essMaxWindGustDir OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The direction of the maximum wind gust recorded
during the 10 minutes preceding the observation at a height as indicated by
essWindSensorHeight; measured in degrees clockwise from true North. The value
361 shall indicate an error condition or missing value.
<SetConstraint>read-only
<Data Concept Type>Data Element
<Unit>degrees"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 11 043."
::=   { essBufrWind 43 }
```


5.6.8 Number of Wind Sensors

```
windSensorTableNumSensors OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of entries in the wind sensor
table.
<Informative>This value may automatically change upon connecting or
disconnecting a sensor; however, the table is still defined as a static table
since the creation/deletion of rows is not managed through SNMP logic.
<SetConstraint>read-only
<DescriptiveName>WindSensorTable.numSensors:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipWind 7 }
```

5.6.9 Wind Sensor Table

```
windSensorTable OBJECT-TYPE
SYNTAX      SEQUENCE OF WindSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Table containing the wind sensor data fields.
<DescriptiveName>WindSensorTable
<Data Concept Type>Class
<TableType> static"
::=   { essNtcipWind 8 }
```

5.6.10 Wind Sensor

```
windSensorEntry OBJECT-TYPE
SYNTAX      WindSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Parameters for specific wind sensor data fields.
<DescriptiveName>WindSensor
<Data Concept Type>Class"
INDEX { windSensorIndex }
::=   { windSensorTable 1 }

WindSensorEntry ::= SEQUENCE {
      windSensorIndex              INTEGER,
      windSensorHeight             INTEGER,
      windSensorLocation           DisplayString,
      windSensorAvgSpeed           INTEGER,
      windSensorAvgDirection       INTEGER,
      windSensorSpotSpeed          INTEGER,
      windSensorSpotDirection      INTEGER,
      windSensorGustSpeed          INTEGER,
      windSensorGustDirection      INTEGER,
      windSensorSituation          INTEGER }
```

5.6.10.1 Wind Sensor Index

```
windSensorIndex OBJECT-TYPE
```

```
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Enumerated list of row entries that will provide
wind sensor data. The first entry shall be that of the primary wind sensor.
<SetConstraint>read-only
<DescriptiveName>WindSensor.index:identifier
<Data Concept Type>Data Element"
::= { windSensorEntry 1 }
```

5.6.10.2 Wind Sensor Height

```
windSensorHeight OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The height of the wind sensor with respect to the
essReferenceHeight in meters.
<SetConstraint>read-only
<DescriptiveName>WindSensor.height:quantity
<Valid Value Rule>
The value of 1001 shall indicate a mssing value.
<Data Concept Type>Data Element
<Unit>meters"
::= { windSensorEntry 2 }
```

5.6.10.3 Wind Sensor Location

```
windSensorLocation OBJECT-TYPE
SYNTAX      DisplayString (SIZE (0..255))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>A textual string indicating the location of the wind
sensor.
<SetConstraint>always
<DescriptiveName>WindSensor.location:text
<Data Concept Type>Data Element"
::= { windSensorEntry 3 }
```

5.6.10.4 Wind Sensor Average Speed

```
windSensorAvgSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A two minute average of the wind speed in tenths of
meters per second as measured by the wind sensor.
<SetConstraint>read-only
<DescriptiveName>WindSensor.avgSpeed:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of meters per second"
REFERENCE "WMO Binary Code Form FM 94 BUFR Table B item 0 11 002."
::= { windSensorEntry 4 }
```


5.6.10.5 Wind Sensor Average Direction

```
windSensorAvgDirection OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A two minute mode (average) of the direction from
which the wind is blowing measured clockwise in degrees from true north as
measured by the wind sensor.
<SetConstraint>read-only
<DescriptiveName>WindSensor.avgDirection:quantity
<Valid Value Rule>
The value of zero (0) shall indicate 'calm', when the associated speed is
zero (0), or 'light and variable,' when the associated speed is greater than
zero (0). Normal observations, as defined by the WMO, shall report a wind
direction in the range of 1 to 360 with 90 meaning from the east and 360
meaning from the north. The value of 361 shall indicate an error condition
and shall always be reported if the associated speed indicates error.
<Data Concept Type>Data Element
<Unit>degrees"
REFERENCE "WMO Code Form FM 94 BUFR Table B item 0 11 001."
::= { windSensorEntry 5 }
```

5.6.10.6 Wind Sensor Spot Speed

```
windSensorSpotSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The wind speed in tenths of meters per second
measured by the wind sensor. For mobile platforms, the wind speed shall be
corrected for vehicle movement.
<SetConstraint>read-only
<DescriptiveName>WindSensor.spotSpeed:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of meters per second"
::= { windSensorEntry 6 }
```

5.6.10.7 Wind Sensor Spot Direction

```
windSensorSpotDirection OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The direction from which the wind is blowing
measured in degrees clockwise from true North as measured by the wind sensor.
For mobile platforms, the wind direction shall be corrected for vehicle
movement.
<SetConstraint>read-only
<DescriptiveName>WindSensor.spotDirection:quantity
<Valid Value Rule>
The value of zero (0) shall indicate 'calm', when the associated speed is
zero (0), or 'light and variable,' when the associated speed is greater than
zero (0). Normal observations, as defined by the WMO, shall report a wind
direction in the range of 1 to 360 with 90 meaning from the east and 360
```

```
meaning from the north. The value of 361 shall indicate an error condition
and shall always be reported if the associated speed indicates error.
<Data Concept Type>Data Element
<Unit>degrees"
::= { windSensorEntry 7 }
```

5.6.10.8 Wind Sensor Gust Speed

```
windSensorGustSpeed OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The maximum wind gust recorded by the wind sensor
during the 10 minutes preceding the observation measured in tenths of meters
per second.
<SetConstraint>read-only
<DescriptiveName>WindSensor.gustSpeed:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of meters per second"
REFERENCE "WMO Code Form FM 94 BUFR Table B item 0 11 041."
::= { windSensorEntry 8 }
```

5.6.10.9 Wind Sensor Gust Direction

```
windSensorGustDirection OBJECT-TYPE
SYNTAX      INTEGER (0..361)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The direction of the maximum wind gust recorded
during the 10 minutes preceding the observation measured in degrees clockwise
from true North by the wind sensor.
<SetConstraint>read-only
<DescriptiveName>WindSensor.gustDirection:quantity
<Valid Value Rule>
The value of zero (0) shall indicate 'calm', when the associated speed is
zero (0), or 'light and variable,' when the associated speed is greater than
zero (0). Normal observations, as defined by the WMO, shall report a wind
direction in the range of 1 to 360 with 90 meaning from the east and 360
meaning from the north. The value of 361 shall indicate an error condition
and shall always be reported if the associated speed indicates error.
<Data Concept Type>Data Element
<Unit>degrees"
REFERENCE "WMO Code Form FM 94 BUFR Table B item 0 11 043."
::= { windSensorEntry 9 }
```

5.6.10.10 Wind Sensor Situation

```
windSensorSituation OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        unknown (2),
                        calm (3),
                        lightBreeze (4),
                        moderateBreeze (5),
                        strongBreeze (6),
                        gale (7),
                        moderateGale (8),
```



```
                        strongGale (9),
                        stormWinds (10),
                        hurricaneForceWinds (11),
                        gustyWinds (12)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Describes the weather and travel situation in terms
of wind from staffed stations only. Specific ranges for these values are
defined in the Glossary of Meteorology.
<DescriptiveName>WindSensor.situation:code
<Valid Value Rule>
Range       Meaning
other       not defined within this standard, see manufacturers documentation
unknown     Unknown conditions
calm        Calm
lightBreeze Light breeze
moderateBreeze    Moderate breeze
strongBreeze      Strong breeze
gale        Gale
moderateGale      Moderate gale
strongGale  Strong gale
stormWinds  Storm winds
hurricaneForceWinds     Hurricane force winds
gustyWinds  defined by a peak and a lull of greater than 46.3 tenths of
            meters per second within a 2 minute period.
<Data Concept Type>Data Element"
::= { windSensorEntry 10 }
```

5.7 TEMPERATURE DATA OBJECTS

```
-- Contains objects used to describe the temperature data that is
-- collected at the ess.
essNtcipTemperature OBJECT IDENTIFIER ::= {essNtcip 5}
```

5.7.1 Number of Temperature Sensors

```
essNumTemperatureSensors OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of entries in the temperature
sensor table.
<Informative>This value may automatically change upon connecting or
disconnecting a sensor; however, the table is still defined as a static table
since the creation/deletion of rows is not managed through SNMP logic.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensorTable.numSensors:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipTemperature 1 }
```

5.7.2 Temperature Sensor Table

```
essTemperatureSensorTable OBJECT-TYPE
SYNTAX      SEQUENCE OF EssTemperatureSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Table containing the temperature sensor data fields.
```

```
<DescriptiveName>TemperatureSensorTable
<Data Concept Type>Class
<TableType> static"
::=   { essNtcipTemperature 2 }
```

5.7.3 Temperature Sensor

```
essTemperatureSensorEntry OBJECT-TYPE
SYNTAX      EssTemperatureSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Parameters for specific temperature sensor as
described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>TemperatureSensor
<Data Concept Type>Class"
INDEX { essTemperatureSensorIndex }
::=   { essTemperatureSensorTable 1 }

EssTemperatureSensorEntry ::= SEQUENCE {
      essTemperatureSensorIndex                 INTEGER,
      essTemperatureSensorHeight                INTEGER,
      essAirTemperature                   INTEGER }
```

5.7.3.1 Temperature Sensor Index

```
essTemperatureSensorIndex OBJECT-TYPE
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Enumerated list of row entries that will provide
temperature sensor data.
<SetConstraint>index
<DescriptiveName>TemperatureSensor.index:identifier
<Data Concept Type>Data Element"
::= { essTemperatureSensorEntry 1 }
```

5.7.3.2 Temperature Sensor Height

```
essTemperatureSensorHeight OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The height of the temperature sensor as measured in
meters above essReferenceHeight.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensor.height:quantity
<Valid Value Rule>
The value 1001 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>meters"
::= { essTemperatureSensorEntry 2 }
```

5.7.3.3 Air Temperature

```
essAirTemperature OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
```



```
STATUS      mandatory
DESCRIPTION "<Definition>The dry-bulb temperature in tenths of degrees
Celsius. The temperature is an instantaneous reading at the height specified
by essTemperatureSensorHeight.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensor.airTemperature:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
REFERENCE "Resolution is based on WMO Binary Code Form FM 94 BUFR Table B
item 0 12 001; temperature in Kelvin is determined by adding 273.15 to this
value."
::= { essTemperatureSensorEntry 3 }
```

5.7.4 Wetbulb Temperature

```
essWetbulbTemp OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The wet-bulb temperature in tenths of degrees
Celsius. The temperature is an instantaneous reading at the height specified
by the essTemperatureSensorHeight as specified in the first row of the
essTemperatureTable.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensorTable.wetBulbTemp:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
REFERENCE   "is based on WMO Binary Code Form FM 94 BUFR Table B item 0 12
002; temperature in Kelvin is determined by adding 273.15 to this value."
::=   { essNtcipTemperature 3 }
```

5.7.5 Dewpoint Temperature

```
essDewpointTemp OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The dewpoint temperature in tenths of degrees
Celsius. The temperature is an instantaneous reading at the height specified
by the essTemperatureSensorHeight as specified in the first row of the
essTemperatureTable.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensorTable.dewpoint:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
REFERENCE   "Resolution is based on WMO Binary Code Form FM 94 BUFR Table B
item 0 12 003; temperature in Kelvin is determined by adding 273.15 to this
value."
::=   { essNtcipTemperature 4 }
```

5.7.6 Maximum Temperature

```
essMaxTemp OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The maximum temperature in tenths of degrees Celsius
recorded during the 24 hours preceding the observation at the height
specified by the essTemperatureSensorHeight as specified in the first row of
the essTemperatureTable.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensorTable.maxTemp:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
REFERENCE   "Resolution is based on WMO Binary Code Form FM 94 BUFR Table B
item 0 12 011; temperature in Kelvin is determined by adding 273.15 to this
value."
::=   { essNtcipTemperature 5 }
```

5.7.7 Minimum Temperature

```
essMinTemp OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The minimum temperature in tenths of degrees Celsius
recorded during the 24 hours preceding the observation at the height
specified by the essTemperatureSensorHeight as specified in the first row of
the essTemperatureTable.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensorTable.minTemp:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
REFERENCE   "Resolution is based on WMO Binary Code Form FM 94 BUFR Table B
item 0 12 012; temperature inKelvin is determined by adding 273.15 to this
value."
::=   { essNtcipTemperature 6 }
```

5.8 HUMIDITY AND PRECIPITATION DATA OBJECTS

```
-- Contains objects used to describe the humidity and precipitation
-- data that is collected by the ess.
essBufrPrecip OBJECT IDENTIFIER ::= {essBufr 13 }
essNtcipPrecip OBJECT IDENTIFIER ::= {essNtcip 6 }
```

5.8.1 Relative Humidity

```
essRelativeHumidity OBJECT-TYPE
SYNTAX      INTEGER (0..101)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The relative humidity in percent.
<SetConstraint>read-only
<DescriptiveName>TemperatureSensorTable.relativeHumidity:quantity
```



```
<Valid Value Rule>
The value of 101 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>percent humidity"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 003."
::=   { essBufrPrecip 3 }
```

5.8.2 Water Depth

-- This object has been deprecated. See Annex D.1.4 for more information.

```
essWaterDepth OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>Indicates the depth of the water from a user defined
point in centimeters. The value of 65535 shall indicate an error condition or
missing value. This may be used for stream depth, depth of water over a
roadway, reservoir depth, or other such uses.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.waterDepth:quantity
<Data Concept Type>Data Element
<Unit>centimeters"
::=   { essNtcipPrecip 1 }
```

5.8.3 Adjacent Snow Depth

```
essAdjacentSnowDepth OBJECT-TYPE
SYNTAX      INTEGER (0..3001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The depth of snow in centimeters on representative
areas other than the highway pavement, avoiding drifts and plowed areas.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.adjacentSnowDepth:quantity
<Valid Value Rule>
The value 3001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>centimeters"
::=   { essNtcipPrecip 2 }
```

5.8.4 Roadway Snow Depth

```
essRoadwaySnowDepth OBJECT-TYPE
SYNTAX      INTEGER (0..3001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The current depth of unpacked snow in centimeters on
the driving surface.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.roadwaySnowDepth:quantity
<Valid Value Rule>

The value 3001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>centimeters"
::=   { essNtcipPrecip 3 }
```

5.8.5 Roadway Snow Pack Depth

```
essRoadwaySnowPackDepth OBJECT-TYPE
SYNTAX      INTEGER (0..3001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The current depth of packed snow in centimeters on
the roadway surface.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.roadwaySnowPackDepth:quantity
<Valid Value Rule>
The value 3001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>centimeters"
::=   { essNtcipPrecip 4 }
```

5.8.6 Precipitation Indicator

```
essPrecipYesNo OBJECT-TYPE
SYNTAX      INTEGER {   precip (1),
                        noPrecip (2),
                        error (3)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates whether or not moisture is detected by the
sensor.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.precipitationIndicator:code
<Valid Value Rule>
precip - Moisture is currently being detected by the precipitation sensor
noPrecip - Moisture is not currently being detected by the precipitation
sensor
error - The sensor is either not connected, not reporting, or is indicating
an error
<Data Concept Type>Data Element"
::=   { essNtcipPrecip 5 }
```

5.8.7 Rainfall or Water Equivalent of Snow

```
essPrecipRate OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The rainfall, or water equivalent of snow, rate in
tenths of grams per square meter per second (for rain, this is approximately
to 0.36 mm/hr).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.precipitationRate:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of grams per square meter per second"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 014."
::=   { essBufrPrecip 14 }
```

5.8.8 Snowfall Accumulation Rate

```
essSnowfallAccumRate OBJECT-TYPE
```



```
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The snowfall accumulation rate in 10^-7 meters per
second (this is equivalent to 0.36 mm/hr).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.snowfallAccumulationRate:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>10^-7 meters per second"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 015."
::=   { essBufrPrecip 15 }
```

5.8.9 Precipitation Situation

```
essPrecipSituation OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        unknown (2),
                        noPrecipitation (3),
                        unidentifiedSlight (4),
                        unidentifiedModerate (5),
                        unidentifiedHeavy (6),
                        snowSlight (7),
                        snowModerate (8),
                        snowHeavy (9),
                        rainSlight (10),
                        rainModerate (11),
                        rainHeavy (12),
                        frozenPrecipitationSlight (13),
                        frozenPrecipitationModerate (14),
                        frozenPrecipitationHeavy (15)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Describes the weather situation in terms of
precipitation.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.precipitationSituation:code
<Valid Value Rule>
Intensity     Meaning
slight      < 2mm/h water equivalent
moderate    >= 2 and < 8 mm/h water equivalent
heavy       >= 8 mm/h water equivalent If one exists, the corresponding BUFR
value is indicated for staffed (BUFRs) and automated (BUFRa) stations. The
indicated value can be found in the BUFR Table referenced below. Defined
values are:
Range   BUFRa   BUFRs  Meaning
1                      other
2                      unknown
3                      no precipitation
4                      unidentified slight
5                      unidentified moderate
6                      unidentified heavy
7       171      85    snow slight
8       172      86    snow moderate
9       173      86    snow heavy
```

```
10              61    rain slight
11      165     63    rain moderate
12      163     65    rain heavy
13                    frozen precipitation slight
14                    frozen precipitation moderate
15                    frozen precipitation heavy
<Data Concept Type>Data Element"
REFERENCE   "The values identified in the above table for BUFRa and BUFRs can
be found in WMO Binary Code Form FM 94 BUFR Table B item 0 20 003."
::=   { essNtcipPrecip 6 }
```

5.8.10 Ice Deposit (Thickness)

```
essIceThickness OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the thickness of the ice in millimeters.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.iceDeposit:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>millimeters"
::=   { essNtcipPrecip 7 }
```

5.8.11 Precipitation Start Time

```
essPrecipitationStartTime OBJECT-TYPE
SYNTAX      INTEGER (0..4294967295)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The time at which the most recent precipitation
event began, measured in seconds since 00:00:00 January 1, 1970 UTC.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.precipitationStartTime:quantity
<Valid Value Rule>
As this standard has been developed long after 1970, a value a 0 for time
should indicate to the management station that the data received is suspect.
<Data Concept Type>Data Element
<Unit>seconds"
::=   { essNtcipPrecip 8 }
```

5.8.12 Precipitation End Time

```
essPrecipitationEndTime OBJECT-TYPE
SYNTAX      INTEGER (0..4294967295)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The time at which the most recently completed
precipitation event ended, measured in seconds since 00:00:00 January 1, 1970
UTC.
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.precipitationEndTime:quantity
<Valid Value Rule>
As this standard has been developed long after 1970, a value of 0 for the
time should indicate to the management station that the data received is
suspect.
```



```
<Data Concept Type>Data Element
<Unit>seconds"
::=   { essNtcipPrecip 9 }
```

5.8.13 Total Precipitation Past One Hour

```
essPrecipitationOneHour OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The total water equivalent precipitation over the
hour preceding the observation in tenths of kilograms per square meter (for
rain, this is approximately tenths of millimeters).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.oneHour:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of kilograms per square meter"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 019."
::=   { essBufrPrecip 19 }
```

5.8.14 Total Precipitation Past Three Hours

```
essPrecipitationThreeHours OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The total water equivalent precipitation over the
three hours preceding the observation in tenths of kilograms per square meter
(for rain, this is approximately tenths of millimeters).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.threeHours:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of kilograms per square meter"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 020."
::=   { essBufrPrecip 20 }
```

5.8.15 Total Precipitation Past Six Hours

```
essPrecipitationSixHours OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The total water equivalent precipitation over the
six hours preceding the observation in tenths of kilograms per square meter
(for rain, this is approximately tenths of millimeters).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.sixHours:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of kilograms per square meter"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 021."
::=   { essBufrPrecip 21 }
```

5.8.16 Total Precipitation Past Twelve Hours

```
essPrecipitationTwelveHours OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The total water equivalent precipitation over the
twelve hours preceding the observation in tenths of kilograms per square
meter (for rain, this is approximately to tenths of millimeters).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.twelveHours:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of kilograms per square meter"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 022."
::=   { essBufrPrecip 22 }
```

5.8.17 Total Precipitation Past Twenty-Four Hours

```
essPrecipitation24Hours OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The total water equivalent precipitation over the
twenty-four hours preceding the observation in tenths of kilograms per square
meter (for rain, this is equivalent to tenths of millimeters).
<SetConstraint>read-only
<DescriptiveName>PrecipitationSensor.24Hours:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of kilograms per square meter"
REFERENCE   "WMO Binary Code Form FM 94 BUFR Table B item 0 13 023."
::=   { essBufrPrecip 23 }
```

5.8.18 Precipitation Sensor Model Information

```
precipitationSensorModelInformation OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>A reference to the row in the Module Table (See
NTCIP 1201) that indicates the manufacturer, model, and version number of the
precipitation sensor.
<DescriptiveName>PrecipitationSensor.modelInformation:identifier
<Valid Value Rule>
The value of zero indicates that this information is not available.
<Data Concept Type>Data Element"
::=   { essNtcipPrecip 10 }
```

5.8.19 Number of Water Level Sensors

```
waterLevelSensorTableNumSensors OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
```



```
DESCRIPTION "<Definition>Indicates the number of entries in the water level
sensor table.
<Informative>This value may automatically change upon connecting or
disconnecting a sensor; however, the table is still defined as a static table
since the creation/deletion of rows is not managed through SNMP logic.
<DescriptiveName>WaterLevelSensorTable.numSensors:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipPrecip 11 }
```

5.8.20 Water Level Sensor Table

```
waterLevelSensorTable OBJECT-TYPE
SYNTAX      SEQUENCE OF WaterLevelSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Table containing the water level sensor data fields.
<DescriptiveName>WaterLevelSensorTable
<Data Concept Type>Class
<TableType> static"
::=   { essNtcipPrecip 12 }
```

5.8.21 Water Level Sensor

```
waterLevelSensorEntry OBJECT-TYPE
SYNTAX      WaterLevelSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Parameters for a specific water level sensor as
described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>WaterLevelSensor
<Data Concept Type>Class"
INDEX { waterLevelSensorIndex }
::=   { waterLevelSensorTable 1 }

WaterLevelSensorEntry ::= SEQUENCE {
waterLevelSensorIndex         INTEGER,
waterLevelSensorReading       INTEGER }
```

5.8.21.1 Water Level Sensor Index

```
waterLevelSensorIndex OBJECT-TYPE
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Enumerated list of row entries that will provide
water level sensor data.
<DescriptiveName>WaterLevelSensor.index:identifier
<Data Concept Type>Data Element"
::= { waterLevelSensorEntry 1 }
```

5.8.21.2 Water Level Sensor Reading

```
waterLevelSensorReading OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
```



```
DESCRIPTION "<Definition>Indicates the depth of the water from a user defined
point in centimeters. This may be used for stream depth, depth of water over
a roadway, reservoir depth, or other such uses.
<DescriptiveName>WaterLevelSensor.reading:quantity
<Valid Value Rule>
The value of 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>centimeters"
::= { waterLevelSensorEntry 2 }
```

5.9 RADIATION OBJECTS

```
-- Contains objects used to describe the data that is collected by the
-- pavement surface sensor.
essBufrRadiation  OBJECT IDENTIFIER ::= {essBufr 14 }
essNtcipRadiation  OBJECT IDENTIFIER ::= {essNtcip 7}
```

5.9.1 Solar Radiation

-- This object has been deprecated. See Annex D.1.5 for more information.

```
essSolarRadiation OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The direct solar radiation integrated over the 24
hours preceding the observation in Joules per square meter. The value of
65535 shall indicate a missing value.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.solarRadiation:quantity
<Data Concept Type>Data Element
<Unit>Joules per square meter"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 14 024."
::=   { essBufrRadiation 24 }
```

5.9.2 Total Sun

```
essTotalSun OBJECT-TYPE
SYNTAX      INTEGER (0..1441)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The total amount of sunshine in minutes over the 24
hour period preceding the observation.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.totalSun:quantity
<Valid Value Rule>
The value of 1441 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>minutes"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 14 031."
::=   { essBufrRadiation 31 }
```

5.9.3 Cloud Cover Situation

```
essCloudSituation OBJECT-TYPE
SYNTAX      INTEGER {   overcast (1),
                        cloudy (2),
                        partlyCloudy (3),
                        mostlyClear (4),
```



```
                        clear (5)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Describes the amount of cloud cover. The associated
percentages of cloud cover are indicated to identify the differences between
the defined values.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.cloudCoverSituation:code
<Valid Value Rule>
Defined values are:
Range  BUFRs  BUFRa   Meaning
1                    other visibility anomaly
2                    unknown
3       0     100    clear
4      44     130    Fog - not patchy
5      41     131    Patchy fog
6      36     127    Blowing snow
7      04     104    Smoke
8      07     207    Sea Spray
9                    Vehicle Spray
10     31     127    Blowing dust or sand
11                   sun glare
12                   Swarms of insects
<Data Concept Type>Data Element"
::=   { essNtcipRadiation 1 }
```

5.9.4 Terrestrial Radiation

```
essInstantaneousTerrestrialRadiation OBJECT-TYPE
SYNTAX      INTEGER (-2048..2049)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The instantaneous infrared (wavelength of 3.5 - 50
micrometers) radiation being emitted from the atmosphere in watts per square
meter.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.instantaneousTerrestrialRadiation:quantity
<Valid Value Rule>
The value of 2049 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>watts per square meter"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 14 017"
::=   { essBufrRadiation 17 }
```

5.9.5 Solar Radiation—Version 2

```
essInstantaneousSolarRadiation OBJECT-TYPE
SYNTAX      INTEGER (-2048..2049)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The instantaneous ultraviolet, visible, and near-
infrared (wavelength of less than 3.0 micrometers) radiation hitting the
earth's surface in watts per square meter.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.instantaneousSolarRadiation:quantity
<Valid Value Rule>
The value of 2049 shall indicate a missing value.
```

```
<Data Concept Type>Data Element
<Unit>watts per square meter"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 14 018"
::=   { essBufrRadiation 18 }
```

5.9.6 Total Radiation

```
essTotalRadiation OBJECT-TYPE
SYNTAX      INTEGER (-2048..2049)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The average total radiation hitting the earth's
surface in watts per square meter during the radiation period.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.totalRadiation:quantity
<Valid Value Rule>
The value of 2049 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>Joules per square meter"
REFERENCE   "WMO Code Form FM 94 BUFR Table B item 0 14 025"
::=   { essBufrRadiation 25 }
```

5.9.7 Total Radiation Period

```
essTotalRadiationPeriod OBJECT-TYPE
SYNTAX      INTEGER (0..86400)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The period, in seconds, that corresponds to the
length of time the essTotalRadiation is averaged.
<SetConstraint>read-only
<DescriptiveName>RadiationSensor.totalRadiationPeriod:quantity
<Data Concept Type>Data Element
<Unit>seconds"
::=   { essNtcipRadiation 2 }
```

5.10 VISIBILITY DATA OBJECTS

```
-- Contains objects used to describe the visibility data that is
-- collected by the ess.
essNtcipVisibility  OBJECT IDENTIFIER ::= {essNtcip 8 }
```

5.10.1 Visibility

```
essVisibility OBJECT-TYPE
SYNTAX      INTEGER (0..1000001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Surface visibility measured in one tenth of a meter.
<SetConstraint>read-only
<DescriptiveName>VisibilitySensor.visibility:quantity
<Valid Value Rule>
The value 1000001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>one tenth of a meter"
REFERENCE   "The value for WMO Code Form FM 94 BUFR Table B item 0 20 001 is
given by this value divided by 100."
::=   { essNtcipVisibility 1 }
```


5.10.2 Visibility Situation

```
essVisibilitySituation OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        unknown (2),
                        clear (3),
                        fogNotPatchy (4),
                        patchyFog (5),
                        blowingSnow (6),
                        smoke (7),
                        seaSpray (8),
                        vehicleSpray (9),
                        blowingDustOrSand (10),
                        sunGlare (11),
                        swarmsOfInsects (12)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Describes the travel environment in terms of
visibility. If one exists, the corresponding BUFR value is indicated for
staffed (BUFRs) and automated (BUFRa) stations. The indicated value can be
found in the BUFR Table referenced below.
<SetConstraint>read-only
<DescriptiveName>VisibilitySensor.visibilitySituation:code
<Valid Value Rule>
Range       BUFRs       BUFRa       Meaning
      1                                   other visibility anomaly
      2                                   unknown
      3           0           100         clear
      4           44          130         Fog - not patchy
      5           41          131         Patchy fog
      6           36          127         Blowing snow
      7           04          104         Smoke
      8           07          207         Sea Spray
      9                                   Vehicle Spray
      10          31          127         Blowing dust or sand
      11                                  sun glare
      12                                  Swarms of insects
<Data Concept Type>Data Element"
REFERENCE   "The values identified in the above table for BUFRa and BUFRs can
be found in WMO Code Form FM 94 BUFR Table B item 0 20 003."
::=   { essNtcipVisibility 3 }
```

5.11 PAVEMENT SENSOR OBJECTS

```
-- Contains objects used to describe the data that is collected by the
-- pavement surface sensor.
essNtcipPavement  OBJECT IDENTIFIER ::= {essNtcip 9}
```

5.11.1 Number of Pavement Sensors

```
numEssPavementSensors OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of entries in the pavement
sensor table.
```

```
<Informative>This value may automatically change upon connecting or
disconnecting a sensor; however, the table is still defined as a static table
since the creation/deletion of rows is not managed through SNMP logic.
<SetConstraint>read-only
<DescriptiveName>PavementSensorTable.numSensors:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipPavement 1 }
```

5.11.2 Pavement Sensor Table

```
essPavementSensorTable OBJECT-TYPE
SYNTAX      SEQUENCE OF EssPavementSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Table containing the pavement sensor data.
<DescriptiveName>PavementSensorTable
<Data Concept Type>Class
<TableType> static"
::=   { essNtcipPavement 2 }
```

5.11.3 Pavement Sensor

```
essPavementSensorEntry OBJECT-TYPE
SYNTAX      EssPavementSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>A pavement sensor is a sensor that reports the
temperature and moisture condition of the roadway pavement. It can be
described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>PavementSensor
<Data Concept Type>Class"
INDEX { essPavementSensorIndex }
::=   { essPavementSensorTable 1 }

EssPavementSensorEntry ::= SEQUENCE {
      essPavementSensorIndex              INTEGER,
      essPavementSensorLocation           DisplayString,
      essPavementType                     INTEGER,
      essPavementElevation                INTEGER,
      essPavementExposure                 INTEGER,
      essPavementSensorType               INTEGER,
      essSurfaceStatus                    INTEGER,
      essSurfaceTemperature               INTEGER,
      essPavementTemperature              INTEGER,
      essSurfaceWaterDepth                INTEGER,
      essSurfaceSalinity                  INTEGER,
      essSurfaceConductivity              INTEGER,
      essSurfaceFreezePoint               INTEGER,
      essSurfaceBlackIceSignal            INTEGER,
      essPavementSensorError              INTEGER,
      essSurfaceIceOrWaterDepth           INTEGER,
      essSurfaceConductivityV2            INTEGER,
      pavementSensorModelInformation      INTEGER,
      pavementSensorTemperatureDepth      INTEGER}
```


5.11.3.1 Pavement Sensor Index

```
essPavementSensorIndex OBJECT-TYPE
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Enumerated list of row entries that will provide
surface sensor data.
<SetConstraint>index
<DescriptiveName>PavementSensor.index:identifier
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 1 }
```

5.11.3.2 Pavement Sensor Location

```
essPavementSensorLocation OBJECT-TYPE
SYNTAX      DisplayString (SIZE (0..255))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>A textual string indicating the location of the
pavement sensor.
<SetConstraint>always
<DescriptiveName>PavementSensor.location:text
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 2 }
```

5.11.3.3 Pavement Type

```
essPavementType OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        unknown (2),
                        asphalt (3),
                        openGradedAsphalt (4),
                        concrete (5),
                        steelBridge (6),
                        concreteBridge (7),
                        asphaltOverlayBridge (8),
                        timberBridge (9)}
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the type of pavement on the roadway.
<SetConstraint>always
<DescriptiveName>PavementSensor.type:code
<Valid Value Rule>
other                 a different type of bridge deck
unknown               the data was never recorded in the system
asphalt               asphalt pavement on ground
concrete              concrete pavement on ground
steelBridgeconcrete   a concrete driving surface on a steel girder bridge
steelBridgeAsphalt    an asphalt driving surface on a steel girder bridge
steelBridge           a steel lattice driving surface on the bridge
concreteBridge        a concrete driving surface on a concrete bridge
concreteBridgeAsphalt an asphalt overlay driving surface on a concrete bridge
timberBridge          a wooden deck driving surface on the bridge
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 3 }
```

5.11.3.4 Pavement Elevation

```
essPavementElevation OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The elevation of the street surface in meters with
respect to the essReferenceHeight.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.elevation:quantity
<Valid Value Rule>
The value 1001 shall indicate a missing value.
<Data Concept Type>Data Element
<Unit>meters"
::= { essPavementSensorEntry 4 }
```

5.11.3.5 Pavement Exposure

```
essPavementExposure OBJECT-TYPE
SYNTAX      INTEGER (0..101)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates a very rough percentage of the solar
energy which will directly hit the sensor.
<SetConstraint>always
<DescriptiveName>PavementSensor.exposure:quantity
<Valid Value Rule>
A value of 100 indicates a fully visible sky. A value of 101 shall indicate a
missing value.
<Data Concept Type>Data Element
<Unit>percent exposure"
::= { essPavementSensorEntry 5 }
```

5.11.3.6 Pavement Sensor Type

```
essPavementSensorType OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        contactPassive (2),
                        contactActive (3),
                        infrared (4),
                        radar (5),
                        vibrating (6),
                        microwave (7),
                        laser (8)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A value indicating the type of pavement sensor.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.sensorType:code
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 6 }
```

5.11.3.7 Surface Status

```
essSurfaceStatus OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        error (2),
                        dry (3),
```



```
                        traceMoisture (4),
                        wet (5),
                        chemicallyWet (6),
                        iceWarning (7),
                        iceWatch (8),
                        snowWarning (9),
                        snowWatch (10),
                        absorption (11),
                        dew (12),
                        frost (13),
                        absorptionAtDewpoint (14)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A value indicating the pavement surface status.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceStatus:code
<Valid Value Rule>
other - The value reported by the sensor is not defined by the standard. See
the manufacturer's documentation for more information.
noReport - The sensor is not providing any reading for surface status and may
not be responding
errorReport - The sensor is providing a reading for surface status, but
either the reading indicates an error code or the data has been deemed
invalid or suspect
dry - The sensor does not detect any moisture or unusual conditions.
trace - The sensor detects some moisture, but it is suspected to be isolated
absorption - A salt chemical is present that is not fully dissolved in water.
As a result, the conductivity readings will result in erroneous calculations
for amount of chemical in the mix.
wet - The sensor detects a significant amount of moisture indicating a wet
roadway.
chemically wet - The sensor detects a signaificant amount of moisture mixed
with a de-icing or anti-icing chemical
dew - The sensor detects moisture that is suspected to be from the formation
of dew
frost - The sensor detects the formation of frost
freezeAdvisory - The risk of the formation of some sort of frozen moisture on
the roadway is elevated, but its occurrence, location, and/or timing is still
uncertain
slushAdvisory - The risk of the accumulation of snow or slush on the roadway
is elevated, but its occurrence, location, and/or timing is still uncertain
iceAdvisory - The risk of the formation of ice or black ice on the roadway is
elevated, but its occurrence, location, and/or timing is still uncertain
freezeHazard - The sensor detects some sort of frozen moisture but is unable
to classify as slush or ice.
slush - The sensor detects snow or slush.
ice - The sensor detects ice or black ice. (See essSurfaceBlackIceSignal)
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 7 }
```

5.11.3.8 Surface Temperature

```
essSurfaceTemperature OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
```

```
DESCRIPTION "<Definition>The current pavement surface temperature in tenths
of degrees Celsius.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceTemperature:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
::= { essPavementSensorEntry 8 }
```

5.11.3.9 Pavement Temperature

```
essPavementTemperature OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The current pavement temperature 2-10 cm below the
pavement surface in tenths of degrees Celsius. The specific depth at which
the reading is taken is defined by pavementSensorTemperatureDepth.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.pavementTemperature:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
::= { essPavementSensorEntry 9 }
```

5.11.3.10 Surface Water Depth

```
-- This object has been deprecated. See Annex D.1.6 for more information.
essSurfaceWaterDepth OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>The current depth of water on the surface of the
roadway measured in millimeters. The value 255 shall indicate an error
condition or missing value.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceWaterDepth:quantity
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 10 }
```

5.11.3.11 Surface Salinity

```
essSurfaceSalinity OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The pavement salinity in parts per one hundred
thousand by weight (i.e., grams of solute per 100,000 grams of solution).
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceSalinity:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per one hundred thousand by weight"
::= { essPavementSensorEntry 11 }
```


5.11.3.12 Surface Conductivity

```
-- This object has been deprecated. See Annex D.1.7 for more information.
essSurfaceConductivity OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>Indicates the conductance of the ice/liquid mixture
on the pavement as detected by the sensor, in mhos, which is the inverse of
ohms. The value  65535 shall indicate an error condition or missing value.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceConductivity:quantity
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 12 }
```

5.11.3.13 Surface Freezing Point

```
essSurfaceFreezePoint OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The temperature in tenths of degrees Celsius at
which the existing solution on the roadway will freeze.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceFreezingPoint:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
::= { essPavementSensorEntry 13 }
```

5.11.3.14 Surface Black Ice Signal

```
essSurfaceBlackIceSignal OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        noIce (2),
                        blackIce (3),
                        detectorError (4)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A value indicating if Black Ice is detected by the
sensor.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceBlackIce:code
<Valid Value Rule>
other - The sensor is reporting a value that is not defined by the standard.
See the manufacturer's documentation for more informaiton.
noIce - The sensor is not currently detecting black ice.
blackIce - The sensor is currently detecting black ice.
detectorError - The sensor is not connected, is not reporting, or is
reporting an error.
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 14 }
```

5.11.3.15 Pavement Sensor Error

```
essPavementSensorError OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
```



```
                        none (2),
                        noResponse (3),
                        cutCable (4),
                        shortCircuit (5),
                        dirtyLens (6)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A value indicating the type of pavement sensor
error.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.pavementSensorError:code
<Valid Value Rule>
other - An error has been detected that is not defined by the standard; see
the manufacturer's documentation for more information.
none - No error is detected, the sensor appears to be working properly
noResponse - The sensor is configured and is believed to be connected, but is
not responding
cutCable - The sensor is not configured, not present or not fully connected,
perhaps because the cable was cut
shortCircuit - The sensor input has detected a short-circuit.
dirtyLens - The lens of the sensor appears to be dirty.
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 15 }
```

5.11.3.16 Surface Water Depth—Version 2

```
essSurfaceIceOrWaterDepth OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The current thickness of ice or depth of water on
the surface of the roadway measured in 1/10th of millimeters.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceIceOrWaterDepth:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>1/10th of millimeters"
::= { essPavementSensorEntry 16 }
```

5.11.3.17 Surface Conductivity—Version 2

```
essSurfaceConductivityV2 OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the conductivity of the ice/liquid mixture
on the pavement as detected by the sensor, in 1/10ths of milli-mhos/cm (mhos
is the inverse of ohms). This value is independent of the size or shape of
the sensor and can be directly translated into a percent concentration of
chemical (e.g. salinity) through look-up tables for a given chemical.
<SetConstraint>read-only
<DescriptiveName>PavementSensor.surfaceConductivityV2:quantity
<Valid Value Rule>
The value  65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>1/10ths of milli-mhos/cm"
```



```
::= { essPavementSensorEntry 17 }
```

5.11.3.18 Pavement Sensor Model Information

```
pavementSensorModelInformation OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the row in the Module Table (See NTCIP
1201) that contains information about the make, model, and version number of
the sensor associated with this row of the Pavement Sensor Table.
<DescriptiveName>PavementSensor.modelInformation:identifier
<Valid Value Rule>
The value of zero indicates that this information is not available.
<Data Concept Type>Data Element"
::= { essPavementSensorEntry 18 }
```

5.11.3.19 Pavement Temperature Depth

```
pavementSensorTemperatureDepth OBJECT-TYPE
SYNTAX      INTEGER (2..11)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The depth at which the pavement temperature is
detected.
<DescriptiveName>PavementSensor.temperatureDepth:quantity
<Valid Value Rule>
The value of 11 indicates that the information is not available.
<Data Concept Type>Data Element
<Unit>centimeters"
::= { essPavementSensorEntry 19 }
```

5.11.4 Number of Subsurface Sensors

```
numEssSubSurfaceSensors OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of entries in the Subsurface
Sensor Table.
<Informative>This value may automatically change upon connecting or
disconnecting a sensor; however, the table is still defined as a static table
since the creation/deletion of rows is not managed through SNMP logic.
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensorTable.numSensors:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipPavement 3 }
```

5.11.5 Subsurface Sensor Table

```
essSubSurfaceSensorTable OBJECT-TYPE
SYNTAX      SEQUENCE OF EssSubSurfaceSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Table containing the subsurface sensor data.
<DescriptiveName>SubSurfaceSensorTable
<Data Concept Type>Class
```

```
<TableType> static"
::=   { essNtcipPavement 4 }
```

5.11.6 Subsurface Sensor

```
essSubSurfaceSensorEntry OBJECT-TYPE
SYNTAX      EssSubSurfaceSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>A subsurface sensor is a sensor that reports the
temperature and moisture condition of the roadway subsurface.  It can be
described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>SubSurfaceSensor
<Data Concept Type>Class"
INDEX { essSubSurfaceSensorIndex }
::=   { essSubSurfaceSensorTable 1 }

EssSubSurfaceSensorEntry ::= SEQUENCE {
      essSubSurfaceSensorIndex                        INTEGER,
      essSubSurfaceSensorLocation                     DisplayString,
      essSubSurfaceType                               INTEGER,
      essSubSurfaceSensorDepth                        INTEGER,
      essSubSurfaceTemperature                        INTEGER,
      essSubSurfaceMoisture                           INTEGER,
      essSubSurfaceSensorError                        INTEGER,
      essSubSurfaceMoistureOrError                    INTEGER}
```

5.11.6.1 Subsurface Sensor Index

```
essSubSurfaceSensorIndex OBJECT-TYPE
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Enumerated list of row entries that will provide
surface sensor data.
<SetConstraint>index
<DescriptiveName>SubSurfaceSensor.index:identifier
<Data Concept Type>Data Element"
::= { essSubSurfaceSensorEntry 1 }
```

5.11.6.2 Subsurface Sensor Location

```
essSubSurfaceSensorLocation OBJECT-TYPE
SYNTAX      DisplayString (SIZE (0..255))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>A textual string indicating the location of the
subsurface sensor.
<SetConstraint>always
<DescriptiveName>SubSurfaceSensor.location:text
<Data Concept Type>Data Element"
::= { essSubSurfaceSensorEntry 2 }
```

5.11.6.3 Subsurface Type

```
essSubSurfaceType OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
```



```
                        unknown (2),
                        concrete (3),
                        asphalt (4),
                        openGradedAsphalt (5),
                        gravel (6),
                        clay (7),
                        loam (8),
                        sand (9),
                        permafrost (10),
                        variousAggregates (11),
                        air (12)}
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the type of subsurface. A value of air
would indicate a bridge.
<SetConstraint>always
<DescriptiveName>SubSurfaceSensor.type:code
<Data Concept Type>Data Element"
::= { essSubSurfaceSensorEntry 3 }
```

5.11.6.4 Subsurface Sensor Depth

```
essSubSurfaceSensorDepth OBJECT-TYPE
SYNTAX      INTEGER (0..1001)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Depth of subsurface sensor in centimeters below the
pavement surface.
<SetConstraint>always
<DescriptiveName>SubSurfaceSensor.depth:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>centimeters"
::= { essSubSurfaceSensorEntry 4 }
```

5.11.6.5 Subsurface Temperature

```
essSubSurfaceTemperature OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The current subsurface temperature in tenths of
degrees Celsius.
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensor.temperature:quantity
<Valid Value Rule>
The value 1001 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>tenths of degrees Celsius"
::= { essSubSurfaceSensorEntry 5 }
```

5.11.6.6 Subsurface Moisture

```
essSubSurfaceMoisture OBJECT-TYPE
SYNTAX      INTEGER (0..101)
ACCESS      read-only
STATUS      mandatory
```

```
DESCRIPTION "<Definition>The subsurface moisture expressed as a percentage
(eg. 0 indicates dry, 100 indicates saturated).
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensor.moisture:quantity
<Valid Value Rule>
The value 101 indicates an error condition or missing value.
<Data Concept Type>Data Element
<Unit>percentage"
::= { essSubSurfaceSensorEntry 7 }
```

5.11.6.7 Subsurface Sensor Error

```
essSubSurfaceSensorError OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        none (2),
                        noResponse (3),
                        cutCable (4),
                        shortCircuit (5)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A value indicating the type of sensor error.
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensor.error:code
<Valid Value Rule>
other - An error has been detected that is not defined by the standard; see
the manufacturer's documentation for more information.
none - No error is detected, the sensor appears to be working properly
noResponse - The sensor is configured and is believed to be connected, but is
not responding
cutCable - The sensor is not configured, not present or not fully connected,
perhaps because the cable was cut
shortCircuit - The sensor input has detected a short-circuit.
<Data Concept Type>Data Element"
::= { essSubSurfaceSensorEntry 8 }
```

5.11.7 Pavement Block

```
-- This object has been deprecated. See Annex D.2.8 for more
-- information.
essPavementBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      deprecated
DESCRIPTION "<Definition>An OER encoded string of the EssPavementData
structure as defined below. This object is used for uploading current
pavement data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssPavementData ::= SEQUENCE OF PavementSensorData
      -- for (
--    x = 1;
--    x < numEssPavementSensors.0;
--    x++)
```



```
PavementSensorData ::= SEQUENCE {
      essPavementSensorIndex.x          OPTIONAL, -- @NTCIP1204-v03
      essSurfaceStatusV2.x              OPTIONAL, -- @NTCIP1204-v03
      essSurfaceTemperature.x           OPTIONAL, -- @NTCIP1204-v03
      essPavementTemperature.x          OPTIONAL, -- @NTCIP1204-v03
      essSurfaceWaterDepth.x            OPTIONAL, -- @NTCIP1204-v03
      essSurfaceSalinity.x              OPTIONAL, -- @NTCIP1204-v03
      essSurfaceConductivity.x          OPTIONAL, -- @NTCIP1204-v03
      essSurfaceFreezePoint.x           OPTIONAL, -- @NTCIP1204-v03
      essSurfaceBlackIceSignal.x        OPTIONAL, -- @NTCIP1204-v03
      essPavementSensorError.x          OPTIONAL  -- @NTCIP1204-v03
}
<Informative>This object has been replaced by essPavementV3Block (See
5.11.9).
<SetConstraint>read-only
<DescriptiveName>PavementSensorTable.pavementBlock:code
<Data Concept Type>Data Element"
::=   { essNtcipPavement 5 }
```

5.11.8 Subsurface Block Object

```
essSubSurfaceBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssSubsurfaceData
structure as defined below. This object is used for uploading current
subsurface data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssSubSurfaceData ::= SEQUENCE OF SubSurfaceSensorData
      -- for (
--    x = 1;
--    x < numEssSubSurfaceSensors.0;
--    x++)

SubSurfaceSensorData ::= SEQUENCE {
      essSubSurfaceSensorIndex.x        OPTIONAL, -- @NTCIP1204-v03
      essSubSurfaceTemperature.x        OPTIONAL, -- @NTCIP1204-v03
      essSubSurfaceMoisture.x           OPTIONAL, -- @NTCIP1204-v03
      essSubSurfaceSensorError.x        OPTIONAL  -- @NTCIP1204-v03
      }
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensorTable.subsurfaceBlock:code
<Data Concept Type>Data Element"
::=   { essNtcipPavement 6 }
```

5.11.9 Pavement Block V3

```
essPavementV3Block OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
```

```
DESCRIPTION "<Definition>An OER encoded string of the EssPavementDataV3
structure as defined below. This object is used for uploading current
pavement data from the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

EssPavementDataV3 ::= SEQUENCE OF PavementSensorDataV3
      -- for (
--    x = 1;
--    x < numEssPavementSensors.0;
--    x++)


PavementSensorDataV3 ::= SEQUENCE {
      essPavementSensorIndex.x            OPTIONAL, -- @NTCIP1204-v03
      essSurfaceStatus.x                  OPTIONAL, -- @NTCIP1204-v03
      essSurfaceTemperature.x             OPTIONAL, -- @NTCIP1204-v03
      essPavementTemperature.x            OPTIONAL, -- @NTCIP1204-v03
      essSurfaceSalinity.x                OPTIONAL, -- @NTCIP1204-v03
      essSurfaceFreezePoint.x             OPTIONAL, -- @NTCIP1204-v03
      essSurfaceBlackIceSignal.x          OPTIONAL, -- @NTCIP1204-v03
      essPavementSensorError.x            OPTIONAL, -- @NTCIP1204-v03
      essSurfaceIceOrWaterDepth.x         OPTIONAL, -- @NTCIP1204-v03
      essSurfaceConductivityV2.x          OPTIONAL  -- @NTCIP1204-v03
}
<SetConstraint>read-only
<DescriptiveName>PavementSensorTable.pavementBlock:code
<Data Concept Type>Data Element"
::=   { essNtcipPavement 7 }
```

5.12 MOBILE PLATFORM OBJECTS

```
-- Contains objects related to monitoring mobile platforms that act as
-- ESS (e.g., specially-equipped maintenance vehicles).
-- There has been limited use of mobile ESS platforms within the
-- surface transportation industry and as such these objects
-- should be considered experimental.
essNtcipMobile  OBJECT IDENTIFIER ::= {essNtcip 10}
```

5.12.1 Detected Friction

```
essMobileFriction OBJECT-TYPE
SYNTAX      INTEGER (0..101)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates measured coefficient of friction in
percent.
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.friction:quantity
<Valid Value Rule>
The value 101 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>percent friction"
::=   { essNtcipMobile 1 }
```


5.12.2 Observed Ground State

```
essMobileObservationGroundState OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        dry (2),
                        moist (3),
                        wet (4),
                        flooded (5),
                        frozen (6),
                        glaze (7),
                        dustySandy (8),
                        veryDry (9),
                        icy (10),
                        patchyWetSnow (11),
                        moderateWetSnowCover (12),
                        fullWetSnowCover (13),
                        patchyDrySnow (14),
                        moderateDrySnowCover (15),
                        fullDrySnowCover (16),
                        driftingSnow (17),
                        unknown (18)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The prevailing observed ground state of the
surrounding environment as determined by the observer. This is an indicator
of past weather conditions.
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.observedGroundState:code
<Data Concept Type>Data Element"
::=   { essNtcipMobile 2 }
```

5.12.3 Observed Pavement State

```
essMobileObservationPavement OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        dry (2),
                        wet (3),
                        puddles (4),
                        shallowStandingWater (5),
                        shallowFlowingWater (6),
                        deepStandingWater (7),
                        deepFlowingWater (8),
                        dustingFreshSnow (9),
                        moderateFreshSnow (10),
                        deepFreshSnow (11),
                        plowedSnow (12),
                        slush (13),
                        packedSnowPatches (14),
                        packedSnow (15),
                        lightSnowDrifts (16),
                        moderateSnowDrifts (17),
                        heavySnowDrifts (18),
                        frost (19),
                        icePatches (20),
                        moderatelyIcy (21),
                        heavyIcing (22),
                        blackIce (23),
```

```
                        sheetIce (24),
                        frozenSlush (25)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The prevailing observed conditions on the driving
surface as determined by the observer.
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.observedPavementState:code
<Data Concept Type>Data Element"
::=   { essNtcipMobile 3 }
```

5.13 PAVEMENT TREATMENT OBJECTS

```
-- Contains objects that monitor the various types and amounts of
-- treatments that are spread on the pavement surface.
essNtcipTreatment OBJECT IDENTIFIER ::= { essNtcip 11 }
```

5.13.1 Number of Treatments

```
numEssTreatments OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of entries in the Pavement
Treatment Table.
<SetConstraint>read-only
<DescriptiveName>PavementTreatmentTable.numTreatments:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipTreatment 1 }
```

5.13.2 Pavement Treatment Table

```
essPavementTreatmentTable OBJECT-TYPE
SYNTAX      SEQUENCE OF EssPavementTreatmentEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>Table containing the pavement treatment data.
<DescriptiveName>PavementTreatmentTable
<Data Concept Type>Class
<TableType> static"
::=   { essNtcipTreatment 2 }
```

5.13.3 Pavement Treatment

```
essPavementTreatmentEntry OBJECT-TYPE
SYNTAX      EssPavementTreatmentEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>A pavement treatment is a chemical that can be
applied to the roadway to de-ice or prevent icing of the pavement. It can be
described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>PavementTreatment
<Data Concept Type>Class"
INDEX { essPavementTreatmentIndex }
::=   { essPavementTreatmentTable 1 }
```



```
EssPavementTreatmentEntry ::= SEQUENCE {
      essPavementTreatmentIndex               INTEGER,
      essPaveTreatProductType                 INTEGER,
      essPaveTreatProductForm                 INTEGER,
      essPercentProductMix                    INTEGER}
```

5.13.3.1 Pavement Treatment Index

```
essPavementTreatmentIndex OBJECT-TYPE
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Enumerated list of row entries that will provide
pavement treatment data.
<SetConstraint>index
<DescriptiveName>PavementTreatement.index:identifier
<Data Concept Type>Data Element"
::= { essPavementTreatmentEntry 1 }
```

5.13.3.2 Pavement Treatment Product Type

```
essPaveTreatProductType OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        sand (2),
                        dirt (3),
                        gravel (4),
                        cinders (5),
                        water (6),
                        enhancedSalts (7),
                        naCl (8),
                        caCl (9),
                        mgCl (10),
                        cMA (11),
                        kAC (12),
                        naFormate (13),
                        naA (14)}
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the type of treatment being applied to the
road. An enhanced definition of some of the values are as follows: other -
any other type of treatment water - used as a diluting agent cMA - Calcium-
Magnesium Acetate kAC - Potassium-Magnesium Acetate naFormate - Sodium
Formate naA - Sodium Acetate
<Informative>Version 02 of this standard incorrectly defined the set
constraint of this read-write object to be read-only; it should be always,
however, the intent is that it would only be set by a local connection when
the new product is being loaded and that remote systems would only read this
object.
<SetConstraint>always
<DescriptiveName>PavementTreatement.type:code
<Valid Value Rule>
An enhanced definition of some of the values are as follows.
other - any other type of treatment
water - used as a diluting agent
cMA - Calcium-Magnesium Acetate
kAC - Potassium-Magnesium Acetate
naFormate - Sodium Formate
```

```
naA - Sodium Acetate
<Data Concept Type>Data Element"
::= { essPavementTreatmentEntry 2 }
```

5.13.3.3 Treatment Product Form

```
essPaveTreatProductForm OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        dry (2),
                        prewet (3),
                        liquid (4)}
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the condition of the treatment being
applied to the road.
<SetConstraint>read-only
<DescriptiveName>PavementTreatement.form:code
<Data Concept Type>Data Element"
::= { essPavementTreatmentEntry 3 }
```

5.13.3.4 Percentage of Treatment Type in Mix

```
essPercentProductMix OBJECT-TYPE
SYNTAX      INTEGER (0..100)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the percentage of the total application
mix by weight that is of the type specified in essPaveTreatProductType.
<SetConstraint>read-only
<DescriptiveName>PavementTreatement.mix:quantity
<Valid Value Rule>
The sum of these percentages within the total mixture shall equal 100.
<Data Concept Type>Data Element
<Unit>percent"
::= { essPavementTreatmentEntry 4 }
```

5.13.4 Treatment Amount

```
essPaveTreatmentAmount OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates quantity of the treatment being applied in
kilograms per lane kilometer.
<SetConstraint>read-only
<DescriptiveName>PavementTreatmentTable.amount:quantity
<Valid Value Rule>
The value of 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>kilograms per lane kilometer"
::=   { essNtcipTreatment 3 }
```

5.13.5 Treatment Width

```
essPaveTreatmentWidth OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
```



```
DESCRIPTION "<Definition>Indicates the width of the spread of treatment in
meters.
<SetConstraint>read-only
<DescriptiveName>PavementTreatmentTable.width:quantity
<Valid Value Rule>
The value of 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>meters"
::=   { essNtcipTreatment 4 }
```

5.13.6 Pavement Treatment Block

```
pavementTreatmentBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the Pavement Treatment
data. This object is used for uploading current pavement treatment data from
the ESS in a bandwidth efficient manner.

The OPTIONAL fields shall be present if the data is supported by the
implementation and is valid. The OPTIONAL fields shall be omitted for any
data that is invalid or not supported by the implementation.

NOTE: The following structure contains pts data elements in addition to the
ess data elements. Although these are not currently required to fulfill any
existing requirements, they are included to maintain backward compatibility
with the version 02 standard.

PavementTreatmentBlock ::= SEQUENCE {
      treatmentInfo     SEQUENCE OF PavementTreatmentData OPTIONAL,
      essPaveTreatmentAmount.0      OPTIONAL, -- @NTCIP1204-v03
      essPaveTreatmentWidth.0       OPTIONAL, -- @NTCIP1204-v03
      ptsOperationalMode.0          OPTIONAL, -- @NTCIP1204-v03
      ptsCommandState.0             OPTIONAL, -- @NTCIP1204-v03
      ptsSprayerState.0             OPTIONAL, -- @NTCIP1204-v03
      ptsSignalDuration.0           OPTIONAL, -- @NTCIP1204-v03
      ptsSignalEventCount.0         OPTIONAL, -- @NTCIP1204-v03
      ptsLastSignalEvent.0          OPTIONAL, -- @NTCIP1204-v03
      ptsActiveEventCount.0         OPTIONAL, -- @NTCIP1204-v03
      ptsInactiveEventCount.0       OPTIONAL, -- @NTCIP1204-v03
      ptsLastactiveEvent.0          OPTIONAL, -- @NTCIP1204-v03
      ptsLastInactiveEvent.0        OPTIONAL, -- @NTCIP1204-v03
      ptsError.0                    OPTIONAL, -- @NTCIP1204-v03
      ptsMonitoringDetectors.0      OPTIONAL  -- @NTCIP1204-v03
}

PavementTreatmentData ::=
      -- for (
      --    x = 1;
      --    x < numEssTreatments.0;
      --    x++)
SEQUENCE {
      essPavementTreatmentIndex.x         OPTIONAL, -- @NTCIP1204-v03
      essPaveTreatProductType.x           OPTIONAL, -- @NTCIP1204-v03
      essPaveTreatProductForm.x           OPTIONAL, -- @NTCIP1204-v03
```

```
      essPercentProductMix.x              OPTIONAL  -- @NTCIP1204-v03
}
<DescriptiveName>PavementTreatmentTable.block:frame
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 5 }
```

5.13.7 Operational Mode

```
ptsOperationalMode OBJECT-TYPE
SYNTAX       INTEGER {   off (1),
                         manual (2),
                         automatic (3)}
ACCESS       read-write
STATUS       obsolete
DESCRIPTION "<Definition>Indicates the operational mode of the Pavement
Treatment System.

When in the 'off' state, the PTS shall not trigger the sprayer even if
commanded to do so and shall always be inactive. The PTS shall transition to
the requested operational mode, upon request.

When in the 'automatic' state, the PTS shall monitor conditions and trigger
the sprayer based on a manufacturer specific algorithm. The algorithm shall
only consider input from the detectors selected in the ptsMonitoringDetectors
object. The PTS shall also trigger the sprayer if commanded to do so via the
ptsCommandState object. The PTS shall transition to the requested operational
mode, upon request.

When in the 'manual' state, the PTS shall trigger the sprayer if commanded to
do so via the ptsCommandState object. The PTS shall transition to the
requested operational mode, upon request.
<SetConstraint>always
<DescriptiveName>PTS.operationalMode.code
<Valid Value Rule>
off   When set to this value the ESS will not trigger the
      bridge sprayer

manual      When set to this value the ESS will only trigger the
      bridge sprayer when manually commanded to do
      so (e.g., see bridgeSprayerMgmtSignalState).

automatic   When set to this value the ESS will trigger the
      bridge sprayer when manually commanded to do
      so or when the internal algorithm determines
      that the sprayer should be triggered.
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 6 }
```

5.13.8 Command State

```
ptsCommandState OBJECT-TYPE
SYNTAX       INTEGER {   other (1),
                         inactive (2),
                         activate (3)}
ACCESS       read-write
STATUS       obsolete
```



```
DESCRIPTION "<Definition>Indicates the operational state of the PTS. When in
the 'inactive' state, the PTS shall not be spraying. Upon entering the
'active' state, either by a manual SET of this object or through an automated
algorithm, the PTS shall trigger the sprayer and spray the chemical for a
duration as defined by the ptsSignalDuration object. Upon expiration of this
duration, the PTS shall automatically transition back to the 'inactive'
state.
<SetConstraint>always
<DescriptiveName>PTS.commandState:code
<Valid Value Rule>
other -
      read  - indicates a unknown or inital state
      write - no effect
inactive  -
      read  - indicates the ess is not signaling the bridge sprayer
      write - no effect
activate  -
      read  - indicates the ess is signaling the bridge sprayer
      write - causes the ess to signal the bridge sprayer
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 7 }
```

5.13.9 Sprayer State

```
ptsSprayerState OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        inactive (2),
                        active (3)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>other     - indicates a unknown or inital state
             inactive  - indicates the bridge sprayer is inactive
             active    - indicates the bridge sprayer is active
<SetConstraint>read-only
<DescriptiveName>PTS.sprayerState:code
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 8 }
```

5.13.10 Signal Duration

```
ptsSignalDuration OBJECT-TYPE
SYNTAX      INTEGER (0..3600000)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>The number of milliseconds of a simple logic level
or state the bridge sprayer needs to detect a signal from the ESS
<SetConstraint>always
<DescriptiveName>PTS.signalDuration:quantity
<Data Concept Type>Data Element
<Unit>milliseconds"
::=   { essNtcipTreatment 9 }
```

5.13.11 Signal Event Count

```
ptsSignalEventCount OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
```



```
DESCRIPTION "<Definition>The count of the number of Signal Events that have
occurred.
<SetConstraint>read-only
<DescriptiveName>PTS.signalEventCount:quantity
<Data Concept Type>Data Element
<Unit>milliseconds"
::=   { essNtcipTreatment 10 }
```

5.13.12 Last Signal Event

```
ptsLastSignalEvent OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The number of seconds since 00:00:00 Jan 1 1970 UTC.
<SetConstraint>read-only
<DescriptiveName>PTS.lastSignalEvent:quantity
<Valid Value Rule>
The value of 0 indicates an unknown or initial value.
<Data Concept Type>Data Element
<Unit>seconds"
::=   { essNtcipTreatment 11 }
```

5.13.13 Active Event Count

```
ptsActiveEventCount OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The count of the number of Active Events that have
occurred.
<SetConstraint>read-only
<DescriptiveName>PTS.activeEventCount:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipTreatment 12 }
```

5.13.14 Inactive Event Count

```
ptsInactiveEventCount OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>A count of the number of Inactive Events that have
occurred.
<SetConstraint>read-only
<DescriptiveName>PTS.inactiveEventCount:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipTreatment 13 }
```

5.13.15 Last Active Event

```
ptsLastActiveEvent OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The number of seconds since 00:00:00 Jan 1 1970 UTC.
```



```
<SetConstraint>read-only
<DescriptiveName>PTS.lastActiveEvent:quantity
<Valid Value Rule>
The value of 0 indicates an unknown or initial value.
<Data Concept Type>Data Element
<Unit>seconds"
::=   { essNtcipTreatment 14 }
```

5.13.16 Last Inactive Event

```
ptsLastInactiveEvent OBJECT-TYPE
SYNTAX      Counter
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The number of seconds since 00:00:00 Jan 1 1970 UTC.
<SetConstraint>read-only
<DescriptiveName>PTS.lastInactiveEvent:quantity
<Valid Value Rule>
The value of 0 indicates an unknown or initial value.
<Data Concept Type>Data Element
<Unit>seconds"
::=   { essNtcipTreatment 15 }
```

5.13.17 PTS Error Code

```
ptsError OBJECT-TYPE
SYNTAX      INTEGER { other (1),
                      ok (2),
                      genericError (3),
                      tankLow (4)}
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the status of the bridge sprayer.
<SetConstraint>read-only
<DescriptiveName>PTS.error:code
<Valid Value Rule>
other        - indicates a unknown or initial state
ok           - indicates the bridge sprayer is operational
genericError - indicates the bridge sprayer has an error
tankLow      - indicates the bridge sprayer's tank is low
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 16 }
```

5.13.18 Monitoring Detectors

```
ptsMonitoringDetectors OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE (4))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the pavement detectors that the PTS shall
use in its algorithm that determines when the PTS will automatically trigger
the sprayer.
<DescriptiveName>PTS.monitoringDetectors:code
<Valid Value Rule>
Each bit indicates whether or not the associated pavement sensor shall be
used within the algorithm. The first (high order) bit in the bit string shall
reference the first pavement sensor. A value of one for any bit shall
```

```
indicate that the sensor input shall be considered, and a value of zero shall
mean that the input shall not be considered.
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 17 }
```

5.13.19 Operational Mode—Version 3

```
ptsOperationalModeV3 OBJECT-TYPE
SYNTAX      INTEGER {   off (1),
                        manual (2),
                        automatic (3)}
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the operational mode of the Pavement
Treatment System.

When in the 'off' state, the PTS shall not trigger the sprayer even if
commanded to do so and shall always be inactive. The PTS shall transition to
the requested operational mode, upon request.

When in the 'automatic' state, the PTS shall monitor conditions and trigger
the sprayer based on a manufacturer specific algorithm. The algorithm shall
only consider input from the detectors selected in the ptsMonitoringDetectors
object. The PTS shall also trigger the sprayer if commanded to do so via the
ptsCommandStateV3 object. The PTS shall transition to the requested
operational mode, upon request.

When in the 'manual' state, the PTS shall trigger the sprayer if commanded to
do so via the ptsCommandStateV3 object. The PTS shall transition to the
requested operational mode, upon request.
<SetConstraint>always
<DescriptiveName>PTS.operationalMode.code
<Valid Value Rule>
off   When set to this value the ESS will not trigger the
      bridge sprayer

manual      When set to this value the ESS will only trigger the
      bridge sprayer when manually commanded to do
      so (e.g., see ptsCommandStateV3).

automatic   When set to this value the ESS will trigger the
      bridge sprayer when manually commanded to do
      so or when the internal algorithm determines
      that the sprayer should be triggered.
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 18 }
```

5.13.20 Command State—Version 3

```
ptsCommandStateV3 OBJECT-TYPE
SYNTAX      INTEGER {   other (1),
                        inactive (2),
                        activate (3)}
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the operational state of the PTS. When in
the 'inactive' state, the PTS shall not be signaling the sprayer. Upon
```



```
entering the 'active' state, either by a manual SET of this object or through
an automated algorithm, the PTS shall trigger the sprayer using a signal that
lasts for a duration as defined by the ptsSignalDuration object. Upon
expiration of this duration, the PTS shall automatically transition back to
the 'inactive' state.
<Informative>This object does not directly control the sprayer, rather it
merely signals an external sprayer.
<SetConstraint>always
<DescriptiveName>PTS.commandState:code
<Valid Value Rule>
other -
      read  - indicates a unknown or inital state
      write - no effect
inactive  -
      read  - indicates the ess is not signaling the bridge sprayer
      write - no effect
activate  -
      read  - indicates the ess is signaling the bridge sprayer
      write - causes the ess to signal the bridge sprayer
<Data Concept Type>Data Element"
::=   { essNtcipTreatment 19 }
```

5.14 AIR QUALITY PARAMETERS

```
-- Contains objects used for monitoring air quality conditions.
essNtcipAirQuality  OBJECT IDENTIFIER ::= { essNtcip 12 }
```

5.14.1 Carbon Monoxide Parameter

```
essCO OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of carbon monoxide in the air,
measured in parts per million.
<SetConstraint>read-only
<DescriptiveName>AirQuality.carbonMonoxide:quantity
<Valid Value Rule>
The value 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per million"
::=   { essNtcipAirQuality 1 }
```

5.14.2 Carbon Dioxide Parameter

```
essCO2 OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of carbon dioxide in the air,
measured in parts per billion.
<SetConstraint>read-only
<DescriptiveName>AirQuality.carbonDioxide:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per billion"
::=   { essNtcipAirQuality 2 }
```

5.14.3 Nitrous Oxide Parameter

```
essNO OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of nitrous oxide in the air,
measured in parts per million.
<SetConstraint>read-only
<DescriptiveName>AirQuality.nitrousOxide:quantity
<Valid Value Rule>
The value 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per million"
::=   { essNtcipAirQuality 3 }
```

5.14.4 Nitrogen Dioxide Parameter

```
essNO2 OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of nitrogen dioxide in the air,
measured in parts per billion.
<SetConstraint>read-only
<DescriptiveName>AirQuality.nitrogenDioxide:quantity
<Valid Value Rule>
The value 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per billion"
::=   { essNtcipAirQuality 4 }
```

5.14.5 Sulfer Dioxide Parameter

```
essSO2 OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of sulfur dioxide in the air,
measured in parts per billion.
<SetConstraint>read-only
<DescriptiveName>AirQuality.sulferDioxide:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per billion"
::=   { essNtcipAirQuality 5 }
```

5.14.6 Ozone Parameter

```
essO3 OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of ozone in the air, measured in
parts per one hundred billion.
<SetConstraint>read-only
```



```
<DescriptiveName>AirQuality.ozone:quantity
<Valid Value Rule>
The value 255 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>parts per one hundred billion"
::=   { essNtcipAirQuality 6 }
```

5.14.7 Particulate Matter Parameter

```
essPM10 OBJECT-TYPE
SYNTAX      INTEGER (0..65535)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The concentration of small particulate matter of 10
micrometers or less in diameter in the air, measured in micrograms per cubic
meter.
<SetConstraint>read-only
<DescriptiveName>AirQuality.particulateMatter:quantity
<Valid Value Rule>
The value 65535 shall indicate an error condition or missing value.
<Data Concept Type>Data Element
<Unit>micrograms per cubic meter"
::=   { essNtcipAirQuality 7 }
```

5.14.8 Air Quality Block Object

```
essAirQualityBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssAirQualityData
structure as defined below. This object is used for uploading current air
quality data from the ESS in a bandwidth efficient manner.

A GET shall return data for all of the fields in the structure (even if they
are indicated as OPTIONAL); unless the data values are not supported by the
controller or are invalid (e.g., the sensor is not attached), in which case
the values shall be omitted.

essAirQualityData ::= SEQUENCE {
      essCO.0                        OPTIONAL,   -- @NTCIP1204-v03
      essCO2.0                       OPTIONAL,   -- @NTCIP1204-v03
      essNO.0                        OPTIONAL,   -- @NTCIP1204-v03
      essNO2.0                       OPTIONAL,   -- @NTCIP1204-v03
      essSO2.0                       OPTIONAL,   -- @NTCIP1204-v03
      essO3.0                        OPTIONAL,   -- @NTCIP1204-v03
      essPM10.0                      OPTIONAL    -- @NTCIP1204-v03
      }
<SetConstraint>read-only
<DescriptiveName>AirQuality.airQualityBlock:code
<Data Concept Type>Data Element"
::=   { essNtcipAirQuality 8 }
```

5.15 WATER QUALITY PARAMETERS

```
--This node contains objects used for monitoring water quality
-- conditions. Reserved for future use.
essNtcipWaterQuality OBJECT IDENTIFIER ::= { essNtcip 13 }
```

5.16 SNAPSHOT PARAMETERS

```
-- Contains objects used to describe the snapshot camera feature
essNtcipSnapshot  OBJECT IDENTIFIER ::= { essNtcip 14 }
```

5.16.1 Number of Snapshot Cameras

```
essSnapshotNumberOfCameras OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of cameras that can be utilized
for capturing snapshots on the ESS.
<Informative>This value may automatically change upon connecting or
disconnecting a camera; however, the table is still defined as a static table
since the creation/deletion of rows is not managed through SNMP logic.
<SetConstraint>read-only
<DescriptiveName>SnapshotCameraTable.numCameras:quantity
<Data Concept Type>Data Element"
::=   { essNtcipSnapshot 1 }
```

5.16.2 Snapshot Camera Table

```
essSnapshotCameraTable OBJECT-TYPE
SYNTAX      SEQUENCE OF EssSnapshotCameraEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>The snapshot camera table provides summary
information about the snapshot cameras supported by the ESS.  It can be
described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>SnapshotCameraTable
<Data Concept Type>Class
<TableType> static"
::=   { essNtcipSnapshot 2 }
```

5.16.3 Snapshot Camera

```
essSnapshotCameraEntry OBJECT-TYPE
SYNTAX      EssSnapshotCameraEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>A snapshot camera is a camera that is able to
capture a picture and store it within the device's memory as a file.  It can
be described through a number of attributes as indicated by the following
subclauses.
<DescriptiveName>SnapshotCamera
<Data Concept Type>Class"
INDEX { essSnapshotCameraIndex }
::=   { essSnapshotCameraTable 1 }

EssSnapshotCameraEntry ::= SEQUENCE {
      essSnapshotCameraIndex        INTEGER,
      essSnapshotCameraDescription  DisplayString,
      essSnapshotCameraStoragePath  DisplayString,
      essSnapshotCameraCommand      INTEGER,
      essSnapshotCameraError        INTEGER,
```



```
      essSnapshotCameraFilename     OCTET STRING }
```

5.16.3.1 Snapshot Camera Index

```
essSnapshotCameraIndex OBJECT-TYPE
SYNTAX      INTEGER (1..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the row number of this entry.
<SetConstraint>index
<DescriptiveName>SnapshotCamera.index:identifier
<Data Concept Type>Data Element"
::= { essSnapshotCameraEntry 1 }
```

5.16.3.2 Snapshot Camera Description

```
essSnapshotCameraDescription OBJECT-TYPE
SYNTAX      DisplayString (SIZE(1..255))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the description of this entry. The
description should include information about the location, direction, and
subject of the camera.
<SetConstraint>read-only
<DescriptiveName>SnapshotCamera.description:text
<Data Concept Type>Data Element"
::= { essSnapshotCameraEntry 2 }
```

5.16.3.3 Snapshot Camera Storage Path

```
essSnapshotCameraStoragePath OBJECT-TYPE
SYNTAX      DisplayString (SIZE(1..255))
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the storage path of snapshots taken from
this camera. The path indicated here shall be relative to the FTP login root.
This path can only include the FTP login root and its subdirectories and
cannot include any parent directories that may exist. The root is specified
by the string '/' (one forward slash. A subdirectory from the root may be
specified by the string '/subdir'.
<SetConstraint>read-only
<DescriptiveName>SnapshotCamera.storagePath:text
<Data Concept Type>Data Element"
DEFVAL { "/" }
::= { essSnapshotCameraEntry 3 }
```

5.16.3.4 Snapshot Camera Command

```
essSnapshotCameraCommand OBJECT-TYPE
SYNTAX      INTEGER {   ready(1),
                        captureSnapshot(2) }
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>A command to control the snapshot feature of the
ESS. Setting this object to a value of captureSnapshot(2) will command the
ESS to take a snapshot and save the image to memory. A Get of this object
will return a value of captureSnapshot(2) while the ESS is in the process of
capturing and saving the image to memory. A Get of this object when the ESS
```

```
is not in the process of capturing and saving the image to memory will return
a value of ready(1). If any errors occur in the process of capturing and
saving the image they shall be noted in essSnapshotError.
<Informative>Version 02 of this standard had a typo for the ACCESS; this
object must be read-write for the camera to capture a snapshot image.
<SetConstraint> Can only be set to 'captureSnapshot' when value is currently
'ready'; any other SET shall return a genErr, including a SET to 'ready'.
<DescriptiveName>SnapshotCamera.command:code
<Data Concept Type>Data Element"
::= { essSnapshotCameraEntry 4 }
```

5.16.3.5 Snapshot Camera Error

```
essSnapshotCameraError OBJECT-TYPE
SYNTAX      INTEGER {   none(1),
                        hardware(2),
                        insufficientMemory(3) }
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the status of the last attempt to capture
a snapshot using essSnapshotCommand.
<SetConstraint>read-only
<DescriptiveName>SnapshotCamera.error:code
<Valid Value Rule>
none - no error was detected
hardware - an error occured with the camera hardware when attempting to
capture a picture.
insufficientMemory - the ESS does not have sufficient memory to store the new
picture.
<Data Concept Type>Data Element"
::= { essSnapshotCameraEntry 5 }
```

5.16.3.6 Snapshot Camera Filename

```
-- EDITORS NOTE. In 2008, the ESS WG was considering a comment to revise the
-- syntax to OCTET STRING (SIZE (1..255)) to prevent zero-length names.
-- Also, to add a DEFVAL clause to initialize the value to ‘x’.
-- A note of caution has been added to the informative subfield.
-- This revision may require deprecation of this object in a future version.

essSnapshotCameraFilename OBJECT-TYPE
SYNTAX      OCTET STRING (SIZE(0..255))
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the filename used when storing new
snapshot images. This parameter shall only consist of upper and lower case
characters (A-Z, a-z), digits (0-9), underscores (_), periods (.), spaces,
and the following case-sensitive field names enclosed in chevrons (<>):

camera: A three digit camera number equal to essSnapshotCameraIndex.
sequence: A three digit value that increments by one each time this
      camera is used to take a picture. This value shall roll-over from '999'
      to '000'. The initial value of this field and its value after a power
      outage is manufacture dependent.
date: A six digit code indicating the UTC date in the format of YYMMDD.
time: A six digit code indicating the UTC time in the format of HHMMSS.
```



```
All implementations shall support all static character filenames with up to
eight characters in the base filename, a period, and up to three characters
in the filename extension. Support for longer filenames and the above fields
are optional.

For example, if this parameter has a value of 'latest.jpg', every snapshot
taken will be saved to 'latest.jpg' and will overwrite any previous image
saved to that filename. If this parameter has a value of '<date>
<sequence>.jpg', each snapshot will have a name (e.g., '060925 038')
indicating the date it was taken along with a relatively unique code. While
the second form prevents the inadvertent overwriting of filenames, it
increases the likelihood of consuming large amounts of field device memory
(eventually resulting in insufficientMemory errors) if not properly managed
by the management station.

<Informative>The filename does not necessarily define file format and it is
left to the users to define an appropriate filename.  For example, this
object may be set to a value of 'latest.bmp'.  All implementations would be
required to save their images with this filename; however, the image saved
may vary.  Some implementations may only save bitmapped images and the file
would have an appropriate name.  Other implementations may only save JPEG
images and the '.bmp' extension may be confusing.  Other implementations may
support both formats and use the extension in this field to select the
storage format, while yet others may support both formats but ignore the
extension assigned by this field (and use some other mechanism to select the
storage format).

A zero length filename may result in unexpected operations.  An
implementation may reject a zero length filename by responding with genErr.

<SetConstraint>always
<DescriptiveName>SnapshotCamera.filename:text
<Data Concept Type>Data Element"
::= { essSnapshotCameraEntry 6 }

  END
```

5.17 SNAPSHOT

A snapshot is an image captured in a computer file. It can be described through a number of attributes as indicated. However, these objects are not SNMP Objects.

5.17.1 Filename

<Definition> The name of the file in which the snapshot image is stored.
<Descriptive Name> Snapshot.filename:text
<Data Concept Type> Data Element

5.17.2 Image

<Definition> The graphic snapshot image. The storage format is not defined by this standard.
<DescriptiveName> Snapshot.image:frame
<Data Concept Type> Data Element

Annex A
REQUIREMENTS TRACEABILITY MATRIX (RTM)
[NORMATIVE]

Table 29 is a Requirements Traceability Matrix (RTM). Table 29 associates each requirement with its standardized dialog and the associated objects. The audience for Table 29 is implementers (vendors and central system developers) and conformance testers. Additionally, other interested parties might use Table 29 to determine how particular functions are to be implemented using the standardized dialogs, interfaces, and object definitions.

To conform to a requirement, an ESS shall implement all objects traced from that requirement; a Management Station shall implement all dialogs traced from the requirement. To be consistent with a requirement, a Management Station shall be able to fulfill the requirement using only objects that a conforming ESS is required to support.

Table 29 Requirements Traceability Matrix (RTM)

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.1		ESS Manager Requirements		
3.5.1.1		ESS Configuration Requirements		
3.5.1.1.1	F.3.1	Retrieve ESS Characteristics		
			5.2.1	essNtcipCategory
			5.2.2	essNtcipSiteDescription
			5.3.1	essTypeofStation
			5.4.1	essLatitude
			5.4.2	essLongitude
			5.5.1	essReferenceHeight
3.5.1.1.2	F.3.1	Retrieve Compressed Station Metadata		
			5.3.8	essStationMetaDataV3Block
3.5.1.1.3	F.3.3	Configure ESS Manager		
			5.2.2	essNtcipSiteDescription
3.5.1.2		ESS Status Monitoring Requirements		
3.5.1.2.1	F.3.1	Retrieve ESS Door Status		
			5.3.2	essDoorStatus

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.1.2.2	F.3.1	Retrieve Battery Status		
			5.3.3	essBatteryStatus
3.5.1.2.3	F.3.1	Retrieve Line Volts		
			5.3.4	essLineVolts
3.5.1.3		ESS Data Retrieval Requirements		
3.5.1.3.1	F.3.1	Retrieve Mobile ESS Movement		
			5.4.1	essLatitude
			5.4.2	essLongitude
			5.4.3	essVehicleSpeed
			5.4.4	essVehicleBearing
			5.4.5	essOdometer
			5.5.1	essReferenceHeight
3.5.1.3.3	F.3.1	Retrieve Compressed Mobile Station Data		
			5.3.7	essMobileBlock
3.5.1.4		ESS Control Requirements		
3.5.2		Sensor Manager Requirements		
3.5.2.1		Sensor Configuration Requirements		
3.5.2.1.1	F.3.1	Retrieve Atmospheric Pressure Height		
			5.5.2	essPressureHeight
3.5.2.1.2	F.4.6	Retrieve Metadata for Each Wind Sensor		
			5.6.8	windSensorTableNumSensors
			5.6.10.1	windSensorIndex
			5.6.10.2	windSensorHeight
			5.6.10.3	windSensorLocation
3.5.2.1.3	F.4.6	Retrieve Temperature Sensor Metadata		
			5.7.1	essNumTemperatureSensors
			5.7.3.1	essTemperatureSensorIndex
			5.7.3.2	essTemperatureSensorHeight

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.2.1.4	F.4.6	Retrieve Pavement Sensor Metadata		
			5.11.1	numEssPavementSensors
			5.11.3.1	essPavementSensorIndex
			5.11.3.2	essPavementSensorLocation
			5.11.3.3	essPavementType
			5.11.3.4	essPavementElevation
			5.11.3.5	essPavementExposure
			5.11.3.6	essPavementSensorType
3.5.2.1.5	F.4.6	Retrieve Subsurface Sensor Metadata		
			5.11.4	numEssSubSurfaceSensors
			5.11.6.1	essSubSurfaceSensorIndex
			5.11.6.2	essSubSurfaceSensorLocation
			5.11.6.3	essSubSurfaceType
			5.11.6.4	essSubSurfaceSensorDepth
3.5.2.1.6	F.4.8	Configure Pavement Sensor		
			5.11.3.1	essPavementSensorIndex
			5.11.3.2	essPavementSensorLocation
			5.11.3.3	essPavementType
			5.11.3.5	essPavementExposure
3.5.2.1.7	F.4.8	Configure Subsurface Sensor		
			5.11.6.1	essSubSurfaceSensorIndex
			5.11.6.2	essSubSurfaceSensorLocation
			5.11.6.3	essSubSurfaceType
			5.11.6.4	essSubSurfaceSensorDepth
3.5.2.1.8	4.2.8	Configure Passive Ice Detection Logic		
			5.13.1	numEssTreatments
			5.13.3.1	essPavementTreatmentIndex
			5.13.3.2	essPaveTreatProductType
			5.13.3.3	essPaveTreatProductForm
			5.13.3.4	essPercentProductMix
3.5.2.1.9	4.2.10	Configure Snapshot Camera		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.16.3.1	essSnapshotCameraIndex
			5.16.3.2	essSnapshotCameraDescription
			5.16.3.6	essSnapshotCameraFilename
3.5.2.2		Sensor Status Monitoring Requirements		
3.5.2.3		Sensor Data Retrieval Requirements		
3.5.2.3.1	F.3.1	Retrieve Weather Profile with Mobile Sources		
			5.3.9	essWeatherV3Block
			5.4.1	essLatitude
			5.4.2	essLongitude
			5.4.3	essVehicleSpeed
			5.4.4	essVehicleBearing
			5.4.5	essOdometer
			5.5.1	essReferenceHeight
			5.12.1	essMobileFriction
			5.12.2	essMobileObservationGroundState
			5.12.3	essMobileObservationPavement
3.5.2.3.2		Monitor Weather Condition		
3.5.2.3.2.1	F.3.1	Retrieve Atmospheric Pressure		
			5.5.4	essAtmosphericPressure
3.5.2.3.2.2	F.4.6	Retrieve Wind Data		
			5.6.8	windSensorTableNumSensors
			5.6.10.1	windSensorIndex
			5.6.10.4	windSensorAvgSpeed
			5.6.10.5	windSensorAvgDirection
			5.6.10.6	windSensorSpotSpeed
			5.6.10.7	windSensorSpotDirection
			5.6.10.8	windSensorGustSpeed
			5.6.10.9	windSensorGustDirection
			5.6.10.10	windSensorSituation
3.5.2.3.2.3	F.4.7	Retrieve Temperature		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.7.3.1	essTemperatureSensorIndex
			5.7.3.3	essAirTemperature
3.5.2.3.2.4	F.3.1	Retrieve Daily Minimum and Maximum Temperature		
			5.7.6	essMaxTemp
			5.7.7	essMinTemp
3.5.2.3.2.5	F.3.1	Retrieve Humidity		
			5.7.4	essWetbulbTemp
			5.7.5	essDewpointTemp
			5.8.1	essRelativeHumidity
3.5.2.3.2.6		Monitor Precipitation		
3.5.2.3.2.6.1	4.2.12	Retrieve Precipitation Presence		
			5.8.6	essPrecipYesNo
			5.8.18	precipitationSensorModelInformation
3.5.2.3.2.6.2	F.3.1	Retrieve Precipitation Rates		
			5.8.7	essPrecipRate
			5.8.8	essSnowfallAccumRate
			5.8.11	essPrecipitationStartTime
			5.8.12	essPrecipitationEndTime
3.5.2.3.2.6.3	F.3.1	Retrieve Precipitation Totals		
			5.8.13	essPrecipitationOneHour
			5.8.14	essPrecipitationThreeHours
			5.8.15	essPrecipitationSixHours
			5.8.16	essPrecipitationTwelveHours
			5.8.17	essPrecipitation24Hours
3.5.2.3.2.7	4.2.13	Retrieve Solar Radiation		
			5.9.2	essTotalSun
			5.9.4	essInstantaneousTerrestrialRadiation
			5.9.5	essInstantaneousSolarRadiation
			5.9.6	essTotalRadiation
			5.9.7	essTotalRadiationPeriod

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.2.3.2.8	F.3.1	Retrieve Visibility		
			5.10.1	essVisibility
3.5.2.3.2.9	F.3.1	Retrieve Compressed Weather Data		
			5.3.9	essWeatherV3Block
3.5.2.3.3		Monitor Pavement Condition		
3.5.2.3.3.1	4.2.14	Retrieve Pavement Surface Condition		
			5.11.3.1	essPavementSensorIndex
			5.11.3.7	essSurfaceStatus
			5.11.3.8	essSurfaceTemperature
			5.11.3.15	essPavementSensorError
			5.11.3.18	pavementSensorModelInformation
3.5.2.3.3.2	4.2.15	Retrieve Icing Conditions—Active		
			5.11.3.1	essPavementSensorIndex
			5.11.3.8	essSurfaceTemperature
			5.11.3.9	essPavementTemperature
			5.11.3.13	essSurfaceFreezePoint
			5.11.3.14	essSurfaceBlackIceSignal
			5.11.3.15	essPavementSensorError
			5.11.3.16	essSurfaceIceOrWaterDepth
			5.11.3.19	pavementSensorTemperatureDepth
3.5.2.3.3.3	4.2.6	Retrieve Icing Conditions—Passive		
			5.11.3.1	essPavementSensorIndex
			5.11.3.8	essSurfaceTemperature
			5.11.3.9	essPavementTemperature
			5.11.3.11	essSurfaceSalinity
			5.11.3.13	essSurfaceFreezePoint
			5.11.3.14	essSurfaceBlackIceSignal
			5.11.3.15	essPavementSensorError
			5.11.3.16	essSurfaceIceOrWaterDepth
			5.11.3.17	essSurfaceConductivityV2
			5.11.3.19	pavementSensorTemperatureDepth

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.13.1	numEssTreatments
			5.13.3.1	essPavementTreatmentIndex
			5.13.3.2	essPaveTreatProductType
			5.13.3.3	essPaveTreatProductForm
			5.13.3.4	essPercentProductMix
3.5.2.3.3.4	F.3.1	Retrieve Adjacent Snow Depth		
			5.8.3	essAdjacentSnowDepth
3.5.2.3.3.5	F.3.1	Retrieve Roadway Snow Depth		
			5.8.4	essRoadwaySnowDepth
3.5.2.3.3.6	F.3.1	Retrieve Roadway Ice Thickness		
			5.8.5	essRoadwaySnowPackDepth
			5.8.10	essIceThickness
3.5.2.3.3.7	F.3.1	Retrieve Compressed Pavement Condition Data		
			5.11.9	essPavementV3Block
3.5.2.3.4		Monitor Subsurface Conditions		
3.5.2.3.4.1	F.4.7	Retrieve Basic Subsurface Conditions		
			5.11.6.1	essSubSurfaceSensorIndex
			5.11.6.5	essSubSurfaceTemperature
			5.11.6.7	essSubSurfaceSensorError
3.5.2.3.4.2	F.4.7	Retrieve Subsurface Moisture		
			5.11.6.1	essSubSurfaceSensorIndex
			5.11.6.6	essSubSurfaceMoisture
3.5.2.3.4.3	F.3.1	Retrieve Compressed Subsurface Condition Data		
			5.11.8	essSubSurfaceBlock
3.5.2.3.5		Monitor Situation Assessments		
3.5.2.3.5.1	F.4.6	Retrieve Wind Situation		
			5.6.10.1	windSensorIndex
			5.6.10.10	windSensorSituation
3.5.2.3.5.2	F.3.1	Retrieve Precipitation Situation		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.8.9	essPrecipSituation
3.5.2.3.5.3	F.3.1	Retrieve Cloud Situation		
			5.9.3	essCloudSituation
3.5.2.3.5.4	F.3.1	Retrieve Visibility Situation		
			5.10.2	essVisibilitySituation
3.5.2.3.5.5	F.3.1	Retrieve Ground State		
			5.12.2	essMobileObservationGroundState
3.5.2.3.5.6	F.3.1	Retrieve Pavement State		
			5.12.3	essMobileObservationPavement
3.5.2.3.6		Monitor Air Quality and Biohazard Conditions		
3.5.2.3.6.1	F.3.1	Retrieve Carbon Monoxide Reading		
			5.14.1	essCO
3.5.2.3.6.2	F.3.1	Retrieve Carbon Dioxide Reading		
			5.14.2	essCO2
3.5.2.3.6.3	F.3.1	Retrieve Nitrous Oxide Reading		
			5.14.3	essNO
3.5.2.3.6.4	F.3.1	Retrieve Nitrogen Dioxide Reading		
			5.14.4	essNO2
3.5.2.3.6.5	F.3.1	Retrieve Sulfur Dioxide Reading		
			5.14.5	essSO2
3.5.2.3.6.6	F.3.1	Retrieve Ozone Reading		
			5.14.6	essO3
3.5.2.3.6.7	F.3.1	Retrieve Small Particulate Matter Reading		
			5.14.7	essPM10
3.5.2.3.6.8	F.3.1	Retrieve Compressed Air Quality Data		
			5.14.8	essAirQualityBlock
3.5.2.3.7	F.4.6	Retrieve Water Level		
			5.8.19	waterLevelSensorTableNumSensors
			5.8.21.1	waterLevelSensorIndex
			5.8.21.2	waterLevelSensorReading

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.2.3.8	4.2.2	Retrieve Snapshot		Upon ESS delivery the FTP username shall be _____________. Upon ESS delivery, the FTP password shall be ___________.
			5.17.1	<not an SNMP Object> Snapshot.filename:text
			5.17.2	<not an SNMP object> Snapshot.image:frame
3.5.2.3.9	4.2.11	Retrieve Snapshot Camera Configuration		
			5.16.1	essSnapshotNumberOfCameras
			5.16.3.1	essSnapshotCameraIndex
			5.16.3.2	essSnapshotCameraDescription
			5.16.3.3	essSnapshotCameraStoragePath
			5.16.3.6	essSnapshotCameraFilename
3.5.2.4		Sensor Control Requirements		
3.5.2.4.1	4.2.1	Capture Snapshot Image		
			5.16.3.1	essSnapshotCameraIndex
			5.16.3.4	essSnapshotCameraCommand
			5.16.3.5	essSnapshotCameraError
3.5.2.4.2	4.2.3	Delete Snapshot		
			5.17.1	<not an SNMP Object> Snapshot.filename:text
			5.17.2	<not an SNMP object> Snapshot.image:frame
3.5.2.4.3	4.2.4	Copy Snapshot		
			5.17.1	<not an SNMP Object> Snapshot.filename:text
			5.17.2	<not an SNMP object> Snapshot.image:frame
3.5.3		PTS Manager Requirements		
3.5.3.1		PTS Configuration Requirements		
3.5.3.1.1	4.2.5	Retrieve Stationary Pavement Treatment Configuration		
			5.13.1	numEssTreatments
			5.13.3.1	essPavementTreatmentIndex
			5.13.3.2	essPaveTreatProductType
			5.13.3.3	essPaveTreatProductForm
			5.13.3.4	essPercentProductMix
			5.13.10	ptsSignalDuration

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.13.18	ptsMonitoringDetectors
3.5.3.1.2	4.2.7	Configure Stationary Pavement Treatment System		
			5.13.1	numEssTreatments
			5.13.3.1	essPavementTreatmentIndex
			5.13.3.2	essPaveTreatProductType
			5.13.3.3	essPaveTreatProductForm
			5.13.3.4	essPercentProductMix
			5.13.10	ptsSignalDuration
			5.13.18	ptsMonitoringDetectors
3.5.3.1.3	F.3.1	Retrieve Mobile Pavement Treatment Configuration		
			5.13.6	pavementTreatmentBlock
3.5.3.1.4	4.2.9	Configure Mobile Pavement Treatment System		
			5.13.1	numEssTreatments
			5.13.3.1	essPavementTreatmentIndex
			5.13.3.2	essPaveTreatProductType
			5.13.3.3	essPaveTreatProductForm
			5.13.3.4	essPercentProductMix
			5.13.4	essPaveTreatmentAmount
			5.13.5	essPaveTreatmentWidth
3.5.3.2		PTS Status Monitoring Requirements		
3.5.3.2.1	F.3.1	Retrieve Pavement Treatment Status		
			5.13.9	ptsSprayerState
			5.13.11	ptsSignalEventCount
			5.13.12	ptsLastSignalEvent
			5.13.13	ptsActiveEventCount
			5.13.14	ptsInactiveEventCount
			5.13.15	ptsLastActiveEvent
			5.13.16	ptsLastInactiveEvent
			5.13.17	ptsError

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.3.3		PTS Data Retrieval Requirements		
3.5.3.4		PTS Control Requirements		
3.5.3.4.1	F.3.3	Set PTS Operational Mode		
			5.13.19	ptsOperationalModeV3
3.5.3.4.2	F.3.3	Manually Activate PTS Sprayer		
			5.13.20	ptsCommandStateV3
3.5.4		Backwards Compatibility Requirements		
3.5.4.1	RFC 1157	Version 1 Wind Sensor Meta Data		
			5.5.3	essWindSensorHeight
3.5.4.2	RFC 1157	Version 1 Average Wind Sensor Data		
			5.6.1	essAvgWindDirection
			5.6.2	essAvgWindSpeed
3.5.4.3	RFC 1157	Version 1 Spot Wind Sensor Data		
			5.6.3	essSpotWindDirection
			5.6.4	essSpotWindSpeed
3.5.4.4	RFC 1157	Version 1 Wind Gust Data		
			5.6.7	essMaxWindGustDir
			5.6.6	essMaxWindGustSpeed
3.5.4.5	RFC 1157	Version 1 Wind Situation		
			5.6.5	essWindSituation
3.5.4.6	RFC 1157	Version 1 Water Depth		
			5.8.2	essWaterDepth
3.5.4.7	RFC 1157	Version 1 Solar Radiation		
			5.9.1	essSolarRadiation

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.4.8	RFC 1157	Version 1 Surface Water Depth		
			5.11.3.10	essSurfaceWaterDepth
3.5.4.9	RFC 1157	Version 1 Surface Conductivity		
			5.11.3.12	essSurfaceConductivity
3.5.4.10	F.3.1	Version 2 Station Meta Data Block		
			5.3.5	essStationMetaDataBlock
3.5.4.11	F.3.1	Version 2 Weather Block		
			5.3.6	essWeatherBlock
3.5.4.12	F.3.1	Version 2 Pavement Block		
			5.11.7	essPavementBlock
F.2.1		Generic Architectural Requirements		
F.2.1.1		Support Basic Communications		
F.2.1.1.1	F.3.1	Retrieve Data		
F.2.1.1.2	F.3.3	Deliver Data		
F.2.1.1.3	F.3.2	Explore Data		
F.2.1.2		Support Logged Data		
F.2.1.2.1	F.4.2	Retrieve Current Configuration of Logging Service		
			NTCIP 1103 v01 Sec. A.7.2.1	eventClassNumber
			NTCIP 1103 v01 Sec. A.7.2.2	eventClassLimit
			NTCIP 1103 v01 Sec. A.7.2.3	eventClassClearTime
			NTCIP 1103 v01 Sec. A.7.2.4	eventClassDescription
			NTCIP 1103 v01 Sec. A.7.2.6	eventClassNumEvents
			NTCIP 1103 v01 A.7.4.1	eventConfigID

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1103 v01 A.7.4.2	eventConfigClass
			NTCIP 1103 v01 A.7.4.3	eventConfigMode
			NTCIP 1103 v01 A.7.4.4	eventConfigCompareValue
			NTCIP 1103 v01 A.7.4.5	eventConfigCompareValue2
			NTCIP 1103 v01 A.7.4.6	eventConfigCompareOID
			NTCIP 1103 v01 A.7.4.7	eventConfigLogOID
			NTCIP 1103 v01 A.7.4.8	eventConfigAction
			NTCIP 1103 v01 A.7.4.9	eventConfigStatus
F.2.1.2.2	F.4.3	Configure Logging Service		
			NTCIP 1103 v01 A.7.2.1	eventClassNumber
			NTCIP 1103 v01 A.7.2.2	eventClassLimit
			NTCIP 1103 v01 A.7.2.3	eventClassClearTime
			NTCIP 1103 v01 A.7.2.4	eventClassDescription
			NTCIP 1103 v01 A.7.4.1	eventConfigID
			NTCIP 1103 v01 A.7.4.2	eventConfigClass
			NTCIP 1103 v01 A.7.4.3	eventConfigMode
			NTCIP 1103 v01 A.7.4.4	eventConfigCompareValue

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1103 v01 A.7.4.5	eventConfigCompareValue2
			NTCIP 1103 v01 A.7.4.6	eventConfigCompareOID
			NTCIP 1103 v01 A.7.4.7	eventConfigLogOID
			NTCIP 1103 v01 A.7.4.8	eventConfigAction
			NTCIP 1103 v01 A.7.4.9	eventConfigStatus
F.2.1.2.3	F.4.1	Retrieve Logged Data		
			NTCIP 1103 v01 A.7.2.1	eventClassNumber
			NTCIP 1103 v01 A.7.2.5	eventClassNumRowsInLog
			NTCIP 1103 v01 A.7.2.6	eventClassNumEvents
			NTCIP 1103 v01 A.7.6.1	eventLogClass
			NTCIP 1103 v01 A.7.6.2	eventLogNumber
			NTCIP 1103 v01 A.7.6.3	eventLogID
			NTCIP 1103 v01 A.7.6.4	eventLogTime
			NTCIP 1103 v01 A.7.6.5	eventLogValue
			NTCIP 1103 v01 A.7.7	numEvents
F.2.1.2.4	F.3.3	Clear Log		
			NTCIP 1103 v01 A.7.2.1	eventClassNumber
			NTCIP 1103 v01 A.7.2.3	eventClassClearTime
F.2.1.2.5	F.3.1	Retrieve Capabilities of Event Logging Service		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1103 v01 A.7.1	maxEventClasses
			NTCIP 1103 v01 A.7.3	maxEventLogConfigs
			NTCIP 1103 v01 A.7.5	maxEventLogSize
F.2.1.2.6	F.3.1	Retrieve Total Number of Logged Events		
			NTCIP 1103 v01 A.7.2.1	eventClassNumber
			NTCIP 1103 v01 A.7.2.5	eventClassNumRowsInLog
			NTCIP 1103 v01 A.7.2.6	eventClassNumEvents
			NTCIP 1103 v01 A.7.7	numEvents
F.2.2		Generic Functional Requirements		
F.2.2.1		Generic Configuration Requirements		
F.2.2.1.1	F.4.6	Retrieve Device Component Information		
			NTCIP 1201 v02 Sec. 2.2.2	globalMaxModules
			NTCIP 1201 v02 Sec. 2.2.3.1	moduleNumber
			NTCIP 1201 v02 Sec. 2.2.3.2	moduleDeviceNode
			NTCIP 1201 v02 Sec. 2.2.3.3	moduleMake
			NTCIP 1201 v02 Sec. 2.2.3.4	moduleModel
			NTCIP 1201 v02 Sec. 2.2.3.5	moduleVersion
			NTCIP 1201 v02 Sec. 2.2.3.6	moduleType
F.2.2.1.2	F.3.1	Retrieve Device Configuration Identifier		
			NTCIP 1201 v02 Sec. 2.2.1	globalSetIDParameter
F.2.2.1.3	F.3.1	Retrieve Supported Standards		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1201 v02 Sec. 2.2.4	controllerBaseStandards
F.2.2.1.4	F.3.1	Retrieve System Name		
			RFC1213 Clause 6	sysName
F.2.2.1.5		Manage Time		
F.2.2.1.5.1	F.3.3	Set Time		
			NTCIP 1201 v02 Sec. 2.4.1	globalTime
F.2.2.1.5.2	F.3.1	Retrieve Current Time		
			NTCIP 1201 v02 Sec. 2.4.1	globalTime
F.2.2.1.6	F.4.6	Retrieve External Port Information		
			NTCIP 1201 v02 Sec. 2.8.2	auxIOTableNumAnalogPorts
			NTCIP 1201 v02 Sec. 2.8.3.1	auxIOPortType
			NTCIP 1201 v02 Sec. 2.8.3.2	auxIOPortNumber
			NTCIP 1201 v02 Sec. 2.8.3.3	auxIOPortDescription
			NTCIP 1201 v02 Sec. 2.8.3.4	auxIOPortResolution
			NTCIP 1201 v02 Sec. 2.8.3.6	auxIOPortDirection
F.2.2.1.7	F.4.8	Configure Port Information		
			NTCIP 1201 v02 Sec. 2.8.3.1	auxIOPortType
			NTCIP 1201 v02 Sec. 2.8.3.2	auxIOPortNumber
			NTCIP 1201 v02 Sec. 2.8.3.3	auxIOPortDescription
F.2.2.2		Generic Status Monitoring Requirements		
F.2.2.2.1	F.4.6	Monitor Status of External Device		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1201 v02 Sec. 2.8.3.5	auxIOPortValue
			NTCIP 1201 v02 Sec. 2.8.3.7	auxIOPortLastCommandedState
F.2.2.3		Generic Data Retrieval Requirements		
F.2.2.4		Generic Control Requirements		
F.2.2.4.1	F.4.8	Control External Device		
			NTCIP 1201 v02 Sec. 2.8.3.1	auxIOPortType
			NTCIP 1201 v02 Sec. 2.8.3.2	auxIOPortNumber
			NTCIP 1201 v02 Sec. 2.8.3.5	auxIOPortValue
3.6		Supplemental Requirements		
3.6.1		Required Number of Atmospheric Pressure Sensors		See Requirement 3.6.1 in PRL
3.6.2		Required Number of Wind Sensors		See Requirement 3.6.2 in PRL
3.6.3		Required Number of Temperature Sensors		See Requirement 3.6.3 in PRL
3.6.4		Required Number of Humidity Sensors		See Requirement 3.6.4 in PRL
3.6.5		Required Number of Precipitation Sensors		See Requirement 3.6.5 in PRL
3.6.6		Required Number of Solar Radiation Sensors		See Requirement 3.6.6 in PRL
3.6.7		Required Number of Visibility Sensors		See Requirement 3.6.7 in PRL
3.6.8		Required Number of Pavement Sensors		See Requirement 3.6.8 in PRL
3.6.9		Active Pavement Treatment Sensors		
3.6.10		Passive Pavement Treatment Sensors		
3.6.11		Required Number of Subsurface Sensors		See Requirement 3.6.11 in PRL
3.6.12		Required Number of Pavement Treatment Products		See Requirement 3.6.12 in PRL
3.6.13		Required Number of Carbon Monoxide Sensors		See Requirement 3.6.13 in PRL
3.6.14		Required Number of Carbon Dioxide Sensors		See Requirement 3.6.14 in PRL
3.6.15		Required Number of Nitrous Oxide Sensors		See Requirement 3.6.15 in PRL

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.6.16		Required Number of Nitrogen Dioxide Sensors		See Requirement 3.6.16 in PRL
3.6.17		Required Number of Sulfur Dioxide Sensors		See Requirement 3.6.17 in PRL
3.6.18		Required Number of Ozone Sensors		See Requirement 3.6.18 in PRL
3.6.19		Required Number of Small Particulate Matter Sensors		See Requirement 3.6.19 in PRL
3.6.20		Required Number of Snapshot Cameras		See Requirement 3.6.20 in PRL
3.6.21		Maximum Response Time for Requests		See Requirement 3.6.21 in PRL
3.6.22		Required Number of Water Level Sensors		See Requirement 3.6.22 in PRL
3.6.23		Support Camera Number in Filename		See Requirement 3.6.23 in PRL
3.6.24		Support Sequence Number in Filename		See Requirement 3.6.24 in PRL
3.6.25		Support Date in Filename		See Requirement 3.6.25 in PRL
3.6.26		Support Time in Filename		See Requirement 3.6.26 in PRL
3.6.27		Support Long Filenames		See Requirement 3.6.27 in PRL
F.2.3		Generic Supplemental Requirements		
F.2.3.1		Supplemental Requirements for Event Monitoring		
F.2.3.1.1		Record and Timestamp Events		
F.2.3.1.2		Support a Number of Event Classes		See Requirement F.2.3.1.2 in PRL
F.2.3.1.3		Support a Number of Event Types to Monitor		See Requirement F.2.3.1.3 in PRL
F.2.3.1.4		Support Monitoring of Event Types		
F.2.3.1.4.1		Support On-Change Events		
F.2.3.1.4.2		Support Greater Than Events		
F.2.3.1.4.3		Support Less Than Events		
F.2.3.1.4.4		Support Hysteresis Events		
F.2.3.1.4.5		Support Periodic Events		
F.2.3.1.4.6		Support Bit-flag Events		
F.2.3.1.5		Support Event Monitoring on Any Data		
F.2.3.1.6		Support a Number of Events to Store in Log		See Requirement F.2.3.1.6 in PRL
F.2.3.2		Required Number of Auxiliary Ports		See Requirement F.2.3.2 in PRL

Annex B
OBJECT TREE
[INFORMATIVE]

Figure 20 provides a pictorial representation of the Environmental Sensor Station Object Tree Structure.

The tree structure identifies how the object definitions are combined under specific nodes.

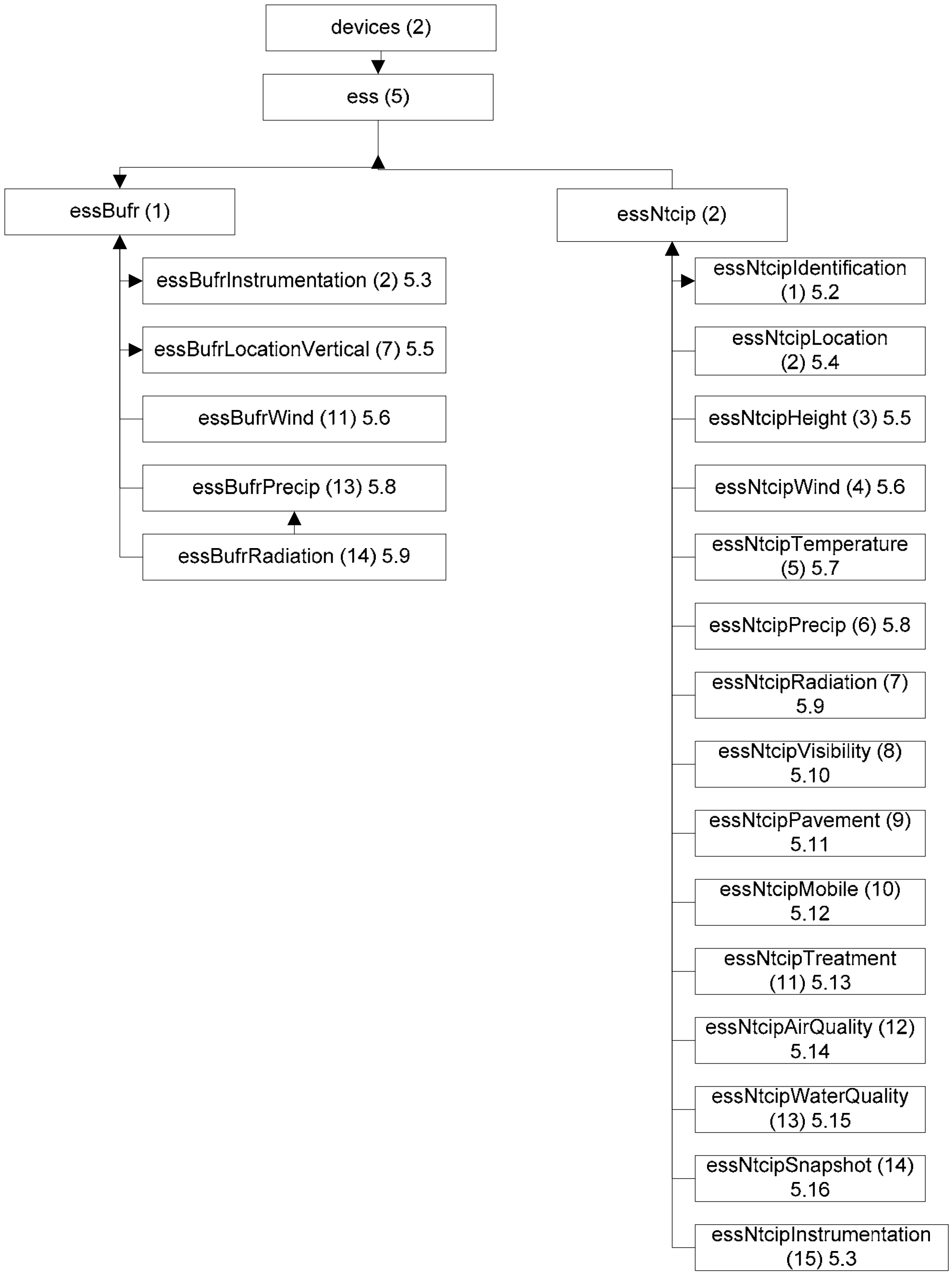

Figure 20 Object Tree for NTCIP 1204 v03

Annex C
TEST PROCEDURES
[NORMATIVE]

C.1 PURPOSE

Annex C defines the detailed, but generic, test procedures for testing an implementation of NTCIP 1204 v03.

C.1.1 Scope

Annex C defines test procedures in a format that is consistent with NTCIP 8007 v01and that covers the entire scope of NTCIP 1204 v03. It includes tests of some of the features defined in NTCIP 1201:2005, but only to the extent that these features have been incorporated by reference in NTCIP 1204 v03.

The procedures are intended to be used as a portion of the overall set of tests that would be performed during the component testing of a device.

NOTE—The NTCIP 8007 v01 test procedure format focuses on the testing of a device and is largely silent on testing a management station. Test procedures for the component testing of a management station and or test procedures for integration testing may be added at a future date.

C.1.2 Keywords

C.1.2.1 Additional Keywords

Keywords are words that are presented in all capital letters within the test procedures. Definitions of keywords are presented below. Keywords that are not defined below are defined in NTCIP 8007 v01.

Keyword	Definition
APPROPRIATE	This keyword is intended to direct the tester to ensure that the device is reporting the value provided by the sensor or set by the management station, not to test the accuracy of the sensor. The exact mechanism used to determine the appropriate value is beyond the scope of NTCIP 1204 v03, but should be defined within a test plan that references these procedures. For example, the appropriate values may be determined by using external software to check the values currently stored in the device, providing a subjective assessment of current conditions, emulating sensor inputs to create known values, etc. The mechanism chosen for a given test may be dependent upon the environment within which the device is being tested.
IF	This keyword causes the user (or application) to perform a comparison and take one action if the comparison evaluates to true and another action if the comparison evaluates to false. It is comparable to the "if…else…" expression in C.
FOR EACH	This keyword causes the user (or application) to begin a looping process that increments through a series of values. It is comparable to the "for…next" expression in C

C.1.2.2 Keyword Combinations

These test procedures frequently use the "SET-UP" and "VERIFY" keywords as defined in NTCIP 8007 v01 in the definition of a single step. When used jointly in these procedures, the failure logic of the "SET-UP" keyword shall override that of the "VERIFY" keyword. In other words, a failure of a step that uses both the "SET-UP" and "VERIFY" keywords means that the test case neither passes nor fails.

C.1.3 Rules for Following Test Procedures

To component test a device for conformance to NTCIP 1204 v03, the user shall follow the steps as written, filling in the pass/fail information in the 'Device' column.

C.2 TESTING REQUIREMENTS

C.2.1 Field Device Test Environment

All Test Cases covered by this Testing Requirements documentation require the Device Under Test (DUT) to be configured for the site and connected to a test application as depicted in Figure 21. A data analyzer may also be used to capture the data exchanged between the two components. The test environment should be designed to minimize any complicating factors that may result in anomalies unrelated to the specific test case. Failure to isolate such variables in the test environment may result in false results to the test. For example, the device may be conformant with NTCIP 1204 v03, but communication delays could result in timeouts and be misinterpreted as failures.

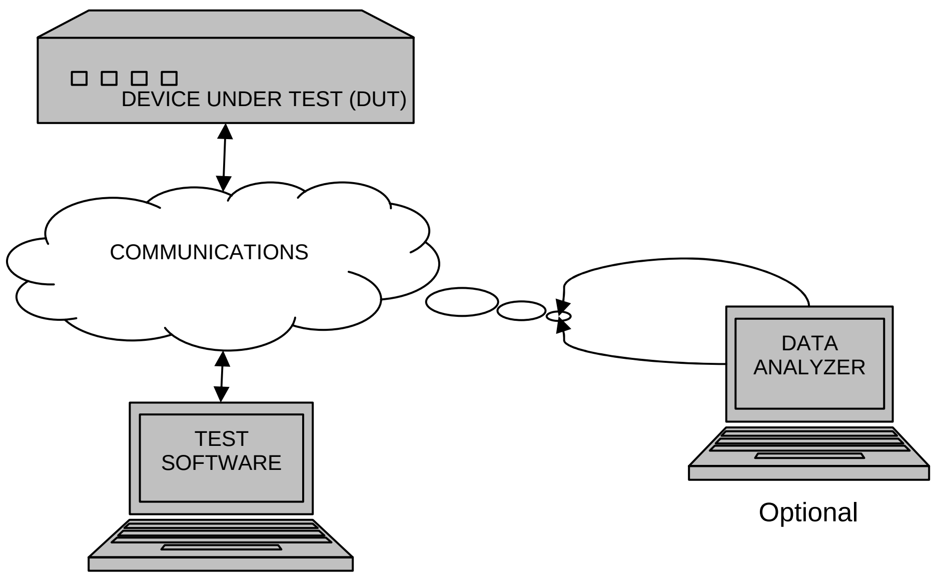

Figure 21 Field Device Test Environment

C.2.2 Traceability Table

Annex C.2.2 defines the relationships between the Requirements of NTCIP 1204 v03 Section 3 and the Test Cases presented in Annex C.2.3.

Table 30 Requirements to Test Case Traceability Table

Requirement		***Test Case***	
ID	***Title***	***ID***	***Title***
3.5	***Data Exchange Requirements***		
3.5.1	***ESS Manager Requirements***		
3.5.1.1	***ESS Configuration Requirements***		
3.5.1.1.1	***Retrieve ESS Characteristics***		
		C.2.3.1.1	***ESS Characteristics***

Requirement		***Test Case***	
ID	***Title***	***ID***	***Title***
3.5.1.1.2	***Retrieve Compressed Station Metadata***		
		C.2.3.1.2	***Retrieve Compressed Station Metadata***
3.5.1.1.3	***Configure ESS Manager***		
		C.2.3.1.1	***ESS Characteristics***
3.5.1.2	***ESS Status Monitoring Requirements***		
3.5.1.2.1	***Retrieve ESS Door Status***		
		C.2.3.1.3	***Retrieve ESS Door Status***
3.5.1.2.2	***Retrieve Battery Status***		
		C.2.3.1.4	***Retrieve Battery Status***
3.5.1.2.3	***Retrieve Line Volts***		
		C.2.3.1.5	***Retrieve Line Volts***
3.5.1.3	***ESS Data Retrieval Requirements***		
3.5.1.3.1	***Retrieve Mobile ESS Movement***		
		C.2.3.1.6	***Retrieve Mobile ESS Movement***
3.5.1.3.3	***Retrieve Compressed Mobile Station Data***		
		C.2.3.1.7	***Retrieve Compressed Mobile Station Data***
3.5.2	***Sensor Manager Requirements***		
3.5.2.1	***Sensor Configuration Requirements***		
3.5.2.1.1	***Retrieve Atmospheric Pressure Height***		
		C.2.3.2.1	***Retrieve Atmospheric Pressure Height***
3.5.2.1.2	***Retrieve Metadata for Each Wind Sensor***		
		C.2.3.2.2	***Retrieve Metadata for Each Wind Sensor***
3.5.2.1.3	***Retrieve Temperature Sensor Meta-Data***		
		C.2.3.2.3	***Retrieve Temperature Sensor Metadata***
3.5.2.1.4	***Retrieve Pavement Sensor Meta-Data***		
		C.2.3.2.4	***Retrieve Pavement Sensor Metadata***
3.5.2.1.5	***Retrieve Subsurface Sensor Meta-Data***		
		C.2.3.2.5	***Retrieve Subsurface Sensor Metadata***
3.5.2.1.6	***Configure Pavement Sensor***		
		C.2.3.2.6	***Configure Pavement Sensor***
3.5.2.1.7	***Configure Subsurface Sensor***		
		C.2.3.2.7	***Configure Subsurface Sensor***
3.5.2.1.8	***Configure Passive Ice Detection Logic***		
		C.2.3.2.8	***Configure Passive Ice Detection Logic***
3.5.2.1.9	***Configure Snapshot Camera***		
		C.2.3.2.9	***Configure Snapshot Camera***
3.5.2.3	***Sensor Data Retrieval Requirements***		
3.5.2.3.1	***Retrieve Weather Profile with Mobile Sources***		
		C.2.3.3.1	***Retrieve Weather Profile with Mobile Sources***
3.5.2.3.2	***Monitor Weather Condition***		
3.5.2.3.2.1	***Retrieve Atmospheric Pressure***		

Requirement		Test Case	
ID	Title	ID	Title
		C.2.3.3.2	Retrieve Atmospheric Pressure
3.5.2.3.2.2	Retrieve Wind Data		
		C.2.3.3.3	Retrieve Wind Data
3.5.2.3.2.3	Retrieve Temperature		
		C.2.3.3.4	Retrieve Temperature
3.5.2.3.2.4	Retrieve Daily Minimum and Maximum Temperature		
		C.2.3.3.5	Retrieve Daily Minimum and Maximum Temperature
3.5.2.3.2.5	Retrieve Humidity		
		C.2.3.3.6	Retrieve Humidity
3.5.2.3.2.6	Monitor Precipitation		
3.5.2.3.2.6.1	Retrieve Precipitation Presence		
		C.2.3.3.7	Retrieve Precipitation Presence
3.5.2.3.2.6.2	Retrieve Precipitation Rates		
		C.2.3.3.8	Retrieve Precipitation Rates
3.5.2.3.2.6.3	Retrieve Precipitation Totals		
		C.2.3.3.9	Retrieve Precipitation Totals
3.5.2.3.2.7	Retrieve Solar Radiation		
		C.2.3.3.10	Retrieve Solar Radiation
3.5.2.3.2.8	Retrieve Visibility		
		C.2.3.3.11	Retrieve Visibility
3.5.2.3.2.9	Retrieve Compressed Weather Data		
		C.2.3.3.12	Retrieve Compressed Weather Data
3.5.2.3.3	Monitor Pavement Condition		
3.5.2.3.3.1	Retrieve Pavement Surface Condition		
		C.2.3.4.1	Retrieve Pavement Surface Condition
3.5.2.3.3.2	Retrieve Icing Conditions—Active		
		C.2.3.4.2	Retrieve Icing Conditions - Active
3.5.2.3.3.3	Retrieve Icing Conditions—Passive		
		C.2.3.4.3	Retrieve Icing Conditions—Passive
3.5.2.3.3.4	Retrieve Adjacent Snow Depth		
		C.2.3.4.4	Retrieve Adjacent Snow Depth
3.5.2.3.3.5	Retrieve Roadway Snow Depth		
		C.2.3.4.5	Retrieve Roadway Snow Depth
3.5.2.3.3.6	Retrieve Roadway Ice Thickness		
		C.2.3.4.6	Retrieve Roadway Ice Thickness
3.5.2.3.3.7	Retrieve Compressed Pavement Condition Data		
		C.2.3.4.7	Retrieve Compressed Pavement Condition Data
3.5.2.3.4	Monitor Subsurface Conditions		
3.5.2.3.4.1	Retrieve Basic Subsurface Conditions		
		C.2.3.4.8	Retrieve Basic Subsurface Conditions
3.5.2.3.4.2	Retrieve Subsurface Moisture		

Requirement		Test Case	
ID	***Title***	***ID***	***Title***
		C.2.3.4.9	***Retrieve Subsurface Moisture***
3.5.2.3.4.3	***Retrieve Compressed Subsurface Condition Data***		
		C.2.3.4.10	***Retrieve Compressed Subsurface Condition Data***
3.5.2.3.5	***Monitor Situation Assessments***		
3.5.2.3.5.1	***Retrieve Wind Situation***		
		C.2.3.5.1	***Retrieve Wind Situation***
3.5.2.3.5.2	***Retrieve Precipitation Situation***		
		C.2.3.5.2	***Retrieve Precipitation Situation***
3.5.2.3.5.3	***Retrieve Cloud Situation***		
		C.2.3.5.3	***Retrieve Cloud Situation***
3.5.2.3.5.4	***Retrieve Visibility Situation***		
		C.2.3.5.4	***Retrieve Visibility Situation***
3.5.2.3.5.5	***Retrieve Ground State***		
		C.2.3.5.5	***Retrieve Ground State***
3.5.2.3.5.6	***Retrieve Pavement State***		
		C.2.3.5.6	***Retrieve Pavement State***
3.5.2.3.6	***Monitor Air Quality and Biohazard Conditions***		
3.5.2.3.6.1	***Retrieve Carbon Monoxide Reading***		
		C.2.3.6.1	***Retrieve Carbon Monoxide Reading***
3.5.2.3.6.2	***Retrieve Carbon Dioxide Reading***		
		C.2.3.6.2	***Retrieve Carbon Dioxide Reading***
3.5.2.3.6.3	***Retrieve Nitrous Oxide Reading***		
		C.2.3.6.3	***Retrieve Nitrous Oxide Reading***
3.5.2.3.6.4	***Retrieve Nitrogen Dioxide Reading***		
		C.2.3.6.4	***Retrieve Nitrogen Dioxide Reading***
3.5.2.3.6.5	***Retrieve Sulfur Dioxide Reading***		
		C.2.3.6.5	***Retrieve Sulfur Dioxide Reading***
3.5.2.3.6.6	***Retrieve Ozone Reading***		
		C.2.3.6.6	***Retrieve Ozone Reading***
3.5.2.3.6.7	***Retrieve Small Particulate Matter Reading***		
		C.2.3.6.7	***Retrieve Small Particulate Matter Reading***
3.5.2.3.6.8	***Retrieve Compressed Air Quality Data***		
		C.2.3.6.8	***Retrieve Compressed Air Quality Data***
3.5.2.3.7	***Retrieve Water Level***		
		C.2.3.7.1	***Retrieve Water Level***
3.5.2.3.8	***Retrieve Snapshot***		
		C.2.3.7.2	***Retrieve Snapshot***
3.5.2.3.9	***Retrieve Snapshot Camera Configuration***		
		C.2.3.7.2	***Retrieve Snapshot***
3.5.2.4	***Sensor Control Requirements***		
3.5.2.4.1	***Capture Snapshot Image***		

Requirement		Test Case	
ID	Title	ID	Title
		C.2.3.7.2	Retrieve Snapshot
3.5.2.4.2	Delete Snapshot		
		C.2.3.7.2	Retrieve Snapshot
3.5.2.4.3	Copy Snapshot		
		C.2.3.7.2	Retrieve Snapshot
3.5.3	PTS Manager Requirements		
3.5.3.1	PTS Configuration Requirements		
3.5.3.1.1	Retrieve Stationary Pavement Treatment Configuration		
		C.2.3.8.1	Retrieve Stationary Pavement Treatment Configuration
3.5.3.1.2	Configure Stationary Pavement Treatment System		
		C.2.3.8.2	Configure Stationary Pavement Treatment System
3.5.3.1.3	Retrieve Mobile Pavement Treatment Configuration		
		C.2.3.8.3	Retrieve Mobile Pavement Treatment Configuration
3.5.3.1.4	Configure Mobile Pavement Treatment System		
		C.2.3.8.4	Configure Mobile Pavement Treatment System
3.5.3.2	PTS Status Monitoring Requirements		
3.5.3.2.1	Retrieve Pavement Treatment Status		
		C.2.3.8.5	Retrieve Pavement Treatment Status
3.5.3.4	PTS Control Requirements		
3.5.3.4.1	Set PTS Operational Mode		
		C.2.3.8.6	Set PTS Operational Mode
3.5.3.4.2	Manually Activate PTS Sprayer		
		C.2.3.8.7	Manually Activate PTS Sprayer
3.5.4	Backwards Compatibility Requirements		
3.5.4.1	Version 1 Wind Sensor Meta Data		
		C.2.3.11.1	Version 1 Wind Sensor Meta Data
3.5.4.2	Version 1 Average Wind Sensor Data		
		C.2.3.11.2	Version 1 Average Wind Sensor Data
3.5.4.3	Version 1 Spot Wind Sensor Data		
		C.2.3.11.3	Version 1 Spot Wind Sensor Data
3.5.4.4	Version 1 Wind Gust Data		
		C.2.3.11.4	Version 1 Wind Gust Data
3.5.4.5	Version 1 Wind Situation		
		C.2.3.11.5	Version 1 Wind Situation
3.5.4.6	Version 1 Water Depth		
		C.2.3.11.6	Version 1 Water Depth
3.5.4.7	Version 1 Solar Radiation		
		C.2.3.11.7	Version 1 Solar Radiation
3.5.4.8	Version 1 Surface Water Depth		
		C.2.3.11.8	Version 1 Surface Water Depth
3.5.4.9	Version 1 Surface Conductivity		

Requirement		***Test Case***	
ID	***Title***	***ID***	***Title***
		C.2.3.11.9	***Version 1 Surface Conductivity***
3.5.4.10	***Version 2 Station Meta Data Block***		
		C.2.3.11.10	***Version 2 Station Meta Data Block***
3.5.4.11	***Version 2 Weather Block***		
		C.2.3.11.11	***Version 2 Weather Block***
3.5.4.12	***Version 2 Pavement Block***		
		C.2.3.11.12	***Version 2 Pavement Block***
3.6	***Supplemental Requirements***		
3.6.1	***Required Number of Atmospheric Pressure Sensors***		
		C.2.3.3.2	***Retrieve Atmospheric Pressure***
3.6.2	***Required Number of Wind Sensors***		
		C.2.3.3.3	***Retrieve Wind Data***
3.6.3	***Required Number of Temperature Sensors***		
		C.2.3.3.4	***Retrieve Temperature***
3.6.4	***Required Number of Humidity Sensors***		
		C.2.3.3.6	***Retrieve Humidity***
3.6.5	***Required Number of Precipitation Sensors***		
		C.2.3.3.7	***Retrieve Precipitation Presence***
3.6.6	***Required Number of Solar Radiation Sensors***		
		C.2.3.3.10	***Retrieve Solar Radiation***
3.6.7	***Required Number of Visibility Sensors***		
		C.2.3.3.11	***Retrieve Visibility***
3.6.8	***Required Number of Pavement Sensors***		
		C.2.3.4.1	***Retrieve Pavement Surface Condition***
3.6.9	***Active Pavement Treatment Sensors***		
		C.2.3.4.2	***Retrieve Icing Conditions—Active***
3.6.10	***Passive Pavement Treatment Sensors***		
		C.2.3.4.3	***Retrieve Icing Conditions—Passive***
3.6.11	***Required Number of Subsurface Sensors***		
		C.2.3.4.8	***Retrieve Basic Subsurface Conditions***
3.6.12	***Required Number of Pavement Treatment Products***		
		C.2.3.8.1	***Retrieve Stationary Pavement Treatment Configuration***
		C.2.3.8.3	***Retrieve Mobile Pavement Treatment Configuration***
3.6.13	***Required Number of Carbon Monoxide Sensors***		
		C.2.3.6.1	***Retrieve Carbon Monoxide Reading***
3.6.14	***Required Number of Carbon Dioxide Sensors***		
		C.2.3.6.2	***Retrieve Carbon Dioxide Reading***
3.6.15	***Required Number of Nitrous Oxide Sensors***		
		C.2.3.6.3	***Retrieve Nitrous Oxide Reading***
3.6.16	***Required Number of Nitrogen Dioxide Sensors***		
		C.2.3.6.4	***Retrieve Nitrogen Dioxide Reading***

Requirement		Test Case	
ID	Title	ID	Title
3.6.17	Required Number of Sulfur Dioxide Sensors		
		C.2.3.6.5	Retrieve Sulfur Dioxide Reading
3.6.18	Required Number of Ozone Sensors		
		C.2.3.6.6	Retrieve Ozone Reading
3.6.19	Required Number of Small Particulate Matter Sensors		
		C.2.3.6.7	Retrieve Small Particulate Matter Reading
3.6.20	Required Number of Snapshot Cameras		
		C.2.3.7.2	Retrieve Snapshot
3.6.21	Maximum Response Time for Requests		
		C.2.3	Test Procedures
3.6.22	Required Number of Water Level Sensors		
		C.2.3.7.1	Retrieve Water Level
3.6.23	Support Camera Number in Filename		
		C.2.3.2.9	Configure Snapshot Camera
3.6.24	Support Sequence Number in Filename		
		C.2.3.2.9	Configure Snapshot Camera
3.6.25	Support Date in Filename		
		C.2.3.2.9	Configure Snapshot Camera
3.6.26	Support Time in Filename		
		C.2.3.2.9	Configure Snapshot Camera
3.6.27	Support Long Filenames		
		C.2.3.2.9	Configure Snapshot Camera
F.2	External Requirements		
F.2.1	Generic Architectural Requirements		
F.2.1.1	Support Basic Communications		
F.2.1.1.1	Retrieve Data		
		C.2.3.1.1	ESS Characteristics
F.2.1.1.2	Deliver Data		
		C.2.3.1.1	ESS Characteristics
F.2.1.1.3	Explore Data		
		C.2.3.10.1	Explore Data
F.2.1.2	Support Logged Data		
F.2.1.2.1	Retrieve Current Configuration of Logging Service		
		C.2.3.9.13	Determine Configuration of Logging Service
F.2.1.2.2	Configure Logging Service		
		C.2.3.9.2	Configure Event Log
		C.2.3.9.7	Verify Support for an On-Change Event
		C.2.3.9.8	Verify Support for a Greater Than Event
		C.2.3.9.9	Verify Support for a Less Than Event
		C.2.3.9.10	Verify Support for a Hysteresis Event
		C.2.3.9.11	Verify Support for a Periodic Event

Requirement		Test Case	
ID	Title	ID	Title
		C.2.3.9.12	Verify Support for a Bit-flag Event
F.2.1.2.3	Retrieve Logged Data		
		C.2.3.9.3	Retrieve Logged Data
F.2.1.2.4	Clear Log		
		C.2.3.9.4	Clear Log
F.2.1.2.5	Retrieve Capabilities of Event Logging Service		
		C.2.3.9.1	Determine Capabilities of Event Logging Service
F.2.1.2.6	Retrieve Total Number of Logged Events		
		C.2.3.9.5	Determine Total Number of Events
F.2.2	Generic Functional Requirements		
F.2.2.1	Generic Configuration Requirements		
F.2.2.1.1	Retrieve Device Component Information		
		C.2.3.10.2	Determine Device Component Information
F.2.2.1.2	Retrieve Device Configuration Identifier		
		C.2.3.10.3	Retrieve Device Configuration Identifier
F.2.2.1.3	Retrieve Supported Standards		
		C.2.3.10.4	Determine Supported Standards
F.2.2.1.4	Retrieve System Name		
		C.2.3.10.5	Retrieve System Name
F.2.2.1.5	Manage Time		
F.2.2.1.5.1	Set Time		
		C.2.3.10.6	Set Time
F.2.2.1.5.2	Retrieve Current Time		
		C.2.3.10.6	Set Time
F.2.2.1.6	Retrieve External Port Information		
		C.2.3.10.7	Monitor External Port Information
F.2.2.1.7	Configure Port Information		
		C.2.3.10.8	Configure External Port
F.2.2.2	Generic Status Monitoring Requirements		
F.2.2.2.1	Monitor Status of External Device		
		C.2.3.10.7	Monitor External Port Information
F.2.2.43	Generic Control Requirements		
F.2.2.43.1	Control External Device		
		C.2.3.10.8	Configure External Port
F.2.3	Generic Supplemental Requirements		
F.2.3.1	Supplemental Requirements for Event Monitoring		
F.2.3.1.1	Record and Timestamp Events		
		C.2.3.9.3	Retrieve Logged Data
		C.2.3.9.6	Verify Log Limit Storage
F.2.3.1.2	Support a Number of Event Classes		
		C.2.3.9.1	Determine Capabilities of Event Logging Service

Requirement		Test Case	
ID	***Title***	***ID***	***Title***
F.2.3.1.3	***Support a Number of Event Types to Monitor***		
		C.2.3.9.1	***Determine Capabilities of Event Logging Service***
F.2.3.1.4	***Support Monitoring of Event Types***		
F.2.3.1.4.1	***Support On-Change Events***		
		C.2.3.9.7	***Verify Support for an On-Change Event***
F.2.3.1.4.2	***Support Greater Than Events***		
		C.2.3.9.8	***Verify Support for a Greater Than Event***
F.2.3.1.4.3	***Support Less Than Events***		
		C.2.3.9.9	***Verify Support for a Less Than Event***
F.2.3.1.4.4	***Support Hysteresis Events***		
		C.2.3.9.10	***Verify Support for a Hysteresis Event***
F.2.3.1.4.5	***Support Periodic Events***		
		C.2.3.9.11	***Verify Support for a Periodic Event***
F.2.3.1.4.6	***Support Bit-flag Events***		
		C.2.3.9.12	***Verify Support for a Bit-flag Event***
F.2.3.1.5	***Support Event Monitoring on Any Data***		
		C.2.3.9.2	***Configure Event Log***
F.2.3.1.6	***Support a number of Events to Store in Log***		
		C.2.3.9.1	***Determine Capabilities of Event Logging Service***
		C.2.3.9.6	***Verify Log Limit Storage***
F.2.3.2	***Required Number of Auxiliary Ports***		
		C.2.3.10.7	***Monitor External Port Information***

C.2.3 Test Procedures

Annex C.2.3 provides detailed test procedures.

C.2.3.1 ESS Manager Tests

C.2.3.1.1 ESS Characteristics

Test Case: 1.1	***Title:***	*ESS Characteristics*
	Description:	*This test case verifies that the ESS accurately reports its type, category and location and allows a management station to edit the site description.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essNtcipCategory.0 »essNtcipSiteDescription.0 »essTypeofStation.0 »essLatitude.0 »essLongitude.0 »essReferenceHeight.0	Pass / Fail (Sec. 3.5.1.1.1)

2	VERIFY that the RESPONSE VALUE for essNtcipCategory.0 is APPROPRIATE. NOTE—See NTCIP 1204 v03 Sec. 5.2.1 for valid enumerated values.	Pass / Fail (Sec. 5.2.1)
3	VERIFY that the RESPONSE VALUE for essNtcipSiteDescription.0 contains only display string characters.	Pass / Fail (Sec. 5.2.2)
4	VERIFY that the RESPONSE VALUE for essTypeofStation.0 is APPROPRIATE. NOTE—See NTCIP 1204 v03 Sec. 5.3.1 for the definition of valid values.	Pass / Fail (Sec. 5.3.1)
5	VERIFY that the RESPONSE VALUE for essLatitude.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.1)
6	VERIFY that the RESPONSE VALUE for essLongitude.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.2)
7	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.1)
8	Determine the RESPONSE VALUE for essNtcipSiteDescription.0. RECORD this information as: »Orig_Description	
9	Calculate a random test value for the site description containing between 1 and 255 DisplayString characters. RECORD this information as: »New_Description	
10	SET the following object(s) to the value(s) shown: »essNtcipSiteDescription.0 = New_Description	Pass / Fail (Sec. 3.5.1.1.3)
11	GET the following object(s): »essNtcipCategory.0 »essNtcipSiteDescription.0 »essTypeofStation.0 »essLatitude.0 »essLongitude.0 »essReferenceHeight.0	Pass / Fail (Sec. 3.5.1.1.1)
12	VERIFY that the RESPONSE VALUE for essNtcipSiteDescription.0 is equal to New_Description.	Pass / Fail (Sec. 3.5.1.1.3)
13	SET the following object(s) to the value(s) shown: »essNtcipSiteDescription.0 = Orig_Description	Pass / Fail (Sec. 3.5.1.1.3)
14	GET the following object(s): »essNtcipSiteDescription.0	Pass / Fail (RFC 1157)
15	VERIFY that the RESPONSE VALUE for essNtcipSiteDescription.0 is equal to Orig_Description.	Pass / Fail (Sec. 3.5.1.1.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.1.2 Retrieve Compressed Station Metadata

Test Case: 1.2	***Title:***	*Retrieve Compressed Station Metadata*	
	Description:	*This test case verifies that the ESS accurately reports its type, category, location, sensor locations, and pavement treatment information in a compressed form.*	
	Variables:	*Pressure_Supported*	*PRL 2.5.2.1.1*
		Wind_Supported	*PRL 2.5.2.1.2*
		Required_Wind_Sensors	*PRL 3.6.2*
		Temperature_Supported	*PRL 2.5.2.1.3*
		Required_Temp_Sensors	*PRL 3.6.3*
		Pavement_Supported	*PRL 2.5.2.2*
		Required_Pavement_Sensors	*PRL 3.6.8*
		Subsurface_Supported	*PRL 2.5.2.3*
		Required_Subsurface_Sensors	*PRL 3.6.11*
		Pavement_Treatment_Supported	*PRL 2.5.3*
		Required_Pavement_Treatment_Products	*PRL 3.6.12*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the ESS is required to monitor atmospheric pressure (PRL 2.5.2.1.1). RECORD this information as: »Pressure_Supported	
2	CONFIGURE: Determine the number of wind sensors that are supported. RECORD this information as: »Wind_Supported (whether the ESS is required to monitor winds—PRL 2.5.2.1.2) »Required_Wind_Sensors (the number of temperature sensors that the ESS is required to support—PRL 3.6.2)	
3	CONFIGURE: Determine whether the ESS is required to support temperature sensors and, if so, how many sensors are required. RECORD this information as: »Temperature_Supported (whether the ESS is required to monitor temperature—PRL 2.5.2.1.3) »Required_Temp_Sensors (the number of temperature sensors that the ESS is required to support—PRL 3.6.3)	
4	CONFIGURE: Determine whether the ESS is required to support pavement sensors and, if so, how many sensors are required. RECORD this information as: »Pavement_Supported (whether the ESS is required to monitor the pavement—PRL 2.5.2.2) »Required_Pavement_Sensors (the number of pavement sensors that the ESS is required to support—PRL 3.6.8)	
5	CONFIGURE: Determine whether the ESS is required to support subsurface sensors and, if so, how many sensors are required. RECORD this information as: »Subsurface_Supported (whether the ESS is required to monitor the subsurface - PRL 2.5.2.3) »Required_Subsurface_Sensors (the number of subsurface sensors that the ESS is required to support—PRL 3.6.11)	
6	CONFIGURE: Determine whether the ESS is required to provide pavement treatment capabilities and, if so, how many treatment products the device is required to support. RECORD this information as:	

	»Pavement_Treatment_Supported (whether pavement treatment capabilities are required—PRL 2.5.3) »Required_Pavement_Treatment_Products (number of pavement treatment products required—PRL 3.6.12)	
7	GET the following object(s): »essStationMetaDataV3Block.0	Pass / Fail (Sec. 3.5.1.1.2)
8	Decode the essStationMetaDataV3Block.0 structure.	
9	VERIFY that the essStationMetaDataV3Block.0 structure was decoded without error.	Pass / Fail (Sec. 5.3.8)
10	VERIFY that the RESPONSE VALUE for the essNtcipCategory.0 field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.1.1.1)
11	VERIFY that the RESPONSE VALUE for the essNtcipCategory.0 field of the essStationMetaDataV3Block.0 object is APPROPRIATE. NOTE—See NTCIP 1204 v03 Sec. 5.2.1 for valid enumerated values.	Pass / Fail (Sec. 5.2.1)
12	VERIFY that the RESPONSE VALUE for the essTypeOfStation.0 field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.1.1.1)
13	VERIFY that the RESPONSE VALUE for the essTypeOfStation.0 field of the essStationMetaDataV3Block.0 object is APPROPRIATE. NOTE—See NTCIP 1204 v03 Sec. 5.3.1 for the definition of valid values.	Pass / Fail (Sec. 5.3.1)
14	VERIFY that the RESPONSE VALUE for the essLatitude.0 field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.1.1.1)
15	VERIFY that the RESPONSE VALUE for the essLatitude.0 field of the essStationMetaDataV3Block.0 object indicates an APPROPRIATE.	Pass / Fail (Sec. 5.4.1)
16	VERIFY that the RESPONSE VALUE for the essLongitude.0 field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.1.1.1)
17	VERIFY that the RESPONSE VALUE for the essLongitude.0 field of the essStationMetaDataV3Block.0 object indicates an APPROPRIATE.	Pass / Fail (Sec. 5.4.2)
18	VERIFY that the RESPONSE VALUE for the essReferenceHeight.0 field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.1.1.1)
19	VERIFY that the RESPONSE VALUE for the essReferenceHeight.0 field of the essStationMetaDataV3Block.0 object is APPROPRIATE.	Pass / Fail (Sec. 5.5.1)
20	IF Pressure_Supported is equal to true, then proceed to Step 20.1; otherwise, proceed to Step 21.	
20.1	VERIFY that the RESPONSE VALUE for the essPressureHeight.0 field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.1.1)
20.2	VERIFY that the RESPONSE VALUE for essPressureHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.2)
21	IF Wind_Supported is equal to true, then proceed to Step 21.1; otherwise, proceed to Step 22.	
21.1	VERIFY that the windMetaData field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.1.2)

21.2	VERIFY that the windMetaData field of the essStationMetaDataV3Block.0 object contains at least Required_Wind_Sensors entries.	Pass / Fail (Sec. 3.6.2)
21.3	FOR EACH value, N, from 1 to Required_Temp_Sensors, perform Steps 21.3.1 through 21.3.4.	
21.3.1	VERIFY that the RESPONSE VALUE for the windSensorIndex.0 field is present in the Nth WindMetaData structure.	Pass / Fail (Sec. 3.5.2.1.2)
21.3.2	VERIFY that the RESPONSE VALUE for the windSensorIndex.x field in the Nth WindMetaData structure is equal to N.	Pass / Fail (Sec. 5.6.10.1)
21.3.3	VERIFY that the RESPONSE VALUE for the windSensorHeight.x field is present in the Nth WindMetaData structure.	Pass / Fail (Sec. 3.5.2.1.2)
21.3.4	VERIFY that the RESPONSE VALUE for windSensorHeight.x field in the Nth WindMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.2)
22	IF Temperature_Supported is equal to true, then proceed to Step 22.1; otherwise, proceed to Step 23.	
22.1	VERIFY that the RESPONSE VALUE for the temperatureMetaData field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.1.3)
22.2	VERIFY that the temperatureMetaData field contains at least Required_Temp_Sensors entries.	Pass / Fail (Sec. 3.6.3)
22.3	FOR EACH value, N, from 1 to Required_Temp_Sensors, perform Steps 22.3.1 through 22.3.4.	
22.3.1	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.0 field is present in the Nth TemperatureMetaData structure.	Pass / Fail (Sec. 3.5.2.1.3)
22.3.2	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.0 field in the Nth TemperatureMetaData structure is equal to N.	Pass / Fail (Sec. 5.7.3.1)
22.3.3	VERIFY that the RESPONSE VALUE for the essTemperatureSensorHeight.0 field is present in the Nth TemperatureMetaData structure.	Pass / Fail (Sec. 3.5.2.1.3)
22.3.4	VERIFY that the RESPONSE VALUE for essTemperatureSensorHeight.0 field in the Nth TemperatureMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.2)
23	IF Pavement_Supported is equal to true, then proceed to Step 23.1; otherwise, proceed to Step 24.	
23.1	VERIFY that the RESPONSE VALUE for the pavementMetaData field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.1.4)
23.2	VERIFY that the pavementMetaData field contains at least Required_Pavement_Sensors entries.	Pass / Fail (Sec. 3.6.8)
23.3	FOR EACH value, N, from 1 to Required_Pavement_Sensors, perform Steps 23.3.1 through 23.3.10.	
23.3.1	VERIFY that the RESPONSE VALUE for the essPavementSensorIndex.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 3.5.2.1.4)
23.3.2	VERIFY that the RESPONSE VALUE for the essPavementSensorIndex.0 field in the Nth PavementMetaData structure is equal to N.	Pass / Fail (Sec. 5.11.3.1)

23.3.3	VERIFY that the RESPONSE VALUE for the essPavementType.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 3.5.2.1.4)
23.3.4	VERIFY that the RESPONSE VALUE for the essPavementType.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.3)
23.3.5	VERIFY that the RESPONSE VALUE for the essPavementElevation.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 3.5.2.1.4)
23.3.6	VERIFY that the RESPONSE VALUE for the essPavementElevation.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.4)
23.3.7	VERIFY that the RESPONSE VALUE for the essPavementExposure.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 3.5.2.1.4)
23.3.8	VERIFY that the RESPONSE VALUE for the essPavementExposure.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.5)
23.3.9	VERIFY that the RESPONSE VALUE for the essPavementSensorType.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 3.5.2.1.4)
23.3.10	VERIFY that the RESPONSE VALUE for the essPavementSensorType.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.6)
24	IF Subsurface_Supported is equal to true, then proceed to Step 24.1; otherwise, proceed to Step 25.	
24.1	VERIFY that the RESPONSE VALUE for the SubSurfaceMetaData field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.1.5)
24.2	VERIFY that the SubSurfaceMetaData field contains at least Required_Subsurface_Sensors entries.	Pass / Fail (Sec. 3.6.11)
24.3	FOR EACH value, N, from 1 to Required_Subsurface_Sensors, perform Steps 24.3.1 through 24.3.6.	
24.3.1	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorIndex.0 field is present in the Nth SubSurfaceMetaData structure.	Pass / Fail (Sec. 3.5.2.1.5)
24.3.2	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorIndex.0 field in the Nth SubSurfaceMetaData structure is equal to N.	Pass / Fail (Sec. 5.11.6.1)
24.3.3	VERIFY that the RESPONSE VALUE for the essSubSurfaceType.0 field is present in the Nth SubSurfaceMetaData structure.	Pass / Fail (Sec. 3.5.2.1.5)
24.3.4	VERIFY that the RESPONSE VALUE for the essSubSurfaceType.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.3)
24.3.5	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorDepth.0 field is present in the Nth SubSurfaceMetaData structure.	Pass / Fail (Sec. 3.5.2.1.5)
24.3.6	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorDepth.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.4)
25	IF Pavement_Treatment_Supported is equal to true, then proceed to Step 25.1; otherwise, proceed to EXIT.	
25.1	VERIFY that the RESPONSE VALUE for the treatmentMetaData field of the essStationMetaDataV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.1.8)

25.2	VERIFY that the treatmentMetaData field contains at least Required_Pavement_Treatment_Products entries.	Pass / Fail (Sec. 3.6.12)
25.3	FOR EACH value, N, from 1 to Required_Pavement_Treatment_Products, perform Steps 25.3.1 through 25.3.11.	
25.3.1	VERIFY that the RESPONSE VALUE for the essPavementTreatmentIndex.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 3.5.2.1.8)
25.3.2	VERIFY that the RESPONSE VALUE for the essPavementTreatmentIndex.0 field in the Nth TreatmentMetaData structure is equal to N.	Pass / Fail (Sec. 5.13.3.1)
25.3.3	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 3.5.2.1.8)
25.3.4	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.0 field in the Nth TreatmentMetaData structure is between 1 and 14, inclusive.	Pass / Fail (Sec. 5.13.3.2)
25.3.5	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.x field in the Nth TreatmentMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.2)
25.3.6	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 3.5.2.1.8)
25.3.7	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.0 field in the Nth TreatmentMetaData structure is between 1 and 4, inclusive.	Pass / Fail (Sec. 5.13.3.3)
25.3.8	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.0 field in the Nth TreatmentMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.3)
25.3.9	VERIFY that the RESPONSE VALUE for the essPercentProductMix.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 3.5.2.1.8)
25.3.10	VERIFY that the RESPONSE VALUE for the essPercentProductMix.0 field in the Nth TreatmentMetaData structure is between 0 and 100, inclusive.	Pass / Fail (Sec. 5.13.3.4)
25.3.11	VERIFY that the RESPONSE VALUE for the essPercentProductMix.0 field in the Nth TreatmentMetaData structure is APPROPRIATE for the subject pavement treatment.	Pass / Fail (Sec. 5.13.3.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.1.3 Retrieve ESS Door Status

Test Case: 1.3	***Title:***	*Retrieve ESS Door Status*
	Description:	*This test case verifies that the ESS allows a management station to determine whether any of the doors related to the ESS are open.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	Verify that all doors associated with the ESS are closed.	
2	GET the following object(s): »essDoorStatus.0	Pass / Fail (Sec. 3.5.1.2.1)
3	VERIFY that the RESPONSE VALUE for essDoorStatus.0 is equal to 0.	Pass / Fail (Sec. 5.3.2)
4	Open at least one door associated with the ESS	
5	GET the following object(s): »essDoorStatus.0	Pass / Fail (Sec. 3.5.1.2.1)
6	VERIFY that the RESPONSE VALUE for essDoorStatus.0 is equal to 1.	Pass / Fail (Sec. 5.3.2)
7	Return all doors to their original state.	

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.1.4 Retrieve Battery Status

Test Case: 1.4	***Title:***	*Retrieve Battery Status*
	Description:	*This test case verifies that the ESS allows a management station to determine the charge status of the battery.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essBatteryStatus.0	Pass / Fail (Sec. 3.5.1.2.2)
2	VERIFY that the RESPONSE VALUE for essBatteryStatus.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.3.3)
3	VERIFY that the RESPONSE VALUE for essBatteryStatus.0 is less than or equal to 101.	Pass / Fail (Sec. 5.3.3)
4	VERIFY that the RESPONSE VALUE for essBatteryStatus.0 is APPROPRIATE.	Pass / Fail (Sec. 5.3.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.1.5 Retrieve Line Volts

Test Case: 1.5	***Title:***	*Retrieve Line Volts*
	Description:	*This test case verifies that the ESS allows a management station to determine the voltage on the incoming A/C power.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essLineVolts.0	Pass / Fail (Sec. 3.5.1.2.3)
2	VERIFY that the RESPONSE VALUE for essLineVolts.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.3.4)
3	VERIFY that the RESPONSE VALUE for essLineVolts.0 is less than or equal to 255.	Pass / Fail (Sec. 5.3.4)
4	VERIFY that the RESPONSE VALUE for essLineVolts.0 is APPROPRIATE.	Pass / Fail (Sec. 5.3.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.1.6 Retrieve Mobile ESS Movement

Test Case: 1.6	***Title:***	*Retrieve Mobile ESS Movement*
	Description:	*This test case verifies that the ESS allows a management station to determine the speed, location and direction of the management station*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essLatitude.0 »essLongitude.0 »essVehicleSpeed.0 »essVehicleBearing.0 »essOdometer.0 »essReferenceHeight.0	Pass / Fail (Sec. 3.5.1.3.1)
2	VERIFY that the RESPONSE VALUE for essLatitude.0 is greater than or equal to -90000000.	Pass / Fail (Sec. 5.4.1)
3	VERIFY that the RESPONSE VALUE for essLatitude.0 is less than or equal to 90000001.	Pass / Fail (Sec. 5.4.1)
4	VERIFY that the RESPONSE VALUE for essLatitude.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.1)
5	VERIFY that the RESPONSE VALUE for essLongitude.0 is greater than or equal to -180000000.	Pass / Fail (Sec. 5.4.2)
6	VERIFY that the RESPONSE VALUE for essLongitude.0 is less than or equal to 180000001.	Pass / Fail (Sec. 5.4.2)
7	VERIFY that the RESPONSE VALUE for essLongitude.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.2)
8	VERIFY that the RESPONSE VALUE for essVehicleSpeed.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.4.3)
9	VERIFY that the RESPONSE VALUE for essVehicleSpeed.0 is less than or equal to 255.	Pass / Fail (Sec. 5.4.3)
10	VERIFY that the RESPONSE VALUE for essVehicleSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.3)
11	VERIFY that the RESPONSE VALUE for essVehicleBearing.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.4.4)
12	VERIFY that the RESPONSE VALUE for essVehicleBearing.0 is less than or equal to 361.	Pass / Fail (Sec. 5.4.4)
13	VERIFY that the RESPONSE VALUE for essVehicleBearing.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.4)
14	VERIFY that the RESPONSE VALUE for essOdometer.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.4.5)
15	VERIFY that the RESPONSE VALUE for essOdometer.0 is less than or equal to	Pass / Fail

	4294967295.	(Sec. 5.4.5)
16	VERIFY that the RESPONSE VALUE for essOdometer.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.5)
17	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is greater than or equal to -400.	Pass / Fail (Sec. 5.5.1)
18	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is less than or equal to 8001.	Pass / Fail (Sec. 5.5.1)
19	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.1.7 Retrieve Compressed Mobile Station Data

Test Case: 1.7	***Title:***	*Retrieve Compressed Mobile Station Data*	
	Description:	*This test case verifies that the ESS allows a management station to determine the location of, speed of, and pavement treatment being applied by the mobile platform*	
	Variables:	*Weather_Profile_Supported*	*PRL 3.5.2.3.1*
		PTS_Supported	*PRL 2.5.3.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the device supports retrieving mobile weather profiles (PRL 3.5.2.3.1). RECORD this information as: »Weather_Profile_Supported	
2	CONFIGURE: Determine whether the device supports pavement treatment operations (PRL 2.5.3.2). RECORD this information as: »PTS_Supported	
3	GET the following object(s): »essMobileBlock.0	Pass / Fail (Sec. 3.5.1.3.3)
4	Decode the essMobileBlock.0 structure.	
5	VERIFY that the essMobileBlock.0 structure was decoded without error.	Pass / Fail (Sec. 5.3.7)
6	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essLatitude.0 field.	Pass / Fail (Sec. 3.5.1.3.1)
7	VERIFY that the essLatitude.0 field has a value between -90000000 and 90000001, inclusive.	Pass / Fail (Sec. 5.4.1)
8	VERIFY that the essLatitude.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.4.1)
9	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essLongitude.0 field.	Pass / Fail (Sec. 3.5.1.3.1)
10	VERIFY that the essLongitude.0 field has a value between -180000000 and 180000001, inclusive.	Pass / Fail (Sec. 5.4.2)
11	VERIFY that the essLongitude.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.4.2)
12	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essReferenceHeight.0 field.	Pass / Fail (Sec. 3.5.1.3.1)
13	VERIFY that the essReferenceHeight.0 field has a value between -400 and 8001, inclusive.	Pass / Fail (Sec. 5.5.1)
14	VERIFY that the essReferenceHeight.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.5.1)
15	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essVehicleSpeed.0 field.	Pass / Fail (Sec. 3.5.1.3.1)

16	VERIFY that the essVehicleSpeed.0 field has a value between 0 and 255, inclusive.	Pass / Fail (Sec. 5.4.3)
17	VERIFY that the essVehicleSpeed.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.4.3)
18	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essVehicleBearing.0 field.	Pass / Fail (Sec. 3.5.1.3.1)
19	VERIFY that the essVehicleBearing.0 field has a value between 0 and 361, inclusive.	Pass / Fail (Sec. 5.4.4)
20	VERIFY that the essVehicleBearing.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.4.4)
21	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essVehicleOdometer.0 field.	Pass / Fail (Sec. 3.5.1.3.1)
22	VERIFY that the essVehicleOdometer.0 field has a value between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.4.5)
23	VERIFY that the essVehicleOdometer.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.4.5)
24	IF Weather_Profile_Supported is equal to true, then proceed to Step 24.1; otherwise, proceed to Step 25.	
24.1	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essMobileFriction.0 field.	Pass / Fail (Sec. 3.5.2.3.1)
24.2	VERIFY that the essMobileFriction.0 field has a value between 0 and 101, inclusive.	Pass / Fail (Sec. 5.12.1)
24.3	VERIFY that the essMobileFriction.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.12.1)
24.4	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essMobileObservationGroundState.0 field.	Pass / Fail (Sec. 3.5.2.3.1)
24.5	VERIFY that the essMobileObservationGroundState.0 field has a value between 1 and 18, inclusive.	Pass / Fail (Sec. 5.12.2)
24.6	VERIFY that the essMobileObservationGroundState.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.12.2)
24.7	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essMobileObservationPavement.0 field.	Pass / Fail (Sec. 3.5.2.3.1)
24.8	VERIFY that the essMobileObservationPavement.0 field has a value between 1 and 25, inclusive.	Pass / Fail (Sec. 5.12.3)
24.9	VERIFY that the essMobileObservationPavement.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.12.3)
25	IF PTS_Supported is equal to true, then proceed to Step 25.1; otherwise, proceed to EXIT.	
25.1	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essPaveTreatmentAmount.0 field.	Pass / Fail (Sec. 3.5.3.1.4)

25.2	VERIFY that the essTreatmentAmount.0 field has a value between 0 and 255, inclusive.	Pass / Fail (Sec. 5.13.4)
25.3	VERIFY that the essTreatmentAmount.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.13.4)
25.4	VERIFY that the RESPONSE VALUE for essMobileBlock.0 contains an essPaveTreatmentWidth.0 field.	Pass / Fail (Sec. 3.5.3.1.4)
25.5	VERIFY that the essPaveTreatmentWidth.0 field has a value between 0 and 255, inclusive.	Pass / Fail (Sec. 5.13.5)
25.6	VERIFY that the essPaveTreatmentWidth.0 field is APPROPRIATE.	Pass / Fail (Sec. 5.13.5)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2 Weather Meta-Data Tests

C.2.3.2.1 Retrieve Atmospheric Pressure Height

Test Case: 2.1	***Title:***	*Retrieve Atmospheric Pressure Height*
	Description:	*This test case verifies that the ESS allows a management station to determine the relative height of the atmospheric pressure sensor*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essPressureHeight.0	Pass / Fail (Sec. 3.5.2.1.1)
2	VERIFY that the RESPONSE VALUE for essPressureHeight.0 is greater than or equal to -1000.	Pass / Fail (Sec. 5.5.2)
3	VERIFY that the RESPONSE VALUE for essPressureHeight.0 is less than or equal to 1001.	Pass / Fail (Sec. 5.5.2)
4	VERIFY that the RESPONSE VALUE for essPressureHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.2 Retrieve Meta-Data for Each Wind Sensor

Test Case: 2.2	***Title:***	*Retrieve Meta-Data for Each Wind Sensor*
	Description:	*This test case verifies that the ESS allows a management station to determine the location and relative height of each wind sensor*
	Variables:	*Required_Wind_Sensors* — *PRL 3.6.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of wind sensors that the ESS is required to support (PRL 3.6.2). RECORD this information as: »Required_Wind_Sensors	
2	GET the following object(s): »windSensorTableNumSensors.0	Pass / Fail (Sec. 3.5.2.1.2)
3	VERIFY that the RESPONSE VALUE for windSensorTableNumSensors.0 is greater than or equal to Required_Wind_Sensors.	Pass / Fail (Sec. 3.6.2)
4	Determine the RESPONSE VALUE for windSensorTableNumSensors.0. RECORD this information as: »Supported_Wind_Sensors	
5	FOR EACH value, N, from 1 to Supported_Wind_Sensors, perform Steps 5.1 through 5.6.	
5.1	GET the following object(s): »windSensorHeight.N »windSensorLocation.N	Pass / Fail (Sec. 3.5.2.1.2)
5.2	VERIFY that the RESPONSE VALUE for windSensorHeight.N is greater than or equal to -1000.	Pass / Fail (Sec. 5.6.10.2)
5.3	VERIFY that the RESPONSE VALUE for windSensorHeight.N is less than or equal to 1001.	Pass / Fail (Sec. 5.6.10.2)
5.4	VERIFY that the RESPONSE VALUE for windSensorHeight.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.2)
5.5	VERIFY that the RESPONSE VALUE for windSensorLocation.N is a valid DisplayString with no more than 255 characters.	Pass / Fail (Sec. 5.6.10.3)
5.6	VERIFY that the RESPONSE VALUE for windSensorLocation.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.3 Retrieve Temperature Sensor Meta-Data

Test Case: 2.3	***Title:***	*Retrieve Temperature Sensor Meta-Data*	
	Description:	*This test case verifies that the ESS allows a management station to determine the relative height of each temperature sensor*	
	Variables:	*Required_Temp_Sensors*	*PRL 3.6.3*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of temperature sensors required by the specification (PRL 3.6.3). RECORD this information as: »Required_Temp_Sensors	
2	GET the following object(s): »essNumTemperatureSensors.0	Pass / Fail (Sec. 3.5.2.1.3)
3	VERIFY that the RESPONSE VALUE for essNumTemperatureSensors.0 is greater than or equal to Required_Temp_Sensors.	Pass / Fail (Sec. 3.6.3)
4	Determine the RESPONSE VALUE for essNumTemperatureSensors.0. RECORD this information as: »Supported_Temp_Sensors	
5	FOR EACH value, N, from 1 to Supported_Temp_Sensors, perform Steps 5.1 through 5.4.	
5.1	GET the following object(s): »essTemperatureSensorHeight.N	Pass / Fail (Sec. 3.5.2.1.3)
5.2	VERIFY that the RESPONSE VALUE for essTemperatureSensorHeight.N is greater than or equal to -1000.	Pass / Fail (Sec. 5.7.3.2)
5.3	VERIFY that the RESPONSE VALUE for essTemperatureSensorHeight.N is less than or equal to 1001.	Pass / Fail (Sec. 5.7.3.2)
5.4	VERIFY that the RESPONSE VALUE for essTemperatureSensorHeight.N is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.4 Retrieve Pavement Sensor Meta-Data

Test Case: 2.4	***Title:***	*Retrieve Pavement Sensor Meta-Data*
	Description:	*This test case verifies that the ESS allows a management station to determine available meta-data for each pavement sensor.*
	Variables:	*Required_Pavement_Sensors* — *PRL 3.6.8*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	GET the following object(s): »numEssPavementSensors.0	Pass / Fail (Sec. 3.5.2.1.4)
3	VERIFY that the RESPONSE VALUE for numEssPavementSensors.0 is greater than or equal to Required_Pavement_Sensors.	Pass / Fail (Sec. 3.6.8)
4	Determine the RESPONSE VALUE for numEssPavementSensors.0. RECORD this information as: »Supported_Pavement_Sensors	
5	FOR EACH value, N, from 1 to Supported_Pavement_Sensors, perform Steps 5.1 through 5.15.	
5.1	GET the following object(s): »essPavementSensorLocation.N »essPavementType.N »essPavementElevation.N »essPavementExposure.N »essPavementSensorType.N	Pass / Fail (Sec. 3.5.2.1.4)
5.2	VERIFY that the RESPONSE VALUE for essPavementSensorLocation.N is a valid DisplayString with no more than 255 characters.	Pass / Fail (Sec. 5.11.3.2)
5.3	VERIFY that the RESPONSE VALUE for essPavementSensorLocation.N is APPROPRIATE.	
5.4	VERIFY that the RESPONSE VALUE for essPavementType.N is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.3)
5.5	VERIFY that the RESPONSE VALUE for essPavementType.N is less than or equal to 9.	Pass / Fail (Sec. 5.11.3.3)
5.6	VERIFY that the RESPONSE VALUE for essPavementType.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.3)
5.7	VERIFY that the RESPONSE VALUE for essPavementElevation.N is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.4)
5.8	VERIFY that the RESPONSE VALUE for essPavementElevation.N is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.4)
5.9	VERIFY that the RESPONSE VALUE for essPavementElevation.N is APPROPRIATE.	Pass / Fail

5.10	VERIFY that the RESPONSE VALUE for essPavementExposure.N is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.5)
5.11	VERIFY that the RESPONSE VALUE for essPavementExposure.N is less than or equal to 101.	Pass / Fail (Sec. 5.11.3.5)
5.12	VERIFY that the RESPONSE VALUE for essPavementExposure.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.5)
5.13	VERIFY that the RESPONSE VALUE for essPavementSensorType.N is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.6)
5.14	VERIFY that the RESPONSE VALUE for essPavementSensorType.N is less than or equal to 8.	Pass / Fail (Sec. 5.11.3.6)
5.15	VERIFY that the RESPONSE VALUE for essPavementSensorType.N is	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.5 Retrieve Subsurface Sensor Meta-Data

Test Case: 2.5	***Title:***	*Retrieve Subsurface Sensor Meta-Data*
	Description:	*This test case verifies that the ESS allows a management station to determine available meta-data for each subsurface sensor.*
	Variables:	*Required_Subsurface_Sensors* — *3.6.11*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of subsurface sensors required by the specification (3.6.11). RECORD this information as: »Required_Subsurface_Sensors	
2	GET the following object(s): »numEssSubSurfaceSensors.0	Pass / Fail (Sec. 3.5.2.1.5)
3	VERIFY that the RESPONSE VALUE for numEssSubSurfaceSensors.0 is greater than or equal to Required_Subsurface_Sensors.	Pass / Fail (Sec. 3.6.11)
4	Determine the RESPONSE VALUE for numEssSubSurfaceSensors.0. RECORD this information as: »Supported_Subsurface_Sensors	
5	FOR EACH value, N, from 1 to Supported_Subsurface_Sensors, perform Steps 5.1 through 5.9.	
5.1	GET the following object(s): »essSubSurfaceSensorLocation.N »essSubSurfaceType.N »essSubSurfaceSensorDepth.N	Pass / Fail (Sec. 3.5.2.1.5)
5.2	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorLocation.N is a valid DisplayString with no more than 255 characters.	Pass / Fail (Sec. 5.11.6.2)
5.3	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorLocation.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.2)
5.4	VERIFY that the RESPONSE VALUE for essSubSurfaceType.N is greater than or equal to 1.	Pass / Fail (Sec. 5.11.6.3)
5.5	VERIFY that the RESPONSE VALUE for essSubSurfaceType.N is less than or equal to 12.	Pass / Fail (Sec. 5.11.6.3)
5.6	VERIFY that the RESPONSE VALUE for essSubSurfaceType.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.3)
5.7	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorDepth.N is greater than or equal to 0.	Pass / Fail (Sec. 5.11.6.4)
5.8	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorDepth.N is less than or equal to 1001.	Pass / Fail (Sec. 5.11.6.4)
5.9	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorDepth.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.6 Configure Pavement Sensor

Test Case: 2.6	***Title:***	*Configure Pavement Sensor*	
	Description:	*This test case verifies that the ESS allows a management station to store configuration information for a specified pavement sensor.*	
	Variables:	*Required_Pavement_Sensors*	*PRL 3.6.8*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	Determine a random value between 1 and Required_Pavement_Senosrs. RECORD this information as: »Subject_Pavement_Sensor	
3	Determine a random DisplayString with a length between 1 and 255 characters. RECORD this information as: »Pavement_Sensor_Location	
4	Determine a random number between 1 and 9. RECORD this information as: »Pavement_Type	
5	Determine a random numbber between 0 and 101. RECORD this information as: »Pavement_Exposure	
6	GET the following object(s): »essPavementSensorLocation.Subject_Pavement_Sensor »essPavementType.Subject_Pavement_Sensor »essPavementExposure.Subject_Pavement_Sensor	Pass / Fail (RFC 1157)
7	Determine the RESPONSE VALUE for essPavementSensorLocation.Subject_Pavement_Sensor. RECORD this information as: »Orig_Pavement_Sensor_Location	
8	Determine the RESPONSE VALUE for essPavementType.Subject_Pavement_Sensor. RECORD this information as: »Orig_Pavement_Type	
9	Determine the RESPONSE VALUE for essPavementExposure.Subject_Pavement_Sensor. RECORD this information as: »Orig_Pavement_Exposure	
10	SET the following object(s) to the value(s) shown: »essPavementSensorLocation.Subject_Pavement_Sensor = Pavement_Sensor_Location »essPavementType.Subject_Pavement_Sensor = Pavement_Type »essPavementExposure.Subject_Pavement_Sensor = Pavement_Exposure	Pass / Fail (Sec. 3.5.2.1.6)
11	GET the following object(s): »essPavementSensorLocation.Subject_Pavement_Sensor »essPavementType.Subject_Pavement_Sensor »essPavementExposure.Subject_Pavement_Sensor	Pass / Fail (RFC 1157)
12	VERIFY that the RESPONSE VALUE for essPavementSensorLocation.Subject_Pavement_Sensor is equal to Pavement_Sensor_Location.	Pass / Fail (Sec. 5.11.3.2)

13	VERIFY that the RESPONSE VALUE for essPavementType.Subject_Pavement_Sensor is equal to Pavement_Type.	Pass / Fail (Sec. 5.11.3.3)
14	VERIFY that the RESPONSE VALUE for essPavementExposure.Subject_Pavement_Sensor is equal to Pavement_Exposure.	Pass / Fail (Sec. 5.11.3.5)
151	SET the following object(s) to the value(s) shown: »essPavementSensorLocation.Subject_Pavement_Sensor = Orig_Pavement_Sensor_Location »essPavementType.Subject_Pavement_Sensor = Orig_Pavement_Type »essPavementExposure.Subject_Pavement_Sensor = Orig_Pavement_Exposure	Pass / Fail (Sec. 3.5.2.1.6)
16	GET the following object(s): »essPavementSensorLocation.Subject_Pavement_Sensor »essPavementType.Subject_Pavement_Sensor »essPavementExposure.Subject_Pavement_Sensor	Pass / Fail (RFC 1157)
17	VERIFY that the RESPONSE VALUE for essPavementSensorLocation.Subject_Pavement_Sensor is equal to Orig_Pavement_Sensor_Location.	Pass / Fail (Sec. 5.11.3.2)
18	VERIFY that the RESPONSE VALUE for essPavementType.Subject_Pavement_Sensor is equal to Orig_Pavement_Type.	Pass / Fail (Sec. 5.11.3.3)
19	VERIFY that the RESPONSE VALUE for essPavementExposure.Subject_Pavement_Sensor is equal to Orig_Pavement_Exposure.	Pass / Fail (Sec. 5.11.3.5)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.7 Configure Subsurface Sensor

Test Case: 2.7	***Title:***	*Configure Subsurface Sensor*	
	Description:	*This test case verifies that the ESS allows a management station to store configuration information for a specified subsurface sensor.*	
	Variables:	*Required_Subsurface_Sensors*	*PRL 3.6.11*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of subsurface sensors as required by the specification (PRL 3.6.11). RECORD this information as: »Required_Subsurface_Sensors	
2	Determine a random value between 1 and Required_Subsurface_Sensors. RECORD this information as: »Subject_Subsurface_Sensor	
3	Determine a random DisplayString with a length between 1 and 255 characters. RECORD this information as: »Subsurface_Sensor_Location	
4	Determine a random number between 1 and 12. RECORD this information as: »Subsurface_Type	
5	Determine a random value between 0 and 1001. RECORD this information as: »Subsurface_Depth	
6	GET the following object(s): »essSubSurfaceSensorLocation.Subject_Subsurface_Sensor »essSubSurfaceType.Subject_Subsurface_Sensor »essSubSurfaceSensorDepth.Subject_Subsurface_Sensor	Pass / Fail (RFC 1157)
7	Determine the retrieved information. RECORD this information as: »Orig_Subsurface_Sensor_Location (the original value of the subsurface sensor location object) »Orig_Subsurface_Type (the original value of the subsurface type object) »Orig_Subsurface_Depth (the original value of the subsurface depth object)	
8	SET the following object(s) to the value(s) shown: »essSubSurfaceSensorLocation.Subject_Subsurface_Sensor = Subsurface_Sensor_Location »essSubSurfaceType.Subject_Subsurface_Sensor = Subsurface_Type »essSubSurfaceSensorDepth.Subject_Subsurface_Sensor = Subsurface_Depth	Pass / Fail (Sec. 3.5.2.1.7)
9	GET the following object(s): »essSubSurfaceSensorLocation.Subject_Subsurface_Sensor »essSubSurfaceType.Subject_Subsurface_Sensor »essSubSurfaceSensorDepth.Subject_Subsurface_Sensor	Pass / Fail (RFC 1157)
10	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorLocation.Subject_Subsurface_Sensor is equal to Subsurface_Sensor_Location.	Pass / Fail (Sec. 5.11.6.2)
11	VERIFY that the RESPONSE VALUE for	Pass / Fail

12	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorDepth.Subject_Subsurface_Sensor is equal to Subsurface_Depth.	Pass / Fail (Sec. 5.11.6.4)
913	SET the following object(s) to the value(s) shown: »essSubSurfaceSensorLocation.Subject_Subsurface_Sensor = Orig_Subsurface_Sensor_Location »essSubSurfaceType.Subject_Subsurface_Sensor = Orig_Subsurface_Type »essSubSurfaceSensorDepth.Subject_Subsurface_Sensor = Orig_Subsurface_Depth	Pass / Fail (Sec. 3.5.2.1.7)
14	GET the following object(s): »essSubSurfaceSensorLocation.Subject_Subsurface_Sensor »essSubSurfaceType.Subject_Subsurface_Sensor »essSubSurfaceSensorDepth.Subject_Subsurface_Sensor	Pass / Fail (RFC 1157)
15	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorLocation.Subject_Subsurface_Sensor is equal to Orig_Subsurface_Sensor_Location.	Pass / Fail (Sec. 5.11.6.2)
16	VERIFY that the RESPONSE VALUE for essSubSurfaceType.Subject_Subsurface_Sensor is equal to Orig_Subsurface_Type.	Pass / Fail (Sec. 5.11.6.3)
17	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorDepth.Subject_Subsurface_Sensor is equal to	Pass / Fail (Sec. 5.11.6.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.8 Configure Passive Ice Detection Logic

Test Case: 2.8	***Title:***	*Configure Passive Ice Detection Logic*
	Description:	*This test case verifies that the ESS allows a management station to store configuration information regarding the pavement treatments being applied.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »numEssTreatments.0	Pass / Fail (RFC 1157)
2	Determine the RESPONSE VALUE for numEssTreatments. RECORD this information as: »Num_Treatments	
3	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 3.1 through 3.2.	
3.1	GET the following object(s): »essPaveTreatProductType.N »essPaveTreatProductForm.N »essPercentProductMix.N	Pass / Fail (RFC 1157)
3.2	Determine the values for the retrieved information. RECORD this information as: »Product_Type[N] (the type of the Nth product read from the device) »Product_Form[N] (the form of the Nth product read from the device) »Product_Percent[N] (the percentage of the Nth product read from the device)	
4	GET the following object(s): »numEssTreatments.0	Pass / Fail (Sec. 3.5.2.1.8)
5	Determine the RESPONSE VALUE for numEssTreatments.0. RECORD this information as: »Num_Treatments	
6	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 6.1 through 6.3.	
6.1	Determine a random value from 1 to 14. RECORD this information as: »Treatment_Type	
6.2	Determine a random value from 1 to 4. RECORD this information as: »Treatment_Form	
6.3	SET the following object(s) to the value(s) shown: »essPaveTreatProductType.N = Treatment_Type »essPaveTreatProductForm.N = Treatment_Form	Pass / Fail (Sec. 3.5.2.1.8)
7	Determine the total percentage of products (i.e., 100). RECORD this information as: »Percentage_Available	
8	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 8.1 through 8.3.	
8.1	IF N is equal to Num_Treatments, then proceed to Step 8.1.1; otherwise, proceed to Step 8.2.1.	

8.1.1	Determine Percentage_Available. RECORD this information as: »Treatment_Percent GO TO Step 8.3.	
8.2.1	Determine a random value between 0 and Percentage_Available. RECORD this information as: »Treatment_Percent	
8.2.2	Determine the remaining percentage available by subtracting Product_Percent from Percentage_Available. RECORD this information as: »Percentage_Available	
8.3	SET the following object(s) to the value(s) shown: »essPercentProductMix.N = Percentage_Available	Pass / Fail (Sec. 3.5.2.1.8)
9	GET the following object(s): »numEssTreatments.0	Pass / Fail (Sec. 3.5.2.1.8)
10	FOR EACH value, N, from 1 to Num_Treatments, perform Step 10.1.	
10.1	SET the following object(s) to the value(s) shown: »essPaveTreatProductType.N = Product_Type[N] »essPaveTreatProductForm.N = Product_Form[N]	Pass / Fail (Sec. 3.5.2.1.8)
11	FOR EACH value, N, from 1 to Num_Treatments, perform Step 11.1.	
11.1	SET the following object(s) to the value(s) shown: »essPercentProductMix.N = Product_Percent[N]	Pass / Fail (Sec. 3.5.2.1.8)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.2.9 Configure Snapshot Camera

Test Case: 2.9	***Title:***	*Configure Snapshot Camera*	
	Description:	*This test case verifies that the ESS allows a management station to store the storage location for newly taken snapshot images.*	
	Variables:	*Required_Cameras*	*PRL 3.6.20*
		Camera_Number_Supported	*PRL 3.6.23*
		Sequence_Numbers_Supported	*PRL 3.6.24*
		Date_Supported	*PRL 3.6.25*
		Time_Supported	*PRL 3.6.26*
		Long_Filenames_Supported	*PRL 3.6.27*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of cameras required by the specification (PRL 3.6.20). RECORD this information as: »Required_Cameras	
2	CONFIGURE: Determine whether the ESS allows the filename to include a camera number field (PRL 3.6.23). RECORD this information as: »Camera_Number_Supported	
3	CONFIGURE: Determine whether the ESS allows the filenames to include sequence numbers (PRL 3.6.24). RECORD this information as: »Sequence_Numbers_Supported	
4	CONFIGURE: Determine whether the camera allows a date to be included in the filename (PRL 3.6.25). RECORD this information as: »Date_Supported	
5	CONFIGURE: Determine whether the ESS allows the filenames to include a time field (PRL 3.6.26). RECORD this information as: »Time_Supported	
6	CONFIGURE: Determine whether the ESS allows the filenames to exceed eight characters in length (PRL 3.6.27). RECORD this information as: »Long_Filenames_Supported	
7	GET the following object(s): »essSnapshotNumberOfCameras.0	Pass / Fail (Sec. 3.5.2.3.9)
8	VERIFY that the RESPONSE VALUE for essSnapshotNumberOfCameras.0 is greater than or equal to Required_Cameras.	Pass / Fail (Sec. 3.6.20)
9	Determine the RESPONSE VALUE for essSnapshotNumberOfCameras.0. RECORD this information as: »Num_Cameras	
10	Determine a random number between 1 and Required_Cameras. RECORD this information as: »Subject_Camera	
11	FOR EACH value, N, from 1 to Num_Cameras, perform Steps 11.1 through 11.4.	
11.1	GET the following object(s): »essSnapshotCameraDescription.N	Pass / Fail (Sec. 3.5.2.3.9)

	»essSnapshotCameraStoragePath.N	
11.2	IF N is equal to Subject_Camera, then proceed to Step 11.2.1; otherwise, proceed to Step 11.3.	
11.2.1	Determine the RESPONSE VALUE for essSnapshotCameraDescription.N. RECORD this information as: »Orig_Description	
11.3	GET the following object(s): »essSnapshotCameraFilename.N	Pass / Fail (Sec. 3.5.2.3.9)
11.4	IF N is equal to Subject_Camera, then proceed to Step 11.4.1; otherwise, proceed to Step 12.	
11.4.1	Determine the RESPONSE VALUE for essSnapshotCameraFilename.N. RECORD this information as: »Orig_Filename	
12	Determine a random string of 3 characters. RECORD this information as: »Base_Filename	
13	Determine a random DisplayString containing between 1 and 255 characters. RECORD this information as: »Camera_Location	
14	SET the following object(s) to the value(s) shown: »essSnapshotCameraDescription.Subject_Camera = Camera_Location	Pass / Fail (Sec. 3.5.2.1.9)
15	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = Base_Filename	Pass / Fail (Sec. 3.5.2.1.9)
16	PERFORM the test case labeled 'Retrieve Snapshot' (2.3.7.2).	Pass / Fail
17	IF Camera_Number_Supported is equal to 1, then proceed to Step 17.1; otherwise, proceed to Step 18.	
17.1	SET the following object(s) to the value(s) shown: »essSnapshotCameraDescription.Subject_Camera = Camera_Location	Pass / Fail (Sec. 3.5.2.1.9)
17.2	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = '.jpg'	Pass / Fail (Sec. 3.5.2.1.9)
17.3	PERFORM the test case labeled 'Retrieve Snapshot' (2.3.7.2).	
18	IF Sequence_Numbers_Supported is equal to 1, then proceed to Step 18.1; otherwise, proceed to Step 19.	
18.1	SET the following object(s) to the value(s) shown: »essSnapshotCameraDescription.Subject_Camera = Camera_Location	Pass / Fail (Sec. 3.5.2.1.9)
18.2	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = '.jpg'	Pass / Fail (Sec. 3.5.2.1.9)
18.3	PERFORM the test case labeled 'Retrieve Snapshot' (2.3.7.2).	
19	IF Date_Supported is equal to 1, then proceed to Step 19.1; otherwise, proceed to Step 20.	
19.1	SET the following object(s) to the value(s) shown:	Pass / Fail

	»essSnapshotCameraDescription.Subject_Camera = Camera_Location	(Sec. 3.5.2.1.9)
19.2	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = '.jpg'	Pass / Fail (Sec. 3.5.2.1.9)
19.3	PERFORM the test case labeled 'Retrieve Snapshot' (2.3.7.2).	
20	IF Time_Supported is equal to 1, then proceed to Step 20.1; otherwise, proceed to Step 21.	
20.1	SET the following object(s) to the value(s) shown: »essSnapshotCameraDescription.Subject_Camera = Camera_Location	Pass / Fail (Sec. 3.5.2.1.9)
20.2	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = '.jpg'	Pass / Fail (Sec. 3.5.2.1.9)
20.3	PERFORM the test case labeled 'Retrieve Snapshot' (2.3.7.2).	
21	IF Long_Filenames_Supported is equal to 1, then proceed to Step 21.1; otherwise, proceed to Step 22.	
21.1	SET the following object(s) to the value(s) shown: »essSnapshotCameraDescription.Subject_Camera = Camera_Location	Pass / Fail (Sec. 3.5.2.1.9)
21.2	CONFIGURE: Determine a random string of 25 characters pre-pended to the string ".jpg". RECORD this information as: »Long_Filename	
21.3	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = Long_Filename	Pass / Fail (Sec. 3.5.2.1.9)
21.4	PERFORM the test case labeled 'Retrieve Snapshot' (2.3.7.2).	
22	SET the following object(s) to the value(s) shown: »essSnapshotCameraDescription.Subject_Camera = Orig_Description	Pass / Fail (Sec. 3.5.2.1.9)
23	SET the following object(s) to the value(s) shown: »essSnapshotCameraFilename.Subject_Camera = Orig_Filename	Pass / Fail (Sec. 3.5.2.1.9)
24	GET the following object(s): »essSnapshotCameraDescription.Subject_Camera	Pass / Fail (RFC 1157)
25	VERIFY that the RESPONSE VALUE for essSnapshotCameraDescription.Subject_Camera is equal to Orig_Description.	Pass / Fail (Sec. 3.5.2.1.9)
26	GET the following object(s): »essSnapshotCameraFilename.Subject_Camera	Pass / Fail (RFC 1157)
27	VERIFY that the RESPONSE VALUE for essSnapshotCameraFilename.Subject_Camera is equal to Orig_Filename.	Pass / Fail (Sec. 3.5.2.1.9)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3 Weather Data Tests

C.2.3.3.1 Retrieve Weather Profile with Mobile Sources

Test Case: 3.1	***Title:***	*Retrieve Weather Profile with Mobile Sources*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve a list of records recorded by the ESS over a period of time.*	
	Variables:	*Pressure_Supported*	*PRL 2.5.2.1.1*
		Wind_Supported	*PRL 2.5.2.1.2*
		Required_Wind_Sensors	*PRL 3.6.2*
		Temperature_Supported	*PRL 2.5.2.1.3*
		Required_Temp_Sensors	*PRL 3.6.3*
		Humidity_Supported	*PRL 2.5.2.1.4*
		Precip_Supported	*PRL 2.5.2.1.5*
		Required_Water_Level_Sensors	*PRL 3.6.22*
		Adjacent_Snow_Depth_Supported	*PRL 3.5.2.3.3.4*
		Roadway_Snow_Depth_Supported	*PRL 3.5.2.3.3.5*
		Ice_Thickness_Supported	*PRL 3.5.2.3.3.6*
		Precip_Rates_Supported	*PRL 3.5.2.3.2.6.2*
		Situation_Supported	*PRL 2.5.2.4*
		Visibility_Supported	*PRL 2.5.2.1.7*
		Radiation_Supported	*PRL 2.5.2.1.6*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the ESS is required to monitor atmospheric pressure (PRL 2.5.2.1.1). RECORD this information as: »Pressure_Supported	
2	CONFIGURE: Determine the number of wind sensors supported. RECORD this information as: »Wind_Supported (whether the ESS is required to monitor winds—PRL 2.5.2.1.2) »Required_Wind_Sensors (the number of wind sensors that the ESS is required to support - PRL 3.6.2)	
3	CONFIGURE: Determine the number of temperature sensors supported. RECORD this information as: »Temperature_Supported (whether the ESS is required to monitor temperature—PRL 2.5.2.1.3) »Required_Temp_Sensors (the number of temperature sensors that the ESS is required to support—PRL 3.6.3) »Humidity_Supported (whether the ESS is required to monitor humidity—PRL 2.5.2.1.4)	

4	CONFIGURE: Determine the number of precipitation sensors supported. RECORD this information as: »Precip_Supported (whether the ESS is required to monitor precipitation—PRL 2.5.2.1.5) »Required_Water_Level_Sensors (the number of water level sensors that the ESS is required to support—PRL 3.6.22) »Adjacent_Snow_Depth_Supported (whether the ESS is required to monitor the adjacent snow depth—PRL 3.5.2.3.3.4) »Roadway_Snow_Depth_Supported (whether the ESS is required to monitor the roadway snow depth—PRL 3.5.2.3.3.5) »Ice_Thickness_Supported (whether the ESS is required to monitor the snow pack depth and ice thickness - PRL 3.5.2.3.3.6) »Precip_Rates_Supported (whether the ESS is required to monitor the precipitation rates—PRL 3.5.2.3.2.6.2)	
5	CONFIGURE: Determine whether the ESS is required to report situation data (PRL 2.5.2.4). RECORD this information as: »Situation_Supported	
6	CONFIGURE: Determine whether the ESS is required to provide visibility capabilities (PRL 2.5.2.1.7). RECORD this information as: »Visibility_Supported	
7	CONFIGURE: Determine whether the ESS is required to monitor solar radiation (PRL 2.5.2.1.6). RECORD this information as: »Radiation_Supported	
8	GET the following object(s): »essWeatherV3Block.0 »essLatitude.0 »essLongitude.0 »essVehicleSpeed.0 »essVehicleBearing.0 »essOdometer.0 »essReferenceHeight.0 »essMobileFriction.0 »essMobileObservationGroundState.0 »essMobileObservationPavement.0	Pass / Fail (Sec. 3.5.2.3.1)
9	Decode the essWeatherV3Block.0 structure.	
10	VERIFY that the essWeatherV3Block.0 structure was decoded without error.	Pass / Fail (Sec. 5.3.9)
11	IF Pressure_Supported is equal to true, then proceed to Step 11.1; otherwise, proceed to Step 12.	
11.1	VERIFY that the RESPONSE VALUE for the essAtmosphericPressure.0 field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.1)
11.2	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.5.4)
11.3	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.4)
12	IF Wind_Supported is equal to true, then proceed to Step 12.1; otherwise, proceed to Step 13.	
12.1	VERIFY that the RESPONSE VALUE for the essWindData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.2)

12.2	VERIFY that the essWindData field contains at least Required_Wind_Sensors entries.	Pass / Fail (Sec. 3.6.2)
12.3	FOR EACH value, N, from 1 to Required_Wind_Sensors, perform Steps 12.3.1 through 12.3.21.	
12.3.1	VERIFY that the windSensorIndex.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.2	VERIFY that the RESPONSE VALUE for windSensorIndex.x in the Nth essWindDataV3 structure is equal to N.	Pass / Fail (Sec. 5.6.10.1)
12.3.3	VERIFY that the windSensorAvgSpeed.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.4	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.x in the Nth essWindDataV3 structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.10.2)
12.3.5	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.2)
12.3.6	VERIFY that the windSensorAvgDirection.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.7	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.x in the Nth essWindDataV3 structure is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.10.5)
12.3.8	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.5)
12.3.9	VERIFY that the windSensorSpotSpeed.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.10	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.x in the Nth essWindDataV3 structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.10.6)
12.3.11	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.6)
12.3.12	VERIFY that the windSensorSpotDirection.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.13	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.x in the Nth essWindDataV3 structure is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.10.7)
12.3.14	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.7)
12.3.15	VERIFY that the windSensorGustSpeed.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.16	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.x in the Nth essWindDataV3 structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.10.8)
12.3.17	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.8)
12.3.18	VERIFY that the windSensorGustDirection.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)

12.3.19	VERIFY that the RESPONSE VALUE for windSensorGustDirection.x in the Nth essWindDataV3 structure is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.10.9)
12.3.20	VERIFY that the RESPONSE VALUE for windSensorGustDirection.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.9)
12.3.21	IF Situation_Supported is equal to true, then proceed to Step 12.3.21.1; otherwise, proceed to Step 13.	
12.3.21.1	VERIFY that the windSensorSituation.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.5.1)
12.3.21.2	VERIFY that the RESPONSE VALUE for windSensorSituation.x in the Nth essWindDataV3 structure is between 1 and 12, inclusive.	Pass / Fail (Sec. 5.6.10.10)
12.3.21.3	VERIFY that the RESPONSE VALUE for windSensorSituation.x is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.10)
13	IF Temperature_Supported is equal to true, then proceed to Step 13.1; otherwise, proceed to Step 14.	
13.1	VERIFY that the essTemperatureData field of the essWeatherV3Block object is present.	Pass / Fail (Sec. 3.5.2.3.2.3)
13.2	VERIFY that the RESPONSE VALUE for the essMaxTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.4)
13.3	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.6)
13.4	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.6)
13.5	Determine the maximum temperature reading reported by the device. RECORD this information as: »Max_Temp	
13.6	VERIFY that the RESPONSE VALUE for the essMinTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.4)
13.7	VERIFY that the RESPONSE VALUE for essMinTemp.0 is between -1000 and Max_Temp, inclusive.	Pass / Fail (Sec. 5.7.7)
13.8	VERIFY that the RESPONSE VALUE for essMinTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.7)
13.9	IF Humidity_Supported is equal to 1, then proceed to Step 13.9.1; otherwise, proceed to Step 13.10.	
13.9.1	VERIFY that the RESPONSE VALUE for the essWetBulbTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.5)
13.9.2	VERIFY that the RESPONSE VALUE for essWetBulbTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.4)
13.9.3	VERIFY that the RESPONSE VALUE for essWetBulbTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.4)
13.9.4	VERIFY that the RESPONSE VALUE for the essDewpointTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.5)

13.9.5	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.5)
13.9.6	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.5)
13.9.7	VERIFY that the RESPONSE VALUE for the essRelativeHumidity.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.5)
13.9.8	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is between 0 and 101, inclusive.	Pass / Fail (Sec. 5.8.1)
13.9.9	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.1)
13.10	VERIFY that the temperatureTable field contains at least Required_Temp_Sensors entries.	Pass / Fail (Sec. 3.6.3)
13.11	FOR EACH value, N, from 1 to Required_Temp_Sensors, perform Steps 13.11.1 through 13.11.5.	
13.11.1	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.x field is present in the Nth Temperature structure.	Pass / Fail (Sec. 3.5.2.3.2.3)
13.11.2	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.x field in the Nth Temperature structure is equal to N.	Pass / Fail (Sec. 5.7.3.1)
13.11.3	VERIFY that the RESPONSE VALUE for the essAirTemperature.x field is present in the Nth Temperature structure.	Pass / Fail (Sec. 3.5.2.3.2.3)
13.11.4	VERIFY that the RESPONSE VALUE for essAirTemperature.x field in the Nth Temperature structure is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.3.3)
13.11.5	VERIFY that the RESPONSE VALUE for essAirTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.3)
14	IF Precip_Supported is equal to true, then proceed to Step 14.1; otherwise, proceed to Step 15.	
14.1	VERIFY that the RESPONSE VALUE for the essPrecipData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.6.1)
14.2	VERIFY that the waterLevelSensorTable field contains at least Required_Water_Level_Sensors entries.	Pass / Fail (Sec. 3.6.22)
14.3	FOR EACH value, N, from 1 to Required_Water_Level_Sensors, perform Steps 14.3.1 through 14.3.5.	
14.3.1	VERIFY that the RESPONSE VALUE for the waterLevelSensorIndex.x field is present in the Nth WaterLevel structure.	Pass / Fail (Sec. 3.5.2.3.7)
14.3.2	VERIFY that the RESPONSE VALUE for the waterLevelSensorIndex.x field in the Nth WaterLevel structure is equal to N.	Pass / Fail (Sec. 5.8.21.1)
14.3.3	VERIFY that the RESPONSE VALUE for the waterLevelSensorReading.x field is present in the Nth WaterLevel structure.	Pass / Fail (Sec. 3.5.2.3.7)
14.3.4	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.x field in the Nth WaterLevel structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.21.2)

14.3.5	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.x is APPROPRIATE.	Pass / Fail (Sec. 5.8.21.2)
14.4	IF Adjacent_Snow_Depth_Supported is equal to 1, then proceed to Step 14.4.1; otherwise, proceed to Step 14.5.	
14.4.1	VERIFY that the RESPONSE VALUE for the essAdjacentSnowDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.4)
14.4.2	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.3)
14.4.3	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.3)
14.5	IF Roadway_Snow_Depth_Supported is equal to 1, then proceed to Step 14.5.1; otherwise, proceed to Step 14.6.	
14.5.1	VERIFY that the RESPONSE VALUE for the essRoadwaySnowDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.5)
14.5.2	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.4)
14.5.3	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.4)
14.6	IF Ice_Thickness_Supported is equal to 1, then proceed to Step 14.6.1; otherwise, proceed to Step 14.7.	
14.6.1	VERIFY that the RESPONSE VALUE for the essRoadwaySnowPackDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.6)
14.6.2	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.5)
14.6.3	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.5)
14.6.4	VERIFY that the RESPONSE VALUE for the essIceThickness.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.6)
14.6.5	VERIFY that the RESPONSE VALUE for essIceThickness.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.10)
14.6.6	VERIFY that the RESPONSE VALUE for essIceThickness.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.10)
14.7	VERIFY that the RESPONSE VALUE for the essPrecipYesNo.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.1)
14.8	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is between 1 and 3, inclusive.	Pass / Fail (Sec. 5.8.6)
14.9	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.6)
14.10	IF Precip_Rates_Supported is equal to 1, then proceed to Step 14.10.1; otherwise, proceed to Step 14.11.	

14.10.1	VERIFY that the RESPONSE VALUE for the essPrecipRate.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.2	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.7)
14.10.3	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.7)
14.10.4	VERIFY that the RESPONSE VALUE for the essSnowfallAccumRate.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.5	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.8)
14.10.6	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.8)
14.10.7	VERIFY that the RESPONSE VALUE for the essPrecipitationStartTime.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.8	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.8.11)
14.10.9	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.11)
14.10.10	VERIFY that the RESPONSE VALUE for the essPrecipitationEndTime.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.11	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.8.12)
14.10.12	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.12)
14.11	IF Situation_Supported is equal to true, then proceed to Step 14.11.1; otherwise, proceed to Step 15.	
14.11.1	VERIFY that the RESPONSE VALUE for the essPrecipSituation.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.5.2)
14.11.2	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is between 1 and 15, inclusive.	Pass / Fail (Sec. 5.8.9)
14.11.3	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.9)
15	IF Visibility_Supported is equal to true, then proceed to Step 15.1; otherwise, proceed to Step 16.	
15.1	VERIFY that the RESPONSE VALUE for the essVisibilityData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.8)
15.2	VERIFY that the RESPONSE VALUE for the essVisibility.0 field of the essVisibilityDataV3 field is present.	Pass / Fail (Sec. 3.5.2.3.2.8)
15.3	VERIFY that the RESPONSE VALUE for essVisibility.0 is between 0 and 1000001, inclusive.	Pass / Fail (Sec. 5.10.1)

15.4	VERIFY that the RESPONSE VALUE for essVisibility.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.1)
15.5	IF Situation_Supported is equal to true, then proceed to Step 15.5.1; otherwise, proceed to Step 16.	
15.5.1	VERIFY that the RESPONSE VALUE for the essVisibilitySituation.0 field of the essVisibilityDataV3 field is present.	Pass / Fail (Sec. 3.5.2.3.5.4)
15.5.2	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is between 1 and 12, inclusive.	Pass / Fail (Sec. 5.10.2)
15.5.3	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.2)
16	IF Radiation_Supported is equal to 1, then proceed to Step 16.1; otherwise, proceed to Step 17.	
16.1	VERIFY that the RESPONSE VALUE for the essRadiationData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.2	VERIFY that the RESPONSE VALUE for the essTotalSun.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.3	VERIFY that the RESPONSE VALUE for essTotalSun.0 is between 0 and 1441, inclusive.	Pass / Fail (Sec. 5.9.2)
16.4	VERIFY that the RESPONSE VALUE for essTotalSun.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.2)
16.5	VERIFY that the RESPONSE VALUE for the essInstantaneousTerrestrialRadiation.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.6	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.9.4)
16.7	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.4)
16.8	VERIFY that the RESPONSE VALUE for the essInstantaneousSolarRadiation.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.9	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.9.5)
16.10	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.5)
16.11	VERIFY that the RESPONSE VALUE for the essTotalRadiation.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.12	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.9.6)
16.13	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.6)
16.14	VERIFY that the RESPONSE VALUE for the essTotalRadiationPeriod.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)

16.15	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is between 0 and 86400, inclusive.	Pass / Fail (Sec. 5.9.7)
16.16	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.7)
16.17	IF Situation_Supported is equal to true, then proceed to Step 16.17.1; otherwise, proceed to Step 17.	
16.17.1	VERIFY that the RESPONSE VALUE for the essCloudSituation.0 field of the essVisibilityDataV3 field is present.	Pass / Fail (Sec. 3.5.2.3.5.3)
16.17.2	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is between 1 and 5, inclusive.	Pass / Fail (Sec. 5.9.3)
16.17.3	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.3)
17	VERIFY that the RESPONSE VALUE for essLatitude.0 is greater than or equal to -90000000.	Pass / Fail (Sec. 5.4.1)
18	VERIFY that the RESPONSE VALUE for essLatitude.0 is less than or equal to 90000001.	Pass / Fail (Sec. 5.4.1)
19	VERIFY that the RESPONSE VALUE for essLatitude.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.1)
20	VERIFY that the RESPONSE VALUE for essLongitude.0 is greater than or equal to -180000000.	Pass / Fail (Sec. 5.4.2)
21	VERIFY that the RESPONSE VALUE for essLongitude.0 is less than or equal to 180000001.	Pass / Fail (Sec. 5.4.2)
22	VERIFY that the RESPONSE VALUE for essLongitude.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.2)
23	VERIFY that the RESPONSE VALUE for essVehicleSpeed.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.4.3)
24	VERIFY that the RESPONSE VALUE for essVehicleSpeed.0 is less than or equal to 255.	Pass / Fail (Sec. 5.4.3)
25	VERIFY that the RESPONSE VALUE for essVehicleSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.3)
26	VERIFY that the RESPONSE VALUE for essVehicleBearing.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.4.4)
27	VERIFY that the RESPONSE VALUE for essVehicleBearing.0 is less than or equal to 361.	Pass / Fail (Sec. 5.4.4)
28	VERIFY that the RESPONSE VALUE for essVehicleBearing.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.4)
29	VERIFY that the RESPONSE VALUE for essOdometer.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.4.5)
30	VERIFY that the RESPONSE VALUE for essOdometer.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.4.5)

31	VERIFY that the RESPONSE VALUE for essOdometer.0 is APPROPRIATE.	Pass / Fail (Sec. 5.4.5)
32	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is greater than or equal to -400.	Pass / Fail (Sec. 5.5.1)
33	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is less than or equal to 8001.	Pass / Fail (Sec. 5.5.1)
34	VERIFY that the RESPONSE VALUE for essReferenceHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.1)
35	VERIFY that the RESPONSE VALUE for essMobileFriction.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.12.1)
36	VERIFY that the RESPONSE VALUE for essMobileFriction.0 is less than or equal to 101.	Pass / Fail (Sec. 5.12.1)
37	VERIFY that the RESPONSE VALUE for essMobileFriction.0 is APPROPRIATE.	Pass / Fail (Sec. 5.12.1)
38	VERIFY that the RESPONSE VALUE for essMobileObservationGroundState.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.12.2)
39	VERIFY that the RESPONSE VALUE for essMobileObservationGroundState.0 is less than or equal to 18.	Pass / Fail (Sec. 5.12.2)
40	VERIFY that the RESPONSE VALUE for essMobileObservationGroundState.0 is APPROPRIATE.	Pass / Fail (Sec. 5.12.2)
41	VERIFY that the RESPONSE VALUE for essMobileObservationPavement.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.12.3)
42	VERIFY that the RESPONSE VALUE for essMobileObservationPavement.0 is less than or equal to 25.	Pass / Fail (Sec. 5.12.3)
43	VERIFY that the RESPONSE VALUE for essMobileObservationPavement.0 is APPROPRIATE.	Pass / Fail (Sec. 5.12.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.2 Retrieve Atmospheric Pressure

Test Case: 3.2	***Title:***	*Retrieve Atmospheric Pressure*
	Description:	*This test case verifies that the ESS allows a management station to determine the current atmospheric pressure.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essAtmosphericPressure.0	Pass / Fail (Sec. 3.5.2.3.2.1)
2	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.5.4)
3	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.5.4)
4	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.3 Retrieve Wind Data

Test Case: 3.3	***Title:***	*Retrieve Wind Data*	
	Description:	*This test case verifies that the ESS allows a management station to determine current wind information.*	
	Variables:	*Required_Wind_Sensors*	*PRL 3.6.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of wind sensors required by the specification (PRL 3.6.2). RECORD this information as: »Required_Wind_Sensors	
2	GET the following object(s): »windSensorTableNumSensors.0	Pass / Fail (Sec. 3.5.2.3.2.2)
3	VERIFY that the RESPONSE VALUE for windSensorTableNumSensors.0 is greater than or equal to Required_Wind_Sensors.	Pass / Fail (Sec. 3.6.2)
4	Determine the RESPONSE VALUE for windSensorTableNumSensors.0. RECORD this information as: »Supported_Wind_Sensors	
5	FOR EACH value, N, from 1 to Supported_Wind_Sensors, perform Steps 5.1 through 5.22.	
5.1	GET the following object(s): »windSensorAvgSpeed.N »windSensorAvgDirection.N »windSensorSpotSpeed.N »windSensorSpotDirection.N »windSensorGustSpeed.N »windSensorGustDirection.N »windSensorSituation.N	Pass / Fail (Sec. 3.5.2.3.2.2)
5.2	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.N is greater than or equal to 0.	Pass / Fail (Sec. 5.6.10.4)
5.3	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.N is less than or equal to 65535.	Pass / Fail (Sec. 5.6.10.4)
5.4	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.4)
5.5	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.N is greater than or equal to 0.	Pass / Fail (Sec. 5.6.10.5)
5.6	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.N is less than or equal to 361.	Pass / Fail (Sec. 5.6.10.5)
5.7	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.5)
5.8	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.N is greater than or	Pass / Fail

5.9	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.N is less than or equal to 65535.	Pass / Fail (Sec. 5.6.10.6)
5.10	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.6)
5.11	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.N is greater than or equal to 0.	Pass / Fail (Sec. 5.6.10.7)
5.12	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.N is less than or equal to 361.	Pass / Fail (Sec. 5.6.10.7)
5.13	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.7)
5.14	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.N is greater than or equal to 0.	Pass / Fail (Sec. 5.6.10.8)
5.15	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.N is less than or equal to 65535.	Pass / Fail (Sec. 5.6.10.8)
5.16	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.8)
5.17	VERIFY that the RESPONSE VALUE for windSensorGustDirection.N is greater than or equal to 0.	Pass / Fail (Sec. 5.6.10.9)
5.18	VERIFY that the RESPONSE VALUE for windSensorGustDirection.N is less than or equal to 361.	Pass / Fail (Sec. 5.6.10.9)
5.19	VERIFY that the RESPONSE VALUE for windSensorGustDirection.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.9)
5.20	VERIFY that the RESPONSE VALUE for windSensorSituation.N is greater than or equal to 1.	Pass / Fail (Sec. 5.6.10.10)
5.21	VERIFY that the RESPONSE VALUE for windSensorSituation.N is less than or equal to 12.	Pass / Fail (Sec. 5.6.10.10)
5.22	VERIFY that the RESPONSE VALUE for windSensorSituation.N is APPROPRIATE.	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.4 Retrieve Temperature

Test Case: 3.4	***Title:***	*Retrieve Temperature*	
	Description:	*This test case verifies that the ESS allows a management station to determine the current temperature from a temperature sensor.*	
	Variables:	*Required_Temperature_Sensors*	*PRL 3.6.3*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of temperature sensors required by the specification (PRL 3.6.3). RECORD this information as: »Required_Temperature_Sensors	
2	GET the following object(s): »essNumTemperatureSensors.0	Pass / Fail (RFC 1157)
3	VERIFY that the RESPONSE VALUE for essNumTemperatureSensors.0 is greater than or equal to Required_Temperature_Sensors.	Pass / Fail (Sec. 3.6.3)
4	Determine a random number between 1 and Required_Temperature_Sensors. RECORD this information as: »Subject_Sensor	
5	GET the following object(s): »essAirTemperature.Subject_Sensor	Pass / Fail (Sec. 3.5.2.3.2.3)
6	VERIFY that the RESPONSE VALUE for essAirTemperature.Subject_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.7.3.3)
7	VERIFY that the RESPONSE VALUE for essAirTemperature.Subject_Sensor is less greater than or equal to 1000.	Pass / Fail (Sec. 5.7.3.3)
8	VERIFY that the RESPONSE VALUE for essAirTemperature.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.5 Retrieve Daily Minimum and Maximum Temperature

Test Case: 3.5	***Title:***	*Retrieve Daily Minimum and Maximum Temperature*
	Description:	*This test case verifies that the ESS allows a management station to determine the minimum and maximum temperatures recorded during the previous 24 hours.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essMaxTemp.0 »essMinTemp.0	Pass / Fail (Sec. 3.5.2.3.2.4)
2	VERIFY that the RESPONSE VALUE for essMinTemp.0 is greater than or equal to -1000.	Pass / Fail (Sec. 5.7.7)
3	VERIFY that the RESPONSE VALUE for essMinTemp.0 is less than or equal to 1001.	Pass / Fail (Sec. 5.7.7)
4	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.7)
5	Determine the RESPONSE VALUE for essMinTemp.0. RECORD this information as: »Minimum_Temp	
6	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is greater than or equal to Minimum_Temp.	Pass / Fail (Sec. 5.7.6)
7	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is less than or equal to 1001.	Pass / Fail (Sec. 5.7.6)
8	VERIFY that the RESPONSE VALUE for essMinTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.6)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.6 Retrieve Humidity

Test Case: 3.6	***Title:***	*Retrieve Humidity*
	Description:	*This test case verifies that the ESS allows a management station to determine the current humidity information.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essWetbulbTemp.0 »essDewpointTemp.0 »essRelativeHumidity.0	Pass / Fail (Sec. 3.5.2.3.2.5)
2	VERIFY that the RESPONSE VALUE for essWetbulbTemp.0 is greater than or equal to -1000.	Pass / Fail (Sec. 5.7.4)
3	VERIFY that the RESPONSE VALUE for essWetbulbTemp.0 is less than or equal to 1001.	Pass / Fail (Sec. 5.7.4)
4	VERIFY that the RESPONSE VALUE for essWetbulbTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.4)
5	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is greater than or equal to -1000.	Pass / Fail (Sec. 5.7.5)
6	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is less than or equal to 1001.	Pass / Fail (Sec. 5.7.5)
7	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.5)
8	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.1)
9	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is less than or equal to 101.	Pass / Fail (Sec. 5.8.1)
10	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.7 Retrieve Precipitation Presence

Test Case: 3.7	***Title:***	*Retrieve Precipitation Presence*
	Description:	*This test case verifies that the ESS allows a management station to determine the presence of precipitation and the make and model of the devicesensor.*
	Variables:	Device_Version — Test Plan
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the version of the NTCIP 1204 standard to which the ESS claims conformance (Test Plan). RECORD this information as: »Device_Version	
12	GET the following object(s): »globalMaxModules.0	Pass / Fail (RFC 1157)
32	Determine the RESPONSE VALUE for globalMaxModules.0. RECORD this information as: »Num_Modules	
34	GET the following object(s): »essPrecipYesNo.0 »precipitationSensorModelInformation.0	Pass / Fail (Sec. 3.5.2.3.2.6.14.2.12)
45	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.8.6)
56	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is less than or equal to 3.	Pass / Fail (Sec. 5.8.6)
76	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.6)
8	IF Device_Version is greater than 1, then proceed to Step 8.1; otherwise, proceed to Step 9.1.	
8.1	GET the following object(s): »precipitationSensorModelInformation.0	Pass / Fail (Sec. 4.2.12)
8.27	VERIFY that the RESPONSE VALUE for precipitationSensorModelInformation.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.8.18)
88.3	VERIFY that the RESPONSE VALUE for precipitationSensorModelInformation.0 is less than or equal to Num_Modules.	Pass / Fail (Sec. 5.8.18)
8.49	VERIFY that the RESPONSE VALUE for precipitationSensorModelInformation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.18)
9.1	GET the following object(s): »precipitationSensorModelInformation.0 VERIFY that the RESPONSE ERROR is equal to ‘NoSuchName’.	Pass / Fail (RFC 1157)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.8 Retrieve Precipitation Rates

Test Case: 3.8	***Title:***	*Retrieve Precipitation Rates*
	Description:	*This test case verifies that the ESS allows a management station to determine the rate at which precipitation is falling and the most recent start and stop times of precipitation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essPrecipRate.0 »essSnowfallAccumRate.0 »essPrecipitationStartTime.0 »essPrecipitationEndTime.0	Pass / Fail (Section 3.5.2.3.2.6.2)
2	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.7)
3	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.7)
4	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.7)
5	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.8)
6	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.8)
7	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.8)
8	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.11)
9	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.8.11)
10	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.11)
11	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.12)
12	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.8.12)
13	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.12)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.9 Retrieve Precipitation Totals

Test Case: 3.9	***Title:***	*Retrieve Precipitation Totals*
	Description:	*This test case verifies that the ESS allows a management station to determine the total amounts of precipitation recorded during previous standardized time intervals.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essPrecipitationOneHour.0 »essPrecipitationThreeHours.0 »essPrecipitationSixHours.0 »essPrecipitationTwelveHours.0 »essPrecipitation24Hours.0	Pass / Fail (Sec. 3.5.2.3.2.6.3)
2	VERIFY that the RESPONSE VALUE for essPrecipitationOneHour.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.13)
3	VERIFY that the RESPONSE VALUE for essPrecipitationOneHour.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.13)
4	VERIFY that the RESPONSE VALUE for essPrecipitationOneHour.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.13)
5	VERIFY that the RESPONSE VALUE for essPrecipitationThreeHours.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.14)
6	VERIFY that the RESPONSE VALUE for essPrecipitationThreeHours.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.14)
7	VERIFY that the RESPONSE VALUE for essPrecipitationThreeHours.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.14)
8	VERIFY that the RESPONSE VALUE for essPrecipitationSixHours.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.15)
9	VERIFY that the RESPONSE VALUE for essPrecipitationSixHours.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.15)
10	VERIFY that the RESPONSE VALUE for essPrecipitationSixHours.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.15)
11	VERIFY that the RESPONSE VALUE for essPrecipitationTwelveHours.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.16)
12	VERIFY that the RESPONSE VALUE for essPrecipitationTwelveHours.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.16)
13	VERIFY that the RESPONSE VALUE for essPrecipitationTwelveHours.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.16)
14	VERIFY that the RESPONSE VALUE for essPrecipitation24Hours.0 is greater than or	Pass / Fail

15	VERIFY that the RESPONSE VALUE for essPrecipitation24Hours.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.17)
16	VERIFY that the RESPONSE VALUE for essPrecipitation24Hours.0 is APPROPRIATE.	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.10 Retrieve Solar Radiation

Test Case: 3.10	***Title:***	*Retrieve Solar Radiation*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the solar radiation data stored in the device.*
	Variables:	Device_Version / Test Plan
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the version of the NTCIP 1204 standard to which the ESS claims conformance (Test Plan). RECORD this information as: »Device_Version	
12	GET the following object(s): »essTotalSun.0 »essInstantaneousTerrestrialRadiation.0 »essInstantaneousSolarRadiation.0 »essTotalRadiation.0 »essTotalRadiationPeriod.0	Pass / Fail (Sec. 3.5.2.3.2.74.2.13)
23	VERIFY that the RESPONSE VALUE for essTotalSun.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.9.2)
34	VERIFY that the RESPONSE VALUE for essTotalSun.0 is less than or equal to 1441.	Pass / Fail (Sec. 5.9.2)
45	VERIFY that the RESPONSE VALUE for essTotalSun.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.2)
6	IF Device_Version is greater than 1, then proceed to Step 6.1; otherwise, proceed to Step 7.1.	
6.1	GET the following object(s): »essInstantaneousTerrestrialRadiation.0 »essInstantaneousSolarRadiation.0 »essTotalRadiation.0 »essTotalRadiationPeriod.0	Pass / Fail (Sec. 4.2.13)
56.2	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is greater than or equal to -2048.	Pass / Fail (Sec. 5.9.4)
66.3	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is less than or equal to 2049.	Pass / Fail (Sec. 5.9.4)
76.4	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.4)
86.5	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is greater than or equal to -2048.	Pass / Fail (Sec. 5.9.5)
96.6	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is less than or equal to 2049.	Pass / Fail (Sec. 5.9.5)
106.7	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.5)
116.8	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is greater than or equal to -2048.	Pass / Fail (Sec. 5.9.6)

126.9	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is less than or equal to 2049.	Pass / Fail (Sec. 5.9.6)
136.10	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.6)
146.11	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.9.7)
156.12	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is less than or equal to 86400.	Pass / Fail (Sec. 5.9.7)
166.13	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.7)
7.1	GET the following object(s): »essInstantaneousTerrestrialRadiation.0 »essInstantaneousSolarRadiation.0 »essTotalRadiation.0 »essTotalRadiationPeriod.0 VERIFY that the RESPONSE ERROR is equal to ‘NoSuchName’.	Pass / Fail (Sec. 4.2.13)
7.2	GET the following object(s): »essSolarRadiation.0	Pass / Fail (RFC 1157)
7.3	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.9.1)
7.4	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.9.1)
7.5	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.11 Retrieve Visibility

Test Case: 3.11	***Title:***	*Retrieve Visibility*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current visibility distance.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essVisibility.0	Pass / Fail (Sec. 3.5.2.3.2.8)
2	VERIFY that the RESPONSE VALUE for essVisibility.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.10.1)
3	VERIFY that the RESPONSE VALUE for essVisibility.0 is less than or equal to 1000001.	Pass / Fail (Sec. 5.10.1)
4	VERIFY that the RESPONSE VALUE for essVisibility.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.3.12 Retrieve Compressed Weather Data

Test Case: 3.12	***Title:***	*Retrieve Compressed Weather Data*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve all current weather information in compressed form.*	
	Variables:	*Pressure_Supported*	*PRL 2.5.2.1.1*
		Wind_Supported	*PRL 2.5.2.1.2*
		Required_Wind_Sensors	*PRL 3.6.2*
		Temperature_Supported	*PRL 2.5.2.1.3*
		Required_Temp_Sensors	*PRL 3.6.3*
		Precip_Supported	*PRL 2.5.2.1.5*
		Required_Water_Level_Sensors	*PRL 3.6.22*
		Adjacent_Snow_Depth_Supported	*PRL 3.5.2.3.3.4*
		Roadway_Snow_Depth_Supported	*PRL 3.5.2.3.3.5*
		Ice_Thickness_Supported	*PRL 3.5.2.3.3.6*
		Precip_Rates_Supported	*PRL 3.5.2.3.2.6.2*
		Humidity_Supported	*PRL 2.5.2.1.4*
		Situation_Supported	*PRL 2.5.2.4*
		Visibility_Supported	*PRL 2.5.2.1.7*
		Radiation_Supported	*PRL 2.5.2.1.6*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the ESS is required to monitor atmospheric pressure (PRL 2.5.2.1.1). RECORD this information as: »Pressure_Supported	
2	CONFIGURE: Determine the number of wind sensors supported. RECORD this information as: »Wind_Supported (whether the ESS is required to monitor winds—PRL 2.5.2.1.2) »Required_Wind_Sensors (the number of wind sensors that the ESS is required to support—PRL 3.6.2)	
3	CONFIGURE: Determine the number of temperature sensors supported. RECORD this information as: »Temperature_Supported (whether the ESS is required to monitor temperature—PRL 2.5.2.1.3) »Required_Temp_Sensors (the number of temperature sensors that the ESS is required to support—PRL 3.6.3)	

4	CONFIGURE: Determine the number of precipitation sensors supported. RECORD this information as: »Precip_Supported (whether the ESS is required to monitor precipitation—PRL 2.5.2.1.5) »Required_Water_Level_Sensors (the number of water level sensors that the ESS is required to support—PRL 3.6.22) »Adjacent_Snow_Depth_Supported (whether the ESS is required to monitor the adjacent snow depth—PRL 3.5.2.3.3.4) »Roadway_Snow_Depth_Supported (whether the ESS is required to monitor the roadway snow depth –PRL 3.5.2.3.3.5) »Ice_Thickness_Supported (whether the ESS is required to monitor the snow pack depth and ice thickness—PRL 3.5.2.3.3.6) »Precip_Rates_Supported (whether the ESS is required to monitor the precipitation rates—PRL 3.5.2.3.2.6.2) »Humidity_Supported (Whether the ESS is required to monitor humidity—PRL 2.5.2.1.4)	
5	CONFIGURE: Determine whether the ESS is required to report situation data (PRL 2.5.2.4). RECORD this information as: »Situation_Supported	
6	CONFIGURE: Determine whether the ESS is required to provide visibility capabilities (PRL 2.5.2.1.7). RECORD this information as: »Visibility_Supported	
7	CONFIGURE: Determine whether the ESS is required to monitor solar radiation (PRL 2.5.2.1.6). RECORD this information as: »Radiation_Supported	
8	GET the following object(s): »essWeatherV3Block.0	Pass / Fail (Sec. 3.5.2.3.2.9)
9	Decode the essWeatherV3Block.0 structure.	
10	VERIFY that the essWeatherV3Block.0 structure was decoded without error.	Pass / Fail (Sec. 5.3.9)
11	IF Pressure_Supported is equal to true, then proceed to Step 11.1; otherwise, proceed to Step 12.	
11.1	VERIFY that the RESPONSE VALUE for the essAtmosphericPressure.0 field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.1)
11.2	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.5.4)
11.3	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.4)
12	IF Wind_Supported is equal to true, then proceed to Step 12.1; otherwise, proceed to Step 13.	
12.1	VERIFY that the RESPONSE VALUE for the essWindData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.2	VERIFY that the essWindData field contains at least Required_Wind_Sensors entries.	Pass / Fail (Sec. 3.6.2)
12.3	FOR EACH value, N, from 1 to Required_Wind_Sensors, perform Steps 12.3.1 through 12.3.21.	

12.3.1	VERIFY that the windSensorIndex.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.2	VERIFY that the RESPONSE VALUE for windSensorIndex.x in the Nth essWindData structure is equal to N.	Pass / Fail (Sec. 5.6.10.1)
12.3.3	VERIFY that the windSensorAvgSpeed.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.4	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.x in the Nth essWindDataV3 structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.10.4)
12.3.5	VERIFY that the RESPONSE VALUE for windSensorAvgSpeed.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.4)
12.3.6	VERIFY that the windSensorAvgDirection.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.7	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.x in the Nth essWindDataV3 structure is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.10.5)
12.3.8	VERIFY that the RESPONSE VALUE for windSensorAvgDirection.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.5)
12.3.9	VERIFY that the windSensorSpotSpeed.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.10	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.x in the Nth essWindDataV3 structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.10.6)
12.3.11	VERIFY that the RESPONSE VALUE for windSensorSpotSpeed.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.6)
12.3.12	VERIFY that the windSensorSpotDirection.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.13	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.x in the Nth essWindDataV3 structure is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.10.7)
12.3.14	VERIFY that the RESPONSE VALUE for windSensorSpotDirection.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.7)
12.3.15	VERIFY that the windSensorGustSpeed.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.16	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.x in the Nth essWindDataV3 structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.10.8)
12.3.17	VERIFY that the RESPONSE VALUE for windSensorGustSpeed.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.8)
12.3.18	VERIFY that the windSensorGustDirection.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.19	VERIFY that the RESPONSE VALUE for windSensorGustDirection.x in the Nth essWindDataV3 structure is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.10.9)
12.3.20	VERIFY that the RESPONSE VALUE for windSensorGustDirection.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.9)

12.3.21	IF Situation_Supported is equal to true, then proceed to Step 12.3.21.1; otherwise, proceed to Step 13.	
12.3.21.1	VERIFY that the windSensorSituation.x field is present in the Nth essWindDataV3 structure.	Pass / Fail (Sec. 3.5.2.3.2.2)
12.3.21.2	VERIFY that the RESPONSE VALUE for windSensorSituation.x in the Nth essWindDataV3 structure is between 1 and 12, inclusive.	Pass / Fail (Sec. 5.6.10.10)
12.3.21.3	VERIFY that the RESPONSE VALUE for windSensorSituation.x in the Nth essWindDataV3 structure is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.10)
13	IF Temperature_Supported is equal to true, then proceed to Step 13.1; otherwise, proceed to Step 14.	
13.1	VERIFY that the essTemperatureData field of the essWeatherV3Block object is present.	Pass / Fail (Sec. 3.5.2.3.2.3)
13.2	VERIFY that the RESPONSE VALUE for the essMaxTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.4)
13.3	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.6)
13.4	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.6)
13.5	Determine the maximum temperature reading reported by the device. RECORD this information as: »Max_Temp	
13.6	VERIFY that the RESPONSE VALUE for the essMinTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.4)
13.7	VERIFY that the RESPONSE VALUE for essMinTemp.0 is between -1000 and Max_Temp, inclusive.	Pass / Fail (Sec. 5.7.7)
13.8	VERIFY that the RESPONSE VALUE for essMinTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.7)
13.9	IF Humidity_Supported is equal to 1, then proceed to Step 13.9.1; otherwise, proceed to Step 13.10.	
13.9.1	VERIFY that the RESPONSE VALUE for the essWetBulbTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.5)
13.9.2	VERIFY that the RESPONSE VALUE for essWetBulbTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.4)
13.9.3	VERIFY that the RESPONSE VALUE for essWetBulbTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.4)
13.9.4	VERIFY that the RESPONSE VALUE for the essDewpointTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.5)
13.9.5	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.5)
13.9.6	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.5)

13.9.7	VERIFY that the RESPONSE VALUE for the essRelativeHumidity.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 3.5.2.3.2.5)
13.9.8	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is between 0 and 101, inclusive.	Pass / Fail (Sec. 5.8.1)
13.9.9	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.1)
13.10	VERIFY that the temperatureTable field contains at least Required_Temp_Sensors entries.	Pass / Fail (Sec. 3.6.3)
13.11	FOR EACH value, N, from 1 to Required_Temp_Sensors, perform Steps 13.11.1 through 13.11.5.	
13.11.1	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.x field is present in the Nth Temperature structure.	Pass / Fail (Sec. 3.5.2.3.2.3)
13.11.2	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.x field in the Nth Temperature structure is equal to N.	Pass / Fail (Sec. 5.7.3.1)
13.11.3	VERIFY that the RESPONSE VALUE for the essAirTemperature.x field is present in the Nth Temperature structure.	Pass / Fail (Sec. 3.5.2.3.2.3)
13.11.4	VERIFY that the RESPONSE VALUE for essAirTemperature.x field in the Nth Temperature structure is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.3.3)
13.11.5	VERIFY that the RESPONSE VALUE for essAirTemperature.x field in the Nth Temperature structure is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.3)
14	IF Precip_Supported is equal to true, then proceed to Step 14.1; otherwise, proceed to Step 15.	
14.1	VERIFY that the RESPONSE VALUE for the essPrecipData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.6.1)
14.2	VERIFY that the waterLevelSensorTable field contains at least Required_Water_Level_Sensors entries.	Pass / Fail (Sec. 3.6.22)
14.3	FOR EACH value, N, from 1 to Required_Water_Level_Sensors, perform Steps 14.3.1 through 14.3.5.	
14.3.1	VERIFY that the RESPONSE VALUE for the waterLevelSensorIndex.x field is present in the Nth WaterLevel structure.	Pass / Fail (Sec. 3.5.2.3.7)
14.3.2	VERIFY that the RESPONSE VALUE for the waterLevelSensorIndex.x field in the Nth WaterLevel structure is equal to N.	Pass / Fail (Sec. 5.8.21.1)
14.3.3	VERIFY that the RESPONSE VALUE for the waterLevelSensorReading.x field is present in the Nth WaterLevel structure.	Pass / Fail (Sec. 3.5.2.3.7)
14.3.4	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.x field in the Nth WaterLevel structure is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.21.2)
14.3.5	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.x field in the Nth WaterLevel structure is APPROPRIATE.	Pass / Fail (Sec. 5.8.21.2)
14.4	IF Adjacent_Snow_Depth_Supported is equal to 1, then proceed to Step 14.4.1; otherwise, proceed to Step 14.5.	

14.4.1	VERIFY that the RESPONSE VALUE for the essAdjacentSnowDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.4)
14.4.2	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.3)
14.4.3	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.3)
14.5	IF Roadway_Snow_Depth_Supported is equal to 1, then proceed to Step 14.5.1; otherwise, proceed to Step 14.6.	
14.5.1	VERIFY that the RESPONSE VALUE for the essRoadwaySnowDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.5)
14.5.2	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.4)
14.5.3	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.4)
14.6	IF Ice_Thickness_Supported is equal to 1, then proceed to Step 14.6.1; otherwise, proceed to Step 14.7.	
14.6.1	VERIFY that the RESPONSE VALUE for the essRoadwaySnowPackDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.6)
14.6.2	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.5)
14.6.3	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.5)
14.6.4	VERIFY that the RESPONSE VALUE for the essIceThickness.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.3.6)
14.6.5	VERIFY that the RESPONSE VALUE for essIceThickness.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.10)
14.6.6	VERIFY that the RESPONSE VALUE for essIceThickness.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.10)
14.7	VERIFY that the RESPONSE VALUE for the essPrecipYesNo.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.1)
14.8	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is between 1 and 3, inclusive.	Pass / Fail (Sec. 5.8.6)
14.9	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.6)
14.10	IF Precip_Rates_Supported is equal to 1, then proceed to Step 14.10.1; otherwise, proceed to Step 14.11.	
14.10.1	VERIFY that the RESPONSE VALUE for the essPrecipRate.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.2	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.7)

14.10.3	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.7)
14.10.4	VERIFY that the RESPONSE VALUE for the essSnowfallAccumRate.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.5	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.8)
14.10.6	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.8)
14.10.7	VERIFY that the RESPONSE VALUE for the essPrecipitationStartTime.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.8	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.8.11)
14.10.9	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.11)
14.10.10	VERIFY that the RESPONSE VALUE for the essPrecipitationEndTime.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.2.6.2)
14.10.11	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.8.12)
14.10.12	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.12)
14.11	IF Situation_Supported is equal to true, then proceed to Step 14.11.1; otherwise, proceed to Step 15.	
14.11.1	VERIFY that the RESPONSE VALUE for the essPrecipSituation.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 3.5.2.3.5.2)
14.11.2	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is between 1 and 15, inclusive.	Pass / Fail (Sec. 5.8.9)
14.11.3	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.9)
15	IF Visibility_Supported is equal to true, then proceed to Step 15.1; otherwise, proceed to Step 16.	
15.1	VERIFY that the RESPONSE VALUE for the essVisibilityData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.8)
15.2	VERIFY that the RESPONSE VALUE for the essVisibility.0 field of the essVisibilityDataV3 field is present.	Pass / Fail (Sec. 3.5.2.3.2.8)
15.3	VERIFY that the RESPONSE VALUE for essVisibility.0 is between 0 and 1000001, inclusive.	Pass / Fail (Sec. 5.10.1)
15.4	VERIFY that the RESPONSE VALUE for essVisibility.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.1)
15.5	IF Situation_Supported is equal to true, then proceed to Step 15.5.1; otherwise, proceed to Step 16.	

15.5.1	VERIFY that the RESPONSE VALUE for the essVisibilitySituation.0 field of the essVisibilityDataV3 field is present.	Pass / Fail (Sec. 3.5.2.3.5.4)
15.5.2	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is between 1 and 12, inclusive.	Pass / Fail (Sec. 5.10.2)
15.5.3	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.2)
16	IF Radiation_Supported is equal to 1, then proceed to Step 16.1; otherwise, proceed to EXIT.	
16.1	VERIFY that the RESPONSE VALUE for the essRadiationData field of the essWeatherV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.2	VERIFY that the RESPONSE VALUE for the essTotalSun.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.3	VERIFY that the RESPONSE VALUE for essTotalSun.0 is between 0 and 1441, inclusive.	Pass / Fail (Sec. 5.9.2)
16.4	VERIFY that the RESPONSE VALUE for essTotalSun.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.2)
16.5	VERIFY that the RESPONSE VALUE for the essInstantaneousTerrestrialRadiation.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.6	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.9.4)
16.7	VERIFY that the RESPONSE VALUE for essInstantaneousTerrestrialRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.4)
16.8	VERIFY that the RESPONSE VALUE for the essInstantaneousSolarRadiation.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.9	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.9.5)
16.10	VERIFY that the RESPONSE VALUE for essInstantaneousSolarRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.5)
16.11	VERIFY that the RESPONSE VALUE for the essTotalRadiation.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.12	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.9.6)
16.13	VERIFY that the RESPONSE VALUE for essTotalRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.6)
16.14	VERIFY that the RESPONSE VALUE for the essTotalRadiationPeriod.0 field of the essRadiationData field is present.	Pass / Fail (Sec. 3.5.2.3.2.7)
16.15	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is between 0 and 86400, inclusive.	Pass / Fail (Sec. 5.9.7)
16.16	VERIFY that the RESPONSE VALUE for essTotalRadiationPeriod.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.7)

16.17	IF Situation_Supported is equal to true, then proceed to Step 16.17.1; otherwise, proceed to EXIT.	
16.17.1	VERIFY that the RESPONSE VALUE for the essCloudSituation.0 field of the essVisibilityDataV3 field is present.	Pass / Fail (Sec. 3.5.2.3.5.3)
16.17.2	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is between 1 and 5, inclusive.	Pass / Fail (Sec. 5.9.3)
16.17.3	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4 Pavement Tests

C.2.3.4.1 Retrieve Pavement Surface Condition

Test Case: 4.1	***Title:***	*Retrieve Pavement Surface Condition*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current temperature and moisture of the pavement.*	
	Variables:	*Required_Pavement_Sensors*	*PRL 3.6.8*
		Device_Version	*Test Plan*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	CONFIGURE: Determine the version of the NTCIP 1204 standard to which the ESS claims conformance (Test Plan). RECORD this information as: »Device_Version	
3	SET-UP: GET the following object(s): »globalMaxModules.0	
4	SET-UP: Determine the RESPONSE VALUE for globalMaxModules.0. RECORD this information as: »Num_Modules	
52	GET the following object(s): »numEssPavementSensors.0	Pass / Fail (RFC 1157)
36	VERIFY that the RESPONSE VALUE for numEssPavementSensors.0 is greater than or equal to Required_Pavement_Sensors.	Pass / Fail (Sec. 5.11.1)
74	Determine a random number between 1 and Required_Pavement_Sensors. RECORD this information as: »Subject_Sensor	
5	GET the following object(s): »globalMaxModules.0	Pass / Fail (RFC 1157)
6	Determine the RESPONSE VALUE for globalMaxModules.0. RECORD this information as: »Num_Modules	
78	GET the following object(s): »essSurfaceStatus.Subject_Sensor »essSurfaceTemperature.Subject_Sensor »essPavementSensorError.Subject_Sensor »pavementSensorModelInformation.Subject_Sensor	Pass / Fail (Sec. 3.5.2.3.3.14.2.14)
89	VERIFY that the RESPONSE VALUE for essSurfaceStatus.Subject_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.7)
910	VERIFY that the RESPONSE VALUE for essSurfaceStatus.Subject_Sensor is less than	Pass / Fail

1011	VERIFY that the RESPONSE VALUE for essSurfaceStatus.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.7)
1112	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Subject_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.8)
1213	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Subject_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.8)
1314	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.8)
1415	VERIFY that the RESPONSE VALUE for essPavementSensorError.Subject_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.15)
1516	VERIFY that the RESPONSE VALUE for essPavementSensorError.Subject_Sensor is less than or equal to 6.	Pass / Fail (Sec. 5.11.3.15)
1617	VERIFY that the RESPONSE VALUE for essPavementSensorError.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.15)
18	IF Device_Version is greater than 1, then proceed to Step 18.1; otherwise, proceed to Step 19.1.	
18.1	GET the following object(s): »pavementSensorModelInformation.Subject_Sensor	Pass / Fail (Sec. 4.2.14)
1718.2	VERIFY that the RESPONSE VALUE for pavementSensorModelInformation.Subject_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.18)
1818.3	VERIFY that the RESPONSE VALUE for pavementSensorModelInformation.Subject_Sensor is less than or equal to Num_Modules.	Pass / Fail (Sec. 5.11.3.18)
1918.4	VERIFY that the RESPONSE VALUE for pavementSensorModelInformation.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.18)
19.1	GET the following object(s): »pavementSensorModelInformation.Subject_Sensor	Pass / Fail (Sec. 4.2.14)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.2 Retrieve Icing Conditions—Active

Test Case: 4.2	***Title:***	*Retrieve Icing Conditions—Active*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve icing condition information from an active sensor.*	
	Variables:	*Active_Pavement_Sensor*	*Test Plan*
		Device_Version	*Test Plan*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the identifier of an active pavement sensor (Test Plan). RECORD this information as: »Active_Pavement_Sensor	
2	CONFIGURE: Determine the version of the NTCIP 1204 standard to which the ESS claims conformance (Test Plan). RECORD this information as: »Device_Version	
23	GET the following object(s): »essSurfaceTemperature.Active_Pavement_Sensor »essPavementTemperature.Active_Pavement_Sensor »essSurfaceFreezePoint.Active_Pavement_Sensor »essSurfaceBlackIceSignal.Active_Pavement_Sensor »essPavementSensorError.Active_Pavement_Sensor »essSurfaceIceOrWaterDepth.Active_Pavement_Sensor »pavementSensorTemperatureDepth.Active_Pavement_Sensor	Pass / Fail (Sec. 3.5.2.3.3.24.2.15)
34	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Active_Pavement_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.8)
45	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Active_Pavement_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.8)
56	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.8)
67	VERIFY that the RESPONSE VALUE for essPavementTemperature.Active_Pavement_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.9)
78	VERIFY that the RESPONSE VALUE for essPavementTemperature.Active_Pavement_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.9)
89	VERIFY that the RESPONSE VALUE for essPavementTemperature.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.9)
910	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.Active_Pavement_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.13)
1011	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.Active_Pavement_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.13)
1112	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.13)
1213	VERIFY that the RESPONSE VALUE for	Pass / Fail

1314	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.Active_Pavement_Sensor is less than or equal to 4.	Pass / Fail (Sec. 5.11.3.14)
1415	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.14)
1516	VERIFY that the RESPONSE VALUE for essPavementSensorError.Active_Pavement_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.15)
1617	VERIFY that the RESPONSE VALUE for essPavementSensorError.Active_Pavement_Sensor is less than or equal to 6.	Pass / Fail (Sec. 5.11.3.15)
1718	VERIFY that the RESPONSE VALUE for essPavementSensorError.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.15)
19	IF Device_Version is greater than 1, then proceed to Step 19.1; otherwise, proceed to Step 20.1.	
19.1	GET the following object(s): »essSurfaceIceOrWaterDepth.Active_Pavement_Sensor »pavementSensorTemperatureDepth.Active_Pavement_Sensor	Pass / Fail (Sec. 4.2.15)
1819.2	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.Active_Pavement_Sensor is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.16)
19.3	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.Active_Pavement_Sensor is less than or equal to 65535.	Pass / Fail (Sec. 5.11.3.16)
2019.4	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.16)
2119.5	VERIFY that the RESPONSE VALUE for pavementSensorTemperatureDepth.Active_Pavement_Sensor is greater than or equal to 2.	Pass / Fail (Sec. 5.11.3.19)
2219.6	VERIFY that the RESPONSE VALUE for pavementSensorTemperatureDepth.Active_Pavement_Sensor is less than or equal to 11.	Pass / Fail (Sec. 5.11.3.19)
2319.7	VERIFY that the RESPONSE VALUE for pavementSensorTemperatureDepth.Active_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.19)
20.1	GET the following object(s): »essSurfaceIceOrWaterDepth.Active_Pavement_Sensor »pavementSensorTemperatureDepth.Active_Pavement_Sensor	Pass / Fail (Sec. 4.2.15)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.3 Retrieve Icing Conditions—Passive

Test Case: 4.3	***Title:***	*Retrieve Icing Conditions—Passive*
	Description:	*This test case verifies that the ESS allows a management station to retrieve icing condition information from an passive sensor.*
	Variables:	*Passive_Pavement_Sensor* — *Test Plan* *Device_Version* — *Test Plan*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the identifier of a passive pavement sensor (Test Plan). RECORD this information as: »Passive_Pavement_Sensor	
2	CONFIGURE: Determine the version of the NTCIP 1204 standard to which the ESS claims conformance (Test Plan). RECORD this information as: »Device_Version	
23	GET the following object(s): »essSurfaceTemperature.Passive_Pavement_Sensor »essPavementTemperature.Passive_Pavement_Sensor »essSurfaceSalinity.Passive_Pavement_Sensor »essSurfaceFreezePoint.Passive_Pavement_Sensor »essSurfaceBlackIceSignal.Passive_Pavement_Sensor »essPavementSensorError.Passive_Pavement_Sensor »essSurfaceIceOrWaterDepth.Passive_Pavement_Sensor »essSurfaceConductivityV2.Passive_Pavement_Sensor »pavementSensorTemperatureDepth.Passive_Pavement_Sensor	Pass / Fail (Sec. 4.2.6.b)
34	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Passive_Pavement_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.8)
45	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Passive_Pavement_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.8)
56	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.8)
67	VERIFY that the RESPONSE VALUE for essPavementTemperature.Passive_Pavement_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.9)
78	VERIFY that the RESPONSE VALUE for essPavementTemperature.Passive_Pavement_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.9)
89	VERIFY that the RESPONSE VALUE for essPavementTemperature.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.9)
910	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.Passive_Pavement_Sensor is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.11)
1011	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.Passive_Pavement_Sensor is less than or equal to 65535.	Pass / Fail (Sec. 5.11.3.11)
1112	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.11)
1213	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.Passive_Pavement_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.3.13)

1314	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.Passive_Pavement_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.3.13)
1415	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.13)
1516	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.Passive_Pavement_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.14)
1617	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.Passive_Pavement_Sensor is less than or equal to 4.	Pass / Fail (Sec. 5.11.3.14)
1718	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.14)
1819	VERIFY that the RESPONSE VALUE for essPavementSensorError.Passive_Pavement_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.3.15)
1920	VERIFY that the RESPONSE VALUE for essPavementSensorError.Passive_Pavement_Sensor is less than or equal to 6.	Pass / Fail (Sec. 5.11.3.15)
2021	VERIFY that the RESPONSE VALUE for essPavementSensorError.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.15)
22	IF Device_Version is greater than 1, then proceed to Step 22.1; otherwise, proceed to Step 23.1.	
22.1	GET the following object(s): »essSurfaceIceOrWaterDepth.Passive_Pavement_Sensor »essSurfaceConductivityV2.Passive_Pavement_Sensor »pavementSensorTemperatureDepth.Passive_Pavement_Sensor	Pass / Fail (Sec. 4.2.6.c)
2122.2	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.Passive_Pavement_Sensor is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.16)
22.3	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.Passive_Pavement_Sensor is less than or equal to 65535.	Pass / Fail (Sec. 5.11.3.16)
2322.4	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.16)
2422.5	VERIFY that the RESPONSE VALUE for essSurfaceConductivityV2.Passive_Pavement_Sensor is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.17)
2522.6	VERIFY that the RESPONSE VALUE for essSurfaceConductivityV2.Passive_Pavement_Sensor is less than or equal to 65535.	Pass / Fail (Sec. 5.11.3.17)
2622.7	VERIFY that the RESPONSE VALUE for essSurfaceConductivityV2.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.17)
2722.8	VERIFY that the RESPONSE VALUE for pavementSensorTemperatureDepth.Passive_Pavement_Sensor is greater than or equal to 2.	Pass / Fail (Sec. 5.11.3.19)
2822.9	VERIFY that the RESPONSE VALUE for pavementSensorTemperatureDepth.Passive_Pavement_Sensor is less than or equal to 11.	Pass / Fail (Sec. 5.11.3.19)
2922.10	VERIFY that the RESPONSE VALUE for pavementSensorTemperatureDepth.Passive_Pavement_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.19)

	GO TO Step 24.	
23.1	GET the following object(s): »essSurfaceIceOrWaterDepth.Passive_Pavement_Sensor »essSurfaceConductivityV2.Passive_Pavement_Sensor »pavementSensorTemperatureDepth.Passive_Pavement_Sensor VERIFY that the RESPONSE ERROR is equal to 'NoSuchName'.	Pass / Fail (Sec. 4.2.6.c)
3024	GET the following object(s): »numEssTreatments.0	Pass / Fail (Sec. 4.2.6.dc)
3125	Determine the RESPONSE VALUE for numEssTreatments.0. RECORD this information as: »Num_Products	
3226	FOR EACH value, N, from 1 to Num_Products, perform Steps 32.1 through 32.10.	
3226.1	GET the following object(s): »essPaveTreatProductType.N »essPaveTreatProductForm.N »essPercentProductMix.N	Pass / Fail (Sec. 4.2.6.de)
3226.2	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.N is greater than or equal to 1.	Pass / Fail (Sec. 5.13.3.2)
3226.3	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.N is less than or equal to 14.	Pass / Fail (Sec. 5.13.3.2)
3226.4	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.N is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.2)
3226.5	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.N is greater than or equal to 1.	Pass / Fail (Sec. 5.13.3.3)
3226.6	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.N is less than or equal to 4.	Pass / Fail (Sec. 5.13.3.3)
3226.7	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.N is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.3)
3226.8	VERIFY that the RESPONSE VALUE for essPercentProductMix.N is greater than or equal to 0.	Pass / Fail (Sec. 5.13.3.4)
3226.9	VERIFY that the RESPONSE VALUE for essPercentProductMix.N is less than or equal to 100.	Pass / Fail (Sec. 5.13.3.4)
3226.10	VERIFY that the RESPONSE VALUE for essPercentProductMix.N is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.4 Retrieve Adjacent Snow Depth

Test Case: 4.4	***Title:***	*Retrieve Adjacent Snow Depth*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current depth of snow adjacent to the traveled way.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essAdjacentSnowDepth.0	Pass / Fail (Sec. 3.5.2.3.3.4)
2	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.3)
3	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is less than or equal to 3001.	Pass / Fail (Sec. 5.8.3)
4	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.5 Retrieve Roadway Snow Depth

Test Case: 4.5	***Title:***	*Retrieve Roadway Snow Depth*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current depth of snow and packed snow on the traveled way.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essRoadwaySnowDepth.0	Pass / Fail (Sec. 3.5.2.3.3.5)
2	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.4)
3	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is less than or equal to 3001.	Pass / Fail (Sec. 5.8.4)
4	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.6 Retrieve Roadway Ice Thickness

Test Case: 4.6	***Title:***	*Retrieve Roadway Ice Thickness*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current thickness of ice on the traveled way.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essRoadwaySnowPackDepth.0 »essIceThickness.0	Pass / Fail (Sec. 3.5.2.3.3.6)
2	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.5)
3	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is less than or equal to 3001.	Pass / Fail (Sec. 5.8.5)
4	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.5)
5	VERIFY that the RESPONSE VALUE for essIceThickness.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.10)
6	VERIFY that the RESPONSE VALUE for essIceThickness.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.10)
7	VERIFY that the RESPONSE VALUE for essIceThickness.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.10)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.7 Retrieve Compressed Pavement Condition Data

Test Case: 4.7	***Title:***	*Retrieve Compressed Pavement Condition Data*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve all current pavement condition data in compressed form.*	
	Variables:	*Required_Pavement_Sensors*	*PRL 3.6.8*
		Support_Icing	*PRL 2.5.2.2.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	CONFIGURE: Determine whether the ESS is required to support icing detection per the specification (PRL 2.5.2.2.2). RECORD this information as: »Support_Icing	
3	GET the following object(s): »essPavementV3Block.0	Pass / Fail (Sec. 3.5.2.3.3.7)
4	Decode the essPavementV3Block.0 structure.	
5	VERIFY that the essPavementV3Block.0 structure was decoded without error.	Pass / Fail (Sec. 5.11.9)
6	VERIFY that the EssPavementDataV3 structure contains at least Required_Pavement_Sensors entries.	Pass / Fail (Sec. 3.6.8)
7	FOR EACH value, N, from 1 to Required_Pavement_Sensors, perform Steps 7.1 through 7.12.	
7.1	VERIFY that the RESPONSE VALUE for the essPavementSensorIndex.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.1)
7.2	VERIFY that the RESPONSE VALUE for essPavementSensorIndex.x is equal to N.	Pass / Fail (Sec. 5.11.3.1)
7.3	VERIFY that the RESPONSE VALUE for the essSurfaceStatus.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.1)
7.4	VERIFY that the RESPONSE VALUE for essSurfaceStatus.x is between 1 and 14, inclusive.	Pass / Fail (Sec. 5.11.3.7)
7.5	VERIFY that the RESPONSE VALUE for essSurfaceStatus.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.7)
7.6	VERIFY that the RESPONSE VALUE for the essSurfaceTemperature.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.1)
7.7	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.3.8)
7.8	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.8)
7.9	IF Support_Icing is equal to true, then proceed to Step 7.9.1; otherwise, proceed to Step	

	7.10.	
7.9.1	VERIFY that the RESPONSE VALUE for the essPavementTemperature.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.2 or 3.5.2.3.3.3)
7.9.2	VERIFY that the RESPONSE VALUE for essPavementTemperature.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.3.9)
7.9.3	VERIFY that the RESPONSE VALUE for essPavementTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.9)
7.9.4	VERIFY that the RESPONSE VALUE for the essSurfaceFreezePoint.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.2 or 3.5.2.3.3.3)
7.9.5	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.3.13)
7.9.6	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.13)
7.9.7	VERIFY that the RESPONSE VALUE for the essSurfaceBlackIceSignal.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.2 or 3.5.2.3.3.3)
7.9.8	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.x is between 1 and 4, inclusive.	Pass / Fail (Sec. 5.11.3.14)
7.9.9	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.14)
7.9.10	VERIFY that the RESPONSE VALUE for the essSurfaceIceOrWaterDepth.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.2 or 3.5.2.3.3.3)
7.9.11	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.x is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.11.3.16)
7.9.12	VERIFY that the RESPONSE VALUE for essSurfaceIceOrWaterDepth.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.16)
7.9.13	GET the following object(s): »essPavementSensorType.N	Pass / Fail (RFC 1157)
7.9.14	IF the RESPONSE VALUE for essPavementSensorType.Subject_Pavement_Sensor is equal to contactPassive, then proceed to Step 7.9.14.1; otherwise, proceed to Step 7.10.	
7.9.14.1	VERIFY that the RESPONSE VALUE for the essSurfaceSalinity.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.3)
7.9.14.2	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.x is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.11.3.11)
7.9.14.3	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.11)
7.9.14.4	VERIFY that the RESPONSE VALUE for the essSurfaceConductivityV2.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.3)
7.9.14.5	VERIFY that the RESPONSE VALUE for essSurfaceConductivityV2.x is between 0 and	Pass / Fail

	65535, inclusive.	(Sec. 5.11.3.17)
7.9.14.6	VERIFY that the RESPONSE VALUE for essSurfaceConductivityV2.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.17)
7.10	VERIFY that the RESPONSE VALUE for the essPavementSensorError.x field of the essPavementV3Block.0 object is present.	Pass / Fail (Sec. 3.5.2.3.3.1)
7.11	VERIFY that the RESPONSE VALUE for essPavementSensorError.x is between 1 and 6, inclusive.	Pass / Fail (Sec. 5.11.3.15)
7.12	VERIFY that the RESPONSE VALUE for essPavementSensorError.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.15)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.8 Retrieve Basic Subsurface Conditions

Test Case: 4.8	***Title:***	*Retrieve Basic Subsurface Conditions*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current subsurface information.*
	Variables:	*Required_Subsurface_Sensors* — *PRL 3.6.11*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of subsurface sensors required by the specification (PRL 3.6.11). RECORD this information as: »Required_Subsurface_Sensors	
2	GET the following object(s): »numEssSubSurfaceSensors.0	Pass / Fail (RFC 1157)
3	VERIFY that the RESPONSE VALUE for numEssSubSurfaceSensors.0 is greater than or equal to Required_Subsurface_Sensors.	Pass / Fail (Sec. 5.11.4)
4	Determine a random number between 1 and Required_Subsurface_Sensors. RECORD this information as: »Subject_Sensor	
5	GET the following object(s): »essSubSurfaceTemperature.Subject_Sensor »essSubSurfaceSensorError.Subject_Sensor	Pass / Fail (Sec. 3.5.2.3.4.1)
6	VERIFY that the RESPONSE VALUE for essSubSurfaceTemperature.Subject_Sensor is greater than or equal to -1000.	Pass / Fail (Sec. 5.11.6.5)
7	VERIFY that the RESPONSE VALUE for essSubSurfaceTemperature.Subject_Sensor is less than or equal to 1001.	Pass / Fail (Sec. 5.11.6.5)
8	VERIFY that the RESPONSE VALUE for essSubSurfaceTemperature.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.5)
9	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorError.Subject_Sensor is greater than or equal to 1.	Pass / Fail (Sec. 5.11.6.7)
10	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorError.Subject_Sensor is less than or equal to 5.	Pass / Fail (Sec. 5.11.6.7)
11	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorError.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.7)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.9 Retrieve Subsurface Moisture

Test Case: 4.9	***Title:***	*Retrieve Subsurface Moisture*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the amount of moisture currently present in the subsurface of the roadway.*
	Variables:	*Required_Subsurface_Sensors* — *PRL 3.6.11*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of subsurface sensors required by the specification (PRL 3.6.11). RECORD this information as: »Required_Subsurface_Sensors	
2	Determine a random number between 1 and Required_Subsurface_Sensors. RECORD this information as: »Subject_Sensor	
3	GET the following object(s): »essSubSurfaceMoisture.Subject_Sensor	Pass / Fail (Sec. 3.5.2.3.4.2)
4	VERIFY that the RESPONSE VALUE for essSubSurfaceMoisture.Subject_Sensor is greater than or equal to 0.	Pass / Fail (Sec. 5.11.6.6)
5	VERIFY that the RESPONSE VALUE for essSubSurfaceMoisture.Subject_Sensor is less than or equal to 101.	Pass / Fail (Sec. 5.11.6.6)
6	VERIFY that the RESPONSE VALUE for essSubSurfaceMoisture.Subject_Sensor is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.6)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.4.10 Retrieve Compressed Subsurface Condition Data

Test Case: 4.10	***Title:***	*Retrieve Compressed Subsurface Condition Data*
	Description:	*This test case verifies that the ESS allows a management station to retrieve all current subsurface condition information in a compressed form.*
	Variables:	*Required_Subsurface_Sensors* — *PRL 3.6.11* *Moisture_Supported* — *PRL 3.5.2.3.4.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of subsurface sensors required by the specification (PRL 3.6.11). RECORD this information as: »Required_Subsurface_Sensors	
2	CONFIGURE: Determine whether the ESS is required to monitor subsurface moisture (PRL 3.5.2.3.4.2). RECORD this information as: »Moisture_Supported	
3	GET the following object(s): »essSubsurfaceDataessSubsurfaceBlock.0	Pass / Fail (Sec. 3.5.2.3.4.3)
4	Decode the essSubsurfaceDataessSubsurfaceBlock.0 structure.	
5	VERIFY that the essSubsurfaceBlock.0 structure was decoded without error.	Pass / Fail (Sec. 5.11.8)
6	VERIFY that the SubSurfaceSensorData structure contains at least Required_Subsurface_Sensors entries.	Pass / Fail (Sec. 3.6.11)
7	FOR EACH value, N, from 1 to Required_Subsurface_Sensors, perform Steps 7.1 through 7.9.	
7.1	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorIndex.x field of the Nth SubSurfaceSensorData field is present.	Pass / Fail (Sec. 3.5.2.3.4.1)
7.2	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorIndex.x is equal to N.	Pass / Fail (Sec. 5.11.6.1)
7.3	VERIFY that the RESPONSE VALUE for the essSubSurfaceTemperature.x field of the Nth SubSurfaceSensorData field is present.	Pass / Fail (Sec. 3.5.2.3.4.1)
7.4	VERIFY that the RESPONSE VALUE for essSubSurfaceTemperature.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.6.5)
7.5	VERIFY that the RESPONSE VALUE for essSubSurfaceTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.5)
7.6	IF Moisture_Supported is equal to 1, then proceed to Step 7.6.1; otherwise, proceed to Step 7.7.	
7.6.1	VERIFY that the RESPONSE VALUE for the essSubSurfaceMoisture.x field of the Nth SubSurfaceSensorData field is present.	Pass / Fail (Sec. 3.5.2.3.4.2)
7.6.2	VERIFY that the RESPONSE VALUE for essSubSurfaceMoisture.x is between 0 and	Pass / Fail

7.6.3	VERIFY that the RESPONSE VALUE for essSubSurfaceMoisture.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.6)
7.7	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorError.x field of the Nth SubSurfaceSensorData field is present.	Pass / Fail (Sec. 3.5.2.3.4.1)
7.8	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorError.x is between 1 and 5, inclusive.	Pass / Fail (Sec. 5.11.6.7)
7.9	VERIFY that the RESPONSE VALUE for essSubSurfaceSensorError.x is	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.5 Situation Tests

C.2.3.5.1 Retrieve Wind Situation

Test Case: 5.1	***Title:***	*Retrieve Wind Situation*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the assessment of the wind situation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	Test Procedure	Device
1	GET the following object(s): »windSensorTableNumSensors.0	Pass / Fail (Sec. 3.5.2.3.5.1)
2	Determine the RESPONSE VALUE for windSensorTableNumSensors.0. RECORD this information as: »Supported_Wind_Sensors	
3	FOR EACH value, N, from 1 to Supported_Wind_Sensors, perform Steps 3.1 through 3.4.	
3.1	GET the following object(s): »windSensorSituation.N	Pass / Fail (Sec. 3.5.2.3.5.1)
3.2	VERIFY that the RESPONSE VALUE for windSensorSituation.N is greater than or equal to 1.	Pass / Fail (Sec. 5.6.10.10)
3.3	VERIFY that the RESPONSE VALUE for windSensorSituation.N is less than or equal to 12.	Pass / Fail (Sec. 5.6.10.10)
3.4	VERIFY that the RESPONSE VALUE for windSensorSituation.N is APPROPRIATE.	Pass / Fail (Sec. 5.6.10.10)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.5.2 Retrieve Precipitation Situation

Test Case: 5.2	***Title:***	*Retrieve Precipitation Situation*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the assessment of the precipitation situation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	Test Procedure	Device
1	GET the following object(s):	Pass / Fail

2	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.8.9)
3	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is less than or equal to 15.	Pass / Fail (Sec. 5.8.9)
4	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is APPROPRIATE.	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.5.3 Retrieve Cloud Situation

Test Case: 5.3	***Title:***	*Retrieve Cloud Situation*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the assessment of the cloud situation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essCloudSituation.0	Pass / Fail (Sec. 3.5.2.3.5.3)
2	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.9.3)
3	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is less than or equal to 5.	Pass / Fail (Sec. 5.9.3)
4	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.5.4 Retrieve Visibility Situation

Test Case: 5.4	***Title:***	*Retrieve Visibility Situation*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the assessment of the visibility situation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essVisibilitySituation.0	Pass / Fail (Sec. 3.5.2.3.5.4)
2	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.10.2)
3	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is less than or equal to 12.	Pass / Fail (Sec. 5.10.2)
4	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.5.5 Retrieve Ground State

Test Case: 5.5	***Title:***	*Retrieve Ground State*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the assessment of the ground state next to the roadway.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essMobileObservationGroundState.0	Pass / Fail (Sec. 3.5.2.3.5.5)
2	VERIFY that the RESPONSE VALUE for essMobileObservationGroundState.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.12.2)
3	VERIFY that the RESPONSE VALUE for essMobileObservationGroundState.0 is less than or equal to 18.	Pass / Fail (Sec. 5.12.2)
4	VERIFY that the RESPONSE VALUE for essMobileObservationGroundState.0 is APPROPRIATE.	Pass / Fail (Sec. 5.12.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.5.6 Retrieve Pavement State

Test Case: 5.6	***Title:***	*Retrieve Pavement State*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the assessment of the pavement state.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essMobileObservationPavement.0	Pass / Fail (Sec. 3.5.2.3.5.6)
2	VERIFY that the RESPONSE VALUE for essMobileObservationPavement.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.12.3)
3	VERIFY that the RESPONSE VALUE for essMobileObservationPavement.0 is less than or equal to 25.	Pass / Fail (Sec. 5.12.3)
4	VERIFY that the RESPONSE VALUE for essMobileObservationPavement.0 is APPROPRIATE.	Pass / Fail (Sec. 5.12.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6 Air Quality Tests

C.2.3.6.1 Retrieve Carbon Monoxide Reading

Test Case: 6.1	***Title:***	*Retrieve Carbon Monoxide Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current carbon monoxide reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essCO.0	Pass / Fail (Sec. 3.5.2.3.6.1)
2	VERIFY that the RESPONSE VALUE for essCO.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.1)
3	VERIFY that the RESPONSE VALUE for essCO.0 is less than or equal to 255.	Pass / Fail (Sec. 5.14.1)
4	VERIFY that the RESPONSE VALUE for essCO.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.2 Retrieve Carbon Dioxide Reading

Test Case: 6.2	***Title:***	*Retrieve Carbon Dioxide Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current carbon dioxide reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essCO2.0	Pass / Fail (Sec. 3.5.2.3.6.2)
2	VERIFY that the RESPONSE VALUE for essCO2.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.2)
3	VERIFY that the RESPONSE VALUE for essCO2.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.14.2)
4	VERIFY that the RESPONSE VALUE for essCO2.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.3 Retrieve Nitrous Oxide Reading

Test Case: 6.3	***Title:***	*Retrieve Nitrous Oxide Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current nitrous oxide reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essNO.0	Pass / Fail (Sec. 3.5.2.3.6.3)
2	VERIFY that the RESPONSE VALUE for essNO.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.3)
3	VERIFY that the RESPONSE VALUE for essNO.0 is less than or equal to 255.	Pass / Fail (Sec. 5.14.3)
4	VERIFY that the RESPONSE VALUE for essNO.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.4 Retrieve Nitrogen Dioxide Reading

Test Case: 6.4	***Title:***	*Retrieve Nitrogen Dioxide Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current nitrogen dioxide reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essNO2.0	Pass / Fail (Sec. 3.5.2.3.6.4)
2	VERIFY that the RESPONSE VALUE for essNO2.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.4)
3	VERIFY that the RESPONSE VALUE for essNO2.0 is less than or equal to 255.	Pass / Fail (Sec. 5.14.4)
4	VERIFY that the RESPONSE VALUE for essNO2.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.5 Retrieve Sulfur Dioxide Reading

Test Case: 6.5	***Title:***	*Retrieve Sulfur Dioxide Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current sulfur dioxide reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essSO2.0	Pass / Fail (Sec. 3.5.2.3.6.5)
2	VERIFY that the RESPONSE VALUE for essSO2.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.5)
3	VERIFY that the RESPONSE VALUE for essSO2.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.14.5)
4	VERIFY that the RESPONSE VALUE for essSO2.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.5)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.6 Retrieve Ozone Reading

Test Case: 6.6	***Title:***	*Retrieve Ozone Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current ozone reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essO3.0	Pass / Fail (Sec. 3.5.2.3.6.6)
2	VERIFY that the RESPONSE VALUE for essO3.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.6)
3	VERIFY that the RESPONSE VALUE for essO3.0 is less than or equal to 255.	Pass / Fail (Sec. 5.14.6)
4	VERIFY that the RESPONSE VALUE for essO3.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.6)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.7 Retrieve Small Particulate Matter Reading

Test Case: 6.7	***Title:***	*Retrieve Small Particulate Matter Reading*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current small particular matter reading.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essPM10.0	Pass / Fail (Sec. 3.5.2.3.6.7)
2	VERIFY that the RESPONSE VALUE for essPM10.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.14.7)
3	VERIFY that the RESPONSE VALUE for essPM10.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.14.7)
4	VERIFY that the RESPONSE VALUE for essPM10.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.7)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.6.8 Retrieve Compressed Air Quality Data

Test Case: 6.8	***Title:***	*Retrieve Compressed Air Quality Data*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve all of the current air quality data in compressed form.*	
	Variables:	*Support_CO*	*PRL3.6.13*
		Support_CO2	*PRL 3.6.14*
		Support_NO	*PRL 3.6.15*
		Support_NO2	*PRL 3.6.16*
		Support_SO2	*PRL 3.6.17*
		Support_O3	*PRL 3.6.18*
		Support_PM10	*PRL 3.6.19*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the device is required to support a carbon monoxide sensor (PRL3.6.13). RECORD this information as: »Support_CO	
2	CONFIGURE: Determine whether the device is required to support a carbon dioxide sensor (PRL 3.6.14). RECORD this information as: »Support_CO2	
3	CONFIGURE: Determine whether the device is required to support a nitrous oxide sensor (PRL 3.6.15). RECORD this information as: »Support_NO	
4	CONFIGURE: Determine whether the device is required to support a nitrogen dioxide sensor (PRL 3.6.16). RECORD this information as: »Support_NO2	
5	CONFIGURE: Determine whether the device is required to support a sulfur dioxide sensor (PRL 3.6.17). RECORD this information as: »Support_SO2	
6	CONFIGURE: Determine whether the device is required to support a ozone sensor (PRL 3.6.18). RECORD this information as: »Support_O3	
7	CONFIGURE: Determine whether the device is required to support a particulate matter sensor (PRL 3.6.19). RECORD this information as: »Support_PM10	
8	GET the following object(s): »essAirQualityDataessAirQualityBlock.0	Pass / Fail (Sec. 3.5.2.3.6.8)
9	Decode the essAirQualityDataessAirQualityBlock.0 structure.	
10	VERIFY that the essAirQualityBlock.0 structure was decoded without error.	Pass / Fail (Sec. 5.14.8)
11	IF Support_CO is equal to true, then proceed to Step 11.1; otherwise, proceed to Step 12.	

11.1	VERIFY that the RESPONSE VALUE for the essCO.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.1)
11.2	VERIFY that the RESPONSE VALUE for essCO.0 is between 0 and 255, inclusive.	Pass / Fail (Sec. 5.14.1)
11.3	VERIFY that the RESPONSE VALUE for essCO.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.1)
12	IF Support_CO2 is equal to true, then proceed to Step 12.1; otherwise, proceed to Step 13.	
12.1	VERIFY that the RESPONSE VALUE for the essCO2.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.2)
12.2	VERIFY that the RESPONSE VALUE for essCO2.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.14.2)
12.3	VERIFY that the RESPONSE VALUE for essCO2.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.2)
13	IF Support_NO is equal to true, then proceed to Step 13.1; otherwise, proceed to Step 14.	
13.1	VERIFY that the RESPONSE VALUE for the essNO.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.3)
13.2	VERIFY that the RESPONSE VALUE for essNO.0 is between 0 and 255, inclusive.	Pass / Fail (Sec. 5.14.3)
13.3	VERIFY that the RESPONSE VALUE for essNO.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.3)
14	IF Support_NO2 is equal to true, then proceed to Step 14.1; otherwise, proceed to Step 15.	
14.1	VERIFY that the RESPONSE VALUE for the essNO2.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.4)
14.2	VERIFY that the RESPONSE VALUE for essNO2.0 is between 0 and 255, inclusive.	Pass / Fail (Sec. 5.14.4)
14.3	VERIFY that the RESPONSE VALUE for essNO2.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.4)
15	IF Support_SO2 is equal to true, then proceed to Step 15.1; otherwise, proceed to Step 16.	
15.1	VERIFY that the RESPONSE VALUE for the essSO2.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.5)
15.2	VERIFY that the RESPONSE VALUE for essSO2.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.14.5)
15.3	VERIFY that the RESPONSE VALUE for essSO2.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.5)
16	IF Support_O3 is equal to true, then proceed to Step 16.1; otherwise, proceed to Step 17.	

16.1	VERIFY that the RESPONSE VALUE for the essO3.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.6)
16.2	VERIFY that the RESPONSE VALUE for essO3.0 is between 0 and 255, inclusive.	Pass / Fail (Sec. 5.14.6)
16.3	VERIFY that the RESPONSE VALUE for essO3.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.6)
17	IF Support_PM10 is equal to true, then proceed to Step 17.1; otherwise, proceed to EXIT.	
17.1	VERIFY that the RESPONSE VALUE for the essPM10.0 field of the essAirQualityData field is present.	Pass / Fail (Sec. 3.5.2.3.6.7)
17.2	VERIFY that the RESPONSE VALUE for essPM10.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.14.7)
17.3	VERIFY that the RESPONSE VALUE for essPM10.0 is APPROPRIATE.	Pass / Fail (Sec. 5.14.7)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.7 Water Level and Snapshot Tests

C.2.3.7.1 Retrieve Water Level

Test Case: 7.1	***Title:***	*Retrieve Water Level*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current depth of water at defined locations.*	
	Variables:	*Required_Sensors*	*PRL 3.6.22*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of water level sensors required by the specification (PRL 3.6.22). RECORD this information as: »Required_Sensors	
2	GET the following object(s): »waterLevelSensorTableNumSensors.0	Pass / Fail (Sec. 3.5.2.3.7)
3	VERIFY that the RESPONSE VALUE for waterLevelSensorTableNumSensors.0 is greater than or equal to Required_Sensors.	Pass / Fail (Sec. 3.6.22)
4	Determine the RESPONSE VALUE for waterLevelSensorTableNumSensors.0. RECORD this information as: »Supported_Sensors	
5	FOR EACH value, N, from 1 to Supported_Sensors, perform Steps 5.1 through 5.4.	
5.1	GET the following object(s): »waterLevelSensorReading.N	Pass / Fail (Sec. 3.5.2.3.7)
5.2	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.N is greater than or equal to 0.	Pass / Fail (Sec. 5.8.21.2)
5.3	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.N is less than or equal to 65535.	Pass / Fail (Sec. 5.8.21.2)
5.4	VERIFY that the RESPONSE VALUE for waterLevelSensorReading.N is APPROPRIATE.	Pass / Fail (Sec. 5.8.21.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.7.2 Retrieve Snapshot

Test Case: 7.2	***Title:***	*Retrieve Snapshot*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve a specified snapshot image.*	
	Variables:	*Required_Cameras*	*PRL 3.6.20*
		FTP_Username	*3.5.2.3.8*
		FTP_Password	*PRL 3.5.2.3.8*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	PRE-CONDITION: The controller shall have at least one image stored.	
2	CONFIGURE: Determine the number of snapshot cameras required by the specification (PRL 3.6.20). RECORD this information as: »Required_Cameras	
3	CONFIGURE: Determine the user name and password for the FTP connection. RECORD this information as: »FTP_Username (the FTP username as defined in Sec. 3.5.2.3.8) »FTP_Password (the FTP password as defined—PRL 3.5.2.3.8)	
4	GET the following object(s): »essSnapshotNumberOfCameras.0	Pass / Fail (Sec. 4.2.11)
5	VERIFY that the RESPONSE VALUE for essSnapshotNumberOfCameras.0 is greater than or equal to Required_Cameras.	Pass / Fail (Sec. 3.6.20)
6	Determine a random number between 1 and Required_Cameras. RECORD this information as: »Subject_Camera	
7	FOR EACH value, N, from 1 to Required_Cameras, perform Steps 7.1 through 7.2.	
7.1	GET the following object(s): »essSnapshotCameraDescription.N »essSnapshotCameraStoragePath.N	Pass / Fail (Sec. 4.2.11)
7.2	GET the following object(s): »essSnapshotCameraFilename.N	Pass / Fail (Sec. 4.2.11)
7.23	IF N is equal to Subject_Camera, then proceed to Step 7.2.1; otherwise, proceed to Step 8.	
7.32.1	Determine the RESPONSE VALUE for essSnapshotCameraStoragePath.N. RECORD this information as: »Subject_Directory	
8	Create an FTP connection to the Subject_Directory directory using FTP_Username as the username and FTP_Password as the password.	Pass / Fail (Sec. 4.2.2)
9	List all of the files in the following directory: Subject_Directory.	Pass / Fail (Sec. 4.2.2)
10	Calculate the file specification for the image files in the subject directory (e.g., Subject_Directory + '*.jpg'). RECORD this information as: »Delete_File_Spec	

11	Delete Delete_File_Spec using FTP.	Pass / Fail (Sec. 4.2.3)
12	List all of the files in the following directory: Subject_Directory. VERIFY that there are 0 files following the file format: "*.jpg".	Pass / Fail (Sec. 4.2.3)
13	SET the following object(s) to the value(s) shown: »essSnapshotCameraCommand.Subject_Camera = 'captureSnapshot' VERIFY that there is no response.	Pass / Fail (Sec. 4.2.1)
14.1	GET the following object(s): »essSnapshotCameraCommand.Subject_Camera	Pass / Fail (Sec. 4.2.1)
14	IF the RESPONSE VALUE for essSnapshotCameraCommand.Subject_Camera is equal to 'captureSnapshot', then proceed to Step 14.1; otherwise, proceed to Step 15.	
154	GET the following object(s): »essSnapshotCameraError.Subject_Camera	Pass / Fail (Sec. 4.2.1)
165	VERIFY that the RESPONSE VALUE for essSnapshotCameraError.Subject_Camera is equal to 'none'.	Pass / Fail (Sec. 4.2.1)
176	List all of the files in the following directory: Subject_Directory.	Pass / Fail (Sec. 4.2.2)
187	Determine the name of the image file in the directory. RECORD this information as: »Subject_File_Name	
198	Retrieve Subject_File_Name using FTP.	Pass / Fail (Sec. 4.2.2)
2019	Delete Subject_File_Name using FTP.	Pass / Fail (Sec. 4.2.3)
210	POST-CONDITION: All image files previously stored on the device have been deleted from the Subject_Camera's directory.	

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8 Pavement Treatment Tests

C.2.3.8.1 Retrieve Stationary Pavement Treatment Configuration

Test Case: 8.1	***Title:***	*Retrieve Stationary Pavement Treatment Configuration*	
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current pavement treatment configuration for a stationary device.*	
	Variables:	*Required_Treatments*	*PRL 3.6.12*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of treatments that the specification requires the device to support (PRL 3.6.12). RECORD this information as: »Required_Treatments	
2	GET the following object(s): »numEssTreatments.0	Pass / Fail (Sec. 4.2.5)
3	VERIFY that the RESPONSE VALUE for numEssTreatments.0 is greater than or equal to Required_Treatments.	Pass / Fail (Sec. 3.6.12)
4	Determine the RESPONSE VALUE for numEssTreatments.0. RECORD this information as: »Num_Treatments	
5	Determine the initial percentage to be equal to 0. RECORD this information as: »Total_Percent	
6	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 6.1 through 6.12.	
6.1	GET the following object(s): »essPaveTreatProductType.N »essPaveTreatProductForm.N »essPercentProductMix.N	Pass / Fail (Sec. 4.2.5)
6.2	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.N is greater than or equal to 1.	Pass / Fail (Sec. 5.13.3.2)
6.3	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.N is less than or equal to 14.	Pass / Fail (Sec. 5.13.3.2)
6.4	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.N is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.2)
6.5	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.N is greater than or equal to 1.	Pass / Fail (Sec. 5.13.3.3)
6.6	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.N is less than or equal to 4.	Pass / Fail (Sec. 5.13.3.3)
6.7	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.N is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.3)
6.8	VERIFY that the RESPONSE VALUE for essPercentProductMix.N is greater than or	Pass / Fail

6.9	VERIFY that the RESPONSE VALUE for essPercentProductMix.N is less than or equal to 100.	Pass / Fail (Sec. 5.13.3.4)
6.10	VERIFY that the RESPONSE VALUE for essPercentProductMix.N is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.4)
6.11	Determine the RESPONSE VALUE for essPercentProductMix.N. RECORD this information as: »Current_Percentage	
6.12	Determine the total of Total_Percentage and Current_Percentage. RECORD this information as: »Total_Percentage	
7	VERIFY that Total_Percentage is greater than or equal to 99.	PASS / Fail (Sec. 5.1.3.3.4)
8	VERIFY that Total_Percentage is less than or equal to 101.	Pass / Fail (Sec. 5.13.3.4)
9	GET the following object(s): »ptsSignalDuration.0 »ptsMonitoringDetectors.0	Pass / Fail (Sec. 4.2.5)
10	VERIFY that the RESPONSE VALUE for ptsSignalDuration.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.10)
11	VERIFY that the RESPONSE VALUE for ptsSignalDuration.0 is less than or equal to 3600000.	Pass / Fail (Sec. 5.13.10)
12	VERIFY that the RESPONSE VALUE for ptsSignalDuration.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.10)
13	VERIFY that ptsMonitoringDetectors.0 is a 4-byte value.	Pass / Fail (Sec. 5.13.18)
14	VERIFY that the RESPONSE VALUE for ptsMonitoringDetectors.0 is APPROPRIATE.	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8.2 Configure Stationary Pavement Treatment System

Test Case: 8.2	***Title:***	*Configure Stationary Pavement Treatment System*	
	Description:	*This test case verifies that the ESS allows a management station to configure the pavement treatment configuration for a stationary system.*	
	Variables:	*Required_Treatments*	*PRL 3.6.12*
		Selected_Detectors	
		Duration	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of treatments that the ESS is required to support per the specification (PRL 3.6.12). RECORD this information as: »Required_Treatments	
2	CONFIGURE: Determine An octet string representing the detector numbers that should be used to control the PTS. RECORD this information as: »Selected_Detectors	
3	CONFIGURE: Determine the duration for which the device should signal the PTS unit. RECORD this information as: »Duration	
4	GET the following object(s): »numEssTreatments.0	Pass / Fail (Sec. 4.2.7)
5	VERIFY that the RESPONSE VALUE for numEssTreatments.0 is greater than or equal to Required_Treatments.	Pass / Fail (Sec. 3.6.12)
6	Determine the RESPONSE VALUE for numEssTreatments.0. RECORD this information as: »Num_Treatments	
7	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 7.1 through 7.4.	
7.1	Determine values to test for product type and form. RECORD this information as: »Product_Type (a random number between 2 and 14) »Product_Form (a random number between 2 and 4)	
7.2	SET-UP: GET the following object(s): »essPaveTreatProductType.N »essPaveTreatProductForm.N	
7.3	Determine the retrieved values. RECORD this information as: »Orig_Type[N] »Orig_Form[N]	
7.4	SET the following object(s) to the value(s) shown: »essPaveTreatProductType.N = Product_Type »essPaveTreatProductForm.N = Product_Form	Pass / Fail (Sec. 4.2.7)
8	Determine the initial percentage available to be one hundred (100). RECORD this information as: »Percentage_Available	
9	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 9.1 through 9.5.	

9.1	IF N is equal to Num_Treatments, then proceed to Step 9.1.1; otherwise, proceed to Step 9.2.1.	
9.1.1	Determine the value stored in Percentage_Available. RECORD this information as: »Current_Percentage GO TO Step 9.3.	
9.2.1	Determine a random number between 0 and Percentage_Available. RECORD this information as: »Current_Percentage	
9.2.2	Determine the value defined by Percentage_Available minus the Current_Percentage. RECORD this information as: »Percentage_Available	
9.3	SET-UP: GET the following object(s): »essPercentProductMix.N	
9.4	Determine the RESPONSE VALUE for essPercentProductMix.N. RECORD this information as: »Orig_Mix[N]	
9.5	SET the following object(s) to the value(s) shown: »essPercentProductMix.N = Current_Percentage	Pass / Fail (Sec. 4.2.7)
10	SET-UP: GET the following object(s): »ptsSignalDuration.0 »ptsMonitoringDetectors.0	
11	Determine the retrieved values. RECORD this information as: »Orig_Duration »Orig_Detectors	
12	SET the following object(s) to the value(s) shown: »ptsSignalDuration.0 = Duration »ptsMonitoringDetectors.0 = Selected_Detectors	Pass / Fail (Sec. 4.2.7)
13	FOR EACH value, N, from 1 to Num_Treatments, perform Step 13.1.	
13.1	SET the following object(s) to the value(s) shown: »essPaveTreatProductType.N = Orig_Type[N] »essPaveTreatProductForm.N = Orig_Form[N]	Pass / Fail (RFC 1157)
14	FOR EACH value, N, from 1 to Num_Treatments, perform Step 14.1.	
14.1	SET the following object(s) to the value(s) shown: »essPercentProductMix.N = Orig_Percentage[N]	Pass / Fail (RFC 1157)
15	SET the following object(s) to the value(s) shown: »ptsSignalDuration.0 = Orig_Duration »ptsMonitoringDetectors.0 = Orig_Detectors	Pass / Fail (RFC 1157)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8.3 Retrieve Mobile Pavement Treatment Configuration

Test Case: 8.3	***Title:***	*Retrieve Mobile Pavement Treatment Configuration*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the current pavement treatment configuration for a mobile platform.*
	Variables:	*Required_Treatments* — *PRL 3.6.12*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of treatments that the specification requires the device to support (PRL 3.6.12). RECORD this information as: »Required_Treatments	
2	GET the following object(s): »numEssTreatments.0	Pass / Fail (RFC 1157)
3	VERIFY that the RESPONSE VALUE for numEssTreatments.0 is greater than or equal to Required_Treatments.	Pass / Fail (Sec. 3.6.12)
4	Determine the RESPONSE VALUE for numEssTreatments.0. RECORD this information as: »Num_Treatments	
5	GET the following object(s): »pavementTreatmentBlock.0	Pass / Fail (Sec. 3.5.3.1.3)
6	Decode the pavementTreatmentBlock.0 structure.	
7	VERIFY that the RESPONSE VALUE for the treatmentInfo field of the pavementTreatmentBlock.0 object is present.	Pass / Fail (Sec. 5.13.6)
8	VERIFY that the treatmentInfo field of the pavementTreatmentBlock.0 object contains Num_Treatments entries.	Pass / Fail (Sec. 3.6.12)
9	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 9.1 through 9.11.	
9.1	VERIFY that the RESPONSE VALUE for the essPavementTreatmentIndex.x field of the treatmentInfo field is present.	Pass / Fail (Sec. 5.13.6)
9.2	VERIFY that the essPavementTreatmentIndex.x field is equal to N.	Pass / Fail (Sec. 5.13.6)
9.3	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.x field of the treatmentInfo field is present.	Pass / Fail (Sec. 5.13.6)
9.4	VERIFY that the essPaveTreatProductType.x field is between 1 and 14.	Pass / Fail (Sec. 5.13.3.2)
9.5	VERIFY that the RESPONSE VALUE for essPaveTreatProductType.x is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.2)
9.6	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.x field of the treatmentInfo field is present.	Pass / Fail (Sec. 5.13.6)
9.7	VERIFY that the essPaveTreatProductForm.x field is between 1 and 4.	Pass / Fail (Sec. 5.13.3.3)

9.8	VERIFY that the RESPONSE VALUE for essPaveTreatProductForm.x is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.3)
9.9	VERIFY that the RESPONSE VALUE for the essPercentProductMix.x field of the treatmentInfo field is present.	Pass / Fail (Sec. 5.13.6)
9.10	VERIFY that the essPercentProductMix.x field is between 0 and 100.	Pass / Fail (Sec. 5.13.3.4)
9.11	VERIFY that the RESPONSE VALUE for essPercentProductMix.x is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.4)
10	VERIFY that the RESPONSE VALUE for the essPaveTreatmentAmount.0 field of the pavementTreatmentBlock.0 object is present.	Pass / Fail (Sec. 5.13.6)
11	VERIFY that the essPaveTreatmentAmount.0 field is between 0 and 255.	Pass / Fail (Sec. 5.13.4)
12	VERIFY that the RESPONSE VALUE for essPaveTreatmentAmount.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.4)
13	VERIFY that the RESPONSE VALUE for the essPaveTreatmentWidth.0 field of the pavementTreatmentBlock.0 object is present.	Pass / Fail (Sec. 5.13.6)
14	VERIFY that the essPaveTreatmentWidth.0 field is between 0 and 255.	Pass / Fail (Sec. 5.13.5)
15	VERIFY that the RESPONSE VALUE for essPaveTreatmentWidth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.5)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8.4 Configure Mobile Pavement Treatment System

Test Case: 8.4	***Title:***	*Configure Mobile Pavement Treatment System*	
	Description:	*This test case verifies that the ESS allows a management station to configure the pavement treatment configuration for a mobile system.*	
	Variables:	*Required_Treatments*	
		Selected_Detectors	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	Test Procedure	Device
1	CONFIGURE: Determine the number of treatments that the ESS is required to support per the specification (). RECORD this information as: »Required_Treatments	
2	CONFIGURE: Determine the detector numbers that should be used to control the PTS. RECORD this information as: »Selected_Detectors	
3	GET the following object(s): »numEssTreatments.0	Pass / Fail (Sec. 4.2.9)
4	VERIFY that the RESPONSE VALUE for numEssTreatments.0 is greater than or equal to Required_Treatments.	Pass / Fail (Sec. 3.6.12)
5	Determine the RESPONSE VALUE for numEssTreatments.0. RECORD this information as: »Num_Treatments	
6	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 6.1 through 6.2.	
6.1	Determine the product type and form to use in the test. RECORD this information as: »Product_Type (a random number between 2 and 14) »Product_Form (a random number between 2 and 4)	
6.2	SET the following object(s) to the value(s) shown: »essPaveTreatProductType.N = Product_Type »essPaveTreatProductForm.N = Product_Form	Pass / Fail (Sec. 4.2.9)
7	Determine the initial percentage available to be one hundred (100). RECORD this information as: »Percentage_Available	
8	FOR EACH value, N, from 1 to Num_Treatments, perform Steps 8.1 through 8.3.	
8.1	IF N is equal to Num_Treatments, then proceed to Step 8.1.1; otherwise, proceed to Step 8.2.1.	
8.1.1	Determine the value stored in Percentage_Available. RECORD this information as: »Current_Percentage GO TO Step 8.3.	
8.2.1	Determine a random number between 0 and Percentage_Available. RECORD this information as: »Current_Percentage	

8.2.2	Determine the value defined by Percentage_Available minus the Current_Percentage. RECORD this information as: »Percentage_Available	
8.3	SET the following object(s) to the value(s) shown: »essPercentProductMix.N = Current_Percentage	Pass / Fail (Sec. 4.2.9)
9	Determine test values for the amount and width. RECORD this information as: »Amount (a random amount of chemical to be sprayed between 0 and 255) »Width (a random width to spray between 0 and 255)	
10	SET the following object(s) to the value(s) shown: »essPaveTreatmentAmount.0 = Amount »essPaveTreatmentWidth.0 = Width	Pass / Fail (Sec. 4.2.9)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8.5 Retrieve Pavement Treatment Status

Test Case: 8.5	***Title:***	*Retrieve Pavement Treatment Status*
	Description:	*This test case verifies that the ESS allows a management station to retrieve the status of the pavement treatment system.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »ptsSprayerState.0 »ptsSignalEventCount.0 »ptsLastSignalEvent.0 »ptsActiveEventCount.0 »ptsInactiveEventCount.0 »ptsLastActiveEvent.0 »ptsLastInactiveEvent.0 »ptsError.0	Pass / Fail (Sec. 3.5.3.2.1)
2	VERIFY that the RESPONSE VALUE for ptsSprayerState.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.13.9)
3	VERIFY that the RESPONSE VALUE for ptsSprayerState.0 is less than or equal to 3.	Pass / Fail (Sec. 5.13.9)
4	VERIFY that the RESPONSE VALUE for ptsSprayerState.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.9)
5	VERIFY that the RESPONSE VALUE for ptsSignalEventCount.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.11)
6	VERIFY that the RESPONSE VALUE for ptsSignalEventCount.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.13.11)
7	VERIFY that the RESPONSE VALUE for ptsSignalEventCount.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.11)
8	VERIFY that the RESPONSE VALUE for ptsLastSignalEvent.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.12)
9	VERIFY that the RESPONSE VALUE for ptsLastSignalEvent.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.13.12)
10	VERIFY that the RESPONSE VALUE for ptsLastSignalEvent.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.12)
11	VERIFY that the RESPONSE VALUE for ptsActiveEventCount.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.13)
12	VERIFY that the RESPONSE VALUE for ptsActiveEventCount.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.13.13)
13	VERIFY that the RESPONSE VALUE for ptsActiveEventCount.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.13)
14	VERIFY that the RESPONSE VALUE for ptsInactiveEventCount.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.14)

15	VERIFY that the RESPONSE VALUE for ptsInactiveEventCount.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.13.14)
16	VERIFY that the RESPONSE VALUE for ptsInactiveEventCount.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.14)
17	VERIFY that the RESPONSE VALUE for ptsLastActiveEvent.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.15)
18	VERIFY that the RESPONSE VALUE for ptsLastActiveEvent.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.13.15)
19	VERIFY that the RESPONSE VALUE for ptsLastActiveEvent.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.15)
20	VERIFY that the RESPONSE VALUE for ptsLastInactiveEvent.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.13.16)
21	VERIFY that the RESPONSE VALUE for ptsLastInactiveEvent.0 is less than or equal to 4294967295.	Pass / Fail (Sec. 5.13.16)
22	VERIFY that the RESPONSE VALUE for ptsLastInactiveEvent.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.16)
23	VERIFY that the RESPONSE VALUE for ptsError.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.13.17)
24	VERIFY that the RESPONSE VALUE for ptsError.0 is less than or equal to 4.	Pass / Fail (Sec. 5.13.17)
25	VERIFY that the RESPONSE VALUE for ptsError.0 is APPROPRIATE.	Pass / Fail (Sec. 5.13.17)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8.6 Set PTS Operational Mode

Test Case: 8.6	***Title:***	*Set PTS Operational Mode*
	Description:	*This test case verifies that the ESS allows a management station to set the operational mode of the pavement treatment system to off, manual, and automatic.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	SET-UP: GET the following object(s): »ptsOperationalModeV3.0	Pass / Fail (RFC 1157)
2	Determine the RESPONSE VALUE for ptsOperationalModeV3.0. RECORD this information as: »Orig_State	
3	SET the following object(s) to the value(s) shown: »ptsOperationalModeV3.0 = 'off'	Pass / Fail (Sec. 3.5.3.4.1)
4	DELAY for 2 seconds.	
5	SET the following object(s) to the value(s) shown: »ptsOperationalModeV3.0 = 'manual'	Pass / Fail (Sec. 3.5.3.4.1)
6	DELAY for 2 seconds.	
7	SET the following object(s) to the value(s) shown: »ptsOperationalModeV3.0 = 'automatic'	Pass / Fail (Sec. 3.5.3.4.1)
8	DELAY for 2 seconds.	
9	SET the following object(s) to the value(s) shown: »ptsOperationalModeV3.0 = Orig_State	Pass / Fail (Sec. 3.5.3.4.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.8.7 Manually Activate PTS Sprayer

Test Case: 8.7	***Title:***	*Manually Activate PTS Sprayer*
	Description:	*This test case verifies that the ESS allows a management station to manually activate the sprayer.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	SET-UP: GET the following object(s): »ptsOperationalModeV3.0 »ptsSignalDuration.0	Pass / Fail (RFC 1157)
2	Determine the RESPONSE VALUE for ptsOperationalModeV3.0. RECORD this information as: »Orig_Mode	
3	Determine the RESPONSE VALUE for ptsSignalDuration.0. RECORD this information as: »Orig_Duration	
4	SET-UP: SET the following object(s) to the value(s) shown: »ptsSignalDuration.0 = 1000	Pass / Fail (RFC 1157)
5	SET-UP: SET the following object(s) to the value(s) shown: »ptsOperationalModeV3.0 = 'manual'	Pass / Fail (RFC 1157)
6	SET the following object(s) to the value(s) shown: »ptsCommandStateV3.0 = 'activate'	Pass / Fail (Sec. 3.5.3.4.2)
7	VERIFY that the PTS signals the sprayer sprayer sprays the product mixture for a period of 1.000 seconds.	Pass / Fail (Sec. 3.5.3.4.2)
8	Delay for 1 seconds. VERIFY that the sprayer sprayed the product mixture.	Pass / Fail (Sec. 3.5.3.4.2)
9	SET-UP: GET the following object(s): »ptsCommandStateV3.0	Pass / Fail (RFC 1157)
10	SET-UP: VERIFY that the RESPONSE VALUE for ptsCommandStateV3.0 is equal to 'inactive'.	Pass / Fail (Sec. 5.13.8)
11	SET the following object(s) to the value(s) shown: »ptsOperationalModeV3.0 = Orig_Mode »ptsSignalDuration.0 = Orig_Duration	Pass / Fail (RFC 1157)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9 Event Logging Tests

C.2.3.9.1 Determine Capabilities of Event Logging Service

Test Case: 9.1	***Title:***	*Determine Capabilities of Event Logging Service*	
	Description:	*This test case verifies that the device indicates that it supports the logging capabilities required by the specification*	
	Variables:	*Required_Event_Classes*	*PRL F.2.3.1.2*
		Required_Event_Configurations	*PRL F.2.3.1.3*
		Required_Event_Log_Size	*PRL F.2.3.1.6*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of event classes required by the specification (PRL F.2.3.1.2). RECORD this information as: »Required_Event_Classes	
2	CONFIGURE: Determine the number of event configurations required by the specification (PRL F.2.3.1.3). RECORD this information as: »Required_Event_Configurations	
3	CONFIGURE: Determine the number of events that the log is required to be able to store (PRL F.2.3.1.6). RECORD this information as: »Required_Event_Log_Size	
4	GET the following object(s): »maxEventClasses.0 »maxEventLogConfigs.0 »maxEventLogSize.0	Pass / Fail (Sec. 3.3.2.5)
5	VERIFY that the RESPONSE VALUE for maxEventClasses.0 is greater than or equal to Required_Event_Classes.	Pass / Fail (Sec. D.3.2.2.2)
6	VERIFY that the RESPONSE VALUE for maxEventLogConfigs.0 is greater than or equal to Required_Event_Configurations.	Pass / Fail (Sec. D.3.2.2.3)
7	VERIFY that the RESPONSE VALUE for maxEventLogSize.0 is greater than or equal to Required_Event_Log_Size.	Pass / Fail (Sec. D.3.2.2.6)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.2 Configure Event Log

Test Case: 9.2	***Title:***	*Configure Event Log*	
	Description:	*This test case configures the event log according to the tester inputs and ensures that the values were accepted and implemented in the device.*	
	Variables:	*Required_Event_Classes*	*PRL F.2.3.1.2*
		Required_Number_Of_Events	*PRL F.2.3.1.6*
		Class_Clear_Time	
		Class_Description	
		Event_Index	
		Event_Mode	
		Event_Compare_Value1	
		Event_Compare_Value2	
		Event_Watch_Object	
		Event_Log_Object	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the event class to use for this test (PRL F.2.3.1.2). RECORD this information as: »Required_Event_Classes	
2	CONFIGURE: Determine the number of events that the device is required to support (PRL F.2.3.1.6). RECORD this information as: »Required_Number_Of_Events	
3	CONFIGURE: Determine the time from which all earlier logs will be cleared (e.g., per the test plan). RECORD this information as: »Class_Clear_Time	
4	CONFIGURE: Determine the description to be used for the log class (e.g., per the test plan). RECORD this information as: »Class_Description	
5	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
6	CONFIGURE: Determine the mode for the event (e.g., the comparison operator) (e.g., per the test plan). RECORD this information as: »Event_Mode	
7	CONFIGURE: Determine the first comparison value for the event (e.g., per the test plan). RECORD this information as: »Event_Compare_Value1	
8	CONFIGURE: Determine the second comparison value for the event (e.g., per the test plan). RECORD this information as: »Event_Compare_Value2	
9	CONFIGURE: Determine the object to which the value will be compared (e.g., per the test plan). RECORD this information as: »Event_Watch_Object	

10	CONFIGURE: Determine the object that should be logged upon the detection of the event (e.g., per the test plan). RECORD this information as: »Event_Log_Object	
11	Calculate a random value between 1 and Required_Event_Classes. RECORD this information as: »Class_Index	
12	Calculate a random value between 1 and Required_Number_Of_Events. RECORD this information as: »Class_Size_Limit	
13	SET the following object(s) to the value(s) shown: »eventClassLimit.Class_Index = Class_Size_Limit »eventClassClearTime.Class_Index = Class_Clear_Time »eventClassDescription.Class_Index = Class_Description	Pass / Fail (Sec. 3.3.2.2)
14	SET the following object(s) to the value(s) shown: »eventConfigClass.Event_Index = Class_Index »eventConfigMode.Event_Index = Event_Mode »eventConfigCompareValue.Event_Index = Event_Compare_Value1 »eventConfigCompareValue2.Event_Index = Event_Compare_Value2 »eventConfigCompareOID.Event_Index = Event_Watch_Object »eventConfigLogOID.Event_Index = Event_Log_Object »eventConfigAction.Event_Index = 'log' NOTE—Valid enumerated values for eventConfigMode are defined in NTCIP 1201 v02 Sec. 2.5.4.3.	Pass / Fail (Sec. 3.3.2.2)
15	GET the following object(s): »eventConfigStatus.Event_Index	Pass / Fail (Sec. 3.3.2.2)
16	POST-CONDITION: An event type has been configured in the controller.	

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.3 Retrieve Logged Data

Test Case: 9.3	***Title:***	*Retrieve Logged Data*
	Description:	*This test case verifies that the device allows a user to retrieve the logged data.*
	Variables:	*Class_Index* *Last_Log_Time* *Last_Log_ID*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the class for which the logged data is to be retrieved (e.g., per the test plan). RECORD this information as: »Class_Index	
2	CONFIGURE: Determine the information about the final log entry; if known (otherwise enter zeros). RECORD this information as: »Last_Log_Time (the time at or before which the last event to be logged occured) »Last_Log_ID (the ID of the last event to be logged)	
3	GET the following object(s): »eventClassNumRowsInLog.Class_Index »eventClassNumEvents.Class_Index	Pass / Fail (Sec. 3.3.2.3)
4	Determine the RESPONSE VALUE for eventClassNumRowsInLog.Class_Index. RECORD this information as: »Rows	
5	FOR EACH value, N, from 1 to Rows, perform Steps 5.1 through 5.2.	
5.1	GET the following object(s): »eventLogID.Class_Index.N »eventLogTime.Class_Index.N »eventLogValue.Class_Index.N	Pass / Fail (Sec. 3.3.2.3)
5.2	IF N is equal to Rows, then proceed to Step 5.2.1; otherwise, proceed to EXIT.	
5.2.1	IF Last_Log_Time is greater than 0, then proceed to Step 5.2.1.1; otherwise, proceed to EXIT.	
5.2.1.1	VERIFY that the RESPONSE VALUE for eventLogTime.Class_Index.N is greater than or equal to Last_Log_Time.	Pass / Fail (Sec. 3.3.2.3)
5.2.1.2	VERIFY that the RESPONSE VALUE for eventLogID.Class_Index.N is equal to Last_Log_ID.	Pass / Fail (Sec. 3.3.2.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.4 Clear Log

Test Case: 9.4	***Title:***	*Clear Log*	
	Description:	*This test case verifies that the device allows the user to clear the log for a specified class.*	
	Variables:	*Class_Index*	
		Class_Clear_Time	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the class of events to be cleared from the log (e.g., per the test plan). RECORD this information as: »Class_Index	
2	CONFIGURE: Determine the time from which all earlier logs will be cleared (e.g., per the test plan). RECORD this information as: »Class_Clear_Time	
3	SET the following object(s) to the value(s) shown: »eventClassClearTime.Class_Index = Class_Clear_Time	Pass / Fail (Sec. 3.3.2.4)
4	GET the following object(s): »eventClassNumRowsInLog.Class_Index	Pass / Fail (RFC 1157)
5	Determine the RESPONSE VALUE for eventClassNumRowsInLog.Class_Index. RECORD this information as: »Rows	
6	FOR EACH value, N, from 1 to Rows, perform Steps 6.1 through 6.2.	
6.1	GET the following object(s): »eventLogTime.Class_Index.N	Pass / Fail (RFC 1157)
6.2	VERIFY that the RESPONSE VALUE for eventLogTime.Class_Index.N is greater than Class_Clear_Time.	Pass / Fail (Sec. 3.3.2.4)
7	POST-CONDITION: Log entries older than Class_Clear_Time have been deleted from the log.	

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.5 Determine Total Number of Events

Test Case: 9.5	***Title:***	*Determine Total Number of Events*
	Description:	*This test case verifies that the device allows the user to determine the total number of events in the log.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »maxEventClasses.0 »maxEventLogConfigs.0 »maxEventLogSize.0	Pass / Fail (Sec. 3.3.2.5)
2	Determine the RESPONSE VALUE for maxEventClasses.0. RECORD this information as: »Max_Event_Classes	
3	Determine the value of 0. RECORD this information as: »Total_Events	
4	FOR EACH value, N, from 1 to Max_Event_Classes, perform Steps 4.1 through 4.2.	
4.1	GET the following object(s): »eventClassNumRowsInLog.N »eventClassNumEvents.N	Pass / Fail (RFC 1157)
4.2	Calculate the sum of Total_Events and eventClassNumRowsInLog.N. RECORD this information as: »Total_Events	
5	GET the following object(s): »numEvents.0	Pass / Fail (Sec. 3.3.2.6)
6	VERIFY that the RESPONSE VALUE for numEvents.0 is equal to Total_Events. NOTE—If an event occurred during this process, this condition will not hold true.	Pass / Fail (Sec. 3.3.2.6)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.6 Verify Log Limit Storage

Test Case: 9.6	***Title:***	*Verify Log Limit Storage*
	Description:	*This test case verifies that the device stores only the latest of the maximum number of events per class.*
	Variables:	*Class_Index* *Class_Size_Limit*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the class for which the logged data is to be retrieved (e.g., per the test plan). RECORD this information as: »Class_Index	
2	CONFIGURE: Determine the log size limit that the test will impose on the class (e.g., per the test plan). RECORD this information as: »Class_Size_Limit	
3	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2).	Pass / Fail
4	Create conditions to cause the device to log the event Class_Size_Limit times. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
5	GET the following object(s): »eventClassNumRowsInLog.Class_Index »eventClassNumEvents.Class_Index	Pass / Fail (Sec. 3.3.2.3)
6	VERIFY that the RESPONSE VALUE for eventClassNumRowsInLog.Class_Index is equal to Class_Size_Limit.	Pass / Fail (Sec. 3.3.2.2)
7	Determine the RESPONSE VALUE for eventClassNumRowsInLog.Class_Index. RECORD this information as: »Rows	
8	FOR EACH value, N, from 1 to Rows, perform Steps 8.1 through 8.3.	
8.1	GET the following object(s): »eventLogID.Class_Index.N »eventLogTime.Class_Index.N »eventLogValue.Class_Index.N	Pass / Fail (Sec. 3.3.2.3)
8.2	IF N is equal to 1, then proceed to Step 8.2.1; otherwise, proceed to Step 8.3.1.	
8.2.1	Determine the RESPONSE VALUE for eventLogTime.Class_Index.N. RECORD this information as: »Old_Timestamp GO TO Step 9 (after any looping logic is completed).	
8.3.1	Determine the RESPONSE VALUE for eventLogTime.Class_Index.N. RECORD this information as: »Limit_Timestamp	
9	Create conditions to cause the device to log the event one more time.	

10	GET the following object(s): »eventClassNumRowsInLog.Class_Index »eventClassNumEvents.Class_Index	Pass / Fail (Sec. 3.3.2.3)
11	VERIFY that the RESPONSE VALUE for eventClassNumRowsInLog.Class_Index is equal to Class_Size_Limit.	Pass / Fail (Sec. 3.3.2.2)
12	Determine the RESPONSE VALUE for eventClassNumRowsInLog.Class_Index. RECORD this information as: »Rows	
13	FOR EACH value, N, from 1 to Rows, perform Steps 13.1 through 13.3.	
13.1	GET the following object(s): »eventLogID.Class_Index.N »eventLogTime.Class_Index.N »eventLogValue.Class_Index.N	Pass / Fail (Sec. 3.3.2.3)
13.2	IF N is equal to 1, then proceed to Step 13.2.1; otherwise, proceed to Step 13.3.1.	
13.2.1	VERIFY that the RESPONSE VALUE for eventLogTime.Class_Index.N is greater than Old_Timestamp. GO TO Step 14 (after any looping logic is completed).	Pass / Fail (Sec. 3.3.2.3)
13.3.1	IF N is equal to Rows, then proceed to Step 13.3.1.1; otherwise, proceed to Step 14.	
13.3.1.1	VERIFY that the RESPONSE VALUE for eventLogTime.Class_Index.N is greater than Limit_Timestamp.	Pass / Fail (Sec. 3.3.2.3)
14	POST-CONDITION: The event log has been filled for subject event class.	

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.7 Verify Support for an On-Change Event

Test Case: 9.7	***Title:***	*Verify Support for an On-Change Event*
	Description:	*This test case verifies that the device allows configuration of an on-change event and the device logs events appropriately.*
	Variables:	*Event_Index*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
2	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2) with the following parameters: »Event_Mode = 2	Pass / Fail
3	GET the following object(s): »globalTime.0	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »Time	
5	Create an event for the device to log. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
6	PERFORM the test case labeled 'Retrieve Logged Data' (2.3.9.3) with the following parameters: »Last_Log_Time = Time »Last_Log_ID = Event_Index	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.8 Verify Support for a Greater Than Event

Test Case: 9.8	***Title:***	*Verify Support for a Greater Than Event*
	Description:	*This test case verifies that the device allows configuration of a greater than event and the device logs events appropriately.*
	Variables:	*Event_Index*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
2	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2) with the following parameters: »Event_Mode = 3	Pass / Fail
3	GET the following object(s): »globalTime.0	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »Time	
5	Create an event for the device to log. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
6	PERFORM the test case labeled 'Retrieve Logged Data' (2.3.9.3) with the following parameters: »Last_Log_Time = Time »Last_Log_ID = Event_Index	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.9 Verify Support for a Less Than Event

Test Case: 9.9	***Title:***	*Verify Support for a Less Than Event*
	Description:	*This test case verifies that the device allows configuration of a less than event and the device logs events appropriately.*
	Variables:	*Event_Index*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
2	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2) with the following parameters: »Event_Mode = 4	Pass / Fail
3	GET the following object(s): »globalTime.0	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »Time	
5	Create an event for the device to log. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
6	PERFORM the test case labeled 'Retrieve Logged Data' (2.3.9.3) with the following parameters: »Last_Log_Time = Time »Last_Log_ID = Event_Index	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.10 Verify Support for a Hysteresis Event

Test Case: 9.10	***Title:***	*Verify Support for a Hysteresis Event*
	Description:	*This test case verifies that the device allows configuration of a hysteresis event and the DMS logs events appropriately.*
	Variables:	*Event_Index*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
2	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2) with the following parameters: »Event_Mode = 5	Pass / Fail
3	GET the following object(s): »globalTime.0	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »Time	
5	Create an event for the device to log. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
6	PERFORM the test case labeled 'Retrieve Logged Data' (2.3.9.3) with the following parameters: »Last_Log_Time = Time »Last_Log_ID = Event_Index	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.11 Verify Support for a Periodic Event

Test Case: 9.11	***Title:***	*Verify Support for a Periodic Event*
	Description:	*This test case verifies that the device allows configuration of a Periodic event and the DMS logs events appropriately.*
	Variables:	*Event_Index*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
2	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2) with the following parameters: »Event_Mode = 6	Pass / Fail
3	GET the following object(s): »globalTime.0	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »Time	
5	Create an event for the device to log. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
6	PERFORM the test case labeled 'Retrieve Logged Data' (2.3.9.3) with the following parameters: »Last_Log_Time = Time »Last_Log_ID = Event_Index	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.12 Verify Support for a Bit-flag Event

Test Case: 9.12	***Title:***	*Verify Support for a Bit-flag Event*
	Description:	*This test case verifies that the device allows configuration of a bit-flag event and the device logs events appropriately.*
	Variables:	*Event_Index*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the index of the event type to configure as a part of the test (e.g., per the test plan). RECORD this information as: »Event_Index	
2	PERFORM the test case labeled 'Configure Event Log' (2.3.9.2) with the following parameters: »Event_Mode = 7	Pass / Fail
3	GET the following object(s): »globalTime.0	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »Time	
5	Create an event for the device to log. NOTE—This may require physically changing a sensor reading or setting an object within the device.	
6	PERFORM the test case labeled 'Retrieve Logged Data' (2.3.9.3) with the following parameters: »Last_Log_Time = Time »Last_Log_ID = Event_Index	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.9.13 Determine Configuration of Logging Service

Test Case: 9.13	***Title:***	*Determine Configuration of Logging Service*
	Description:	*This test case verifies that the device returns the configuration of the logging service.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »maxEventClasses.0 »maxEventLogConfigs.0 »maxEventLogSize.0	Pass / Fail (Sec. 3.3.2.5)
2	Determine the values retreived from the device. RECORD this information as: »Max_Event_Classes »Max_Configs »Max_Log_Size	
3	FOR EACH value, N, from 1 to Max_Event_Classes, perform Step 3.1.	
3.1	GET the following object(s): »eventClassLimit.N »eventClassClearTime.N »eventClassDescription.N	Pass / Fail (Sec. 3.3.2.1)
4	FOR EACH value, N, from 1 to Max_Configs, perform Step 4.1.	
4.1	GET the following object(s): »eventConfigClass.N »eventConfigMode.N »eventConfigCompareValue.N »eventConfigCompareValue2.N »eventConfigCompareOID.N »eventConfigLogOID.N »eventConfigAction.N »eventConfigStatus.N	Pass / Fail (Sec. 3.3.2.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10 Global Tests

C.2.3.10.1 Explore Data

Test Case: 10.1	***Title:***	*Explore Data*
	Description:	*This test case verifies that the device properly responds to a GET-NEXT request.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	Determine the OID value of 'null'. RECORD this information as: »Last_Object	
2.1	Send a GET-NEXT request for the following object(s): »Last_Object VERIFY that the RESPONSE ERROR is equal to 'noError'or 'noSuchName'.	Pass / Fail (Sec. F.3.2)
2.2	Determine whether the OID of the returned object is lexigraphically larger than the OID contained in the request. RECORD this information as: »Continue	
2.3	IF the RESPONSE ERROR is equal to noError, then proceed to Step 2.3.1; otherwise, proceed to Step 2.4.1.	
2.3.1	VERIFY that Continue is equal to true. GO TO Step 2.5.	Pass / Fail (RFC 1157)
2.4.1	VERIFY that the returned OID is identical to that sent in the request.	Pass / Fail (RFC 1157)
2.54	Determine the OID of the retrieved object. RECORD this information as: »Last_Object	
2	IF the RESPONSE ERROR is equal to noError and Continue equals true, then proceed to Step 2.1; otherwise, proceed to Step 3EXIT.	

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.2 Determine Device Component Information

Test Case: 10.2	***Title:***	*Determine Device Component Information*
	Description:	*This test case verifies that the data stored in the module table reflects the information about the device.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »globalMaxModules.0	Pass / Fail (Sec. D.3.1.1)
2	Determine the RESPONSE VALUE for globalMaxModules.0. RECORD this information as: »Num_Modules	
3	FOR EACH value, N, from 1 to Num_Modules, perform Steps 3.1 through 3.6.	
3.1	GET the following object(s): »moduleDeviceNode.N »moduleMake.N »moduleModel.N »moduleVersion.N »moduleType.N	Pass / Fail (Sec. D.3.1.1)
3.2	VERIFY that the RESPONSE VALUE for moduleDeviceNode.N is APPROPRIATE. NOTE—Should be equal to '1.3.6.1.4.1.1206.4.2.3' for DMS	Pass / Fail (Sec. D.3.1.1)
3.3	VERIFY that the RESPONSE VALUE for moduleMake.N is APPROPRIATE.	Pass / Fail (Sec. D.3.1.1)
3.4	VERIFY that the RESPONSE VALUE for moduleModel.N is APPROPRIATE.	Pass / Fail (Sec. D.3.1.1)
3.5	VERIFY that the RESPONSE VALUE for moduleVersion.N is APPROPRIATE.	Pass / Fail (Sec. D.3.1.1)
3.6	VERIFY that the RESPONSE VALUE for moduleType.N is APPROPRIATE.	Pass / Fail (Sec. D.3.1.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.3 Retrieve Device Configuration Identifier

Test Case: 10.3	***Title:***	*Retrieve Device Configuration Identifier*
	Description:	*This test case verifies that the device allows the user to retrieve a code that only changes when changes are made to the controller configuration.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »globalSetIDParameter.0	Pass / Fail (Sec. F.2.2.1.2)
2	Determine the RESPONSE VALUE for globalSetIDParameter.0. RECORD this information as: »ID	
3	DELAY for 5 seconds.	
4	GET the following object(s): »globalSetIDParameter.0	Pass / Fail (Sec. F.2.2.1.2)
5	VERIFY that the RESPONSE VALUE for globalSetIDParameter.0 is equal to ID.	Pass / Fail (NTCIP 1201 2.2.1)
6	GET the following object(s): »essNtcipSiteDescription.0	Pass / Fail (RFC 1157)
7	Determine the RESPONSE VALUE for essNtcipSiteDescription.0. RECORD this information as: »Description	
8	SET the following object(s) to the value(s) shown: »essNtcipSiteDescription.0 = 'Test Location'	Pass / Fail (Sec. 3.5.1.1.3)
9	GET the following object(s): »globalSetIDParameter.0	Pass / Fail (RFC 1157)
10	VERIFY that the RESPONSE VALUE for globalSetIDParameter.0 is not equal to ID.	Pass / Fail (NTCIP 1201 v02 2.2.1)
11	SET the following object(s) to the value(s) shown: »essNtcipSiteDescription.0 = Description	Pass / Fail (Sec. 3.5.1.1.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.4 Determine Supported Standards

Test Case: 10.4	***Title:***	*Determine Supported Standards*
	Description:	*This test case verifies that the device indicates the standards that it supports.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »controllerBaseStandards.0 NOTE—NTCIP 1203 uses hyphenated names when referring to some NTCIP 1201 objects; these test procedures have been updated to use the actual object names without hyphens.	Pass / Fail (Sec. D.3.1.4)
2	VERIFY that the RESPONSE VALUE for controllerBaseStandards.0 properly identifies the standards that the device supports and the information is presented in the correct format.	Pass / Fail (Sec. D.3.1.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.5 Retrieve System Name

Test Case: 10.5	***Title:***	*Retrieve System Name*
	Description:	*This test case verifies that the device allows the user to retrieve the system name of the device.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »sysName.0	Pass / Fail (Sec. F.2.2.1.4)
2	VERIFY that the RESPONSE VALUE for sysName.0 is APPROPRIATE.	Pass / Fail (RFC 1213 Clause 6)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.6 Set Time

Test Case: 10.6	***Title:***	*Set Time*
	Description:	*This test case verifies that the devcie allows a set to the UTC time to a new value and ensure that the new value was accepted, implemented and that the clock is still working.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	Determine the time the test started according to the test computer. RECORD this information as: »Test_Time	
2	GET the following object(s): »globalTime.0	Pass / Fail (Sec. D.3.1.2.4)
3	Determine the RESPONSE VALUE for globalTime.0. RECORD this information as: »UTC_Time	
4	Calculate the value of UTC_Time plus 7200 seconds. RECORD this information as: »New_UTC_Time	
5	SET the following object(s) to the value(s) shown: »globalTime.0 = New_UTC_Time NOTE—This advances the clock by two hours.	Pass / Fail (Sec. D.3.1.2.1)
6	Calculate UTC_Time plus 7200 plus the amount of time that has elapsed since Step 1. RECORD this information as: »Expected_Time	
7	GET the following object(s): »globalTime.0	Pass / Fail (Sec. D.3.1.2.4)
8	VERIFY that the RESPONSE VALUE for globalTime.0 is roughly equal to Expected_Time.	Pass / Fail (Sec. D.3.1.2.4)
9	DELAY for 15 seconds.	
10	GET the following object(s): »globalTime.0 NOTE—NTCIP 1203 uses hyphenated names when referring to some NTCIP 1201 objects; these test procedures have been updated to use the actual object names without hyphens.	Pass / Fail (Sec. D.3.1.2.4)
11	VERIFY that the RESPONSE VALUE for globalTime.0 is roughly equal to Expected_Time plus 15 seconds.	Pass / Fail (Sec. D.3.1.2.4)
12	Calculate the time to set in the agent to restore the original value. RECORD this information as: »Restore_UTC_Time	
13	SET the following object(s) to the value(s) shown: »globalTime.0 = Restore_UTC_Time	Pass / Fail (Sec. D.3.1.2.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.7 Monitor External Port Information

Test Case: 10.7	***Title:***	*Monitor External Port Information*	
	Description:	*This test case verifies that the device allows the user to retrieve information about the device's external port(s).*	
	Variables:	*Required_Ports*	*F.2.3.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of external analog ports required by the specification (F.2.3.2). RECORD this information as: »Required_Ports	
2	GET the following object(s): »auxIOTableNumAnalogPorts.0 VERIFY that there is no response.	Pass / Fail (Sec. F.2.2.1.6)
3	Determine the RESPONSE VALUE for auxIOTableNumAnalogPorts.0. RECORD this information as: »Num_Ports	
4	FOR EACH value, N, from 1 to Num_Ports, perform Steps 4.1 through 4.17.	
4.1	GET the following object(s): »auxIOPortType.2.N »auxIOPortNumber.2.N »auxIOPortDescription.2.N »auxIOPortResolution.2.N »auxIOPortDirection.2.N	Pass / Fail (Sec. F.2.2.1.6)
4.2	VERIFY that the RESPONSE VALUE for auxIOPortType.2.N is equal to 2.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.1)
4.3	VERIFY that the RESPONSE VALUE for auxIOPortNumber.2.N is equal to N.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.2)
4.4	VERIFY that the RESPONSE VALUE for auxIOPortDescription.2.N is no more than 255 characters and contains only DisplayString characters.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.3)
4.5	VERIFY that the RESPONSE VALUE for auxIOPortDescription.2.N is APPROPRIATE.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.3)
4.6	VERIFY that the RESPONSE VALUE for auxIOPortResolution.2.N is greater than or equal to 1.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.4)
4.7	VERIFY that the RESPONSE VALUE for auxIOPortResolution.2.N is less than or equal to 32.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.4)
4.8	VERIFY that the RESPONSE VALUE for auxIOPortResolution.2.N is APPROPRIATE.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.4)

4.9	Determine the RESPONSE VALUE for auxIOPortResolution.2.N. RECORD this information as: »Resolution	
4.10	VERIFY that the RESPONSE VALUE for auxIOPortDirection.2.N is greater than or equal to 1.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.6)
4.11	VERIFY that the RESPONSE VALUE for auxIOPortDirection.2.N is less than or equal to 3.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.6)
4.12	VERIFY that the RESPONSE VALUE for auxIOPortDirection.2.N is APPROPRIATE.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.6)
4.13	GET the following object(s): »auxIOPortValue.2.N »auxIOPortLastCommandedState.2.N	Pass / Fail (Sec. F.2.2.2.1)
4.14	VERIFY that the RESPONSE VALUE for auxIOPortValue.2.N is greater than 0 and less than the port's maximum value as defined by auxIOPortResolution.2.N (i.e., (2^Resolution) - 1).	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.5)
4.15	VERIFY that the RESPONSE VALUE for auxIOPortValue.2.N is APPROPRIATE.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.5)
4.16	VERIFY that the RESPONSE VALUE for auxIOPortLastCommandedState.2.N is greater than 0 and less than the port's maximum value as defined by auxIOPortResolution.2.N (i.e., (2^Resolution) - 1).	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.7)
4.17	VERIFY that the RESPONSE VALUE for auxIOPortLastCommandedState.2.N is APPROPRIATE.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.7)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.10.8 Configure External Port

Test Case: 10.8	***Title:***	*Configure External Port*	
	Description:	*This test case verifies that the device allows the user to configure the external port.*	
	Variables:	*Required_Ports*	*F.2.3.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of external analog ports required by the specification (F.2.3.2). RECORD this information as: »Required_Ports	
2	Determine a random port from 1 to Required_Ports. RECORD this information as: »Subject_Port	
3	GET the following object(s): »auxIOPortDescription.2.Subject_Port »auxIOPortResolution.2.Subject_Port	Pass / Fail (RFC 1157)
4	Determine the RESPONSE VALUE for auxIOPortDescription.2.Subject_Port. RECORD this information as: »Description	
5	Determine the RESPONSE VALUE for auxIOPortResolution.2.Subject_Port. RECORD this information as: »Resolution	
6	Calculate a random value between 0 and the maximum value supported by the device. RECORD this information as: »Value	
7	SET the following object(s) to the value(s) shown: »auxIOPortDescription.2.Subject_Port = 'Test'	Pass / Fail (Sec. F.2.2.1.7)
8	GET the following object(s): »auxIOPortDescription.2.Subject_Port	Pass / Fail (RFC 1157)
9	VERIFY that the RESPONSE VALUE for auxIOPortDescription.2.Subject_Port is equal to "Test".	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.3)
10	SET the following object(s) to the value(s) shown: »auxIOPortDescription.2.Subject_Port = Description	Pass / Fail (Sec. F.2.2.1.7)
11	SET the following object(s) to the value(s) shown: »auxIOPortValue.2.Subject_Port = Value	Pass / Fail (Sec. F.2.2.4.1)
12	GET the following object(s): »auxIOPortValue.2.Subject_Port »auxIOPortLastCommandedState.2.Subject_Port	Pass / Fail (Sec. F.2.2.2.1)
13	VERIFY that the RESPONSE VALUE for auxIOPortLastCommandedState.2.Subject_Port is equal to Value.	Pass / Fail (NTCIP 1201 v02 Sec. 2.8.3.7)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11 Backward Compatibility Tests

C.2.3.11.1 Version 1 Wind Sensor Meta Data

Test Case: 11.1	***Title:***	*Version 1 Wind Sensor Meta Data*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 wind sensor meta data.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essWindSensorHeight.0	Pass / Fail (Sec. 3.5.4.1)
2	VERIFY that the RESPONSE VALUE for essWindSensorHeight.0 is greater than or equal to -1000.	Pass / Fail (Sec. 5.5.3)
3	VERIFY that the RESPONSE VALUE for essWindSensorHeight.0 is less than or equal to 1001.	Pass / Fail (Sec. 5.5.3)
4	VERIFY that the RESPONSE VALUE for essWindSensorHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.3)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.2 Version 1 Average Wind Sensor Data

Test Case: 11.2	***Title:***	*Version 1 Average Wind Sensor Data*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 average wind data.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essAvgWindDirection.0 »essAvgWindSpeed.0	Pass / Fail (Sec. 3.5.4.2)
2	VERIFY that the RESPONSE VALUE for essAvgWindDirection.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.6.1)
3	VERIFY that the RESPONSE VALUE for essAvgWindDirection.0 is less than or equal to 361.	Pass / Fail (Sec. 5.6.1)
4	VERIFY that the RESPONSE VALUE for essAvgWindDirection.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.1)
5	VERIFY that the RESPONSE VALUE for essAvgWindSpeed.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.6.2)
6	VERIFY that the RESPONSE VALUE for essAvgWindSpeed.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.6.2)
7	VERIFY that the RESPONSE VALUE for essAvgWindSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.3 Version 1 Spot Wind Sensor Data

Test Case: 11.3	***Title:***	*Version 1 Spot Wind Sensor Data*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 spot wind data.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essSpotWindDirection.0 »essSpotWindSpeed.0	Pass / Fail (Sec. 3.5.4.3)
2	VERIFY that the RESPONSE VALUE for essSpotWindDirection.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.6.3)
3	VERIFY that the RESPONSE VALUE for essSpotWindDirection.0 is less than or equal to 361.	Pass / Fail (Sec. 5.6.3)
4	VERIFY that the RESPONSE VALUE for essSpotWindDirection.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.3)
5	VERIFY that the RESPONSE VALUE for essSpotWindSpeed.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.6.4)
6	VERIFY that the RESPONSE VALUE for essSpotWindSpeed.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.6.4)
7	VERIFY that the RESPONSE VALUE for essSpotWindSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.4 Version 1 Wind Gust Data

Test Case: 11.4	***Title:***	*Version 1 Wind Gust Data*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 wind gust data.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essMaxWindGustSpeed.0 »essMaxWindGustDir.0	Pass / Fail (Sec. 3.5.4.4)
2	VERIFY that the RESPONSE VALUE for essMaxWindGustSpeed.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.6.5)
3	VERIFY that the RESPONSE VALUE for essMaxWindGustSpeed.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.6.6)
4	VERIFY that the RESPONSE VALUE for essMaxWindGustSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.6)
5	VERIFY that the RESPONSE VALUE for essMaxWindGustDir.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.6.7)
6	VERIFY that the RESPONSE VALUE for essMaxWindGustDir.0 is less than or equal to 361.	Pass / Fail (Sec. 5.6.7)
7	VERIFY that the RESPONSE VALUE for essMaxWindGustDir.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.7)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.5 Version 1 Wind Situation

Test Case: 11.5	***Title:***	*Version 1 Wind Situation*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 wind situation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essWindSituation.0	Pass / Fail (Sec. 3.5.4.5)
2	VERIFY that the RESPONSE VALUE for essWindSituation.0 is greater than or equal to 1.	Pass / Fail (Sec. 5.6.5)
3	VERIFY that the RESPONSE VALUE for essWindSituation.0 is less than or equal to 12.	Pass / Fail (Sec. 5.6.5)
4	VERIFY that the RESPONSE VALUE for essWindSituation.0 is APPROPRIATE. NOTE—Valid enumerated values are defined in NTCIP 1204 v03 Sec. 5.6.5	Pass / Fail (Sec. 5.6.5)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.6 Version 1 Water Depth

Test Case: 11.6	***Title:***	*Version 1 Water Depth*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 water depth.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essWaterDepth.0	Pass / Fail (Sec. 3.5.4.6)
2	VERIFY that the RESPONSE VALUE for essWaterDepth.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.8.2)
3	VERIFY that the RESPONSE VALUE for essWaterDepth.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.8.2)
4	VERIFY that the RESPONSE VALUE for essWaterDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.7 Version 1 Solar Radiation

Test Case: 11.7	***Title:***	*Version 1 Solar Radiation*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 solar radiation.*
	Variables:	
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	GET the following object(s): »essSolarRadiation.0	Pass / Fail (Sec. 3.5.4.7)
2	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is greater than or equal to 0.	Pass / Fail (Sec. 5.9.1)
3	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is less than or equal to 65535.	Pass / Fail (Sec. 5.9.1)
4	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.1)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.8 Version 1 Surface Water Depth

Test Case: 11.8	***Title:***	*Version 1 Surface Water Depth*
	Description:	*This test case verifies that the device allows the user to retreive the version 1 surface water depth.*
	Variables:	*Required_Pavement_Sensors* — *PRL 3.6.8*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	FOR EACH value, N, from 1 to Required_Pavement_Sensors, perform Steps 2.1 through 2.4.	
2.1	GET the following object(s): »essSurfaceWaterDepth.N	Pass / Fail (Sec. 3.5.4.8)
2.2	VERIFY that the RESPONSE VALUE for essSurfaceWaterDepth.N is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.10)
2.3	VERIFY that the RESPONSE VALUE for essSurfaceWaterDepth.N is less than or equal to 255.	Pass / Fail (Sec. 5.11.3.10)
2.4	VERIFY that the RESPONSE VALUE for essSurfaceWaterDepth.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.10)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.9 Version 1 Surface Conductivity

Test Case: 11.9	***Title:***	*Version 1 Surface Conductivity*	
	Description:	*This test case verifies that the device allows the user to retreive the version 1 surface conductivity.*	
	Variables:	*Required_Pavement_Sensors*	*PRL 3.6.8*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	FOR EACH value, N, from 1 to Required_Pavement_Sensors, perform Steps 2.1 through 2.4.	
2.1	GET the following object(s): »essSurfaceConductivity.N	Pass / Fail (Sec. 3.5.4.9)
2.2	VERIFY that the RESPONSE VALUE for essSurfaceConductivity.N is greater than or equal to 0.	Pass / Fail (Sec. 5.11.3.12)
2.3	VERIFY that the RESPONSE VALUE for essSurfaceConductivity.N is less than or equal to 65535.	Pass / Fail (Sec. 5.11.3.12)
2.4	VERIFY that the RESPONSE VALUE for essSurfaceConductivity.N is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.12)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.10 Version 2 Station Meta Data Block

Test Case: 11.10	***Title:***	*Version 2 Station Meta Data Block*	
	Description:	*This test case verifies that the device allows the user to retreive the version 2 Station Meta Data Block.*	
	Variables:	*Pressure_Supported*	*PRL 2.5.2.1.1*
		Wind_Supported	*PRL 2.5.2.1.2*
		Required_Wind_Sensors	*PRL 3.6.2*
		Temperature_Supported	*PRL 2.5.2.1.3*
		Required_Temp_Sensors	*PRL 3.6.3*
		Pavement_Supported	*PRL 2.5.2.2*
		Required_Pavement_Sensors	*PRL 3.6.8*
		Subsurface_Supported	*PRL 2.5.2.3*
		Required_Subsurface_Sensors	*PRL 3.6.11*
		Pavement_Treatment_Supported	*PRL 2.5.3*
		Required_Pavement_Treatment_Products	*PRL 3.6.12*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the ESS is required to monitor atmospheric pressure (PRL 2.5.2.1.1). RECORD this information as: »Pressure_Supported	
2	CONFIGURE: Determine . RECORD this information as: »Wind_Supported (whether the ESS is required to monitor winds—PRL 2.5.2.1.2) »Required_Wind_Sensors (the number of temperature sensors that the ESS is required to support - PRL 3.6.2)	
3	CONFIGURE: Determine whether the ESS is required to support temperature sensors and, if so, how many sensors are required. RECORD this information as: »Temperature_Supported (whether the ESS is required to monitor temperature - PRL 2.5.2.1.3 - PRL 2.5.2.1.3) »Required_Temp_Sensors (the number of temperature sensors that the ESS is required to support - PRL 3.6.3 - PRL 3.6.3)	
4	CONFIGURE: Determine whether the ESS is required to support pavement sensors and, if so, how many sensors are required. RECORD this information as: »Pavement_Supported (whether the ESS is required to monitor the pavement - PRL 2.5.2.2—PRL 2.5.2.2) »Required_Pavement_Sensors (the number of pavement sensors that the ESS is required to support—PRL 3.6.8 through PRL 3.6.8)	
5	CONFIGURE: Determine whether the ESS is required to support subsurface sensors and, if so, how many sensors are required. RECORD this information as: »Subsurface_Supported (whether the ESS is required to monitor the subsurface—PRL 2.5.2.3 through PRL 2.5.2.3) »Required_Subsurface_Sensors (the number of subsurface sensors that the ESS is required to support—PRL 3.6.11 through PRL 3.6.11)	
6	CONFIGURE: Determine whether the ESS is required to provide pavement treatment capabilities and, if so, how many treatment products the device is required to support. RECORD this information as: »Pavement_Treatment_Supported (whether pavement treatment capabilities are required—PRL 2.5.3 through PRL 2.5.3) »Required_Pavement_Treatment_Products (number of pavement treatment products	

	required--PRL 3.6.12 through PRL 3.6.12)	
7	GET the following object(s): »essStationMetaDataBlock.0	Pass / Fail (Sec. 3.5.4.10)
8	Decode the essStationMetaDataBlock.0 structure.	
9	VERIFY that the essStationMetaDataBlock.0 structure was decoded without error.	Pass / Fail (Sec. 5.3.5)
10	VERIFY that the RESPONSE VALUE for the essNtcipCategory.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
11	VERIFY that the RESPONSE VALUE for the essNtcipCategory.0 field of the essStationMetaDataBlock.0 object is APPROPRIATE. NOTE—See NTCIP 1204 v03 Sec. 5.2.1 for valid enumerated values.	Pass / Fail (Sec. 5.2.13.5)
12	VERIFY that the RESPONSE VALUE for the essTypeOfStation.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
13	VERIFY that the RESPONSE VALUE for the essTypeOfStation.0 field of the essStationMetaDataBlock.0 object is APPROPRIATE. NOTE—See NTCIP 1204 v03 Sec. 5.3.1 for the definition of valid values.	Pass / Fail (Sec. 5.3.15)
14	VERIFY that the RESPONSE VALUE for the essLatitude.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
15	VERIFY that the RESPONSE VALUE for the essLatitude.0 field of the essStationMetaDataBlock.0 object is APPROPRIATE.	Pass / Fail (Sec. 5.4.13.5)
16	VERIFY that the RESPONSE VALUE for the essLongitude.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
17	VERIFY that the RESPONSE VALUE for the essLongitude.0 field of the essStationMetaDataBlock.0 object is APPROPRIATE.	Pass / Fail (Sec. 5.4.23.5)
18	VERIFY that the RESPONSE VALUE for the essReferenceHeight.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
19	VERIFY that the RESPONSE VALUE for the essReferenceHeight.0 field of the essStationMetaDataBlock.0 object is APPROPRIATE.	Pass / Fail (Sec. 5. 3.55.1)
20	IF Pressure_Supported is equal to true, then proceed to Step 20.1; otherwise, proceed to Step 21.	
20.1	VERIFY that the RESPONSE VALUE for the essPressureHeight.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
20.2	VERIFY that the RESPONSE VALUE for essPressureHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.23.5)
21	IF Wind_Supported is equal to true, then proceed to Step 21.1; otherwise, proceed to Step 22.	
21.1	VERIFY that the RESPONSE VALUE for the essWindSensorHeight.0 field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
21.2	VERIFY that the RESPONSE VALUE for essWindSensorHeight.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.33.5)

22	IF Temperature_Supported is equal to true, then proceed to Step 22.1; otherwise, proceed to Step 23.	
22.1	VERIFY that the RESPONSE VALUE for the temperatureMetaData field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
22.2	VERIFY that the temperatureMetaData field contains at least Required_Temp_Sensors entries.	Pass / Fail (Sec. 5.3.5)
22.3	FOR EACH value, N, from 1 to Required_Temp_Sensors, perform Steps 22.3.1 through 22.3.4.	
22.3.1	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.0 field is present in the Nth TemperatureMetaData structure.	Pass / Fail (Sec. 5.3.5)
22.3.2	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.0 field in the Nth TemperatureMetaData structure is equal to N.	Pass / Fail (Sec. 5.7.3.13.5)
22.3.3	VERIFY that the RESPONSE VALUE for the essTemperatureSensorHeight.0 field is present in the Nth TemperatureMetaData structure.	Pass / Fail (Sec. 5.3.5)
22.3.4	VERIFY that the RESPONSE VALUE for essTemperatureSensorHeight.0 field in the Nth TemperatureMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.23.5)
23	IF Pavement_Supported is equal to true, then proceed to Step 23.1; otherwise, proceed to Step 24.	
23.1	VERIFY that the RESPONSE VALUE for the pavementMetaData field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
23.2	VERIFY that the pavementMetaData field contains at least Required_Pavement_Sensors entries.	Pass / Fail (Sec. 5.3.5)
23.3	FOR EACH value, N, from 1 to Required_Pavement_Sensors, perform Steps 23.3.1 through 23.3.10.	
23.3.1	VERIFY that the RESPONSE VALUE for the essPavementSensorIndex.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 5.3.5)
23.3.2	VERIFY that the RESPONSE VALUE for the essPavementSensorIndex.0 field in the Nth PavementMetaData structure is equal to N.	Pass / Fail (Sec. 5.11.3.13.5)
23.3.3	VERIFY that the RESPONSE VALUE for the essPavementType.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 5.3.5)
23.3.4	VERIFY that the RESPONSE VALUE for the essPavementType.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5. 11.3.33.5)
23.3.5	VERIFY that the RESPONSE VALUE for the essPavementElevation.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 5.3.5)
23.3.6	VERIFY that the RESPONSE VALUE for the essPavementElevation.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.43.5)
23.3.7	VERIFY that the RESPONSE VALUE for the essPavementExposure.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 5.3.5)
23.3.8	VERIFY that the RESPONSE VALUE for the essPavementExposure.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.53.5)

23.3.9	VERIFY that the RESPONSE VALUE for the essPavementSensorType.0 field is present in the Nth PavementMetaData structure.	Pass / Fail (Sec. 5.3.5)
23.3.10	VERIFY that the RESPONSE VALUE for the essPavementSensorType.0 field in the Nth PavementMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.63.5)
24	IF Subsurface_Supported is equal to true, then proceed to Step 24.1; otherwise, proceed to Step 25.	
24.1	VERIFY that the RESPONSE VALUE for the SubSurfaceMetaData field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
24.2	VERIFY that the SubSurfaceMetaData field contains at least Required_Subsurface_Sensors entries.	Pass / Fail (Sec. 5.3.5)
24.3	FOR EACH value, N, from 1 to Required_Subsurface_Sensors, perform Steps 24.3.1 through 24.3.6.	
24.3.1	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorIndex.0 field is present in the Nth SubSurfaceMetaData structure.	Pass / Fail (Sec. 5.3.5)
24.3.2	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorIndex.0 field in the Nth SubSurfaceMetaData structure is equal to N.	Pass / Fail (Sec. 5.11.6.13.5)
24.3.3	VERIFY that the RESPONSE VALUE for the essSubSurfaceType.0 field is present in the Nth SubSurfaceMetaData structure.	Pass / Fail (Sec. 5.3.5)
24.3.4	VERIFY that the RESPONSE VALUE for the essSubSurfaceType.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.33.5)
24.3.5	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorDepth.0 field is present in the Nth SubSurfaceMetaData structure.	Pass / Fail (Sec. 5.3.5)
24.3.6	VERIFY that the RESPONSE VALUE for the essSubSurfaceSensorDepth.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.11.6.43.5)
25	IF Pavement_Treatment_Supported is equal to true, then proceed to Step 25.1; otherwise, proceed to EXIT.	
25.1	VERIFY that the RESPONSE VALUE for the treatmentMetaData field of the essStationMetaDataBlock.0 object is present.	Pass / Fail (Sec. 5.3.5)
25.2	VERIFY that the treatmentMetaData field contains at least Required_Pavement_Treatment_Products entries.	Pass / Fail (Sec. 5.3.5)
25.3	FOR EACH value, N, from 1 to Required_Pavement_Treatment_Products, perform Steps 25.3.1 through 25.3.11.	
25.3.1	VERIFY that the RESPONSE VALUE for the essPavementTreatmentIndex.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 5.3.5)
25.3.2	VERIFY that the RESPONSE VALUE for the essPavementTreatmentIndex.0 field in the Nth TreatmentMetaData structure is equal to N.	Pass / Fail (Sec. 5.13.3.13.5)
25.3.3	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 5.3.5)
25.3.4	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.0 field in the Nth TreatmentMetaData structure is between 1 and 14, inclusive.	Pass / Fail (Sec. 5.13.3.23.5)

25.3.5	VERIFY that the RESPONSE VALUE for the essPaveTreatProductType.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.2)
25.3.6	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 5.3.5)
25.3.7	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.0 field in the Nth TreatmentMetaData structure is between 1 and 4, inclusive.	Pass / Fail (Sec. 5.13.3.33.5)
25.3.8	VERIFY that the RESPONSE VALUE for the essPaveTreatProductForm.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.3)
25.3.9	VERIFY that the RESPONSE VALUE for the essPercentProductMix.0 field is present in the Nth TreatmentMetaData structure.	Pass / Fail (Sec. 5.3.5)
25.3.10	VERIFY that the RESPONSE VALUE for the essPercentProductMix.0 field in the Nth TreatmentMetaData structure is between 0 and 100, inclusive.	Pass / Fail (Sec. 5.13.3.43.5)
25.3.11	VERIFY that the RESPONSE VALUE for the essPercentProductMix.0 field in the Nth SubSurfaceMetaData structure is APPROPRIATE.	Pass / Fail (Sec. 5.13.3.4)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.11 Version 2 Weather Block

Test Case: 11.11	***Title:***	*Version 2 Weather Block*	
	Description:	*This test case verifies that the device allows the user to retreive the version 2 Weather Data Block Object.*	
	Variables:	*Pressure_Supported*	*PRL 2.5.2.1.1*
		Wind_Supported	*PRL 2.5.2.1.2*
		Required_Wind_Sensors	*PRL 3.6.2*
		Temperature_Supported	*PRL 2.5.2.1.3*
		Required_Temp_Sensors	*PRL 3.6.3*
		Precip_Supported	*PRL 2.5.2.1.5*
		Required_Water_Level_Sensors	*PRL 3.6.22*
		Situation_Supported	*PRL 2.5.2.4*
		Visibility_Supported	*PRL 2.5.2.1.7*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine whether the ESS is required to monitor atmospheric pressure (PRL 2.5.2.1.1). RECORD this information as: »Pressure_Supported	
2	CONFIGURE: Determine . RECORD this information as: »Wind_Supported (whether the ESS is required to monitor winds—PRL 2.5.2.1.2) »Required_Wind_Sensors (the number of wind sensors that the ESS is required to suppor—PRL 3.6.2)	
3	CONFIGURE: Determine . RECORD this information as: »Temperature_Supported (whether the ESS is required to monitor temperature - PRL 2.5.2.1.3) »Required_Temp_Sensors (the number of temperature sensors that the ESS is required to support—PRL 3.6.3)	
4	CONFIGURE: Determine . RECORD this information as: »Precip_Supported (whether the ESS is required to monitor precipitation—PRL 2.5.2.1.5) »Required_Water_Level_Sensors (the number of water level sensors that the ESS is required to support—PRL 3.6.22)	
5	CONFIGURE: Determine whether the ESS is required to report situation data (PRL 2.5.2.4). RECORD this information as: »Situation_Supported	
6	CONFIGURE: Determine whether the ESS is required to provide visibility capabilities (PRL 2.5.2.1.7). RECORD this information as: »Visibility_Supported	
7	GET the following object(s): »essWeatherBlock.0	Pass / Fail (Sec. 3.5.4.11)
8	Decode the essWeatherBlock.0 structure.	
9	VERIFY that the essWeatherBlock.0 structure was decoded without error.	Pass / Fail (Sec. 5.3.6)

10	IF Pressure_Supported is equal to true, then proceed to Step 10.1; otherwise, proceed to Step 11.	
10.1	VERIFY that the RESPONSE VALUE for the essAtmosphericPressure.0 field of the essWeatherBlock.0 object is present.	Pass / Fail (Sec. 5.3.6)
10.2	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.5.43.6)
10.3	VERIFY that the RESPONSE VALUE for essAtmosphericPressure.0 is APPROPRIATE.	Pass / Fail (Sec. 5.5.4)
11	IF Wind_Supported is equal to true, then proceed to Step 11.1; otherwise, proceed to Step 12.	
11.1	VERIFY that the RESPONSE VALUE for the essWindData field of the essWeatherBlock.0 object is present.	Pass / Fail (Sec. 5.3.6)
11.2	VERIFY that the essAvgWindDirection.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.3	VERIFY that the RESPONSE VALUE for essAvgWindDirection.0 is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.6.13.6)
11.4	VERIFY that the RESPONSE VALUE for essAvgWindDirection.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.1)
11.5	VERIFY that the essAvgWindSpeed.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.6	VERIFY that the RESPONSE VALUE for essAvgWindSpeed.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.23.6)
11.7	VERIFY that the RESPONSE VALUE for essAvgWindSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.2)
11.8	VERIFY that the essMaxWindGustSpeed.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.9	VERIFY that the RESPONSE VALUE for essMaxWindGustSpeed.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.63.6)
11.10	VERIFY that the RESPONSE VALUE for essMaxWindGustSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.6)
11.11	VERIFY that the essMaxWindGustDir.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.12	VERIFY that the RESPONSE VALUE for essMaxWindGustDir.0 is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.3.66.7)
11.13	VERIFY that the RESPONSE VALUE for essMaxWindGustDir.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.7)
11.14	VERIFY that the essSpotWindDirection.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.15	VERIFY that the RESPONSE VALUE for essSpotWindDirection.0 is between 0 and 361, inclusive.	Pass / Fail (Sec. 5.3.66.3)

11.16	VERIFY that the RESPONSE VALUE for essSpotWindDirection.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.3)
11.17	VERIFY that the essSpotWindSpeed.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.18	VERIFY that the RESPONSE VALUE for essSpotWindSpeed.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.6.43.6)
11.19	VERIFY that the RESPONSE VALUE for essSpotWindSpeed.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.4)
11.20	IF Situation_Supported is equal to true, then proceed to Step 11.20.1; otherwise, proceed to Step 12.	
11.20.1	VERIFY that the essWindSituation.0 field is present in the essWindData structure.	Pass / Fail (Sec. 5.3.6)
11.20.2	VERIFY that the RESPONSE VALUE for essWindSituation.0 is between 1 and 12, inclusive.	Pass / Fail (Sec. 5.6.53.6)
11.20.3	VERIFY that the RESPONSE VALUE for essWindSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.6.5)
12	IF Temperature_Supported is equal to true, then proceed to Step 12.1; otherwise, proceed to Step 13.	
12.1	VERIFY that the essTemperatureData field of the essWeatherBlock object is present.	Pass / Fail (Sec. 5.3.6)
12.2	VERIFY that the RESPONSE VALUE for the essWetBulbTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 5.3.6)
12.3	VERIFY that the RESPONSE VALUE for essWetBulbTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.43.6)
12.4	VERIFY that the RESPONSE VALUE for essWetBulbTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.4)
12.5	VERIFY that the RESPONSE VALUE for the essDewpointTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 5.3.6)
12.6	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.53.6)
12.7	VERIFY that the RESPONSE VALUE for essDewpointTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.5)
12.8	VERIFY that the RESPONSE VALUE for the essMaxTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 5.3.6)
12.9	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.73.6)
12.10	VERIFY that the RESPONSE VALUE for essMaxTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.6)
12.11	Determine the maximum temperature reading reported by the device. RECORD this information as: »Max_Temp	

12.12	VERIFY that the RESPONSE VALUE for the essMinTemp.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 5.3.6)
12.13	VERIFY that the RESPONSE VALUE for essMinTemp.0 is between -1000 and Max_Temp, inclusive.	Pass / Fail (Sec. 5.7.73.6)
12.14	VERIFY that the RESPONSE VALUE for essMinTemp.0 is APPROPRIATE.	Pass / Fail (Sec. 5.7.7)
12.15	VERIFY that the RESPONSE VALUE for the essRelativeHumidity.0 field of the essTemperatureData field is present.	Pass / Fail (Sec. 5.3.6)
12.16	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is between 0 and 101, inclusive.	Pass / Fail (Sec. 5.8.13.6)
12.17	VERIFY that the RESPONSE VALUE for essRelativeHumidity.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.1)
12.18	VERIFY that the temperatureTable field contains at least Required_Temp_Sensors entries.	Pass / Fail (Sec. 5.3.6)
12.19	FOR EACH value, N, from 1 to Required_Temp_Sensors, perform Steps 12.19.1 through 12.19.5.	
12.19.1	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.x field is present in the Nth Temperature structure.	Pass / Fail (Sec. 5.3.6)
12.19.2	VERIFY that the RESPONSE VALUE for the essTemperatureSensorIndex.x field in the Nth Temperature structure is equal to N.	Pass / Fail (Sec. 5.7.3.13.6)
12.19.3	VERIFY that the RESPONSE VALUE for the essAirTemperature.x field is present in the Nth Temperature structure.	Pass / Fail (Sec. 5.3.6)
12.19.4	VERIFY that the RESPONSE VALUE for essAirTemperature.x field in the Nth Temperature structure is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.7.3.33.6)
12.19.5	VERIFY that the RESPONSE VALUE for essAirTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.7.3.3)
13	IF Precip_Supported is equal to true, then proceed to Step 13.1; otherwise, proceed to Step 14.	
13.1	VERIFY that the RESPONSE VALUE for the essPrecipData field of the essWeatherBlock.0 object is present.	Pass / Fail (Sec. 5.3.6)
13.2	VERIFY that the RESPONSE VALUE for the essWaterDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.3	VERIFY that the RESPONSE VALUE for essWaterDepth.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.23.6)
13.4	VERIFY that the RESPONSE VALUE for essWaterDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.2)
13.5	VERIFY that the RESPONSE VALUE for the essAdjacentSnowDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.6	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.33.6)

13.7	VERIFY that the RESPONSE VALUE for essAdjacentSnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.3)
13.8	VERIFY that the RESPONSE VALUE for the essRoadwaySnowDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.9	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.43.6)
13.10	VERIFY that the RESPONSE VALUE for essRoadwaySnowDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.4)
13.11	VERIFY that the RESPONSE VALUE for the essRoadwaySnowPackDepth.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.12	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is between 0 and 3001, inclusive.	Pass / Fail (Sec. 5.8.53.6)
13.13	VERIFY that the RESPONSE VALUE for essRoadwaySnowPackDepth.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.5)
13.14	VERIFY that the RESPONSE VALUE for the essPrecipYesNo.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.15	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is between 1 and 3, inclusive.	Pass / Fail (Sec. 5.83.6)
13.16	VERIFY that the RESPONSE VALUE for essPrecipYesNo.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.6)
13.17	VERIFY that the RESPONSE VALUE for the essPrecipRate.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.18	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.73.6)
13.19	VERIFY that the RESPONSE VALUE for essPrecipRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.7)
13.20	VERIFY that the RESPONSE VALUE for the essSnowfallAccumRate.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.21	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.83.6)
13.22	VERIFY that the RESPONSE VALUE for essSnowfallAccumRate.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.8)
13.23	IF Situation_Supported is equal to true, then proceed to Step 13.23.1; otherwise, proceed to Step 13.24.	
13.23.1	VERIFY that the RESPONSE VALUE for the essPrecipSituation.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.23.2	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is between 1 and 15, inclusive.	Pass / Fail (Sec. 5.8.93.6)
13.23.3	VERIFY that the RESPONSE VALUE for essPrecipSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.9)

13.24	VERIFY that the RESPONSE VALUE for the essIceThickness.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.25	VERIFY that the RESPONSE VALUE for essIceThickness.0 is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.8.103.6)
13.26	VERIFY that the RESPONSE VALUE for essIceThickness.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.10)
13.27	VERIFY that the RESPONSE VALUE for the essPrecipitationStartTime.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.28	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.8.113.6)
13.29	VERIFY that the RESPONSE VALUE for essPrecipitationStartTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.11)
13.30	VERIFY that the RESPONSE VALUE for the essPrecipitationEndTime.0 field of the essPrecipData field is present.	Pass / Fail (Sec. 5.3.6)
13.31	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is between 0 and 4294967295, inclusive.	Pass / Fail (Sec. 5.8.123.6)
13.32	VERIFY that the RESPONSE VALUE for essPrecipitationEndTime.0 is APPROPRIATE.	Pass / Fail (Sec. 5.8.12)
14	IF Visibility_Supported is equal to true, then proceed to Step 14.1; otherwise, proceed to EXIT.	
14.1	VERIFY that the RESPONSE VALUE for the essVisibilityData field of the essWeatherBlock.0 object is present.	Pass / Fail (Sec. 5.3.6)
14.2	VERIFY that the RESPONSE VALUE for the essSolarRadiation.0 field of the essVisibilityData field is present.	Pass / Fail (Sec. 5.3.6)
14.3	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is between -2048 and 2049, inclusive.	Pass / Fail (Sec. 5.3.69.1)
14.4	VERIFY that the RESPONSE VALUE for essSolarRadiation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.1)
14.5	VERIFY that the RESPONSE VALUE for the essTotalSun.0 field of the essVisibilityData field is present.	Pass / Fail (Sec. 5.3.6)
14.6	VERIFY that the RESPONSE VALUE for essTotalSun.0 is between 0 and 1441, inclusive.	Pass / Fail (Sec. 5.3.69.2)
14.7	VERIFY that the RESPONSE VALUE for essTotalSun.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.2)
14.8	VERIFY that the RESPONSE VALUE for the essVisibility.0 field of the essVisibilityData field is present.	Pass / Fail (Sec. 5.3.6)
14.9	VERIFY that the RESPONSE VALUE for essVisibility.0 is between 0 and 1000001, inclusive.	Pass / Fail (Sec. 5.3.610.1)
14.10	VERIFY that the RESPONSE VALUE for essVisibility.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.1)

14.11	IF Situation_Supported is equal to true, then proceed to Step 14.11.1; otherwise, proceed to EXIT.	
14.11.1	VERIFY that the RESPONSE VALUE for the essCloudSituation.0 field of the essVisibilityData field is present.	Pass / Fail (Sec. 5.3.6)
14.11.2	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is between 1 and 5, inclusive.	Pass / Fail (Sec. 5.9.33.6)
14.11.3	VERIFY that the RESPONSE VALUE for essCloudSituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.9.3)
14.11.4	VERIFY that the RESPONSE VALUE for the essVisibilitySituation.0 field of the essVisibilityData field is present.	Pass / Fail (Sec. 5.3.6)
14.11.5	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is between 1 and 12, inclusive.	Pass / Fail (Sec. 5.10.23.6)
14.11.6	VERIFY that the RESPONSE VALUE for essVisibilitySituation.0 is APPROPRIATE.	Pass / Fail (Sec. 5.10.2)

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

C.2.3.11.12 Version 2 Pavement Block

Test Case: 11.12	***Title:***	*Version 2 Pavement Block*	
	Description:	*This test case verifies that the device allows the user to retreive the version 2 Pavement Block Object.*	
	Variables:	*Required_Pavement_Sensors*	*PRL 3.6.8*
		Support_Icing	*PRL 2.5.2.2.2*
	Pass/Fail Criteria:	*The device under test (DUT) shall pass every verification step included within the Test Case to pass the Test Case.*	

Step	***Test Procedure***	***Device***
1	CONFIGURE: Determine the number of pavement sensors required by the specification (PRL 3.6.8). RECORD this information as: »Required_Pavement_Sensors	
2	CONFIGURE: Determine whether the ESS is required to support icing detection per the specification (PRL 2.5.2.2.2). RECORD this information as: »Support_Icing	
3	GET the following object(s): »essPavementBlock.0	Pass / Fail (Sec. 3.5.4.12)
4	Decode the essPavementBlock.0 structure.	
5	VERIFY that the EssPavementData structure contains at least Required_Pavement_Sensors entries.	Pass / Fail (Sec. 5.11.7)
6	FOR EACH value, N, from 1 to Required_Pavement_Sensors, perform Steps 6.1 through 6.12.	
6.1	VERIFY that the RESPONSE VALUE for the essPavementSensorIndex.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.2	VERIFY that the RESPONSE VALUE for essPavementSensorIndex.x is equal to N.	Pass / Fail (Sec. 5.11.3.17)
6.3	VERIFY that the RESPONSE VALUE for the essSurfaceStatus.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.4	VERIFY that the RESPONSE VALUE for essSurfaceStatus.x is between 1 and 14, inclusive.	Pass / Fail (Sec. 5.11.3.7)
6.5	VERIFY that the RESPONSE VALUE for essSurfaceStatus.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.7)
6.6	VERIFY that the RESPONSE VALUE for the essSurfaceTemperature.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.7	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.73.8)
6.8	VERIFY that the RESPONSE VALUE for essSurfaceTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.8)
6.9	IF Support_Icing is equal to true, then proceed to Step 6.9.1; otherwise, proceed to Step	

6.9.1	VERIFY that the RESPONSE VALUE for the essPavementTemperature.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.9.2	VERIFY that the RESPONSE VALUE for essPavementTemperature.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.3.97)
6.9.3	VERIFY that the RESPONSE VALUE for essPavementTemperature.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.9)
6.9.4	VERIFY that the RESPONSE VALUE for the essSurfaceFreezePoint.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.9.5	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.x is between -1000 and 1001, inclusive.	Pass / Fail (Sec. 5.11.73.13)
6.9.6	VERIFY that the RESPONSE VALUE for essSurfaceFreezePoint.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.13)
6.9.7	VERIFY that the RESPONSE VALUE for the essSurfaceBlackIceSignal.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.9.8	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.x is between 1 and 4, inclusive.	Pass / Fail (Sec. 5.11.3.147)
6.9.9	VERIFY that the RESPONSE VALUE for essSurfaceBlackIceSignal.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.14)
6.9.10	VERIFY that the RESPONSE VALUE for the essSurfaceWaterDepth.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.9.11	VERIFY that the RESPONSE VALUE for essSurfaceWaterDepth.x is between 0 and 255, inclusive.	Pass / Fail (Sec. 5.11.3.167)
6.9.12	VERIFY that the RESPONSE VALUE for essSurfaceWaterDepth.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.16)
6.9.13	GET the following object(s): »essPavementSensorType.N	Pass / Fail (Sec. RFC 1157)
6.9.14	IF the RESPONSE VALUE for essPavementSensorType.Subject_Pavement_Sensor is equal to contactPassive, then proceed to Step 6.9.14.1; otherwise, proceed to Step 6.10.	
6.9.14.1	VERIFY that the RESPONSE VALUE for the essSurfaceSalinity.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.9.14.2	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.x is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.11.3.117)
6.9.14.3	VERIFY that the RESPONSE VALUE for essSurfaceSalinity.x is APPROPRIATE.	Pass / Fail (Sec. 5.11.3.11)
6.9.14.4	VERIFY that the RESPONSE VALUE for the essSurfaceConductivity.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.9.14.5	VERIFY that the RESPONSE VALUE for essSurfaceConductivity.x is between 0 and 65535, inclusive.	Pass / Fail (Sec. 5.11.3.127)
6.9.14.6	VERIFY that the RESPONSE VALUE for essSurfaceConductivity.x is APPROPRIATE.	Pass / Fail

6.10	VERIFY that the RESPONSE VALUE for the essPavementSensorError.x field of the essPavementBlock.0 object is present.	Pass / Fail (Sec. 5.11.7)
6.11	VERIFY that the RESPONSE VALUE for essPavementSensorError.x is between 1 and 6, inclusive.	Pass / Fail (Sec. 5.11.3.157)
6.12	VERIFY that the RESPONSE VALUE for essPavementSensorError.x is APPROPRIATE.	Pass / Fail

Test Case Results		
Tested By:	***Date Tested:***	***Pass / Fail***
Test Case Notes:		

Annex D
DOCUMENTATION OF REVISIONS
[INFORMATIVE]

Annex D identifies the changes that have been made to NTCIP 1204 v03. The NTCIP effort makes reasonable efforts to ensure that standards are as backward compatible as possible, but the primary purpose of NTCIP 1204 v03 is to provide interoperability by developing standards in a consensus environment. When changes are required to meet these objectives, the problematic objects are refined (if the issue is primarily editorial) or deprecated and, in most cases, replaced with new objects. Annex D identifies why each of these changes has been made. New implementations should support the new/replacement objects; they may also support deprecated objects.

D.1 NTCIP 1204 V01 TO NTCIP 1204 V02

General edits were made to the MIB header to reflect updates to other MIBs from which this MIB imports data.

All DESCRIPTION fields were updated to conform to NTCIP 8004 v02.

The STATUS of all objects was changed to "mandatory" to reflect the fact that conformance is now measured through the use of the PRL as contained in Section 3 and the RTM contained in Annex A.

References to Global Objects are now made through the RTM rather than through comments in the MIB.

Several objects were added to reflect new user needs.

The following identify additional edits that were made.

D.1.1 Station Category

The definition of essNtcipCategory was modified to reflect the fact that this parameter relates to "category" rather than "type".

D.1.2 Latitude

The definition of the essLatitude object was modified to reference the datum set to be used.

D.1.3 Wind Sensor Information

The various wind objects were deprecated and replaced with a table to allow the standard to support multiple wind sensors. The deprecated objects and their replacements are listed as follows:

a) essWindSensorHeight was replaced with windSensorHeight and windSensorLocation
b) essAvgWindDirection was replaced with windSensorAvgDirection
c) essAvgWindSpeed was replaced with windSensorAvgSpeed
d) essSpotWindDirection was replaced with windSensorSpotDirection
e) essSpotWindSpeed was replaced with windSensorSpotSpeed
f) essWindSituation was replaced with windSensorSituation
g) essMaxWindGustSpeed was replaced with windSensorGustSpeed
h) essMaxWindGustDir was replaced with windSensorGustDirection

In addition, the meaning of zero was clarified for the wind direction objects.

New implementations should support the replacement objects but may also support the original objects for backwards compatibility purposes. If the original objects are supported, they should report the values reported by the first wind sensor.

D.1.4 Water Depth

The essWaterDepth object was deprecated to reflect the fact that the ESS may support multiple Water Level Sensors. New implementations should support the replacement objects (i.e., those associated with the Water Level Sensor Table) but may also support essWaterDepth for backwards compatibility purposes.

D.1.5 Solar Radiation

The essSolarRadiation object was deprecated to reflect the fact that this value should be an instantaneous value rather than a value integrated over 24 hours to provide maximum compatibility with BUFR. New implementations should support the replacement object (essInstantaneousSolarRadiation) but may also support essSolarRadiation for backwards compatibility purposes.

D.1.6 Surface Water Depth

The essSurfaceWaterDepth object was deprecated to allow more precise measurements as required in practice. New implementations should support the replacement object (essSurfaceIceOrWaterDepth) but may also support essSurfaceWaterDepth for backwards compatibility purposes.

D.1.7 Surface Conductivity

The essSurfaceConductivity object was deprecated to correct the units of the object. New implementations should support the replacement object (essSurfaceConductivityV2) but may also support essSurfaceConductivity for backwards compatibility purposes.

D.1.8 NTCIP 1204 V02 Errata

Prior to publishing NTCIP 1204 v02, a few minor errors were identified in the document that led to the release of an Errata Sheet that corrected the issues as identified in Annex D.1.9 and D.1.10.

D.1.9 Correct OID Numbers

The Object Identifiers for essSubSurfaceMoisture and essSubSurfaceSensorError were inadvertantly changed in NTCIP 1204 v02. This change was corrected in the Errata by returning the assigned Object Identifiers to their original values (as defined in NTCIP 1204 v01).

D.1.10 Correct Snapshot Camera Details

Sections 4.2.1 (Capture Snapshot Image) and 5.16.3.4 (essSnapshotCameraCommand) contained slight inconsistencies within their definitions of how a snapshot camera would operate. The dialog in Section 4.2.1 was corrected to allow the operation suggested by the essSnapshotCameraCommand object, namely that the SNMP response to the snapshot command does not have to wait for action of capturing and storing the snapshot image. Likewise, the ACCESS of essSnapshotCameraCommand (Section 5.16.3.4) was corrected from read-only to read-write to allow for the operation discussed in both the object DESCRIPTION as well as Section 4.2.1 (Capture Snapshot Image).

D.2 NTCIP 1204 V02 TO NTCIP 1204 V03

The primary change from NTCIP 1204 v02 to NTCIP 1204 v03 was the addition of the test procedures in Annex C. However, adding these procedures also identified a few other anomalies, which are addressed.

D.2.1 NTCIP References

Annex D.2.1 identifies changes that have been made in relation to referenced standards.

References to the MIB Module name (especially in block objects) has been revised from “NTCIP1204-200x,” where the suffix was the year of Joint Approval, to “NTCIP1204-v03,” with the suffix as the Major Version number, as an editorial change.

D.2.1.1 NTCIP 1201 Reference

The reference for the definition of global objects was updated to NTCIP 1201:2005 with Amendment 2. The only changes that this created are as follows:

a) Event Logging Objects—These objects are unchanged, but have been moved from NTCIP 1201 to NTCIP 1103 v01.
b) Auxiliary I/O Objects—These objects were renamed to prevent any naming conflicts from objects previously defined in NTCIP 1203 that have now been deprecated. This change has no effect on deployments (i.e., the textual name is not used within a data exchange.).

D.2.1.2 NTCIP 1103 v01 Reference

Because Auxiliary objects have been moved to NTCIP 1103 v01, NTCIP 1103 v01 has been made a normative reference.

D.2.1.3 NTCIP 8004 v02 Reference

NTCIP 8004 v02 publication is anticipated shortly. NTCIP 8004 v02 does not require any changes to NTCIP 1204 v03.

D.2.1.4 NTCIP 8007 v01 Reference

Since NTCIP 1204 v03 includes test procedures, 1204 v03 now includes a reference to NTCIP 8007 v01.

D.2.2 Deletion of Requirement 3.5.1.3.2

A review of NTCIP 1204 v02 revealed that Requirement 3.5.1.3.2 (Retrieve Mobile Treatment Information) was redundant with Requirement 3.5.3.1.3 (Retrieve Mobile Pavement Treatment Configuration). After consideration, it was determined that the requirement was more appropriately defined in Section 3.5.3.1.3 (i.e., according to the terms used in this standard, the ESS Manager does not include pavement treatment information; that information would be stored in the PTS Manager); therefore Section 3.5.1.3.2 was deleted.

D.2.3 Clarify Snapshot Camera Configuration

Requirements 3.5.2.1.9 and 3.5.2.3.9 were edited to more accurately reflect the intent and design of NTCIP 1204 v03.

D.2.4 Remove Copy Snapshot Requirement

Requirement 3.5.2.4.3 and its corresponding dialog defined in Section 4.2.4 were deprecated because the defined FTP mechanism does not support the copy operation. After discussing this issue, it was determined that there was not really a need to support the copy operation if the snapshot filename can be defined to use a sequence field (see Requirement 3.6.24).

D.2.5 Correct Stationary Pavement Treatment System Configuration

Item b of Requirements 3.5.3.1.1.b, 3.5.3.1.2.b, and 3.5.3.4.2 were revised to reflect the design of the MIB objects. Originally, the requirement stated that the "spray duration" was a configurable parameter. This statement did not properly capture the intent; however, the intent was properly implemented in the design. The intent was to define how long the PTS is required to signal the external sprayer device to ensure that the PTS sprayer acts upon the signal. The duration of the spray is not a configurable parameter. This change did not require any change to the ptsSignalDuration object, but did require a change to the ptsCommandState object (Section 5.13.8). related object definition (Section 5.13.10). Further, the change to the ptsCommandState object required a change to the ptsOperationalMode object (Section 5.13.7), since it references ptsCommandState within its definition. Therefore, both of these objects were changed to obsolete status, since there were no known deployments at the time of the change, and new objects were added to accommodate the change in a backward compatible fashion.

D.2.6 Required Number of Water Level Sensors

Version 3 added Requirement 3.6.22 to allow a user of NTCIP 1204 v03 an easy way to specify the number of water level sensors to be included in a deployment when using the PRL.

D.2.7 Addition of Configurable Filenames for Snapshot Images

A number of requirements (Sections 3.6.23 through 3.6.27) were added to allow a management station to configure the name that the ESS should use when saving snapshot images.

D.2.8 Modify Set Constraints

A review of NTCIP 1204 v02 indicated that the following objects were defined as read-write, but had a set constraint that prevented it from being set:

a) essNtcipSiteDescription
b) essPaveTreatProductType

These typos were corrected.

D.2.9 Clarify essTypeofStation

The definition of the essTypeofStation object was clarified with an <Informative> statement to clarify that the value of 2 is reserved and that a hybrid station requires two instances of the MIB.

D.2.10 Correct Block Objects

A review of NTCIP 1204 v02 revealed that the following newly created block objects referenced deprecated objects within the block object structure:

a) Station Meta Data Block
b) Weather Block
c) Pavement Block

To rectify this problem, the existing block objects were deprecated and new replacement block objects were created.

D.2.11 Correct Access of essSnapshotCameraCommand

Shortly after NTCIP 1204 v02 was relaeased, implementers realized that the essSnapshotCameraCommand object has to be read-write for the snapshot feature to have any useful functionality. While this was a significant technical change on paper, the object has no use when implemented read-only, converting it to read-write should not create any backwards compatibility problems with existing deployments, many implementers have already implemented it as read-write, and it has been determined to be the approach that results in the smallest impact to existing deployments.

D.2.12 Added a Backward Compatibility User Need and Associated Requirements

Additional user needs and requirements were added to address backward compatibility concerns.

D.2.13 Added Dialogs to Separate NTCIP 1204 v01 and NTCIP 1204 v02 Objects

A few of the standardized dialogs in NTCIP 1204 v02 combined NTCIP 1204 v01 and NTCIP 1204 v02 objects into a single request. As a result, if the standardized dialog was used to interface with an NTCIP 1204 v01 device, the request would fail with a noSuchName error.

This version v03 has modified these dialogs to separate the NTCIP 1204 v01 objects from the NTCIP 1204 v02 objects. This separation has affected the standardized dialogs for the following requirements:

a) 3.5.2.3.2.6.1 Retrieve Precipitation Presence
b) 3.5.2.3.2.7 Retrieve Solar Radiation
c) 3.5.2.3.3.1 Retrieve Pavement Surface Condition
d) 3.5.2.3.3.2 Retrieve Icing Conditions – Active
e) 3.5.2.3.3.3 Retrieve Icing Conditions – Passive

Table 31 summarizes the impact that this change has on backward and forward compatibility.

Table 31 Backward and Forward Compatiblity Impacts

Central	Device	Interoperable	Notes
v01	v01	Maybe [1]	1204 version v01 did not define any standardized dialogs; however, interoperability is provided through the rules of Section 4 (for v02 and v03) and RFC 1157. Interoperability will be obtained as long as the v02/v03 device supports the deprecated objects used by the central, if any.
v01	v02	Maybe	
v01	v03	Maybe	
v02	v01	Maybe—With additional logic in central	A few of the standardized dialogs grouped v01 and v02 data into a single request; using these dialogs would result in a noSuchName error. This is a forward-compatible response since the v02 central could then implement logic that would only request v01 data; however, such logic would be outside the scope of the standardized dialog.
v02	v02	Yes	
v02	v03	Yes	While v03 defines a slightly different standardized dialog, the rules of Section 4 and RFC 1157 require the device to support requests that combine data in any combination that do not break the rules defined in Section 4.3. Thus, interoperability is maintained.
v03	v01	Maybe	Data defined in different versions are separated into different requests with logic used to determine if and when deprecated objects or v02 objects should be requested. Thus, interoperability is maintained and automatically provided.
v03	v02	Yes	While v03 defines a slightly different standardized dialog, the rules of Section 4 and RFC 1157 require the device to support requests that combine data in any combination that do not break the rules defined in Section 4.3. Thus, interoperability is maintained.
v03	v03	Yes	
[Note 1] If the v01 to v02 change was because of an ambiguity, the original interoperability problems remain with any implementation of v01 objects.			

D.2.14 Clarified Handling of globalSetIDParameter

Annex H was added to clarify exactly what parameters would affect the value of the globalSetIDParameter object. This was deemed to be a simple clarification since the definition of these objects did not change the written intent of the object definition.

D.2.15 Clarify Meaning of Static Tables

Objects that define how many rows exist in static tables were clarified with an informative statement to explain that the device may have internal logic to add and delete rows based on sensors being connected and disconnected from the device. These tables are still considered static from an NTCIP perspective since the size of the table is not managed using the protocol.

D.2.16 Seven Corrections Prior to Publication

0—Edited section 3.6.21 and other sections to modify the term "Response Time" to "Maximum Response Time".
1—Edited the SYNTAX for waterLevelSensorIndex and essSnapshotCameraIndex to INTEGER (1..255) from INTEGER (0..255), as an editorial change.
2—Edited all occurrences of <table object>.0 to read <table object>.x, as an editorial change.
3—Edited the <Definition> of essSnapshotCameraFilename to read "and the following <u>case-sensitive</u> field names enclosed in chevrons (<>):," as an editorial change.
4—Added an informative subfield to the DESCRIPTION field of essSnapshotCameraFilename which partially reads "<Informative>The filename ...," as an editorial change.
5—Added comments to the object definition, and an informative subfield, regarding the SYNTAX for essSnapshotCameraFilename, to alert implementors that syntax may change to OCTET STRING (SIZE (1..255)) from OCTET STRING (SIZE (0..255)), to prevent zero length names.
6—Edited the occurrences of the MIB Module name (especially in block objects) from "NTCIP1204-200x" to "NTCIP1204-v03," as an editorial change.

Annex E
USER REQUESTS
[INFORMATIVE]

Annex E identifies features that were suggested for NTCIP 1204 v03, but are either supported by mechanisms that may not be readily obvious, or are not supported by NTCIP 1204 v03.

E.1 FEATURES INDIRECTLY SUPPORTED

The following identifies how certain features are supported by NTCIP 1204 v03.

E.1.1 Archiving Data on a Periodic Basis for Dial-up Operations

Some users wish to configure their ESS to archive data into memory on a periodic basis so that multiple readings may be retrieved in bulk at a later time (e.g., because of a long polling cycle over a dial-up link). To operate in a meaningful manner, each entry into the archive needs to have a timestamp that identifies when the measurement was taken.

This capability is provided through the "Provide Off-Line Log Data" Architectural Need defined in Annex F.1.1.3. This architectural need can be used in conjunction with any data supported by the device.

E.2 FEATURES NOT SUPPORTED BY THIS VERSION

E.2.1 User Defined Sampling Periods

Some users have requested the ability to configure the details about how a device calculates the current reading. For example, some have requested the ability to configure an overall sampling period that is used to archive data and then, for each entry into the archive, a second sampling period over which measurements are actually taken and averaged.

ESS WG discussed this feature and concluded that it would:

a) result in a standard that was not backward compatible with NTCIP 1204 v01.
b) result in a standard that was roughly three times the size and complexity of NTCIP 1204 v03
c) be difficult to implement and test

Instead, ESS WG has followed an approach that allows all data to be monitored, measured, and archived continuously using averaging periods that are appropriate and in wide use for each parameter. This data can be uploaded to a central system for further statistical analysis, if needed.

E.2.2 Exception Reporting

Many users have requested the ability to configure the device to automatically notify the central system upon the detection of certain events (e.g., the detection of ice on the pavement, a cabinet door being opened, etc.). This is a particularly useful feature for environments where a device may be polled on a relatively infrequent basis (e.g., because of communication resource requirements).

ESS WG is committed to supporting this capability; however, certain key services need to be supported by the underlying protocol before this can be properly supported. The NTCIP Base Standards, Protocols, and Profiles WG is in the process of developing this capability as a part of a planned NTCIP 1103 v03. When NTCIP 1103 v03 is completed, this service can be provided without requiring changes to NTCIP 1204 v03.

Annex F
GENERIC CLAUSES
[NORMATIVE]

Annex F contains user needs, requirements, and dialogs that are considered to be generic to many types of NTCIP field devices. It is expected that the text contained in Annex F may eventually be defined in a separate standard. However, Annex F serves as a placeholder until this is achieved.

F.1 EXTERNAL CONCEPT OF OPERATIONS

F.1.1 Generic Architectural Needs

NTCIP 1204 v03 addresses the interface between an ESS and one or more management stations (e.g., central computers, laptops, etc.). The data collected by the ESS may include data from multiple sensors. When communicating with a management station, each reading needs to be clearly associated with a specific sensor. After the management station has retrieved the data of interest, the operator can use the retrieved data to make decisions and initiate other events (such as changes to DMS messages) to better manage the transportation system.

To enable communications between these components, the transportation system manager needs to establish a communication system that links the ESS with a management station. For some systems, the resources required for communications may be minimal and as such the system may be designed for constant polling; other systems may require significant resources for communicating with the ESS and as such the system may be designed to minimize data exchanges. When deploying an ESS, the system designer needs to consider which of the following operational environments need to be supported.

F.1.1.1 Provide Live Data

The typical operational environment allows the management system to monitor and control the device by issuing requests (e.g., requests to access information, alter information, or control the device). In this environment, the device responds to requests from the management station immediately (e.g., through the provision of live data, success/failure notice of information alteration, or success/failure of the command).

F.1.1.2 Provide Compressed Data

Some operational environments have limited data capacity because of limitations in the data rates of the media and/or because of multiple devices sharing the same communications channel. In such environments, compressed data provides the capability for grouping sets of data together so that data can be transmitted more efficiently over telecommunications networks, thereby conserving the limited data capacity of the channel.

F.1.1.3 Provide Off-line Log Data

Some operational environments do not have always-on connections (e.g., dial-up links). In such environments, a transportation system operator may wish to define conditions under which data will be placed into a log, which can then be uploaded at a later time. For example, the operator may wish to maintain a log of when the cabinet door is opened.

F.1.2 Generic Features

Annexes F.1.2.1and F.1.2.2 document features of an ESS that are generic to most devices.

F.1.2.1 Retrieve Device Identity

A transportation system operator may need to determine basic information about the device, such as its location, and the make, model, and version of the device components.

F.1.2.2 Control External Devices

A transportation system operator may need to turn simple auxiliary devices on and off. For example, the ESS may be co-located with a warning sign equipped with flashing beacons; this feature would allow the ESS controller to activate and deactivate the beacons rather than requiring an additional controller at the site.

F.2 EXTERNAL REQUIREMENTS

F.2.1 Generic Architectural Requirements

Requirements for communication capabilities follow.

F.2.1.1 Support Basic Communications

Requirements for making requests follow.

F.2.1.1.1 Retrieve Data

A management station shall be able to retrieve any set of data from the device at any time.

F.2.1.1.2 Deliver Data

A management station shall be able to deliver data (e.g., configuration data, commands, etc.) to the device at any time.

NOTE—Other requirements may place restrictions on how the device may respond under certain scenarios.

F.2.1.1.3 Explore Data

A management station shall be able to dynamically discover what data and data instances are supported by the device.

F.2.1.2 Support Logged Data

Requirements for managing the logged data follow.

F.2.1.2.1 Retrieve Current Configuration of Logging Service

Upon request from a management station, the device shall return the current configuration of the event logging service, including the classes and types of events that are currently configured.

F.2.1.2.2 Configure Logging Service

Upon request from a management station, the device shall configure the event logging service as requested, including configuration of the event classes and event types to log.

F.2.1.2.3 Retrieve Logged Data

Upon request from a management station, the device shall return the event log.

F.2.1.2.4 Clear Log

Upon request from a management station, the device shall clear the indicated log entries of a given event class.

F.2.1.2.5 Retrieve Capabilities of Event Logging Service

Upon request from a management station, the device shall return the capabilities of the event logging service, including the number of classes, number of event types, and number of events that can be supported by the device.

F.2.1.2.6 Retrieve Total Number of Logged Events

Upon request from a management station, the device shall return the total number of events that the device has detected.

F.2.2 Generic Functional Requirements

Requirements for data exchange capabilities follow.

F.2.2.1 Generic Configuration Requirements

Requirements for configuring a device controller follow.

F.2.2.1.1 Retrieve Device Component Information

Upon request from a management station, the device shall return identification information for each module contained in the device, including:

a) an indication of the type of device
b) the manufacturer of the module
c) the model number or firmware reference of the module
d) the version of the module
e) an indication of whether it is a software or hardware module

F.2.2.1.2 Retrieve Device Configuration Identifier

Upon request from a management station, the device shall return a code that only changes when changes are made to the controller configuration. The controller configuration shall consist of the objects identified in Annex H.

F.2.2.1.3 Retrieve Supported Standards

Upon request from a management station, the device shall return the NTCIP standards that it supports.

F.2.2.1.4 Retrieve System Name

Upon request from a management station, the device shall return the system name of the device.

F.2.2.1.5 Manage Time

Requirements for managing the controller's clock follow.

F.2.2.1.5.1 Set Time

Upon request from a management station, the device shall set the coordinated universal time to that requested.

F.2.2.1.5.2 Retrieve Current Time

Upon request from a management station, the device shall return the current time settings within the controller.

F.2.2.1.6 Retrieve External Port Information

Upon request from a management station, the device shall return the number of auxiliary ports and the following information for each port:

a) an indication of whether the port is analog or digital
b) a description of the port
c) an indication of the port resolution
d) an indication of whether the port can be used for input, output, or both

F.2.2.1.7 Configure Port Information

Upon request from a management station, the device shall store the indicated description for the indicated auxiliary port.

F.2.2.2 Generic Status Monitoring Requirements

Requirements for monitoring the status of a device controller follow.

F.2.2.2.1 Monitor Status of External Device

Upon request from a management station, the device shall return the following information for the indicated auxiliary port:

a) Current state
b) Last commanded state

F.2.2.3 Generic Data Retrieval Requirements

There are no data retrieval requirements for a generic device controller.

F.2.2.4 Generic Control Requirements

Requirements for controlling a device controller follow.

F.2.2.4.1 Control External Device

Upon request from a management station, the device shall activate or de-activate, as requested, a simple external device connected through an analog auxiliary port.

F.2.3 Generic Supplemental Requirements

Supplemental requirements follow.

F.2.3.1 Supplemental Requirements for Event Monitoring

Supplemental requirements for monitoring for the occurrence of certain events follow.

F.2.3.1.1 Record and Timestamp Events

Upon detection of a configured event, the device shall record the event type, the current time, and the configured log information in a local log (log contained in the device controller).

F.2.3.1.2 Support a Number of Event Classes

The device shall support the number of event classes as defined by the specification. If the specification does not define the number of event classes, the device shall support at least one event class.

F.2.3.1.3 Support a Number of Event Types to Monitor

The device shall support the number of event types as defined by the specification. If the specification does not define the number of event types, the device shall support at least one event type.

F.2.3.1.4 Support Monitoring of Event Types

Supplemental requirements for monitoring types of events follow.

F.2.3.1.4.1 Support On-Change Events

The device shall allow any event type configuration to monitor data for changes in value.

F.2.3.1.4.2 Support Greater Than Events

The device shall allow any event type configuration to monitor data for values exceeding a defined threshold for a period of time.

F.2.3.1.4.3 Support Less Than Events

The device shall allow any event type configuration to monitor data for values falling below a defined threshold for a period of time.

F.2.3.1.4.4 Support Hysteresis Events

The device shall allow any event type configuration to monitor data for values exceeding an upper limit or dropping below a lower limit.

F.2.3.1.4.5 Support Periodic Events

The device shall allow any event type configuration to monitor data on a periodic basis.

F.2.3.1.4.6 Support Bit Flag Events

The device shall allow any event type configuration to monitor one or more bits of a value becoming true (i.e., obtaining a value of one).

F.2.3.1.5 Support Event Monitoring on Any Data

The device shall allow any event type configuration to monitor any piece of data in the device within the logical rules of the type of event (e.g., ASCII strings should not be monitored with greater than or less than conditions).

F.2.3.1.6 Support a Number of Events to Store in Log

The device event log shall support the number of events as defined by the specification. If the specification does not define the number of events for the log, the device shall support at least one event in the log.

F.2.3.2 Required Number of Auxiliary Ports

The device shall support the number of analog auxiliary ports of the resolution and direction (input, output, or bidirectional) specified in the specification. If the specification does not define the number, resolution, or direction of analog ports, the device shall support at least one binary analog output port for external device control.

F.3 SNMP INTERFACE

The NTCIP field device shall conform to the requirements for the Simple Network Management Protocol (SNMP) as defined in NTCIP 1103 v01. [In RFC 1157, sections 4.1.2 through 4.1.5 (GetRequest, GetNextRequest, GetResponse, and SetRequest),] provide a description of the key services offered by SNMP assuming no errors; precise rules and procedures are defined in NTCIP 1103 v01. [RFC 1157 section 4.1.6 (Trap)] extends the requirements of NTCIP 1103 v01 by providing additional requirements that supplement, but do not replace, any requirements of NTCIP 1103 v01.

[Editor’s NOTE—the prior v03.07b text which referenced “Clause C.4.1.1,” et al, was assumed to refer to RFC 1157 sections. The references have been revised in v03.08.]

NOTE—To promote interoperability and to reflect marketplace realities, NTCIP 1204 v03 requires support for the Simple Network Management Protocol. Use of the other protocols defined in NTCIP 1103 v01 (i.e., the Simple Transportation Management Protocol and the Simple Fixed Message Protocol) is discouraged for ESS as these have not been widely implemented in ESS, and thus would likely result in decreased interoperability, limited competition, and increased resource needs for testing, integration, and maintenance.

F.3.1 Generic SNMP Get Interface

SNMP defines a generic process by which a management station can retrieve data from a device. This process consists of a Get request (GET) and a GetResponse as depicted in Figure 22. Both the Get request and the GetResponse messages contain a list of objects as defined by the varBindingList structure (see Annex C.4.1.4).

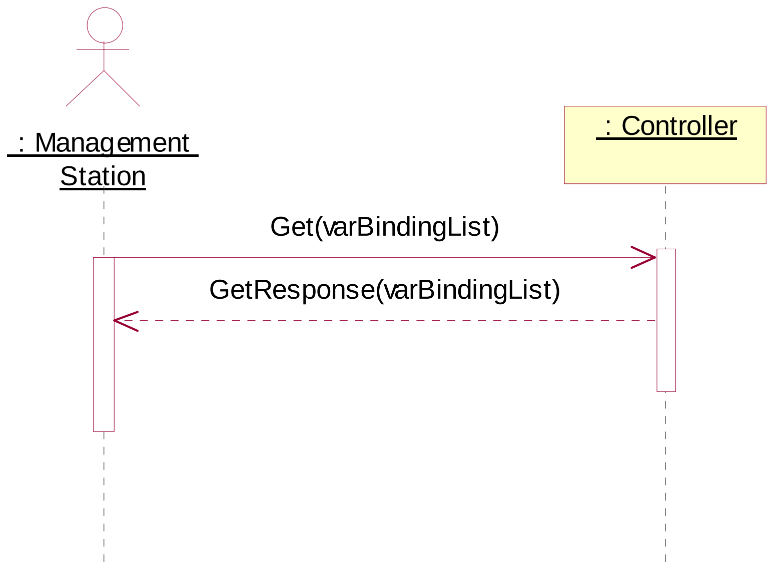

Figure 22 SNMP Get Interface

The RTM (Annex A) customizes this generic process by calling out the appropriate objects to meet specific requirements as defined in Section 3.

F.3.2 Generic SNMP Get-Next Interface

SNMP defines a process by which a management station can explore data within a device to fulfill the requirement as defined in Annex C.3.1.1.2. This process consists of a GetNext request and a GetResponse as depicted in Figure 23. Both the GetNext request and the GetResponse messages contain a list of objects as defined by the varBindingList structure (see Annex C.4.1.4).

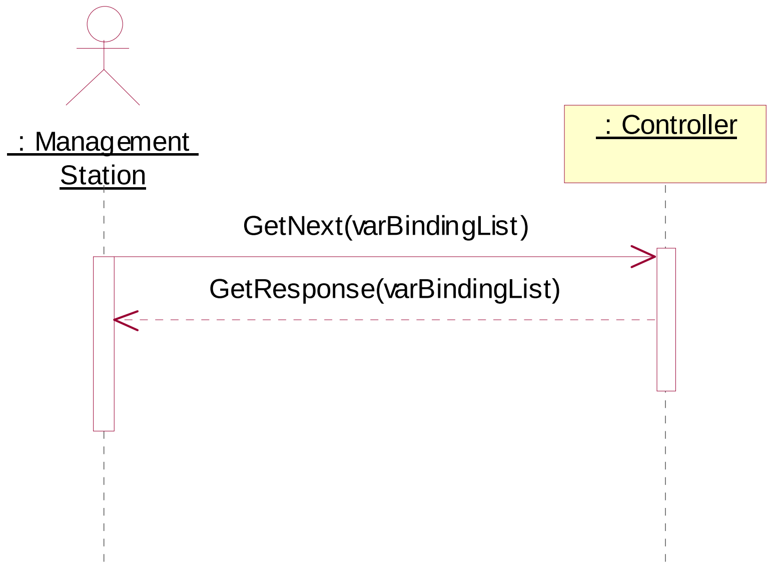

Figure 23 SNMP GetNext Interface

F.3.3 Generic SNMP Set Interface

SNMP defines a generic process by which a management station can send data to a device. This process consists of a Set request and a GetResponse [sic] as depicted in Figure 24. Both the Set request and the GetResponse messages contain a list of objects as defined by the varBindingList structure (see Annex C.4.1.4).

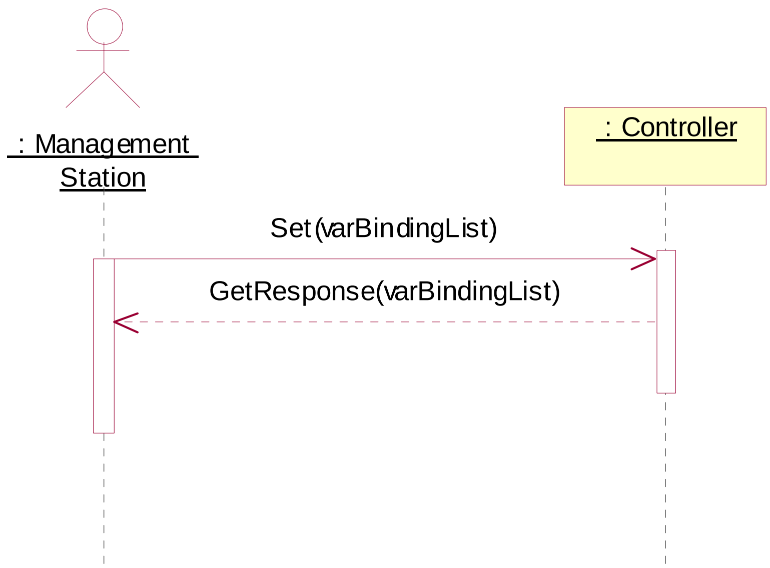

Figure 24 SNMP Set Interface

NOTE—The response message issued to an SNMP Set request is the same message structure as used to respond to an SNMP Get request. The SNMP standard calls this response message a GetResponse, but it is in fact a response to either a GET or a SET.

This generic process is customized by subsequent sections of NTCIP 1204 v03, by referencing the SET operation, and directly by the RTM, by section number, to fulfill a wide range of the requirements defined in Section 3. Additional rules for SETs are defined by the Control Mode State Machine. (See Section 4.4.5.3.)

F.3.4 Variable Binding List Structure

The requests and responses for the Get, Get Next, and Set operations all use the varBindingList structure. NTCIP 1103 v01 defines this structure as containing zero or more varBindings, where each varBinding is defined to consist of an object name (as indicated by an Object Identifier (oid)) and the associated object value. This relationship is depicted in Figure 25.

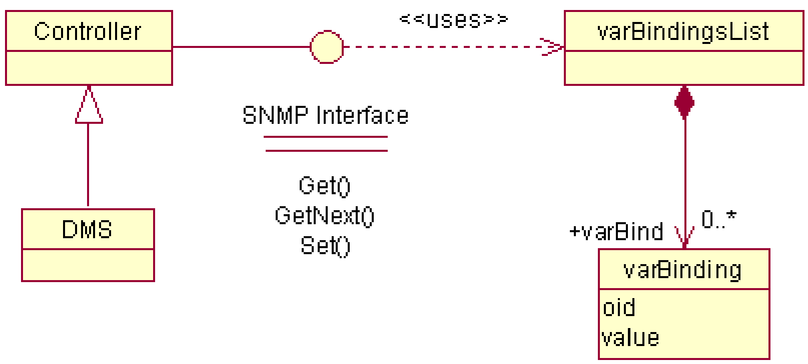

Figure 25 SNMP Interface—View of Participating Classes

Additional requirements related to the contents of the variable binding list follow.

F.3.4.1 Grouping of Objects in a Request

The NTCIP field device shall allow the management station to perform a single Get, GetNext, or Set operation on any combination of supported objects with the objects listed in any order within the message, unless otherwise restricted by this standard.

The NTCIP field device shall not associate any semantics to the ordering of objects within the varBindingsList. As required by RFC 1157 section 4.1.5, each object shall be affected “as if simultaneously set with respect to all other assignments specified in the same message.”

F.3.4.2 Support of Get

The NTCIP field device shall allow the management station to perform the Get operation on any supported object for which the ACCESS field indicates “read-only”’or “read-write” in Section 5.

F.3.4.3 Support of GetNext

The NTCIP field device shall allow the management station to perform the GetNext operation on any OBJECT IDENTIFIER.

F.3.4.4 Support of Set

The NTCIP field device shall allow the management station to perform the Set operation on any supported object for which the ACCESS field indicates “read-write” in Section 5; however, the ability to perform a set may be restricted by the object definition itself or rules defined in [NTCIP 1204 v03] Annex F.4.3.

F.3.4.5 Properly Defined Objects

Every supported object shall be defined in a manner that conforms to RFC 1212 and shall have a unique OBJECT IDENTIFIER properly registered under the ISO Naming Tree. If the definition of the supported object is controlled by parties within the ITS community, the object definition should also conform to NTCIP 8004 v02.

F.4 GLOBAL CUSTOM DIALOGS

F.4.1 Retrieve Logged Data

The standardized dialog for a management station to retrieve logged data shall be as follows:

a) (Precondition) The management station shall be aware of the number of events that had previously been reported for the device for the subject event class (e.g., from the previous performance of this operation).
b) The management station shall GET the following data:
    1) eventClassNumRowsInLog.x
    2) eventClassNumEvents.x
c) If eventClassNumEvents.x has not changed since the previous reading, the management station shall exit the process. Otherwise, the management station shall determine the additional number of events that have occured since the last read.

    NOTE—This is generally determined by subtracting the previous number of events from eventClassNumEvents; however, since this object wraps at 65535, the management station should be prepared to determine the differential if eventClassNumEvents is less than the previous number.

d) The management station shall determine the lesser of eventClassNumRowsInLog and the additional number of events that have occured since the last read. This number shall be termed the Events to Read.
e) Starting with y = eventClassNumRowsInLog and working down until y = (eventClassNumRowsInLog - Events to Read), the management station shall GET the following data:
    1) eventLogID.x.y
    2) eventLogTime.x.y
    3) eventLogValue.x.y
f) Repeat the same GET operation with y decremented by one (1) for each set of duplicated values (until y reaches a value of zero (0)).

    NOTE—If the event class is full and another event occurs, the new event is recorded in the last entry and all previously logged data is moved to one index lower with index 1 being deleted from the table. Thus, if a duplicate row is detected (i.e., same event at same time), it is likely an indication that the same event is being read and that a new event was added to the log.

    NOTE—The management station may wish to clear the event log after the read to minimize the above problem.

Where:
    x = event log class
    y = event log number

F.4.2 Retrieve Current Configuration of Logging & Exception Reporting Service

The standardized dialog for a management station to determine the current configuration of the logging service and/or exception reporting events shall be as follows:

a) (Precondition) The management station shall be aware of the number of classes and event configurations supported by the device.

b) For each row of the event class table, the management station shall GET the following data:
   1) eventClassLimit.x
   2) eventClassClearTime.x
   3) eventClassDescription.x
c) For each row of the event configuration table, the management station shall GET the following data:
   1) eventConfigClass.y
   2) eventConfigMode.y
   3) eventConfigCompareValue.y
   4) eventConfigCompareValue2.y
   5) eventConfigCompareOID.y
   6) eventConfigLogOID.y
   7) eventConfigAction.y
   8) eventConfigStatus.y

Where:
x = event class number
y = event configuration identifier

F.4.3 Configure Logging Service

The standardized dialog for a management station to configure the logging service or events to be reported shall be as follows:

a) (Precondition) The management station shall ensure that there are sufficient rows in the event configuration and event class tables to download the proposed configuration.
b) The management station shall SET the following data to the desired values to configure each desired event class:
   1) eventClassLimit.x
   2) eventClassClearTime.x
   3) eventClassDescription.x

   NOTE—Each event type to be monitored is classified into one event class. For example, critical events may be grouped into Class 1 events and warnings may be grouped into Class 2 events. This step, defines the structure of each class of events.

c) The management station shall SET the following data to the desired values to configure each desired event to be monitored:
   1) eventConfigClass.y
   2) eventConfigMode.y
   3) eventConfigCompareValue.y
   4) eventConfigCompareValue2.y
   5) eventConfigCompareOID.y
   6) eventConfigLogOID.y
   7) eventConfigAction.y

   NOTE—Depending on the value of eventConfigMode, not all other objects may be necessary for the event to be defined, however, they shall always be SET as a part of this standardized dialog.

d) The management station shall GET eventConfigStatus.y to ensure that there is not an error in the configuration.

Where:
x = event class number
y = event configuration identifier

F.4.4 Configure Events

See NTCIP 1103 v01 for the definition of how events shall be managed.

F.4.5 Manage Exception Reporting

See NTCIP 1103 v01 for the definition of how events shall be managed.

F.4.6 Generic Retrieve Table Dialog

NOTE—This is a generic dialog that is referenced by requirements in the RTM with specific object names. The list of objects provided by the specific dialog shall include:

a) an object that indicates the number of rows in the table,
b) the object(s) that serve as the index field of the table row, and
c) the list of columnar objects to be retrieved from the table.

The standardized dialog for a management station to retrieve a table shall be as follows:
a) The management station shall GET the number of rows in the table.
b) For each row of the table, the management station shall GET all objects referenced by the specific dialog that references this generic dialog, except for the number of rows object and the index object(s).

F.4.7 Generic Retrieve Table Row Dialog

NOTE—This is a generic dialog that is referenced by other dialogs with specific object names. The list of objects provided by the specific dialog shall include:

a) the object(s) that serve as the index field of the table row, and
b) the list of columnar objects to be retrieved from the table.

The standardized dialog for a management station to retrieve a table shall be as follows:

a) (Precondition) The management station shall be aware of which row of the table is to be retrieved.
b) For the specified row, the management station shall GET all objects referenced by the specific dialog that references this generic dialog, except for the index object(s).

F.4.8 Generic Configure Table Row

NOTE--This is a generic dialog that is referenced by other dialogs with specific object names. The list of objects provided by the specific dialog shall include:

a) the object(s) that serve as the index field of the table row, and
b) the list of columnar objects to be configured and their desired values.

The standardized dialog for a management station to configure a table row shall be as follows:

a) (Precondition) The management station shall be aware of which row in the table is to be configured.
b) For the specified row, the management station shall SET all objects (to their desired values) referenced by the specific dialog that references this generic dialog, except for the index object(s).

Annex G
ENCODING OF SAMPLE BLOCK OBJECTS
[INFORMATIVE]

This Annex provides a sample encoding for each block object defined within this standard in an effort to further clarify the intent of their definitions.

G.1 STATION META DATA BLOCK

FF E0	*Preamble; everything present*
02	*essNtcipCategory.0 = 2 = 'permanent'*
00	*essTypeOfStation.0 = 0 = 'automatic'*
02 51 F0 D4	*essLatitude = 38924500 = 38.9245*
FB 63 0B C8	*essLongitude = -77395000 = -77.3950*
00 32	*essReferenceHeight.0 = 50*
00 02	*essPressureHeight.0 = 2*
00 05	*essWindSensorHeight.0 = 5*
	temperatureMetaData
01 01	*1-byte quantity field indicating 1 instance of component*
C0	*Preamble of SEQUENCE; both fields are present*
01	*essTemperatureSensorIndex.x = 1*
00 02	*essTemperatureSensorHeight.x = 2*
	pavementMetaData
01 02	*1-byte quantity field indicating 2 instances of component*
F8	*Preamble of first SEQUENCE; all 5 fields are present*
01	*essPavementSensorIndex.x = 1*
03	*essPavementType.x = 3 = 'asphalt'*
FF FE	*essPavementElevation.x = -2*
32	*essPavementExposure.x = 50*
03	*essPavementSensorType.x = 3 = 'contactActive'*
F8	*Preamble of second SEQUENCE; all 5 fields are present*
02	*essPavementSensorIndex.x = 2*
03	*essPavementType.x = 3 = 'asphalt'*
00 03	*essPavementElevation.x = 3*
64	*essPavementExposure.x = 100*
03	*essPavementSensorType.x = 3 = 'contactActive'*
	subSurfaceMetaData
01 01	*1-byte quantity field indicating 1 instance of component*
E0	*Preamble of SEQUENCE; all 3 fields are present*
01	*essSubSurfaceSensorIndex.x = 1*
03	*essSubSurfaceType.x = 3 = 'concrete'*
00 05	*essSubSurfaceSensorDepth.x = 5*
	treatmentMetaData
01 02	*1-byte quantity field indicating 2 instances of component*
F0	*Preamble of SEQUENCE; all fields are present*
01	*essPavementTreatmentIndex = 1*
02	*essPaveTreatProductType = 2 (sand)*
02	*essPaveTreatProductForm = 2 (dry)*
60	*essPercentProductMix = 96*
F0	*Preamble of SEQUENCE; all fields are present*
01	*essPavementTreatmentIndex = 1*

08	*essPaveTreatProductType = 8 (naCl)*
02	*essPaveTreatProductForm = 2 (dry)*
04	*essPercentProductMix = 4*

G.2 STATION META DATA BLOCK V3

FF C0	*Preamble; first 10 fields are present, treatmentMetaData is not*
02	*essNtcipCategory.0 = 2 = 'permanent'*
00	*essTypeOfStation.0 = 0 = 'automatic'*
02 51 F0 D4	*essLatitude = 38924500 = 38.9245*
FB 63 0B C8	*essLongitude = -77395000 = -77.3950*
00 32	*essReferenceHeight.0 = 50*
00 02	*essPressureHeight.0 = 2*
	windMetaData
01 01	*1-byte quantity field indicating 1 instance of component*
C0	*Preamble of SEQUENCE; both fields are present*
01	*windSensorIndex.x = 1*
00 02	*windSensorHeight.x = 2*
	temperatureMetaData
01 01	*1-byte quantity field indicating 1 instance of component*
C0	*Preamble of SEQUENCE; both fields are present*
01	*essTemperatureSensorIndex.x = 1*
00 02	*essTemperatureSensorHeight.x = 2*
	pavementMetaData
01 02	*1-byte quantity field indicating 2 instances of component*
F8	*Preamble of first SEQUENCE; all 5 fields are present*
01	*essPavementSensorIndex.x = 1*
03	*essPavementType.x = 3 = 'asphalt'*
FF FE	*essPavementElevation.x = -2*
32	*essPavementExposure.x = 50*
03	*essPavementSensorType.x = 3 = 'contactActive'*
F8	*Preamble of second SEQUENCE; all 5 fields are present*
02	*essPavementSensorIndex.x = 2*
03	*essPavementType.x = 3 = 'asphalt'*
00 03	*essPavementElevation.x = 3*
64	*essPavementExposure.x = 100*
03	*essPavementSensorType.x = 3 = 'contactActive'*
	subSurfaceMetaData
01 01	*1-byte quantity field indicating 1 instance of component*
E0	*Preamble of SEQUENCE; all 3 fields are present*
01	*essSubSurfaceSensorIndex.x = 1*
03	*essSubSurfaceType.x = 3 = 'concrete'*
00 05	*essSubSurfaceSensorDepth.x = 5*

G.3 MOBILE DATA BLOCK

FC 00	*Preamble; first 6 fields are present; weather profile and PTS are not supported*
02 51 F0 D4	*essLatitude.0 = 38924500 = 38.9245*
FB 63 0B C8	*essLongitude.0 = -77395000 = -77.3950*
00 32	*essReferenceHeight.0 = 50*
50	*essVehicleSpeed.0 = 80 kph*
00 5A	*essVehicleBearing.0 = 90 = due east*
00 00 C3 50	*essVehicleOdemeter.0 = 50,000 meters*

G.4 PAVEMENT TREATMENT BLOCK

E0 00	*Preamble; first 3 fields are present, others are not*
	treatmentInfo
01 02	*1-byte quantity field indicating 2 instances of component*
0F	*Preamble of SEQUENCE; all four fields are present*
01	*essPavementTreatmentIndex.x = 1*
02	*essPaveTreatProductType.x = 'sand'*
02	*essPaveTreatProductForm.x = 'dry'*
5F	*essPercentProductMix.x = 95*
0F	*Preamble of SEQUENCE; all four fields are present*
02	*essPavementTreatmentIndex.x = 1*
08	*essPaveTreatProductType.x = 'naCl'*
02	*essPaveTreatProductForm.x = 'dry'*
05	*essPercentProductMix.x = 5*
0A	*essPaveTreatmentAmount.0 = 10*
03	*essPaveTreatmentWidth.0 = 3*

G.5 WEATHER BLOCK

F8	*Preamble; everything present*
10 00	*essAtmosphericPressure.0 = 4096*
	essWindData
FE	*Preamble – all fields present*
00 00	*essAvgWindDirection.0 = 0*
00 10	*essAvgWindSpeed.0 = 16*
03	*essWindSituation.0 = 3 (calm)*
00 20	*essMaxWindGustSpeed.0 = 32*
00 05	*essMaxWindGustDir.0 = 5*
00 15	*essSpotWindDirection.0 = 21*
01 63	*essSpotWindSpeed.0 = 355*
	essTemperatureData
FC	*Preamble of SEQUENCE; all 6 fields are present*
00 A0	*essWetBulbTemp.0 = 160*
00 A0	*essDewpointTemp.0 = 160*
01 40	*essMaxTemp.0 = 320*
00 50	*essMinTemp.0 = 80*
00 50	*essRelativeHumidity.0 = 80*
	temperatureTable
01 01	*1-byte quantity field indicating 1 instance of component*
C0	*Preamble of SEQUENCE; both fields are present*
01	*essTemperatureSensorIndex.1 = 1*
00 A0	*essAirTemperature.1 = 160*
	essPrecipData
8D 60	*Preamble – device supports essWaterDepth, PrecipYesNo, PrecipRate, PrecipSituation, StartTime and EndTime*
01 02	*essWaterDepth.0 = 258*
01	*essPrecipYesNo.0 = 1 (precip)*
02 EE	*essPrecipRate.0 = 750*
03	*essPrecipSituation.0 = 5 = 'unidentifiedModerate'*
45 44 51 8E	*essPrecipitationStartTime.0 = 1162105230 = 10/29/06 7:00:30*
45 44 5F 81	*essPrecipitationEndTime.0 = 1162108801 = 10/29/06 8:00:01*
	essVisibilityData
F8	*Preamble – device supports all fields*

01 00	*essSolarRadiation.0 = 256*
01 00	*essTotalSun.0 = 256*
01	*essCloudSituation.0 = 1 (overcast)*
00 00 A0 00	*essVisibility.0 = 40960*
00 03	*essVisibilitySituation.0 = 3 'clear'*

G.6 WEATHER BLOCK V3

F8	*Preamble; everything present*
10 00	*essAtmosphericPressure.0 = 4096*
	essWindData
01 01	*1-byte quantity field indicating 1 instance of component*
FF	*Preamble – everything present*
01	*windSensorIndex.1 = 1*
00 10	*windSensorAvgSpeed.1 = 16*
00 00	*windSensorAvgDirection.1 = 0*
01 63	*windSensorSpotSpeed.1 = 355*
00 15	*windSensorSpotDirection.1 = 21*
00 20	*windSensorGustSpeed.1 = 32*
00 05	*windSensorGustDirection.1 = 5*
03	*essWindSituation.1 = 3 (calm)*
	essTemperatureData
FC	*Preamble of SEQUENCE; all 6 fields are present*
00 A0	*essWetBulbTemp.0 = 160*
00 A0	*essDewpointTemp.0 = 160*
01 40	*essMaxTemp.0 = 320*
00 50	*essMinTemp.0 = 80*
00 50	*essRelativeHumidity.0 = 80*
	temperatureTable
01 01	*1-byte quantity field indicating 1 instance of component*
C0	*Preamble of SEQUENCE; both fields are present*
01	*essTemperatureSensorIndex.1 = 1*
00 A0	*essAirTemperature.1 = 160*
	essPrecipData
8D 60	*Preamble – device supports waterLevelSensorTable, PrecipYesNo, PrecipRate, PrecipSituation, StartTime and EndTime*
	waterLevelSensorTable
01 01	*1-byte quantity field indicating 1 instance of component*
C0	*Preamble – all fields present*
01	*waterLevelSensorIndex = 1*
01 02	*waterLevelSensorReading.1 = 258*
01	*essPrecipYesNo.0 = 1 (precip)*
02 EE	*essPrecipRate.x = 750*
03	*essPrecipSituation.x = 5 = 'unidentifiedModerate'*
45 44 51 8E	*essPrecipitationStartTime.x = 1162105230 = 10/29/06 7:00:30*
45 44 5F 81	*essPrecipitationEndTime.x = 1162108801 = 10/29/06 8:00:01*
	essVisibilityData
C0	*Preamble – device supports all fields*
00 00 A0 00	*essVisibility.0 = 40960*
00 03	*essVisibilitySituation.0 = 3 'clear'*
	essRadiationData
01 00	*essTotalSun.0 = 256*
FF FF	*essInstantaneousTerrestrialRadiation.0 = -1*
01 00	*essInstantaneousSolarRadiation.0 = 256*

00 FF	*essTotalRadiation.0 = 255*
0E 10	*essTotalRadiationPeriod.0 = 3600*
01	*essCloudSituation.0 = 1 (overcast)*

G.7 PAVEMENT BLOCK

01 02	*1-byte quantity field indicating 2 instances of component*
FF C0	*Preamble; everything present*
01	*essPavementSensorIndex.x = 1*
03	*essSurfaceStatusV2.x = 3 'dry'*
00 C8	*essSurfaceTemperature.x = 200*
00 B4	*essPavementTemperature.x = 180*
00	*essSurfaceWaterDepth.x = 0*
00 10	*essSurfaceSalinity.x = 16*
00 00	*essSurfaceConductivity.x = 0*
FF F6	*essSurfaceFreezePoint.x = -10*
02	*essSurfaceBlackIceSignal.x = 2 'noIce'*
02	*essPavementSensorError.x = 2 'none'*
FF C0	*Preamble; everything present*
02	*essPavementSensorIndex.x = 2*
03	*essSurfaceStatusV2.x = 3 'dry'*
00 C8	*essSurfaceTemperature.x = 200*
00 B4	*essPavementTemperature.x = 180*
00	*essSurfaceWaterDepth.x = 0*
00 10	*essSurfaceSalinity.x = 16*
00 00	*essSurfaceConductivity.x = 0*
FF F6	*essSurfaceFreezePoint.x = -10*
02	*essSurfaceBlackIceSignal.x = 2 'noIce'*
02	*essPavementSensorError.x = 2 'none'*

G.8 PAVEMENT BLOCK V3

01 02	*1-byte quantity field indicating 2 instances of component*
FF C0	*Preamble; everything present*
01	*essPavementSensorIndex.x = 1*
03	*essSurfaceStatus.x = 3 'dry'*
00 C8	*essSurfaceTemperature.x = 200*
00 B4	*essPavementTemperature.x = 180*
00 10	*essSurfaceSalinity.x = 16*
FF F6	*essSurfaceFreezePoint.x = -10*
02	*essSurfaceBlackIceSignal.x = 2 'noIce'*
02	*essPavementSensorError.x = 2 'none'*
00 00	*essSurfaceIceOrWaterDepth.x = 0*
00 00	*essSurfaceConductivityV2.x = 0*
FF C0	*Preamble; everything present*
02	*essPavementSensorIndex.x = 2*
03	*essSurfaceStatus.x = 3 'dry'*
00 C8	*essSurfaceTemperature.x = 200*
00 B4	*essPavementTemperature.x = 180*
00 10	*essSurfaceSalinity.x = 16*
FF F6	*essSurfaceFreezePoint.x = -10*
02	*essSurfaceBlackIceSignal.x = 2 'noIce'*
02	*essPavementSensorError.x = 2 'none'*
00 00	*essSurfaceIceOrWaterDepth.x = 0*

00 00	*essSurfaceConductivityV2.x = 0*

G.9 SUBSURFACE BLOCK

01 02	*1-byte quantity field indicating 2 instances of component*
F0	*Preamble; everything present*
01	*essSubSurfaceSensorIndex.x = 1*
00 50	*essSubSurfaceTemperature.x = 80*
14	*essSubSurfaceMoisture.x = 20*
02	*essSubSurfaceSensorError.x = 2 'none'*
F0	*Preamble; everything present*
02	*essSubSurfaceSensorIndex.x = 2*
00 50	*essSubSurfaceTemperature.x = 80*
14	*essSubSurfaceMoisture.x = 20*
02	*essSubSurfaceSensorError.x = 2 'none'*

G.10 AIR QUALITY BLOCK

FE	*Preamble; everything present*
64	*essCO.0 = 100*
0F A0	*essCO2.0 = 4000*
32	*essNO.0 = 50*
A0	*essNO2.0 = 160*
A0 00	*essSO2.0 = 40960*
50	*essO3.0 = 80*
00 20	*essPM10.0 = 32*

Annex H
CONTROLLER CONFIGURATION OBJECTS
[NORMATIVE]

The controller configuration shall be defined by all instances of the following objects that are supported by the device:

a) globalMaxModules
b) moduleDeviceNode
c) moduleMake
d) moduleModel
e) moduleVersion
f) moduleType
g) essNtcipCategory
h) essNtcipSiteDescription
i) essTypeOfStation
j) essPressureHeight
k) windSensorTableNumSensors
l) windSensorHeight
m) windSensorLocation
n) essNumTemperatureSensors
o) essTemperatureSensorHeight
p) precipitationSensorModelInformation
q) waterLevelSensorTableNumSensors
r) numEssPavementSensors
s) essPavementSensorLocation
t) essPavementType
u) essPavementElevation
v) essPavementExposure
w) essPavementSensorType
x) pavementSensorModelInformation
y) pavementSensorTemperatureDepth
z) numEssSubSurfaceSensors
aa) essSubSurfaceSensorLocation
bb) essSubSurfaceType
cc) essSubSurfaceSensorDepth
dd) numEssTreatments
ee) essPaveTreatProductType
ff) essPaveTreatProductForm
gg) essPercentProductMix
hh) essPaveTreatmentAmount
ii) essPaveTreatmentWidth
jj) ptsSignalDuration
kk) ptsMonitoringDetectors
ll) essSnapshotNumberOfCameras
mm) essSnapshotCameraDescription

A change to any of these object values shall also cause a change to the value for globalSetIDParameter.

§

