A Joint Standard of AASHTO, ITE, and NEMA

NTCIP 1204 version v02

National Transportation Communications for ITS Protocol

Object Definitions for Environmental Sensor Station v02

published in May 2008

A major revision of NTCIP 1204 v01

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 22209-3806

file version 1204 v02.24r rita 080512

NOTICES

Copyright Notice

PDF File License Agreement

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

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

Data Dictionary and MIB Distribution Permission

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

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

commercial hardware or software products intended for field implementation;
(iii) use of the DD or MIB is restricted in that the syntax fields may be modified only to reflect a more restrictive subrange or enumerated values;
(iv) the description field may be modified but only to the extent that: (a) only those bit values or enumerated values that are supported are listed; and (b) the more restrictive subrange is expressed.

These materials are delivered "AS IS" without any warranties as to their use or performance.

AASHTO / ITE / NEMA and their suppliers do not warrant the performance or results you may obtain by using these materials. AASHTO / ITE / NEMA and their suppliers make no warranties, express or implied, as to noninfringement of third party rights, merchantability, or fitness for any particular purpose. In no event will AASHTO / ITE / NEMA or their suppliers be liable to you or any third party for any claim or for any consequential, incidental or special damages, including any lost profits or lost savings, arising from your reproduction or use of these materials, even if an AASHTO / ITE / NEMA representative has been advised of the possibility of such damages.

Some states or jurisdictions do not allow the exclusion or limitation of incidental, consequential, or special damages, or the exclusion of implied warranties, so the above limitations may not apply to a given user.

Use of these materials does not constitute an endorsement or affiliation by or between AASHTO, ITE, or NEMA and the user, the user's company, or the products and services of the user's company.

If the user is unwilling to accept the foregoing restrictions, he or she should immediately return these materials.

PRL and RTM Distribution Permission

To the extent that these materials are distributed by AASHTO / ITE / NEMA in the form of a Profile Requirements List ("PRL") or a Requirements Traceability Matrix ("RTM"), AASHTO / ITE / NEMA extend the following permission:

(i) you may make or distribute unlimited copies, including derivative works of the PRL (then known as a Profile Implementation Conformance Statement ("PICS")) or the RTM, provided that each copy you make or distribute contains the citation "Based on NTCIP 0000 [insert the standard number] PRL or RTM. Used by permission. Original text © AASHTO / ITE / NEMA.";
(ii) you may only modify the PRL or the RTM by adding text in the Project Requirements column, which is the only column that may be modified to show a product's implementation or the project-specific requirements;
(iii) you may only modify the PRL or the RTM by adding additional table columns or table rows that are clearly labeled as ADDITIONAL for project-unique or vendor-unique features; and
(iv) if the PRL or RTM excerpt is made from an unapproved draft, add to the citation "PRL (or RTM) excerpted from a draft standard containing preliminary information that is subject to change."

This limited permission does not include reuse in works offered by other standards developing organizations or publishers, and does not include reuse in works-for-hire, compendiums, or electronic storage devices that are not associated with procurement documents, or commercial hardware, or commercial software products intended for field installation.

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

Content and Liability Disclaimer

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

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

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

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

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

Trademark Notice

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

ACKNOWLEDGEMENTS

This publication was prepared by the NTCIP Environmental Sensor Station (ESS) Working Group, 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.

At the time that this standard was prepared, the following individuals were active members of the NTCIP ESS Working Group:

- Gene Martin (Chair)
- Peter Davies
- Kenney Fisher
- Sharon Jeffers
- Phil Labanca
- Dan Roosevelt
- Larry Senn
- Brent Trainor
- Ken Vaughn
- John Whited

Other individuals providing input to the standard included:

- Chris Bates
- Blake Christie
- Paul Gowing
- Steve Howe
- Dale Keep
- Paul Pisano
- Colin Rayman

In addition to the many volunteers, recognition is also given to those organizations which supported the effort of the working group by providing funding for the standard, providing travel support, and submitting comments, including:

- U.S. Department of Transportation, Research and Innovative Technology Administration
- Aurora
- Castle Rock Consulting
- Colorado DOT
- Environment Canada
- Ice and Snow Technologies
- Iowa DOT
- Ministry of Ontario
- Mitretek
- Qualimetrics
- Surface Systems, Incorporated
- Trevilon Corporation
- Vaisala
- Virginia DOT
- Washington State DOT

FOREWORD

NTCIP 1204 v02 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-compliant fashion.

This NTCIP standards publication was developed, maintained, and approved in accordance with the Joint AASHTO, ITE, and NEMA NTCIP policies and procedures. This NTCIP standards publication has been Jointly Approved and published, and shall be cited as the major version NTCIP 1204 v02. This major version is the last and final revision in a series of major-plus-minor version numbers. The major-plus-minor version number is in the header on each page to identify this major version.

NTCIP 1204 v02 is an NTCIP Data Dictionary Standard. Data Dictionary Standards provide definitions of data elements for use within NTCIP systems. A Joint NTCIP Data Dictionary Standard is equivalent to these document types at the standards organizations:

AASHTO – Standard Specification
ITE – Software Standard
NEMA – Standard

There are 3 normative and 3 informative annexes in NTCIP 1204 v02:

(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 this standard.
(c) Annex C is normative and is a placeholder for test procedures that may be added to this standard at a later date.
(d) Annex D is informative and identifies the significant revisions in NTCIP 1204 v02 that have been made since NTCIP 1204 v01.
(e) Annex E is informative and responds to user requests by providing an explanation as to how certain complex features can be supported by the standard and why certain other features are not supported by this version of the standard.
(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.

NTCIP 1204 v02 uses only metric units.

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

User Comment Instructions

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

All User Comments will be referred to the committee responsible for developing or maintaining this standards publication. The committee chairperson, or their designee, may contact the submitter for

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

A User Comment should be submitted to this address:

NTCIP Coordinator
National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, Virginia 22209-3806
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

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

AASHTO – Standard Specification; April 2006
ITE – Software Standard; October 2005
NEMA – Standard; October 2005

History

The development history of NTCIP 1204 is:

TS 3.7 version 97.01.11. Approved by AASHTO, ITE, and NEMA in 1998.

TS 3.7 Amendment 1, version 98.01.02. Approved by AASHTO and NEMA in 1999 and approved by ITE in 2001.

NTCIP 1204:1998 v01.12. Combined TS 3.7 version 01.11 and Amendment 1 version 02, converted to the 1204 numbering scheme, and made changes in sections: 3.2.2, 3.3.1, 3.8.11, 3.8.12, 3.11.2.10, 3.11.2.11, 3.11.2.12, 3.11.4.3, 3.11.4.4, 3.14.1 through 3.14.7, and 4.2.1.

NTCIP 1204:1998 v01.13. November 2001 edition – Reformatted for printing: incremented version number, updated document to NTCIP 8002 v04 Template format, added approvals and history to Foreword, and converted references to TS 3 Standards to renumbered NTCIP Standards. The underline and strikethrough format used to indicate the amendment changes was removed. The normative reference to WMO BUFR was corrected and References were updated.

NTCIP 1204 v02. Starting in 2001, version 02 was developed to reflect lessons learned, to update to new document formats, and to add new features such as the control of automated de-icing equipment. In May 2004, the Joint Committee on the NTCIP accepted v02.18 as a User Comment Draft. Standards Bulletin B0098 sent v02.19b out for comment.

NTCIP 1204 v02.22 – In March 2005, the Joint Committee on the NTCIP accepted v02.22 as a Recommended Standard. In April 2005, User Comment UC0354 was considered and disposed to produce v02.23b.

NTCIP 1204 v02.23b was approved by AASHTO in April 2006, approved by ITE in October 2005, and approved by NEMA in October 2005. NTCIP 1204 v02 was Jointly Approved in 2006.

NTCIP 1204 v02.24. In November 2007, errata were approved to revise NTCIP 1204 v02 for:
section 4.2.1 Capture Snapshot Image and the UML diagram in Figure 4-1;
section 5.11.6.6 Sub-Surface Moisture OID;
section 5.11.6.7 Sub-Surface Sensor Error OID; and
section 5.16.3.4 Snapshot Camera Command ACCESS field.
The approved errata are included in v02.24. See Annex D for the Documentation of Revisions.
In May 2008, the standard was edited and formatted for publication.

INTRODUCTION

NTCIP 1204 v02 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 the defined NTCIP format defined in NTCIP 8004. This data would typically be exchanged between a management station and a field device using one of the NTCIP 1103 recognized Application Layers (e.g., SNMP). The data may also be exchanged among management stations using other protocols.

This standard defines requirements that are applicable to all NTCIP environments and it also contains optional and conditional clauses that are applicable to specific environments for which they are intended.

The following keywords apply to this document: AASHTO, ITE, NEMA, NTCIP, ESS, data, data dictionary, object.

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 in order to develop the object definitions 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 this version 02.

< This page is intentionally left blank. >

CONTENTS

< This page is intentionally left blank. >

Section 1
GENERAL
[Informative]

1.1 SCOPE

Environmental sensors include a wide array of sensors, including those which 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 the purpose of this standard, all three types of stations are called ESS. In the transportation community, these devices are frequently used in order 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 the purpose of this standard, the term ESS may also include a PTS.

NOTE: The PTS portion of this standard 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 costs. This additional cost inhibits information sharing within and between various potential users of the data and prevents vendor independence. Without vendor independence, costs further increase due to 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, the NTCIP is developing a family of open standards for communications between field devices and central management systems.

This standard is a 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 this standard to allow for new functions as they may be needed. It is expected that this will support the deployment of ESS from one or more vendors in a consistent and cost-efficient way.

This standard only addresses a subset of the requirements required 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.

This standard standardizes the communications interface by identifying the various operational needs of the users (Section 2) and subsequently identifying the requirements (Section 3) that must be supported for each need. The standard 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 must be supported for each requirement.

The 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 this standard are provided in section 3.3.

An implementation of this standard requires lower level services to structure, encode, and exchange the data concepts defined by this standard. This standard assumes that the data concepts will be exchanged by one of the protocols defined in NTCIP 2301.

1.2 REFERENCES

For approved amendments, contact:

NTCIP Coordinator
National Electrical Manufacturers Association
1300 North 17th Street, Suite 1752
Rosslyn, VA 22209-3801
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 at www.ntcip.org.

1.2.1 NORMATIVE REFERENCES

The following standards contain provisions, which, through references in this text, constitute provisions of this Standard. By reference herein, these standards are adopted, in whole or in part as indicated, in this publication. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this Standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Identifier	Title	Summary
Glossary of Meteorology	*Glossary of Meteorology, Second Edition; American Meteorological Society, 2000.*	This document defines common meteorological terms.
AASHTO / ITE / NEMA NTCIP 1201:2005 v02.32	*NTCIP Global Object Definitions – version 02*	NTCIP 1201 v02 defines data elements that are used by multiple types of devices (e.g., signs, sensor stations, signals, etc.). Many of these objects, such as time and scheduling objects, are referenced by this standard in order to fulfill user needs.
AASHTO / ITE / NEMA NTCIP 2301:2001 v01.08	*NTCIP – Simple Transportation Management Framework (STMF) Application Profile*	The STMF Application Profile as restricted by this standard, defines the mechanisms by which the data defined in this standard is exchanged.

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.*	BUFR is a major standard of the meteorological community that provides standardized definitions of various pieces of data as well as encoding forms. The NTCIP ESS standard uses common data definitions where appropriate; however, the NTCIP uses different encoding (as defined in NTCIP 2301) and customizes many terms in order to reflect the unique needs of the transportation community.

1.2.2 OTHER REFERENCES

The following documents and standards may provide the reader with a more complete understanding of the entire protocol and the relations between all parts of the protocol. However, these documents do not contain direct provisions that are required by this standard. 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 below.

Identifier	Title	Summary
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	This standard defines the generic format for defining an SNMP Management Information Base (MIB). The NTCIP standards have further refined this in NTCIP 8004.
National ITS Architecture, Version 5.0	*National ITS Architecture, FHWA, 2003*	The National ITS Architecture is used to define the how the contents of this standard relate to the overall scope of ITS.
AASHTO / ITE / NEMA NTCIP 1103 v01.26 (Recommended Std, June 2005)	*Transportation Management Protocols*	This standard defines the protocols used to exchange information between management systems and most field devices. It is a normative reference of NTCIP 2301, which is a normative reference to this standard.
AASHTO / ITE / NEMA NTCIP 2201:2003 v01.15	*Transportation Transport Profile (T2)*	This standard defines a specialized, low-bandwidth transport mechanism for exchanging data.
AASHTO / ITE / NEMA NTCIP 2202:2001 v01.05	*Transport Profile for Internet (TCP/IP and UDP/IP)*	This standard defines how the NTCIP community uses the TCP/IP and UDP/IP standards.

AASHTO / ITE / NEMA NTCIP 8004 v01.38 Jointly Approved December 2005	*Structure and Identification of Management Information*	This standard defines and explains the structure used by the NTCIP effort to document data definitions.
OMG Unified Modeling Language Specification, Version 1.5	OMG Unified Modeling Language Specification, Object Management Group, 2003	The standard that formally defines the meaning of each symbol used in the UML diagrams contained in this standard. However, as these diagrams are informational, this standard is not normative.
Publication No. FHWA-SA-98-008	*Assessing the Results of the Strategic Highway Research Program, FHWA, 1998 http://www.fhwa.dot.gov/winter/ roadsvr/sumrept.pdf*	This document describes how ESS can be used within transportation management systems in order to improve operations.
Weather-Responsive Traffic Management	*Weather-Responsive Traffic Management – Concept of Operations. FHWA, 2003.* http://ops.fhwa.dot.gov/weather/ *best_practices/WeatherConOps0103.pdf*	An overview of how weather data can be used in traffic management applications.

1.2.3 CONTACT INFORMATION

1.2.3.1 American Meteorological Society and World Meteorological Organization 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

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 http://itsarch.iteris.com/itsarch/

1.2.3.4 NTCIP Standards

Copies of NTCIP standards may be obtained from:

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

1.2.3.5 Object Management Group Documents

Copies of OMG standards may be obtained electronically from:

Object Management Group
www.omg.org

1.2.3.6 OFCM Documents

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

1.3 GENERAL STATEMENTS

< In the opinion of the ESS WG, this clause does not apply in the context of NTCIP 1204 v02. >

1.4 TERMS

For the purposes of this standard, 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 in this section are used in accordance with their definitions in IEEE Std 100-2000. English words not defined in this section 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 (IEEE Std. 610.12-1990: IEEE Standard Glossary of Software Engineering Terminology).
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	[To be revised in 1204 v03 to refer to NTCIP 8004 v02] The 'deprecated' value in the STATUS field of an 'OBJECT-TYPE' macro (see Section 5) indicates that the subject object was included in a previous version of the standard but no longer represents the preferred design.
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	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.
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. (National Telecommunications and Information Administration, U.S. Department of Commerce)
Interoperable	[To be revised in 1204 v03 to refer to NTCIP 8004 v02] The ability of two or more systems or components to exchange information and use the information that has been exchanged (IEEE Std. 610.12-1990: IEEE Standard Glossary of Software Engineering Terminology).
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.
National Transportation Communications for ITS Protocol	The NTCIP is a family of protocols that provide common control and data collection services as well as accommodating various system topologies and data routing duties. The NTCIP will support not only currently deployed systems, but new systems and technologies as they become available.

Obsolete	[To be revised in 1204 v03 to refer to NTCIP 8004 v02] The 'obsolete' value in the STATUS field of an 'OBJECT-TYPE' macro (see Section 5) indicates that the object was included in a previous version of the standard but is no longer in significant use within the industry.
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 must be accepted and used to understand each other.
Remote Processor Unit	A field processor which 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 must conform; either derived directly from user needs, or stated in a contract, standard, specification, or other formally imposed 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' shall be understood that the data is sent to the requester.
Road Weather Data Collection Market Package	A set of components which 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 which is capable of detecting a condition and reporting the result to an RPU.
Simple Network Management Protocol	A communications protocol developed by the IETF, used for configuration and monitoring of network devices.
Simple Transportation Management Framework	Describes the organization of the information within devices and the methods of retrieving or modifying any information within the device. STMF also explains how to generate and utilize computer readable information organization descriptions.

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 who will use the system that is developed.
User Need	The business or operational problem (opportunity) that must be fulfilled in order to justify purchase or use. While this is termed a 'user need' within the NTCIP community, it reflects needs of all stakeholders.

1.5 ABBREVIATIONS AND ACRONYMS

The abbreviations and acronyms used in NTCIP 1204 v02 are defined as follows:

IAB STD	Internet Architecture Board Standard
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISO	International Organization for Standardization
MIB	Management Information Base
NTCIP	National Transportation Communications for Intelligent Transportation Systems (ITS) Protocol
OID	OBJECT IDENTIFIER
PRL	Profile Requirements List
RFC	Request for Comments
SNMP	Simple Network Management Protocol
STMF	Simple Transportation Management Framework
T2	Transportation Transport Profile
TCP	Transmission Control Protocol
TMC	Traffic Management Center
TMP	Transportation Management Protocol
UDP	User Datagram Protocol
WG	Working Group
WMO	World Meteorological Organization

Section 2
CONCEPT OF OPERATIONS
[Normative]

This section defines the user needs that subsequent sections within this standard will 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 will be used. In the case of this standard, this entails identifying the various ways in which transportation operations personnel may use ESS information in order 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 the standard.

This section is intended for all readers of the document, including:

a. Transportation operations managers
b. Transportation operations personnel
c. Transportation engineers
d. System integrators
e. Device manufacturers

The first three categories of readers will find this section useful in order to understand how ESS equipment can be used in their system. For this audience, this section serves as the starting point in the procurement process. They will be able to become familiar with each feature covered by the standard and determine whether that feature is appropriate for their implementation. If it is, then their procurement specification will need to require support for the feature and all of the mandatory requirements related to that feature.

The last two categories of readers will find this section useful in order to gain a more thorough understanding as to why the more detailed requirements (as specified in later sections of this standard) 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 the system developers understand the users' needs. Within the context of NTCIP standards it is used to document the intent of each feature for which the standard 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 the standard and to the development of the standard 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 the:

a. Reference physical architecture – The reference physical architecture defines the overall context of the proposed system and defines which specific interface is addressed by this standard. 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 this standard.
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 form the basis of the traceability tables contained in Section 3 and Annex A.

The architectural needs and features are collectively called the *user needs*. Section 3 uses these user needs in the analysis of the system in order to define the various functional requirements of an ESS. Each user need must be traced to one or more functional requirements and each functional requirement must 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 the standard 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. In order to obtain a device that meets specific needs, the user will need to first identify 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 the standardized dialogs and data element definitions provided in Sections 4 and 5.

A conformant device may support other user needs, as long as they are conformant with the requirements of this standard and the standards it references (i.e., NTCIP 2301 and NTCIP 8004). For example, a device may support data that has not been defined by this standard; however, when exchanged via one of the NTCIP 2301 protocols, the data must 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 will inhibit the goals of interoperability and interchangeability, especially if the documentation of these user needs is not available for distribution to system integrators. The standards allow implementations to support additional user needs in order 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 standard and by providing a description of how this standard 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:[1]

a. Sharing the 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

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 due to precipitation, fog, high winds, snow, ice and/or flooding. When travel is becoming hazardous due to 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 due to 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 is 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 will 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 due to 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 in order to determine the accuracy of predictions. Finally, some research has suggested that air-quality hot-spots could be monitored in order 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

[1] Additional information about how this data can be used is provided in *Weather-Responsive Traffic Management – Concept of Operations.*

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]

This standard addresses the communications interface between a management station and a controller. The relationship between these and other logical components is depicted in Figure 2-1. However, one should realize that 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.

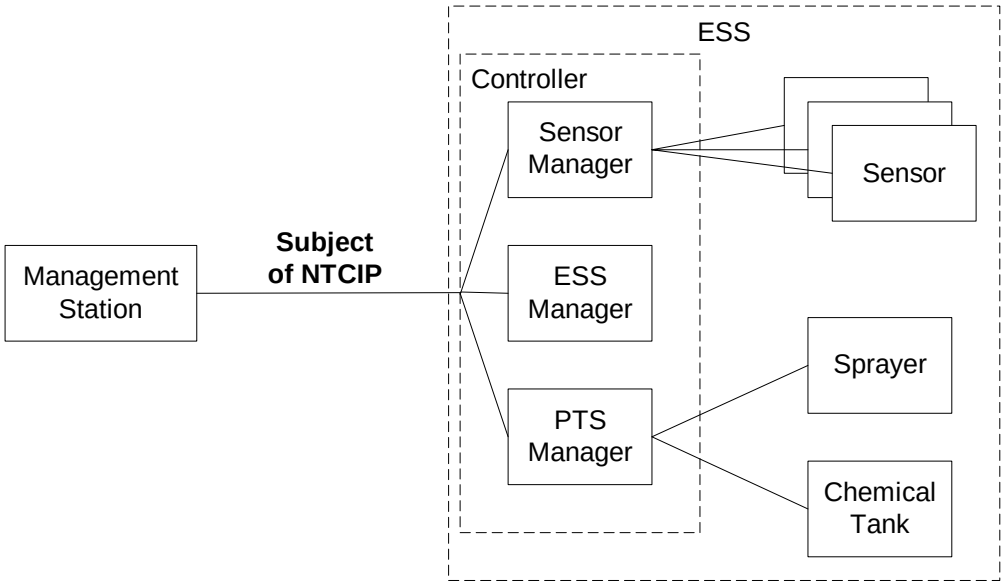

Figure 2-1
Reference Architecture

The major components of the system are as follows:

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 below.

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. (*Merriam-Webster On-Line Dictionary*, 2004)

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.

The following subsections describe sample physical architectures that are supported by this reference architecture.

2.3.1 Remote Weather Station

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

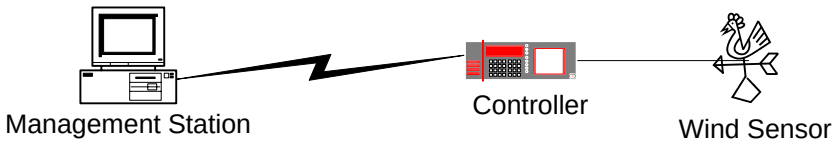

Figure 2-2
Remote Weather Station

2.3.2 Sprayer Combined with a Pavement Sensor

Figure 2-3 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.

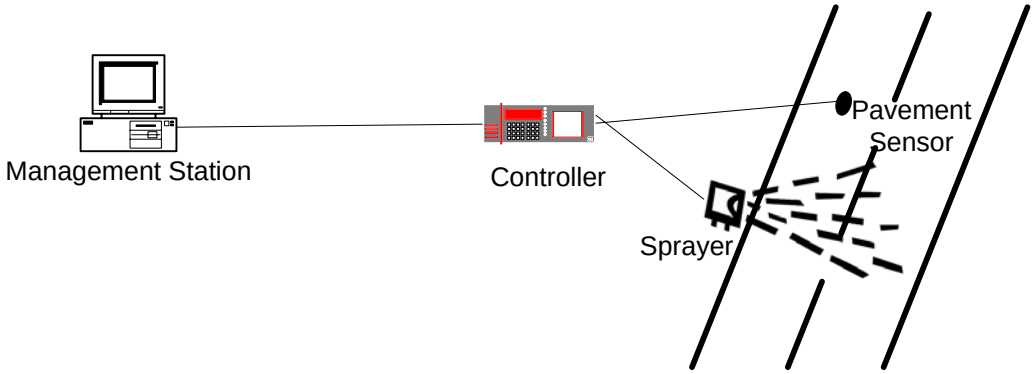

Figure 2-3
Sprayer Combined with a Pavement Sensor

2.3.3 Pavement Treatment Station

Figure 2-4 depicts an ESS that only consists of sprayers for pavement treatment.

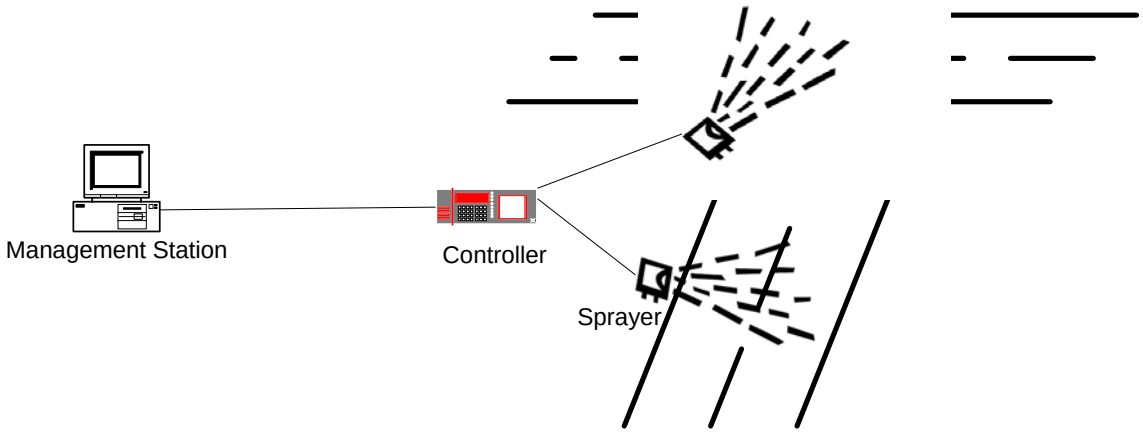

Figure 2-4
Pavement Treatment Station

2.4 ARCHITECTURAL NEEDS

The following subsections define the communications environment within which an ESS is expected to operate.

2.4.1 Generic Architectural Needs

The features defined in Section F.1.1 of Annex F 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 will 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 this separate document.

2.5 FEATURES

The following subsections identify and describe the various features that may be offered by the ESS. It is divided into the following major subsections:

a. ESS Manager Features
b. Sensor Manager Features

c. Pavement Treatment System Manager Features

2.5.1 ESS Manager Features

The following subsections identify and describe 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 Section F.1.2 of Annex F 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 will 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 this separate document.

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

The following subsections identify and describe the various features that may be offered by the Sensor Manager. It consists of the following features:

a. Monitor weather conditions
b. Monitor pavement
c. Monitor subsurface conditions
d. Monitor human readings
e. Monitor water levels
f. Monitor air quality and bio-hazards
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 contains 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's location.

2.5.2.1.4 Monitor Humidity

A transportation system operator may need to monitor the humidity at the ESS's 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 will need to configure and monitor the parameters defining the current treatments being applied in order 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 s/he 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 Bio-hazards

A transportation system operator may need to monitor the current air quality in the vicinity of the ESS and determine whether there are airborne bio-hazards 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 subsections 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 SECURITY

This standard 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.

2.7 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 above, are related to the National ITS Architecture Flows in the following manner:

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]

This section defines the Functional Requirements based on the user needs identified in the Concept of Operations (see Section 2). This section provides the reader with:

a. A tutorial
b. Architectural Requirements – These are requirements related to the architectural needs defined in Section 2.4.
c. Data Exchange Requirements – These are requirements related 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 the Concept of Operations that do not fall into one of the above two categories. 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.

This section is intended for all readers of the document, including:

a. Transportation operations managers
b. Transportation operations personnel
c. Transportation engineers
d. System integrators
e. Device manufacturers

The first three categories of readers will find this section useful in order to understand the details of what the standard requires of an ESS. This audience will find Section 3.3 to be particularly useful in preparing procurement specifications and will be able to map the various rows of this table to the more detailed text contained within the other sections.

The last two categories of readers will find this section useful in order to fully understand what is required of equipment meeting this interface standard. They will also be able to use the table in Section 3.3 to document the capabilities of their implementations.

3.1 TUTORIAL [INFORMATIVE]

The Functional Requirements Section defines the formal 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 defined in this section. 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 the Concept of Operations.

b. Data Exchange Requirements – These requirements define the required behavior of the system in exchanging data across the communications interface based upon the features identified in the Concept of Operations.

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 (e.g., how many ____ are supported by the device).

3.2 SCOPE OF THE INTERFACE [INFORMATIVE]

<In the opinion of the responsible NTCIP working group, this section does not apply in the context of this standard publication.>

3.3 PROTOCOL REQUIREMENTS LIST

The PRL, provided in 3.3.7 and 3.3.8, maps the user needs defined in Section 2 to the requirements defined in Section 3. The table 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 within Section 2 and the PRL is based upon the user need sections within that Section. The section identifier and section name are indicated within these columns.

3.3.2 Requirements Column

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

3.3.3 Conformance Column

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

3.3.3.1 Status Symbols

The following symbols are used to indicate status:

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 scope of the standard)
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 must 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 following predicate notations may be used:

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

The <predicate>: notation means that the status following it applies only when the PRL 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 or subsection. 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 this standard map to the following clauses:

PREDICATE	CLAUSE
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 must be 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 support column can be used by 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 support column:

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 product 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 in order 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 requirements’ column, each response shall be selected either from the indicated set of responses (for example: Yes / No / NA), or it shall reference additional items that are to be attached.

If a conditional requirement is inapplicable, use the Not Applicable (NA) choice. If a mandatory requirement is not satisfied, exception information must be supplied by entering a reference Xi, where i is a unique identifier, to an accompanying rationale for the non-conformance. When the status is expressed as a two-character combination (as defined in 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: A specification can allow for flexibility in a deliverable by leaving the selection in the Project Requirement column blank for a given row. For example, a 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	
		3.5.1.3.2	Retrieve Mobile Treatment Information	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	
		3.6.1	Required Number of Atmospheric Pressure	M	Yes / NA	The ESS shall support at least ____ atmospheric pressure sensors.

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
			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	
		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	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		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).
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	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		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	
		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	M	Yes / NA	The ESS shall support at least ____

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
			Snapshot Cameras			snapshot cameras (1..255).
2.5.2.3 (Subsurface)	Monitor Subsurface Conditions			O.2 (1..*)	Yes / No / NA	
		3.5.2.1.7	Configure Sub-Surface Sensor	Subsurface:M	Yes / NA	
		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	
2.5.2.6 (Air)	Monitor Air Quality and Bio-hazards			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	Retrieve Nitrous Oxide	O.4 (1..*)	Yes / No /	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		(N2O)	Reading		NA	
		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	
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	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
		F.2.1.1.2	Deliver Data	M	Yes	
		F.2.1.1.3	Explore Data	M	Yes	
		3.6.21	Response Time for Requests	M	Yes	The 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 Meta-Data	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	Response Time for Requests	M	Yes	The 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	Response Time for Requests	M	Yes	The 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 Sub-Surface 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	
		F.2.2.2.1	Monitor Status of External	M	Yes / NA	

User Need ID	User Need	FR ID	Functional Requirement	Conformance	Project Requirement	Additional Project Requirements
			Device			
		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 for this standard. 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 are provided in the following subsections.

3.5.1 ESS Manager Requirements

Requirements for managing an ESS Manager are provided in the following subsections.

3.5.1.1 ESS Configuration Requirements

Requirements for configuring an ESS Manager are provided in the following subsections.

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 Meta-Data

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 are provided in the following subsections.

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.

3.5.1.3 ESS Data Retrieval Requirements

Requirements for retrieving data from an ESS Manager are provided in the following subsections.

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

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

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 are provided in the following subsections.

3.5.2.1 Sensor Configuration Requirements

Requirements for configuring a Sensor Manager are provided in the following subsections.

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 Meta-Data 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 Meta-Data

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 Meta-Data

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 Sub-Surface Sensor Meta-Data

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 Sub-Surface Sensor

Upon request, the ESS shall store configuration information for a specified sub-surface 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 the storage location for newly taken snapshot images.

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 are provided in the following subsections.

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 are provided in the following subsections.

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 are provided in the following subsections.

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 by Sections 3.5.2.3.2.1 through 3.5.2.3.2.8, that is supported by the device.

3.5.2.3.3 Monitor Pavement Condition

Requirements for monitoring pavement conditions are provided in the following subsections.

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 (i.e., 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 by 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 are provided in the following subsections.

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 by sections 3.5.2.3.4.1 through 3.5.2.3.4.2, supported by the device.

3.5.2.3.5 Monitor Situation Assessments

Requirements for monitoring situation assessments are provided in the following subsections.

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 Bio-Hazard Conditions

Requirements for monitoring air quality and bio-hazard conditions are provided in the following subsections.

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 being stored.

3.5.2.4 Sensor Control Requirements

Requirements for controlling a Sensor Manager are provided in the following subsections.

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

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

3.5.3 PTS Manager Requirements

Requirements for managing a PTS Manager are provided in the following subsections.

3.5.3.1 PTS Configuration Requirements

Requirements for configuring a PTS Manager are provided in the following subsections.

3.5.3.1.1 Retrieve Stationary Pavement Treatment Configuration

Upon request, the PTS shall return:

a. The sensors that the PTS will monitor to determine when to trigger the sprayers,
b. The spray duration, and
c. The mix of chemicals to use when spaying.

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 will monitor to determine when to trigger the sprayers,
b. The spray duration, and
c. The mix of chemicals to use when spaying.

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 spaying.

3.5.3.2 PTS Status Monitoring Requirements

Requirements for monitoring the status of a PTS Manager are provided in the following subsections.

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 occurred.

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 are provided in the following subsections.

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 for the configured duration period.

3.6 SUPPLEMENTAL REQUIREMENTS

Supplemental requirements for ESS are provided in the following subsections. 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 must fulfill the stated requirement in order to claim conformance to this standard.

3.6.1 Required Number of Atmospheric Pressure Sensors

The communication 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 specification. If the 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 specification. If the 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 specification. If the 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 communication 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 specification. If the specification does not define the number of humidity sensors, the ESS shall support at least one humidity sensor.**Required Number of Precipitation Sensors**

The communication 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 specification. If the 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. In order to overcome problems that may result from this variation, the standard 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 specification. If the 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 communication 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 specification. If the 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 specification. If the specification does not define the number of snapshot cameras, the ESS shall support at least one snapshot camera.

3.6.21 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 and NTCIP 2303), 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 Response Time. If the specification does not indicate the Response Time, the Response Time shall be 100 milliseconds. The 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.

Section 4
DIALOGS
[Normative]

This section 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 must 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 in order to conform to this standard.

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 will 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). In order 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.

In order to overcome this complication, this section 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 this standard. Such a management station is termed a ‘consistent management station’. A consistent management station will interoperate with any ‘conformant’ device. However, since an agency can not 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 will only use the standardized dialogs as defined in this section. 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 will provide an agency with a much greater chance of achieving interoperability with off-the-shelf devices that have been tested against this standard and the designation of such a system is intended to provide a guaranteed base level of interoperability.

The rules for the standardized dialogs are as follows:

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 Section F.3.1 and Section F.3.3 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. Formal definitions of each object type are provided in Section 5 of this standard and NTCIP 1201. 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 this standard in order to provide for 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 this standard.
f. After sending a message, the management station shall not transmit any other data across the communications channel until the earlier of:

   fa. The management station receiving a response from the device or
   fb. The expiration of the 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 write-able) 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) presented 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 the appropriate response per the generic dialog specification.

This section 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 as follows:

a. The management station shall SET essSnapshotCameraCommand.x to the a 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 essSnapshotStoragePath and to the file essSnapshotFilename.
d. The management station shall repeatedly GET essSnapshotCameraCommand.x until it equals ready (1). [revised in 1204 v02.24 per Errata]
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 4-1.

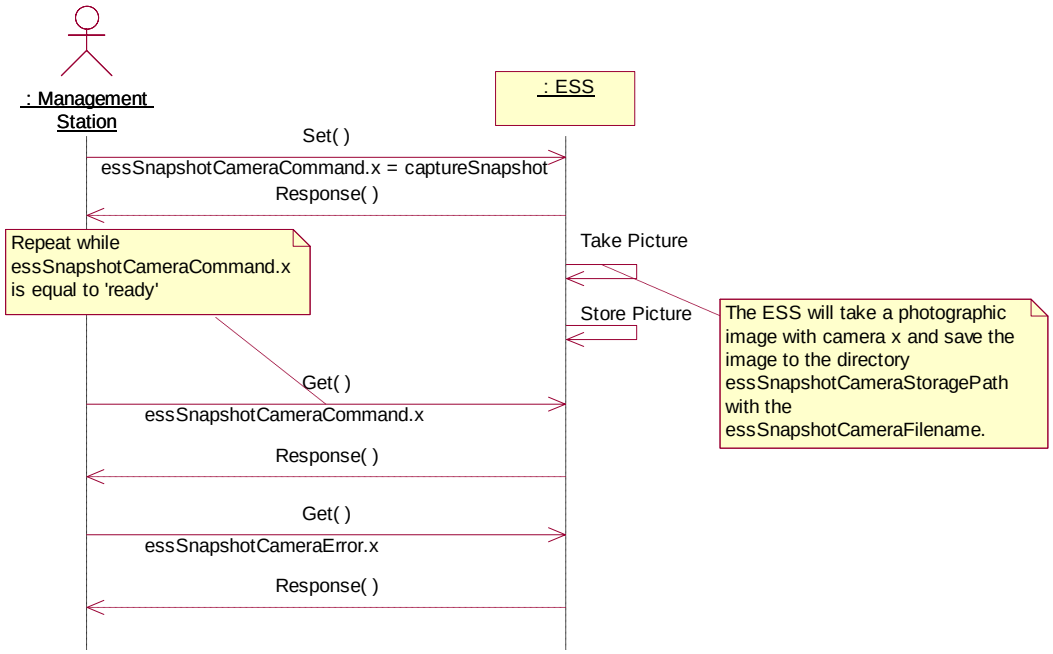

Figure 4-1
Dialog for Capture Snapshot Image
[revised in 1204 v02.24 per Errata]

4.2.2 Retrieve Snapshot

The standardized dialog for a management station to retrieve a snapshot image shall conform to NTCIP 2303 (File Transfer Protocol).

The following are rules that the device shall adhere to:

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 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 (File Transfer Protocol) and to the rules defined in Section 4.2.

4.2.4 Copy Snapshot

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 Section 4.2.

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 will be aware from which sensor 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
    7. essSurfaceIceOrWaterDepth.x
    8. essSurfaceConductivityV2.x
    9. pavementSensorTemperatureDepth.x

c. The management station shall GET numEssTreatments.0.
d. 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

4.3 STATE TRANSITION DIAGRAMS

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

4.3.1 Pavement Treatment System State Transition Diagram

Figure 4-2 depicts the state transition diagram for the Pavement Treatment System class.

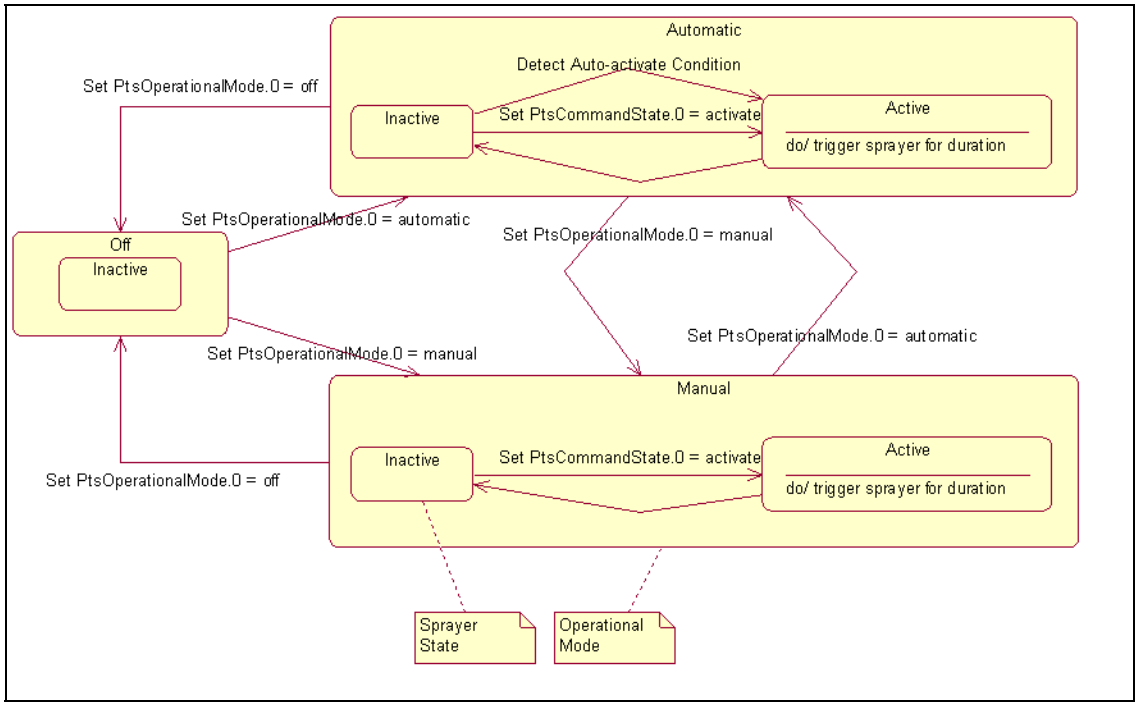

Figure 4-2
Pavement Treatment System State Machine Diagram

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 ptsCommandState 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 ptsCommandState 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 pysSignalDuration object. Upon expiration of this duration, the PTS shall automatically transitoin back to the "inactive" state.**CLASS DIAGRAMS**

The relationships between data elements are described through the use of UML class diagrams. Figure

4-3 provides a sample class diagram.

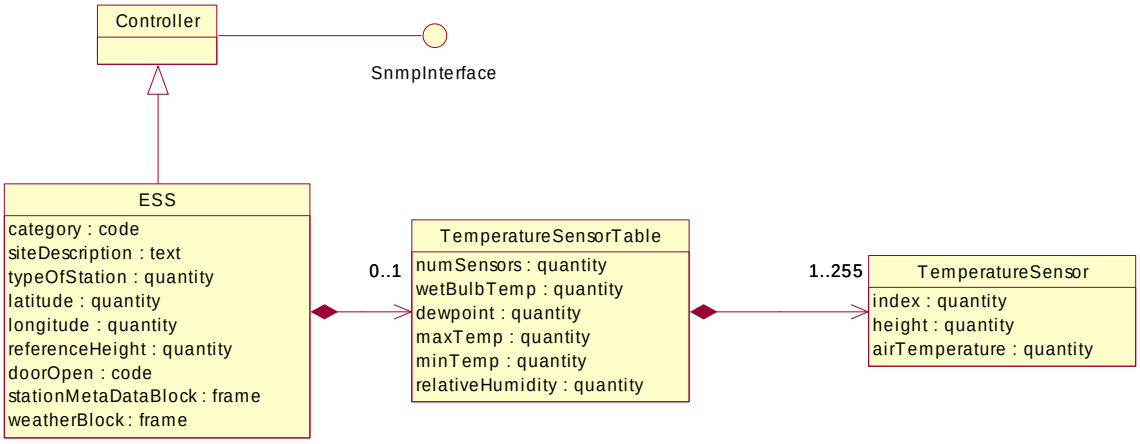

Figure 4-3
Sample Class Diagram for Temperature Sensors

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 this standard, there are two interfaces, the SNMP Interface (as shown in the sample diagram), and the FTP Interface (not shown here, but shown later in this section).

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

Each piece of data referenced in this section 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 formally 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 4-4. 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.

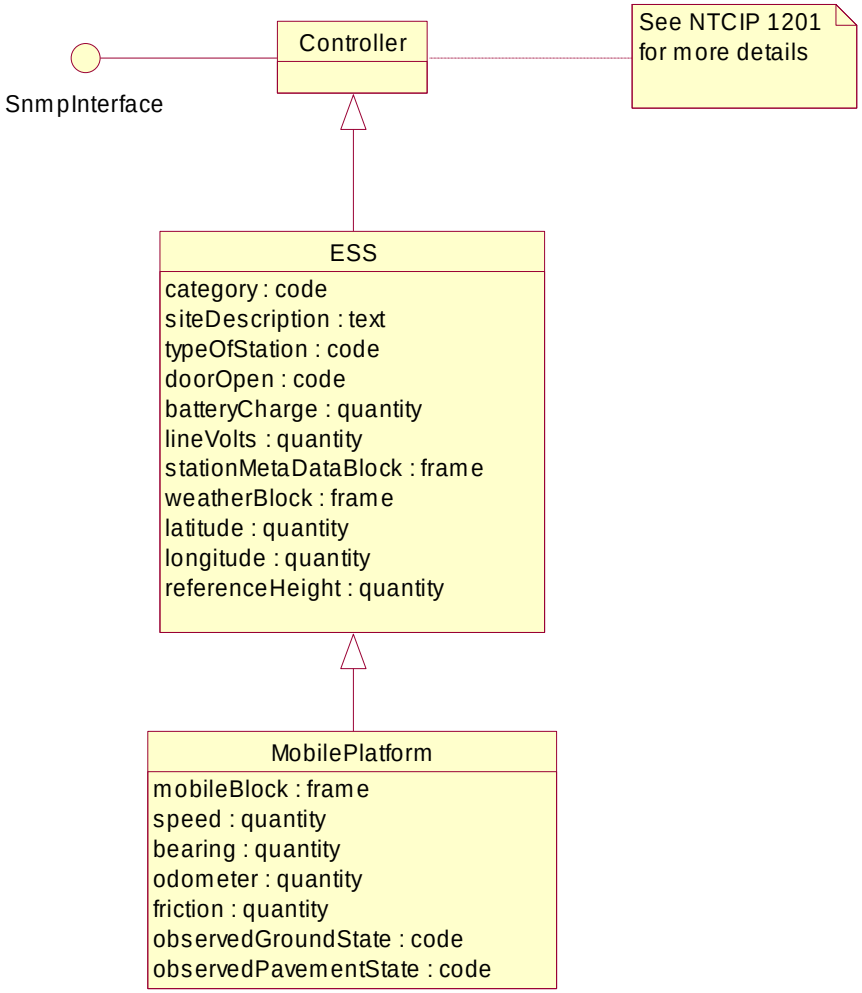

Figure 4-4
ESS Characteristics Class Diagram

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 below.

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 below.

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 specification. This information is depicted in Figure 4-5.

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

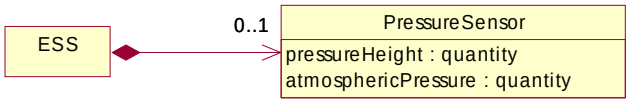

Figure 4-5
Pressure Sensor Class Diagram

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 below.

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 specification. The wind sensor table shall be associated with the number of wind sensors as defined in the specification.The information supported by the wind sensor is depicted in Figure 4-6.

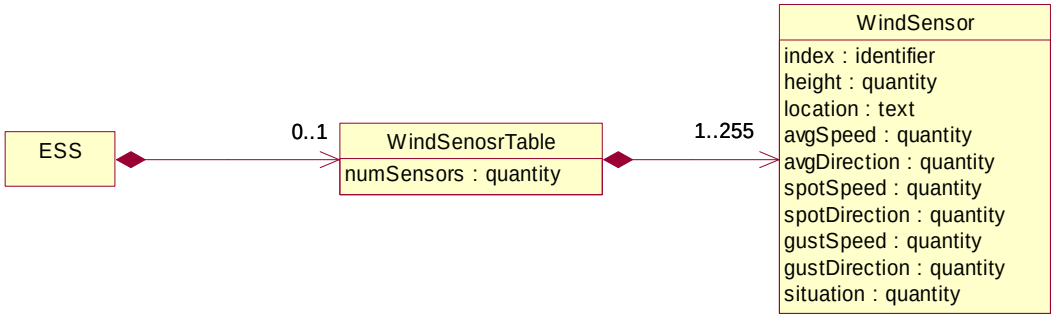

Figure 4-6
Wind Sensor Class Diagram

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 below.

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 below.

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 specification. The temperature sensor table shall be associated with the number of temperature sensors as defined in the specification. The information supported by the temperature sensor is depicted in Figure 4-7.

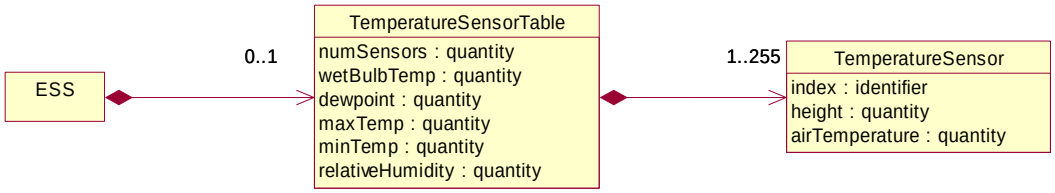

Figure 4-7
Temperature Sensor Class Diagram

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 below.

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 below.

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 specification. The ESS shall support one water level sensor table if required by the specification. The water level sensor table shall be associated with the number of water level sensors as defined in the specification. The information supported by these sensors are depicted in Figure 4-8.

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

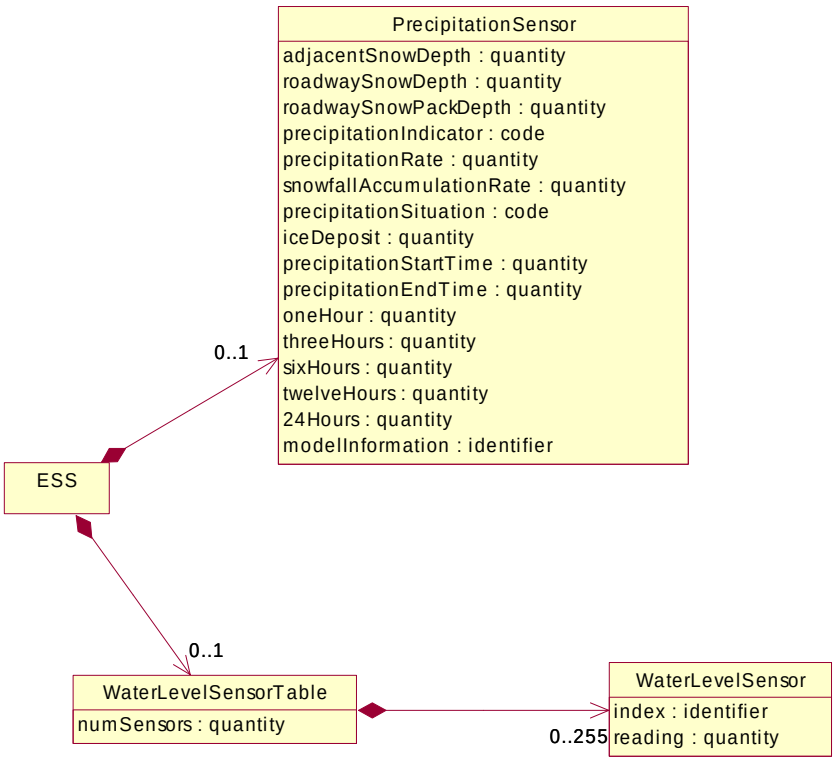

Figure 4-8
Precipitation Sensor Class Diagram

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 below.

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 below.

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 below.

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 specification. The information supported by the radiation sensor is depicted in Figure 4-9.

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

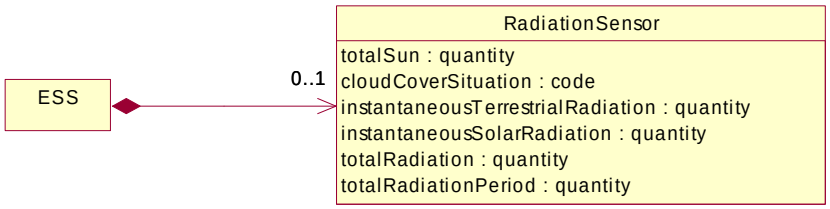

Figure 4-9
Radiation Sensor Class Diagram

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 by the following subsections.

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 specification. The information supported by the visibility sensor is depicted in Figure 4-10.

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

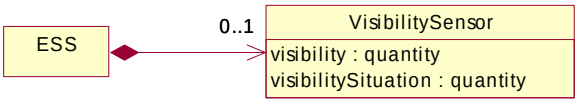

Figure 4-10
Visibility Sensor Class Diagram

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 by the following subsections.

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 specification. The pavement sensor table shall be associated with the number of pavement sensors as defined in the specification. The information supported by these sensors are depicted in Figure 4-11.

If the ESS uses a passive pavement sensor to predict the temperature at which ice will form, 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 specification. See Section 4.4.12 for more information.

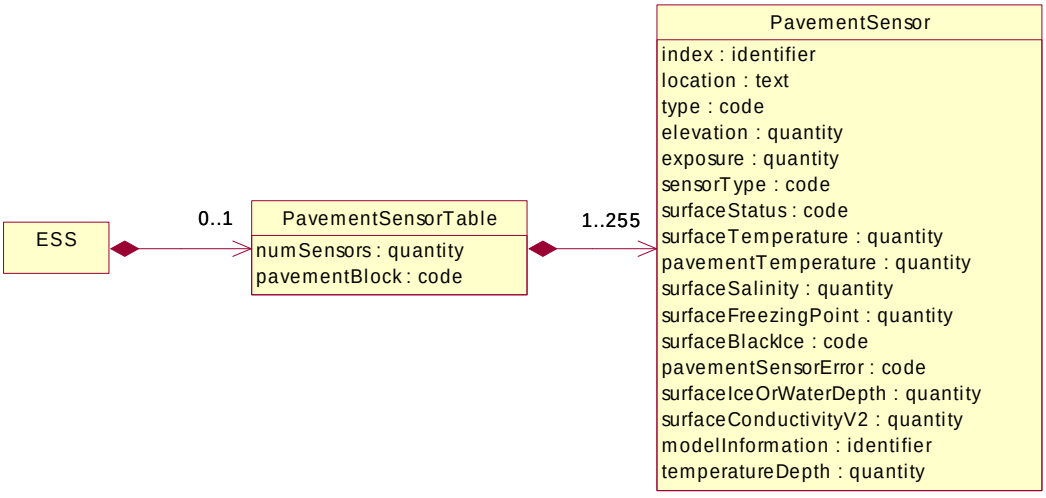

Figure 4-11
Pavement Sensor Class Diagram

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 below.

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.surfaceIceOrWaterDepth: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	pavementSensorTemperatureDepth

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 below.

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 sub-surface sensor table if required by the specification. The sub-surface sensor table shall be associated with the number of sub-surface sensors as defined in the specification. The information supported by these sensors are depicted in Figure 4-12.

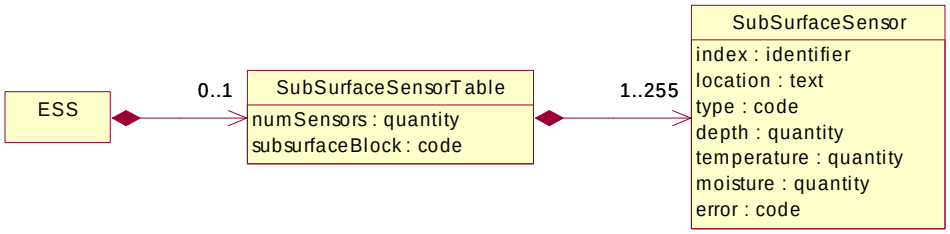

Figure 4-12
Subsurface Sensor Class Diagram

4.4.9.2 Subsurface Sensor Table

The sub-surface sensor table provides summary information related to the sub-surface sensors supported by the ESS. It can be described through a number of attributes as indicated below.

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 sub-surface sensor provides information related to the state of the pavement sub-surface. It can be described through a number of attributes as indicated below.

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 specification. The information supported by this sensor is depicted in Figure 4-13.

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

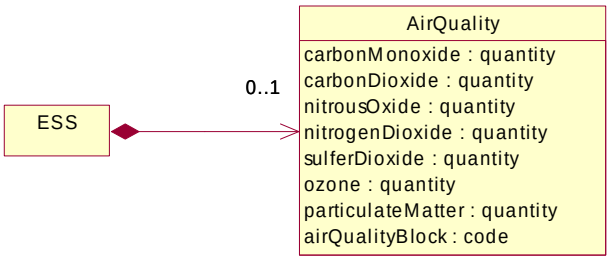

Figure 4-13
Air Quality Sensor Class Diagram

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 by the following subsections.

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 specification. The snapshot camera table shall be associated with the number of snapshot cameras as defined in the specification. The information supported by these entities are depicted in Figure 4-14.

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

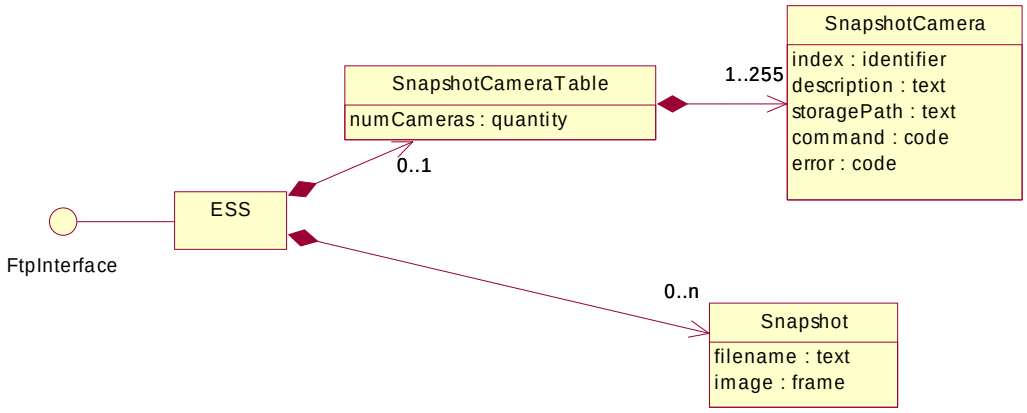

Figure 4-14
Snapshot Class Diagram

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 below.

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

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 below.

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 below.

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 specification. The information supported by the PTS is depicted in Figure 4-15.

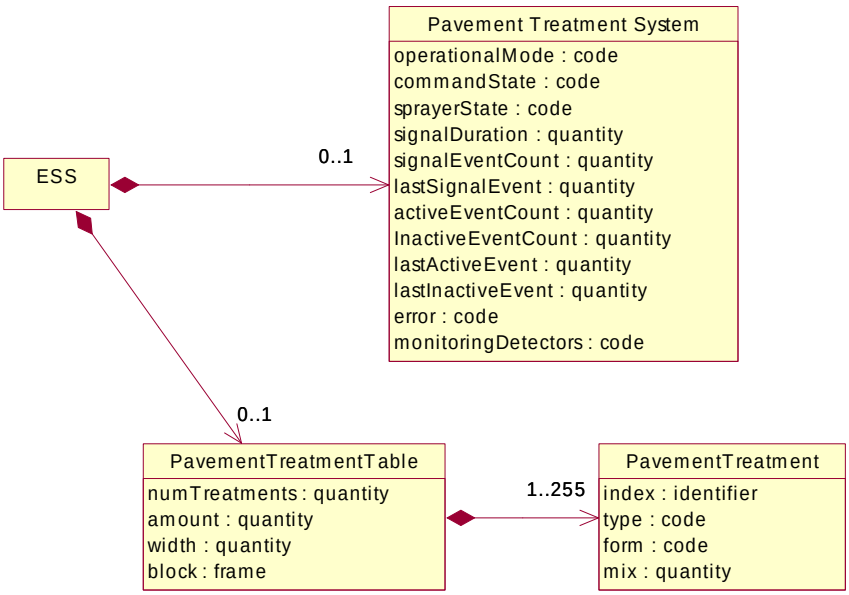

Figure 4-15
Pavement Treatment Class Diagram

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 below.

Descriptive Name	Section	Object Name
PTS.operationalMode.code	5.13.7	ptsOperationalMode
PTS.commandState:code	5.13.8	ptsCommandState
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 below.

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 in order inhibit ice formation or promote ice melting. It can be described through a number of attributes as indicated below.

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]

This section defines those objects which 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. The text provided from Section 5.0 through the end of Section 5.16 (except the section headings) constitutes the standard NTCIP1204-v02 MIB.

All of the objects defined in this document 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. In order 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 this standard defined data elements that have been replaced in order 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 WG decided to deprecate the various objects.

5.0 MIB COMMENT HEADER

```
--*****************************************************************
-- Filename:    1204-v02.MIB
-- Description: This MIB defines the Environmental Sensor Station
--              Objects
--*****************************************************************
```

5.1 MIB HEADER

```
NTCIP1204-v02 DEFINITIONS ::= BEGIN
IMPORTS
  Counter
  -- Deleted reference to IpAddress as it is not used
      FROM RFC1155-SMI
  DisplayString
      FROM RFC1213-MIB   -- Updated reference from RFC1158-SMI
  OBJECT-TYPE
      FROM RFC-1212
  ess, OerString
  -- Deleted unneeded references to experimental
  -- and devices and added references to ess and OerString
      FROM NTCIP8004-A-2004;
```



```
-- 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>read-only
<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.
<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
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

```
essStationMetaDataBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssStationMetaData
structure as defined in Section 4.  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-v02
essTypeOfStation.0,                 OPTIONAL,   -- @NTCIP1204-v02
essLatitude.0,                      OPTIONAL,   -- @NTCIP1204-v02
essLongitude.0,               OPTIONAL,   -- @NTCIP1204-v02
essReferenceHeight.0,               OPTIONAL,   -- @NTCIP1204-v02
essPressureHeight.0,                OPTIONAL,   -- @NTCIP1204-v02
essWindSensorHeight.0,        OPTIONAL,   -- @NTCIP1204-v02
temperatureMetaData           SEQUENCE OF TemperatureMetaData     OPTIONAL,
pavementMetaData        SEQUENCE OF PavementMetaData  OPTIONAL,
subSurfaceMetaData            SEQUENCE OF SubSurfaceMetaData      OPTIONAL,
treatmentMetaData       SEQUENCE OF TreatmentMetaData OPTIONAL
}

TemperatureMetaData ::= SEQUENCE {
essTemperatureSensorIndex.0         OPTIONAL,   -- @NTCIP1204-v02
essTemperatureSensorHeight.0        OPTIONAL    -- @NTCIP1204-v02
}

PavementMetaData ::= SEQUENCE {
essPavementSensorIndex.0,           OPTIONAL,   -- @NTCIP1204-v02
essPavementType.0,                  OPTIONAL,   -- @NTCIP1204-v02
essPavementElevation.0        OPTIONAL,   -- @NTCIP1204-v02
essPavementExposure.0         OPTIONAL,   -- @NTCIP1204-v02
essPavementSensorType.0       OPTIONAL    -- @NTCIP1204-v02
}
```



```
SubSurfaceMetaData ::= SEQUENCE {
essSubSurfaceSensorIndex.0          OPTIONAL,   -- @NTCIP1204-v02
essSubSurfaceType.0                 OPTIONAL,   -- @NTCIP1204-v02
essSubSurfaceSensorDepth.0          OPTIONAL    -- @NTCIP1204-v02
}

TreatmentMetaData ::= SEQUENCE {
      essPavementTreatmentIndex.0         OPTIONAL,   -- @NTCIP1204-v02
essPaveTreatProductType.0           OPTIONAL,   -- @NTCIP1204-v02
      essPaveTreatProductForm.0           OPTIONAL,   -- @NTCIP1204-v02
essPercentProductMix.0        OPTIONAL    -- @NTCIP1204-v02
}
<SetConstraint>read-only
<DescriptiveName>ESS.stationMetaDataBlock:frame
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 4 }
```

5.3.6 Weather Block

```
essWeatherBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
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-v02
essWindData                   EssWindData       OPTIONAL,
essTemperatureData                  EssTemperatureData      OPTIONAL,
essPrecipData                       EssPrecipData           OPTIONAL,
essVisibilityData             EssVisibilityData OPTIONAL
}

EssWindData ::= SEQUENCE {
essAvgWindDirection.0               OPTIONAL,   -- @NTCIP1204-v02
essAvgWindSpeed.0             OPTIONAL,   -- @NTCIP1204-v02
essWindSituation.0                  OPTIONAL,   -- @NTCIP1204-v02
essMaxWindGustSpeed     .0          OPTIONAL,   -- @NTCIP1204-v02
essMaxWindGustDir.0                 OPTIONAL,   -- @NTCIP1204-v02
essSpotWindDirection.0              OPTIONAL,   -- @NTCIP1204-v02
essSpotWindSpeed.0                  OPTIONAL    -- @NTCIP1204-v02
}

EssTemperatureData ::= SEQUENCE {
essWetBulbTemp.0              OPTIONAL,   -- @NTCIP1204-v02
essDewpointTemp.0             OPTIONAL,   -- @NTCIP1204-v02
essMaxTemp.0                        OPTIONAL,   -- @NTCIP1204-v02
essMinTemp.0                        OPTIONAL,   -- @NTCIP1204-v02
essRelativeHumidity.0               OPTIONAL,   -- @NTCIP1204-v02
      -- for (
```



```
--    x = 1;
--    x < essNumTemperatureSensors.0;
--    x++)
temperatureTable              SEQUENCE OF Temperature OPTIONAL
}

Temperature ::= SEQUENCE {
essTemperatureSensorIndex.x         OPTIONAL,   -- @NTCIP1204-v02
essAirTemperature.x                 OPTIONAL    -- @NTCIP1204-v02
}

EssPrecipData ::= SEQUENCE {
essWaterDepth.0               OPTIONAL,   -- @NTCIP1204-v02
essAdjacentSnowDepth.0        OPTIONAL,   -- @NTCIP1204-v02
essRoadwaySnowDepth.0         OPTIONAL,   -- @NTCIP1204-v02
essRoadwaySnowPackDepth.0           OPTIONAL,   -- @NTCIP1204-v02
essPrecipYesNo.0              OPTIONAL,   -- @NTCIP1204-v02
essPrecipRate.0               OPTIONAL,   -- @NTCIP1204-v02
essSnowfallAccumRate.0        OPTIONAL,   -- @NTCIP1204-v02
essPrecipSituation.0                OPTIONAL,   -- @NTCIP1204-v02
essIceThickness.0             OPTIONAL,   -- @NTCIP1204-v02
essPrecipitationStartTime.0         OPTIONAL,   -- @NTCIP1204-v02
essPrecipitationEndTime.0           OPTIONAL    -- @NTCIP1204-v02
}

Editor's Note - I think we decided to remove the following yes?
essPrecipitationOneHour.0           OPTIONAL,   -- @NTCIP1204-v02
essPrecipitationThreeHours.0        OPTIONAL,   -- @NTCIP1204-v02
essPrecipitationSixHours.0          OPTIONAL,   -- @NTCIP1204-v02
essPrecipitationTwelveHours.0       OPTIONAL,   -- @NTCIP1204-v02
essPrecipitation24Hours.0           OPTIONAL    -- @NTCIP1204-v02

EssVisibilityData ::= SEQUENCE {
essSolarRadiation.0                 OPTIONAL,   -- @NTCIP1204-v02
essTotalSun.0                       OPTIONAL,   -- @NTCIP1204-v02
essCloudSituation.0                 OPTIONAL,   -- @NTCIP1204-v02
essVisibility.0                     OPTIONAL,   -- @NTCIP1204-v02
essVisibilitySituation.0                  OPTIONAL    -- @NTCIP1204-v02
}
<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-v02
      essLongitude.0          OPTIONAL, -- @NTCIP1204-v02
      essReferenceHeight.0    OPTIONAL, -- @NTCIP1204-v02
      essVehicleSpeed.0       OPTIONAL, -- @NTCIP1204-v02
      essVehicleBearing.0     OPTIONAL, -- @NTCIP1204-v02
      essVehicleOdemeter.0    OPTIONAL, -- @NTCIP1204-v02
      essMobileFriction.0           OPTIONAL, -- @NTCIP1204-v02
      essMobileObservationGroundState.0   OPTIONAL, -- @NTCIP1204-v02
      essMobileObservationPavement.0      OPTIONAL, -- @NTCIP1204-v02
      essPaveTreatmentAmount.0            OPTIONAL, -- @NTCIP1204-v02
      essPaveTreatmentWidth.0       OPTIONAL -- @NTCIP1204-v02
      }
<SetConstraint>read-only
<DescriptiveName>MobilePlatform.mobileBlock:frame
<Data Concept Type>Data Element"
::=   { essNtcipInstrumentation 6 }
```

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 Section D.4 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 Section D.4 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 Section D.4 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 Section D.4 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 Section D.4 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 Section D.4 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 Section D.4 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 Section D.4 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.
<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.
<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
subsections.
<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 Section D.5 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.
<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
subsections.
<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 (0..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 Section D.6 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 v2

```
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.
<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
subsections.
```

```
<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 Section D.7 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 Section D.8 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 Sub-Surface Sensors

```
numEssSubSurfaceSensors OBJECT-TYPE
SYNTAX      INTEGER (0..255)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>Indicates the number of entries in the Sub-Surface
Sensor Table.
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensorTable.numSensors:quantity
<Data Concept Type>Data Element
<Unit>count"
::=   { essNtcipPavement 3 }
```

5.11.5 Sub-Surface 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 Sub-Surface Sensor

```
essSubSurfaceSensorEntry OBJECT-TYPE
SYNTAX      EssSubSurfaceSensorEntry
ACCESS      not-accessible
STATUS      mandatory
DESCRIPTION "<Definition>A sub-surface sensor is a sensor that reports the
temperature and moisture condition of the roadway sub-surface.   It can be
described through a number of attributes as indicated by the following
subsections.
<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}
```

5.11.6.1 Sub-Surface 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 Sub-Surface 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 Sub-Surface 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 sub-surface.  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 Sub-Surface Sensor Depth

```
essSubSurfaceSensorDepth OBJECT-TYPE
SYNTAX      INTEGER (0..1001)
ACCESS      read-write
STATUS      mandatory
DESCRIPTION "<Definition>Depth of sub-surface 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 Sub-Surface Temperature

```
essSubSurfaceTemperature OBJECT-TYPE
SYNTAX      INTEGER (-1000..1001)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The current sub-surface 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 Sub-Surface Moisture

```
essSubSurfaceMoisture OBJECT-TYPE
SYNTAX      INTEGER (0..101)
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>The sub-surface moisture expressed as a percentage
(eg. 0 indicates dry, 100 indicates saturated) (OID revised per errata).
<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 Sub-Surface 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 (OID
revised per errata).
<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

```
essPavementBlock OBJECT-TYPE
SYNTAX      OerString
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "<Definition>An OER encoded string of the EssPavementData
structure as defined in Section 4.  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-v02
      essSurfaceStatusV2.x                OPTIONAL, -- @NTCIP1204-v02
      essSurfaceTemperature.x       OPTIONAL, -- @NTCIP1204-v02
      essPavementTemperature.x            OPTIONAL, -- @NTCIP1204-v02
```



```
      essSurfaceWaterDepth.x        OPTIONAL, -- @NTCIP1204-v02
      essSurfaceSalinity.x                OPTIONAL, -- @NTCIP1204-v02
      essSurfaceConductivity.x            OPTIONAL, -- @NTCIP1204-v02
      essSurfaceFreezePoint.x       OPTIONAL, -- @NTCIP1204-v02
      essSurfaceBlackIceSignal.x          OPTIONAL, -- @NTCIP1204-v02
      essPavementSensorError.x            OPTIONAL -- @NTCIP1204-v02
}
<SetConstraint>read-only
<DescriptiveName>PavementSensorTable.pavementBlock:code
<Data Concept Type>Data Element"
::=   { essNtcipPavement 5 }
```

5.11.8 Sub-Surface Block Object

```
essSubsurfaceData 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-v02
      essSubSurfaceTemperature.x          OPTIONAL, -- @NTCIP1204-v02
      essSubSurfaceMoisture.x       OPTIONAL, -- @NTCIP1204-v02
      essSubSurfaceSensorError.x          OPTIONAL, -- @NTCIP1204-v02
      }
<SetConstraint>read-only
<DescriptiveName>SubSurfaceSensorTable.subsurfaceBlock:code
<Data Concept Type>Data Element"
::=   { essNtcipPavement 6 }
```

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 in order to de-ice or prevent icing of the pavement.
It can be described through a number of attributes as indicated by the
following subsections.
<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
<SetConstraint>read-only
<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.

PavementTreatmentBlock ::= SEQUENCE {
      treatmentInfo     SEQUENCE OF PavementTreatmentData OPTIONAL
      essPaveTreatmentAmount.0      OPTIONAL, -- @NTCIP1204-v02
      essPaveTreatmentWidth.0 OPTIONAL, -- @NTCIP1204-v02
      ptsOperationalMode.0    OPTIONAL, -- @NTCIP1204-v02
      ptsCommandState.0       OPTIONAL, -- @NTCIP1204-v02
      ptsSprayerState.0       OPTIONAL, -- @NTCIP1204-v02
      ptsSignalDuration.0           OPTIONAL -- @NTCIP1204-v02
      ptsSignalEventCount.0   OPTIONAL, -- @NTCIP1204-v02
      ptsLastSignalEvent.0    OPTIONAL, -- @NTCIP1204-v02
      ptsActiveEventCount.0   OPTIONAL, -- @NTCIP1204-v02
      ptsInactiveEventCount.0 OPTIONAL, -- @NTCIP1204-v02
      ptsLastactiveEvent.0    OPTIONAL, -- @NTCIP1204-v02
      ptsLastInactiveEvent.0  OPTIONAL, -- @NTCIP1204-v02
      ptsError.0              OPTIONAL, -- @NTCIP1204-v02
      ptsMonitoringDetectors.0      OPTIONAL -- @NTCIP1204-v02
}

PavementTreatmentData ::=
      -- for (
      --    x = 1;
      --    x < numEssTreatments.0;
      --    x++)
SEQUENCE {
      essPavementTreatmentIndex.x         OPTIONAL, -- @NTCIP1204-v02
      essPaveTreatProductType.x                 OPTIONAL, -- @NTCIP1204-v02
      essPaveTreatProductForm.x           OPTIONAL, -- @NTCIP1204-v02
      essPercentProductMix.x        OPTIONAL }-- @NTCIP1204-v02
<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      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
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      mandatory
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 transitoin 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 }
```

5.14 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

```
essAirQualityData 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-v02
      essCO2.0                      OPTIONAL,   -- @NTCIP1204-v02
      essNO.0                       OPTIONAL,   -- @NTCIP1204-v02
      essNO2.0                      OPTIONAL,   -- @NTCIP1204-v02
      essSO2.0                      OPTIONAL,   -- @NTCIP1204-v02
      essO3.0                       OPTIONAL,   -- @NTCIP1204-v02
      essPM10.0                     OPTIONAL    -- @NTCIP1204-v02
      }
<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.
<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
subsections.
<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 ppicture 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
subsections.
<DescriptiveName>SnapshotCamera
<Data Concept Type>Class"
INDEX { essSnapshotCameraIndex }
::=   { essSnapshotCameraTable 1 }

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


5.16.3.1 Snapshot Camera Index

```
essSnapshotCameraIndex OBJECT-TYPE
SYNTAX      INTEGER (0..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 snapshot's 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.  (ACCESS revised
per errata)
<SetConstraint>read-only
```

```
<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 }

  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 by the following subsections. 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
[Normative]

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

In order 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. In order 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.

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 Meta-Data		
			5.3.5	essStationMetaDataBlock
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
3.5.1.2.2	F.3.1	Retrieve Battery Status		
			5.3.3	essBatteryStatus

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
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
3.5.1.3.2	F.3.1	Retrieve Mobile Treatment Information		
			5.13.6	pavementTreatmentBlock
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 Meta-Data 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 Meta-Data		
			5.7.1	essNumTemperatureSensors
			5.7.3.1	essTemperatureSensorIndex
			5.7.3.2	essTemperatureSensorHeight
3.5.2.1.4	F.4.6	Retrieve Pavement Sensor Meta-Data		
			5.11.1	numEssPavementSensors

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			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 Sub-Surface Sensor Meta-Data		
			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 Sub-Surface 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	F.4.8	Configure Snapshot Camera		
			5.16.3.1	essSnapshotCameraIndex
			5.16.3.2	essSnapshotCameraDescription

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
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		
			5.3.6	essWeatherBlock
			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		
			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		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			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	F.3.1	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	F.3.1	Retrieve Solar Radiation		
			5.9.2	essTotalSun
			5.9.4	essInstantaneousTerrestrialRadiation
			5.9.5	essInstantaneousSolarRadiation
			5.9.6	essTotalRadiation
			5.9.7	essTotalRadiationPeriod
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		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.3.6	essWeatherBlock
3.5.2.3.3		Monitor Pavement Condition		
3.5.2.3.3.1	F.4.7	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	F.4.7	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
			5.13.1	numEssTreatments
			5.13.3.1	essPavementTreatmentIndex
			5.13.3.2	essPaveTreatProductType

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			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		
			5.11.7	essPavementBlock
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	essSubsurfaceData
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		
			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		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			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 Bio-Hazard 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	essAirQualityData
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
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

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
3.5.2.3.9	F.4.6	Retrieve Snapshot Camera Configuration		
			5.16.1	essSnapshotNumberOfCameras
			5.16.3.1	essSnapshotCameraIndex
			5.16.3.2	essSnapshotCameraDescription
			5.16.3.3	essSnapshotCameraStoragePath
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
			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

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			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
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		
			5.13.7	ptsOperationalMode
3.5.3.4.2	F.3.3	Manually Activate PTS Sprayer		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			5.13.8	ptsCommandState
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 1201 Section 2.5.2.1	eventClassNumber
			NTCIP 1201 Section 2.5.2.2	eventClassLimit
			NTCIP 1201 Section 2.5.2.3	eventClassClearTime
			NTCIP 1201 Section 2.5.2.4	eventClassDescription
			NTCIP 1201 Section 2.5.2.6	eventClassNumEvents
			NTCIP 1201 Section 2.5.4.1	eventConfigID
			NTCIP 1201 Section 2.5.4.2	eventConfigClass
			NTCIP 1201 Section 2.5.4.3	eventConfigMode
			NTCIP 1201 Section 2.5.4.4	eventConfigCompareValue
			NTCIP 1201 Section 2.5.4.5	eventConfigCompareValue2
			NTCIP 1201 Section 2.5.4.6	eventConfigCompareOID
			NTCIP 1201 Section 2.5.4.7	eventConfigLogOID
			NTCIP 1201 Section	eventConfigAction

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			2.5.4.8	
			NTCIP 1201 Section 2.5.4.9	eventConfigStatus
F.2.1.2.2	F.4.3	Configure Logging Service		
			NTCIP 1201 Section 2.5.2.1	eventClassNumber
			NTCIP 1201 Section 2.5.2.2	eventClassLimit
			NTCIP 1201 Section 2.5.2.3	eventClassClearTime
			NTCIP 1201 Section 2.5.2.4	eventClassDescription
			NTCIP 1201 Section 2.5.4.1	eventConfigID
			NTCIP 1201 Section 2.5.4.2	eventConfigClass
			NTCIP 1201 Section 2.5.4.3	eventConfigMode
			NTCIP 1201 Section 2.5.4.4	eventConfigCompareValue
			NTCIP 1201 Section 2.5.4.5	eventConfigCompareValue2
			NTCIP 1201 Section 2.5.4.6	eventConfigCompareOID
			NTCIP 1201 Section 2.5.4.7	eventConfigLogOID
			NTCIP 1201 Section 2.5.4.8	eventConfigAction
			NTCIP 1201 Section 2.5.4.9	eventConfigStatus
F.2.1.2.3	F.4.1	Retrieve Logged Data		
			NTCIP 1201 Section 2.5.2.1	eventClassNumber
			NTCIP 1201 Section	eventClassNumRowsInLog

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			2.5.2.5	
			NTCIP 1201 Section 2.5.2.6	eventClassNumEvents
			NTCIP 1201 Section 2.5.6.1	eventLogClass
			NTCIP 1201 Section 2.5.6.2	eventLogNumber
			NTCIP 1201 Section 2.5.6.3	eventLogID
			NTCIP 1201 Section 2.5.6.4	eventLogTime
			NTCIP 1201 Section 2.5.6.5	eventLogValue
			NTCIP 1201 Section 2.5.7	numEvents
F.2.1.2.4	F.3.3	Clear Log		
			NTCIP 1201 Section 2.5.2.1	eventClassNumber
			NTCIP 1201 Section 2.5.2.3	eventClassClearTime
F.2.1.2.5	F.3.1	Retrieve Capabilities of Event Logging Service		
			NTCIP 1201 Section 2.5.1	maxEventClasses
			NTCIP 1201 Section 2.5.3	maxEventLogConfigs
			NTCIP 1201 Section 2.5.5	maxEventLogSize
F.2.1.2.6	F.3.1	Retrieve Total Number of Logged Events		
			NTCIP 1201 Section 2.5.2.1	eventClassNumber
			NTCIP 1201 Section 2.5.2.5	eventClassNumRowsInLog
			NTCIP 1201 Section 2.5.2.6	eventClassNumEvents

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1201 Section 2.5.7	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 Section 2.2.2	globalMaxModules
			NTCIP 1201 Section 2.2.3.1	moduleNumber
			NTCIP 1201 Section 2.2.3.2	moduleDeviceNode
			NTCIP 1201 Section 2.2.3.3	moduleMake
			NTCIP 1201 Section 2.2.3.4	moduleModel
			NTCIP 1201 Section 2.2.3.5	moduleVersion
			NTCIP 1201 Section 2.2.3.6	moduleType
F.2.2.1.2	F.3.1	Retrieve Device Configuration Identifier		
			NTCIP 1201 Section 2.2.1	globalSetIDParameter
F.2.2.1.3	F.3.1	Retrieve Supported Standards		
			NTCIP 1201 Section 2.2.4	controllerBaseStandards
F.2.2.1.4	F.3.1	Retrieve System Name		
			RFC1213.1	sysName
F.2.2.1.5		Manage Time		
F.2.2.1.5.1	F.3.3	Set Time		
			NTCIP 1201 Section 2.4.1	globalTime
F.2.2.1.5.2	F.3.1	Retrieve Current Time		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1201 Section 2.4.1	globalTime
F.2.2.1.6	F.4.6	Retrieve External Port Information		
			NTCIP 1201 Section 2.8.2	auxIOTableNumAnalogPorts
			NTCIP 1201 Section 2.8.3.1	auxIOPortType
			NTCIP 1201 Section 2.8.3.2	auxIOPortNumber
			NTCIP 1201 Section 2.8.3.3	auxIODescription
			NTCIP 1201 Section 2.8.3.4	auxIOResolution
			NTCIP 1201 Section 2.8.3.6	auxIOPortDirection
F.2.2.1.7	F.4.8	Configure Port Information		
			NTCIP 1201 Section 2.8.3.1	auxIOPortType
			NTCIP 1201 Section 2.8.3.2	auxIOPortNumber
			NTCIP 1201 Section 2.8.3.3	auxIODescription
F.2.2.2		Generic Status Monitoring Requirements		
F.2.2.2.1	F.4.6	Monitor Status of External Device		
			NTCIP 1201 Section 2.8.3.5	auxIOValue
			NTCIP 1201 Section 2.8.3.7	auxIOLastCommandedState
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 Section 2.8.3.1	auxIOPortType

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
			NTCIP 1201 Section 2.8.3.2	auxIOPortNumber
			NTCIP 1201 Section 2.8.3.5	auxIOValue
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
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		Response Time for Requests		See Requirement 3.6.21 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		

Req ID	Dialog	Requirement	Object ID	Add'l Requirements/Object
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]

The following figure 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.

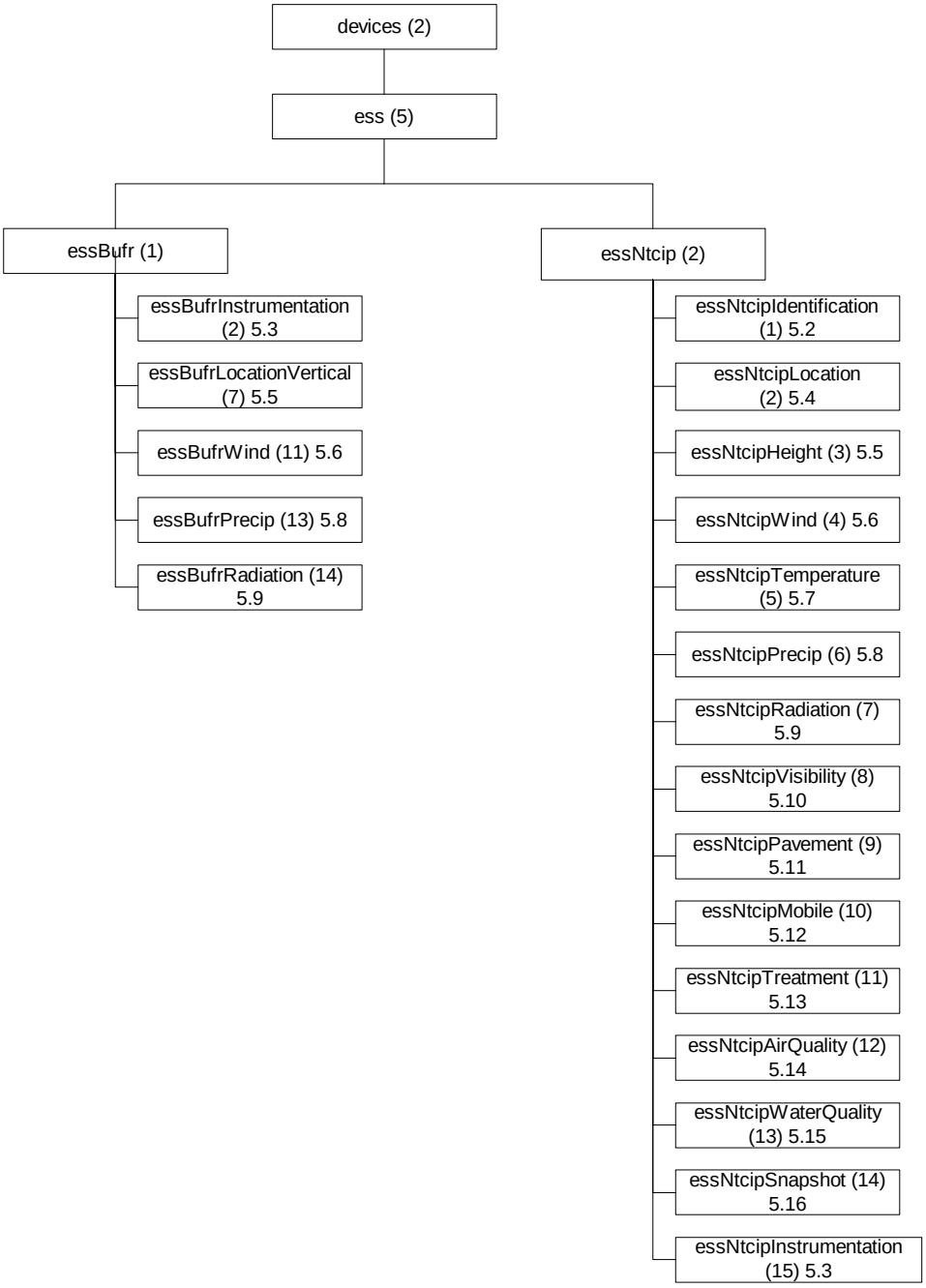

Figure B-1
Object Tree for NTCIP 1204 v02

Annex C
TEST PROCEDURES
[Normative]

< This annex will include additional content in a future version of this standard. >

Annex D
DOCUMENTATION OF REVISIONS
[Informative]

This annex identifies the changes made in NTCIP 1204 v02 that have required the deprecation of objects. The NTCIP effort makes reasonable efforts to ensure that the standards are as backwards compatible as possible, but the primary purpose of the standard is to provide interoperability by developing standards in a consensus environment. When changes are required to meet these objectives, the problematic objects are deprecated and, in most cases, are replaced with new objects. This annex identifies why each of these changes have been made. New implementations shall support the new and replacement objects; and as a default condition shall also support deprecated objects.

D.1. NTCIP 1204 V02

General edits have been made to the MIB header in NTCIP 1204 v02 (this standard) in order to reflect updates to other MIBs from which this MIB imports data.

All DESCRIPTION fields have been updated to conform to the NTCIP 8004 standard.

The STATUS of all objects has been changed to "mandatory" in order to reflect the fact that conformance is now measured through the use of the PRL as contained in NTCIP 1204 v02 and the RTM contained in Annex A of this standard.

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.

D.2. STATION CATEGORY

The definition of essNtcipCategory was modified in NTCIP 1204 v02 (this standard) to reflect the fact that this parameter relates to "category" rather than "type".

D.3. LATITUDE

The definition of the essLatitude object was modified in NTCIP 1204 v02 (this standard) in order to reference the datum set to be used.

D.4. WIND SENSOR INFORMATION

The various wind objects were deprecated in NTCIP 1204 v02 (this standard) and replaced with a table in order 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.5. WATER DEPTH

The essWaterDepth object was deprecated in NTCIP 1204 v02 (this standard) in order 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.6. SOLAR RADIATION

The essSolarRadiation object was deprecated in NTCIP 1204 v02 (this standard) in order 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 (essInstantaneousRadiation) but may also support essSolarRadiation for backwards compatibility purposes.

D.7. SURFACE WATER DEPTH

The essSurfaceWaterDepth object was deprecated in NTCIP 1204 v02 (this standard) in order 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.8. SURFACE CONDUCTIVITY

The essSurfaceConductivity object was deprecated in NTCIP 1204 v02 (this standard) in order 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.9. APPROVED ERRATA FROM 2007 [INCLUDED IN 1204 V02.24]

In November 2007, the NTCIP ESS WG approved four errata:

4.2.1 Capture Snapshot Image

The text of the section shall be modified to read:

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

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. [MODIFIED TO READ] <u>The management station shall repeatedly GET essSnapshotCameraCommand.x until it equals ready (1)</u>.

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 4-1.

5.11.6.6 Sub-Surface Moisture

The Object Identifier for essSubSurfaceMoisture shall be changed as follows:
::= { essSubSurfaceSensorEntry ~~6~~ 7 }

5.11.6.7 Sub-Surface Sensor Error

The Object Identifier for essSubSurfaceSensorError shall be changed as follows:
::= { essSubSurfaceSensorEntry ~~7~~ 8 }

5.16.3.4 Snapshot Camera Command

The ACCESS field shall be modified to read:
ACCESS read-~~only~~write

Annex E
User Requests
[Informative]

This annex identifies features that were suggested for this standard but either are supported by mechanisms that may not be readily obvious or are not supported by this version of the standard.

E.1. FEATURES INDIRECTLY SUPPORTED

The following sections identify how certain features are supported by the standard.

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., due to a long polling cycle over a dial-up link). In order for this to operate in a meaningful manner, each entry into the archive must 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 Section 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.

The WG discussed this feature and concluded that it would
a. Result in a standard that was not backwards compatible with version 1
b. Result in a standard that was roughly three times the size and complexity of the current standard
c. Be difficult to implement and test

Instead, the 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., due to communication costs).

The 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 Base Standards, Protocols, and Profiles WG is in the process of developing this capability as a part of NTCIP 1103 version 2. As soon as this standard is completed, this service can be provided without requiring any changes to this standard.

Annex F
Generic Clauses
[Normative]

This Annex 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 within this Annex will eventually be defined in a separate standard. However, this section serves as a placeholder until this is achieved.

F.1. EXTERNAL CONCEPT OF OPERATIONS

F.1.1. Generic Architectural Needs

This standard 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 must be clearly associated with a specific sensor. Once the management station has retrieved the data of interest, the operator can use this information to make decisions and initiate other events (such as changes to DMS messages) to better manage the transportation system.

In order to enable communications between these components, the transportation system manager will need to establish a communication system that links the ESS with a management station. For some systems, the cost of communications may be minimal and as such the system may be designed for constant polling; other systems may encounter significant costs for communicating with the ESS and as such the system may be designed to minimize data exchanges. When deploying an ESS, the system designer must 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 due to limitations in the data rates of the media and/or due to 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

The following subsections document features of an ESS that are generic to most devices.

F.1.2.1. Retrieve the 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 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 are provided in the following subsections.

F.2.1.1. Support Basic Communications

Requirements for making requests are provided in the following subsections.

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 are provided in the following subsections.

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 are provided in the following subsections.

F.2.2.1. Generic Configuration Requirements

Requirements for configuring a device controller are provided in the following subsections.

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 will only change when changes are made to the controller configuration.

F.2.2.1.3. Retrieve Supported Standards

Upon request from a management station, the device shall return the NTCIP standards which 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 are provided in the following subsections.

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 are provided in the following subsections.

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 are provided in the following subsections.

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 are provided in the following subsections.

F.2.3.1. Supplemental Requirements for Event Monitoring

Supplemental requirements for monitoring for the occurrence of certain events are provided in the following subsections.

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 are provided in the following subsections.

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 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. Sections C.4.1.1 through C.4.1.4 provide a description of the key services offered by SNMP assuming no errors; precise rules and procedures are defined in NTCIP 1103. Section C.4.1.5 extends the requirements of NTCIP 1103 by providing additional requirements that supplement but do not replace any requirements of NTCIP 1103.

NOTE: In order to promote interoperability and to the reflect marketplace realities, this standard requires support for the Simple Network Management Protocol. Use of the other protocols defined in NTCIP 1103 (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 costs 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 C-1. Both the Get request and the GetResponse messages contain a list of objects as defined by the varBindingList structure (see Section C.4.1.4).

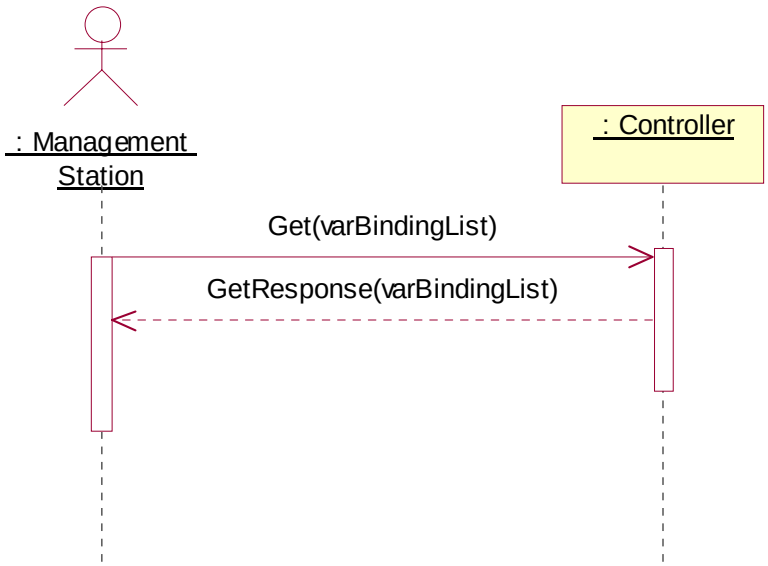

Figure C-1
SNMP Get Interface

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 Section C.3.1.1.2. This process consists of a GetNext request and a GetResponse as depicted in Figure C-2. Both the GetNext request and the GetResponse messages contain a list of objects as defined by the varBindingList structure (see Section C.4.1.4).

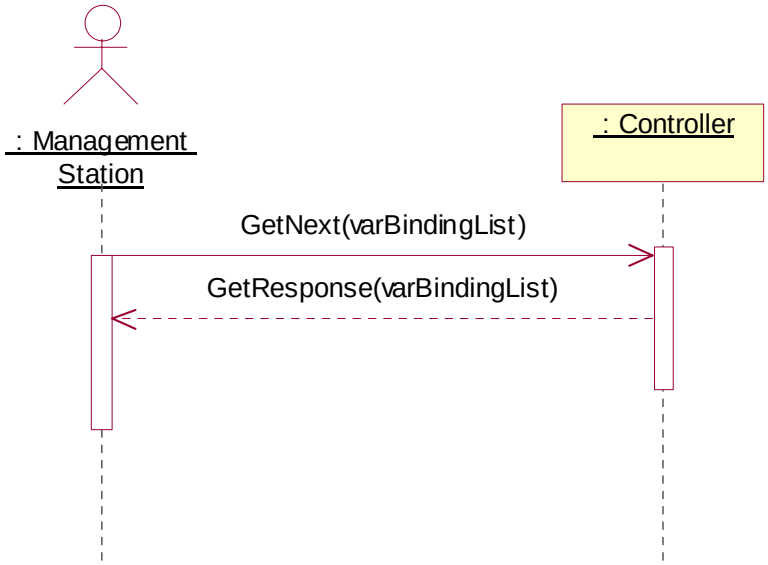

Figure C-2
SNMP GetNext Interface

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 C-3. Both the Set request and the GetResponse messages contain a list of objects as defined by the varBindingList structure (see Section C.4.1.4).

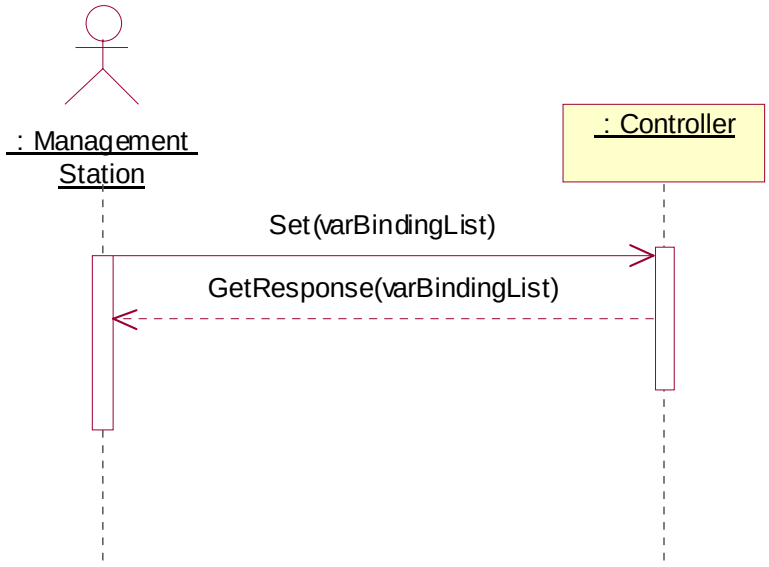

Figure C-3
SNMP Set Interface

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 this standard, by referencing the 'SET' operation, and directly by the RTM, by section number, in order 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 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 is relationship is depicted in Figure C-4.

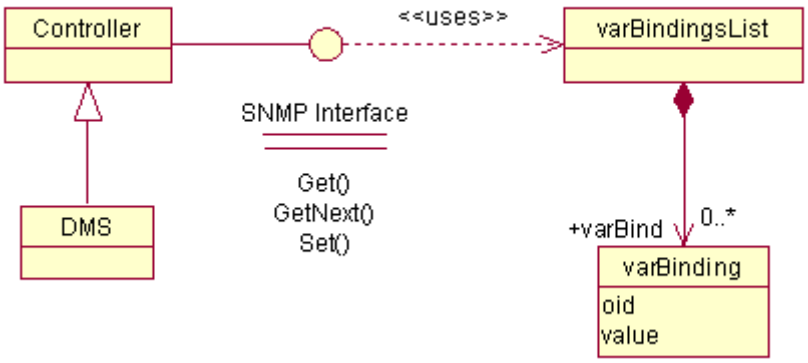

Figure C-4
SNMP Interface - View of Participating Classes

Additional requirements related to the contents of the variable binding list are provided in the following subsections.

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 Section 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.

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 in order 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 in order 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 in order 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 in order 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 for the definition of how events shall be managed.

F.4.5. Manage Exception Reporting

See NTCIP 1103 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 other dialogs with specific object names. The list of objects provided by the specific dialog shall include (1) an object that indicates the number of rows in the table, (2) the object(s) that serve as the index field of the table row, and (3) 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 (1) the object(s) that serve as the index field of the table row, and (2) 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 (1) the object(s) that serve as the index field of the table row, and (2) 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).

§