

DATE: November 2, 2001

TO: AASHTO, ITE, and NEMA

SUBJECT: **Recommended Amendment 1 to NTCIP 1203:1997, Object Definitions for Dynamic Message Signs (DMS), part of the NTCIP**

ACTION: Request for Ballot to Approve

RECOMMENDATION. This Bulletin reports that the Joint Committee on the NTCIP accepted the proposed Amendment 1 to NTCIP 1203:1997, *Object Definitions for Dynamic Message Signs (DMS)*, as a Recommended Amendment, and agreed to refer the document to AASHTO, ITE, and NEMA for balloting and approval. The recommendation was made at the scheduled Joint Committee meeting in Monterey, California on March 21-22, 2001, by the required two-thirds majority of the 18-member committee. The motion to recommend the amendment carried with 14 affirmative votes, no negative votes, and no abstentions.

The Joint Committee on the NTCIP contributes to the joint standards development process by accepting a proposed draft document at two stages in its development – first as a User Comment Draft and then again as a Recommended Standard. Since the amendment was drafted in response to user comments, the document was advanced to the Recommended Amendment stage.

BACKGROUND. This document was developed to amend a Joint AASHTO / ITE / NEMA Standards Publication, part of the NTCIP family of standards and profiles. The NTCIP 1203 Amendment 1 document defines corrections and revisions to the DMS device data dictionary.

The NTCIP Dynamic Message Sign Working Group managed a user comment database of all comments received from many sources: user experience in four years of implementing 1203, the Battelle testing project, and from the working group members. The comments were divided into two groups -- the first group identified comments dealing with ambiguities that needed to be addressed in the amendment, and the second group identified new functionalities to be addressed in 1203 version 2. The responses to



the first group of comments led to the development of the amendment. The Amendment 1 was developed and revised over a nine-month period in 2000 and 2001.

NTCIP 1203 Amendment 1 was developed by the NTCIP DMS Working Group, chaired by Layne Mostad (Daktronics, Inc.) Mr. Mostad was subsequently replaced by Chris Bates, also of Daktronics. When the document was prepared, the DMS WG had 31 members from both the public and private sectors.

Also attached to this NTCIP Standards Bulletin is supporting information, called the Standards Development Report, required by the ITE standards development process. The ITE requires a list of members and user comments.

An electronic copy of this Recommended Amendment is available in the LIBRARY area of the NTCIP Website, at <http://www.ntcip.org>.

NOTES. The Joint Committee accepted 1203 Amendment 1 version 20Feb01. Subsequent changes to make version 07 included: added the title page, corrected typographic errors, and revised the document reference numbers.

the distribution of this document was delayed from April 2001 until now by the corrections, preparing the SDR, and administrative delay.

The NTCIP standards bulletin, SDR, and standards publication are being distributed in the Adobe™ PDF format.

BALLOT INSTRUCTIONS. The eligible voting representatives who receive this Bulletin are asked to follow the ballot instructions of their standards organization.

**ATTACH-
MENT:** 1. Standards Development Report, August 20, 2001,
with Appendices A and B.

2. NTCIP 1203 Amendment 1, version 07,
July 03, 2001, ~21 pages.

FROM: NTCIP Coordinator
NEMA
1300 North 17th Street, Suite 1847
Rosslyn, VA 22209-3801
fax (703) 841-3331
email <ntcip@nema.org>

CC: Email distribution with PDF attachment:
A. For redistribution to AASHTO staff; ITE staff
B. Joint Committee on the NTCIP; WG reflector
C. NEMA members
D. FHWA; JPL; Mitretek; ASTM; IEEE; IMSA; ITS-A; SAE;
SIA; TSC; ITS press editors
Email distribution without attachment:
E. NTCIP community via fullroster reflector

###

Standards Development Report

NTCIP 1203:1997
NTCIP - Object Definitions for Dynamic Message Signs (DMS)

AMENDMENT 1

Prepared By : NTCIP Dynamic Message Sign Working Group

Date: August 20, 2001. Revised November 2, 2001.

Table of Contents

1. DRAFT PROPOSED STANDARD	3
2. STATUS REPORT	3
2.1 SUBMIT REQUEST TO INITIATE NTCIP ACTIVITY	3
2.2 JOINT COMMITTEE VOTES TO FORM WORKING GROUP	4
2.3 JOINT COMMITTEE FORMS WORKING GROUP	4
2.4 WG DEVELOPS WORKING GROUP DRAFT FOR INFORMAL REVIEW.....	4
2.5 WG PREPARES USER COMMENT DRAFT	4
2.6 JOINT COMMITTEE VOTES ON DISTRIBUTION OF USER COMMENT DRAFT	4
2.7 JOINT COMMITTEE DISTRIBUTES USER COMMENT DRAFT AND RECEIVES COMMENTS	4
2.8 WORKING GROUP RESOLVES USER COMMENTS.....	4
2.9 WG DECIDES TO SUBMIT A "RESOLVED" VERSION AS A RECOMMNEDED STANDARD	4
2.10 WORKING GROUP PREPARES A DRAFT RECOMMENDED STANDARD.....	4
2.11 JOINT COMMITTEE VOTES ON SUBMISSION OF THE RECOMMENDED STANDARD TO SDO.....	4
2.12 JOINT COMMITTEE FORWARDS RECOMMENDED STANDARD TO SDOs.....	5
2.13 SDO APPROVE THE RECOMMENDED STANDARD AND THEREBY CREATE THE STANDARD.....	5
2.14 SDOs MAINTAIN THE STANDARD.....	5
3. USER COMMENTS	5
4. COMMITTEE OBJECTIVES	5
5. COMMITTEE MEMBERSHIP	5
6. HARMONIZATION DECLARATION	6
7. ABSTRACT	6
8. ALTERNATE PROCESSES.....	6
APPENDIX A – MEMBERSHIP LIST	7
APPENDIX B – COMMENT RESOLUTION LIST	8

1. DRAFT PROPOSED STANDARD

Name: NATIONAL TRANSPORTATION COMMUNICATIONS FOR ITS PROTOCOL (NTCIP)
Object Definitions for Dynamic Message Signs (DMS) – Amendment 1
Reference Number: NTCIP 1203 - Amendment 1

2. STATUS REPORT

The Joint AASHTO/ITE/NEMA Committee on the NTCIP uses a 14 step process in the development of standards. These steps and dates of completion in the development of –Amendment 1 to NTCIP 1203 are as follows:

1.	Submit request to initiate NTCIP Activity	9/1999
2.	Joint Committee votes to form Working Group	previously established; re-established in 5/2000
3.	Joint Committee forms Working Group	previously established; reactivated in 5/2000
4.	WG develops Working Group Draft for informal review	2000
5.	WG prepares User Comment Draft	9/2000
6.	Joint Committee votes on distribution of User Comment Draft	N/A *
7.	Joint Committee distributes User Comment Draft and receives comments	N/A *
8.	Working group resolves user comments	N/A *
9.	WG decides to submit a "resolved" version as a Recommended Standard	07/2000
10.	Working Group prepares a Draft Recommended Standard	8/2000
11.	Joint Committee votes on submission of the Recommended Standard to SDO.	3/2001
12.	Joint Committee forwards Recommended Standard to SDOs	3/2001
13.	SDO approve the Recommended Standard and thereby create the Standard	TBD
14.	SDOs maintain the Standard	TBD

* = the Technical WGs have proposed and the JC has agreed that Amendments don't need to follow the entire process. Especially, the User Comment Draft period is being skipped. The argument is that for Amendments, which should not introduce any new features but clarifications, time is of essence and every interested user can still comment during the ballot.

Additional detail is provided in the following subclauses.

2.1 Submit request to initiate NTCIP Activity

The request to initiate the development of Amendment 1 to NTCIP 1203 was first formulated within the NTCIP Joint Committee in 1999.

2.2 Joint Committee votes to form Working Group

The NTCIP Joint Committee requested the re-establishment of the existing DMS WG to prepare an amendment that clarifies the ambiguities within the published standard. These ambiguities were determined through actual implementations and the Battelle Testing effort.

2.3 Joint Committee forms Working Group

Working sponsor: Mike Forbis (WSDOT)

The DMS WG was already established, but needed to be re-activated. However, additional vendors, public agency representatives and system integrators/consultants were approached to provide input to the effort.

2.4 WG develops Working Group Draft for informal review

The working group developed a user comment database that contained all comments received from actual implementations, the Battelle Testing effort as well as from other observations. These comments were split into 2 groups: the first one identified comments that needed to be addressed in the amendment since those dealt with ambiguities, while the second group identified new functionalities that will be addressed in Version 2 of the 1203 standard. The part of the UCD comment database including the WG responses to the comments are attached to this SDR.

The responses to the comments lead to the development of the amendment. Several versions of the amendments were developed, reviewed and discussed within and among the WG members.

2.5 WG prepares User Comment Draft

Not applicable for Amendments (see note under Table 1)

2.6 Joint Committee votes on distribution of User Comment Draft

Not applicable for Amendments (see note under Table 1)

2.7 Joint Committee distributes User Comment Draft and receives comments

Not applicable for Amendments (see note under Table 1)

2.8 Working group resolves user comments

Not applicable for Amendments (see note under Table 1).

2.9 WG decides to submit a "resolved" version as a Recommended Standard

The WG incorporated user comments received during the 4 years of 1203 implementations, comments from the Battelle Testing, as well as internal comments relating to consistency in the document. The WG voted in July 2000 to forward the amendment as 'recommended standard' (RS) to the Joint committee.

2.10 Working Group prepares a Draft Recommended Standard

The Amendment 1 as Recommended Standard was completed and finalized in August 2000.

2.11 Joint Committee votes on submission of the Recommended Standard to SDO.

On the question of accepting Amendment 1 of NTCIP 1203, Dynamic Message Signs (DMS), as a Recommended Standard, the teller's report is:

At the March 21-22, 2001 NTCIP JC meeting in Monterey, the minutes report:

Mike Forbis introduced the subject of how the comments on the Amendment have been incorporated into the document. Ken Vaughn discussed the addition of additional objects to the DMS Amendment and other comments: typos and editorial changes. Responses were submitted to the commenter. All comments received to-date have been addressed that do not add features. All comments dealing with features and adding objects will be included in Version 2 of the Standards. The committee continued to discuss these issues associated with the revised standard. Warren feels that the specification should have some clear direction as to how the users should use the standard. New features for version 2 will have graphics, multiple power supply, area, improve and provide some informative, colors, message durations, multiple signs, review the pics for the standards. There will be some substantive changes to the document based on a proposal by Ken. There was discussion concerning questions that were submitted by Bob DeRoche. These questions have been addressed in changes in the document. The Working Group is concerned that the document has to go back through the USERS COMMENT Stage. James Cheeks discussed the ITE Request for Revision Process. Tom Brahm advised the JC that the ITE process can be changed to add a statement: "The Community feels that the Changes should be made and the Standard should be moved forward to Ballot".

*In response to the needs of user community a motion was submitted to advance the Amendments to NTCIP 1203. Dynamic Message Sign Objects to the SDO for approval. A note will be added that the JC is responding to the needs of the user community to get this document out for use. Motion: Colin Rayman
Second: Mike Forbis Yea: 14 Nay: 0*

2.12 Joint committee forwards Recommended Standard to SDOs

Joint Committee awaiting SDR from Working Group for submittal to SDOs.

2.13 SDO approve the Recommended Standard and thereby create the Standard

TBD

2.14 SDOs maintain the Standard

TBD

3. USER COMMENTS

The NTCIP User Comments relevant to this amendment are provided in Appendix B. Working group member comments were handled the same way as 'formal' user comments.

4. COMMITTEE OBJECTIVES

The main objective of the committee was to develop clarifications and remove ambiguities of Version 1 of NTCIP 1203. The WG was to review all received comments and to address those comments relevant to ambiguities (additional function requests and comments to that respect were to be addressed in Version 2 of NTCIP 1203). After provision of this Amendment, the WG is to proceed to address the other comments and to prepare a Version 2.

5. COMMITTEE MEMBERSHIP

See Appendix A.

6. HARMONIZATION DECLARATION

The Dynamic Message Sign (DMS) application performs unique traffic management functions but every effort was made to ensure that there was no duplication with existing standards such as the Global Object Definitions (1201:1997). Additionally, the clarifications defined in this Amendment were to remove ambiguities and to keep the modifications of the objects to the minimum possible. It is understood that due to these modifications, certain real implementations would become 'non-compliant' to Amendment 1. However, this drawback needed to be taken to ensure future interoperability between vendor products.

7. ABSTRACT

This document provides an amendment for a Dynamic Message Sign (DMS) system. A DMS is used to provide traveler information en-route to various types of travelers.

8. ALTERNATE PROCESSES

The process used in the development of this standard is defined in the Joint AASHTO/ITE/NEMA Committee on the NTCIP: Joint committee Standards Development Process.

APPENDIX A – MEMBERSHIP LIST

NTCIP Dynamic Message Sign Working Group Membership List

Company/Agency	Representation	Contact
Daktronics, Inc.	NTCIP Committee/ Working Group Chair	Chris Bates
3M Dynamic Message Signs	Private	Kenneth Bare
ADDCO	Private	Jerry Meyers
ADDCO	Private	David Woosley
American Signal Company	Private	Patrick Ryan
Delcan Corporation	Private	Richard Chylinski
DMJM-FR Harris	Private	Robert Pulver
Fiberoptic Display Systems	Private	Robert Blasi
FHWA	Public/FHWA	Emiliano Lopez
IBI Group	Private	James Barbosa
ITE	ITE	James Cheeks
ITS America	Public	Richard Taylor
LEDStar, Inc.	Private	Milan Patel
Minnesota DOT	Public	Erik Engstrom
MTA Bridges and Tunnels	Public	Patrick Chan
Mitretek Systems, Inc.	Private	Blake Christie
MTA Bridges and Tunnels	Public	Assem Ramadan
NEMA	NEMA	Bruce Schopp
PB Farradyne, Inc.	Private	Joerg ‘Nu’ Rosenbohm
Port Authority of NY and NJ	Public	Ira Huttner
Rijkswaterstaat AVV	Private	Job Klignhout
Southwest Research Institute	Private	Amit Misra
Skyline Products, Inc.	Private	Dan Vanada
TransCore	Private	Joeseeph Knapka
Transdyn	Private	Chris Gilbert
Trevilon Corporation	Private	Kenneth Vaughn
Texas Transportation Institute	Private	Edward Seymour
Virginia DOT	Public	Michael Winn
Vultron, Inc.	Private	Don Davis
Washington DOT	Public	Michael Forbis
Washington State DOT	Public	Mark Morse

APPENDIX B – COMMENT RESOLUTION LIST

Responses to Comments on NTCIP 1203:1997

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
<p>What to do if an installation (field device) supports objects whose function is not supported, i.e., for backwards compatibility reasons or RFP responses. Not clarifying this function will lead to ambiguous interpretations of the returned values.</p>	
<i>Response</i>	
<p>In order to claim compliance to an object, at least one value of the object must be supported and the device must be able to properly implement that value within the device. Thus, the statement that a device would support an object without supporting the functionality is an impossibility.</p>	

<i>Page</i>	<i>Paragraph</i>
	Areas
<i>Comment</i>	
<p>We need a way to define areas within MULTI</p>	
<i>Response</i>	
<p>The concept of defining areas on a sign will be considered in the development of version 2.</p>	

<i>Page</i>	<i>Paragraph</i>
	Graphics
<i>Comment</i>	
<p>We need a way to define and display graphics within MULTI.</p>	
<i>Response</i>	
<p>The ability to support graphics will be considered in the development of version 2.</p>	

<i>Page</i>	<i>Paragraph</i>
	Scope
<i>Comment</i>	
<p>Does this document cover portable signs? If so, this should be clarified.</p>	
<i>Response</i>	
<p>Yes, the document does cover portable signs as already reflected within the definition of the dmsSignType object.</p>	

<i>Page</i>	<i>Paragraph</i>
	Graphics
<i>Comment</i>	
There should be a way to post graphics on the sign.	
<i>Response</i>	
The ability to display graphics will be a major issue in the development of version 2 of the standard.	

<i>Page</i>	<i>Paragraph</i>
	New Objects
<i>Comment</i>	
The MIB should include objects to monitor heaters and ventilation conditions	
<i>Response</i>	
Objects for heating and ventlation will be considered in the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
Provide better descriptions of error conditions. For example, describe the error codes to be returned when modifying the illumination.	
<i>Response</i>	
<p>A few of the error objects should have explanations added to their DESCRIPTIONS. These will be investigated for the next revision. For your example, the new DESCRIPTION for dmsIllumBrightnessValuesError will look something like:</p> <p>DESCRIPTION "Indicates the error encountered when the brightness table was SET. other(1) is for a manufacturer specific indication when none of the other possible values can be used. none(2) indicates that no error was encountered. photocellGap(3) indicates that certain brightness levels are undefined. negativeSlope(4) indicates that the photocell range used to select a brighter brightness level is lower or overlaps the photocell range used to select a dimmer brightness level. tooManyLevels(5) indicates that more brightness levels are defined than are reported by dmsIllumNumBrightLevels. invalidData(6) indicates a manufacturer defined condition of invalid data not described by the other options."</p>	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
Develop standards that are easier to code.	
<i>Response</i>	
It is the belief of the WG that the long-term benefits of the standard outweigh the short-term costs of implementation; wheather this is accurate will be demonstrated through the success of the standard in the open marketplace.	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
Develop standards that require less bandwidth.	
<i>Response</i>	
It is the belief of the WG that the long-term benefits of the standard outweigh the short-term costs of implementation; wheather this is accurate will be demonstrated through the success of the standard in the open marketplace.	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
Develop standards that enable more cost effective devices and system implementations.	
<i>Response</i>	
It is the belief of the WG that the long-term benefits of the standard outweigh the short-term costs of implementation; wheather this is accurate will be demonstrated through the success of the standard in the open marketplace.	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
Documentation of the existing NTCIP standards should be enhanced to include descriptions of the intended use of the communications protocols (e.g., difference between PPP and PMPP, for which (combination of) standards are off-the-shelf software packages available). A specific suggestion is to describe the use of object definitions (with examples).	
<i>Response</i>	
This is not an issue for the DMS standard, the comment has been forwarded to the NTCIP Joint Committee for their consideration for a potential enhancement to the NTCIP Guide.	

<i>Page</i>	<i>Paragraph</i>
	Memory Type
<i>Comment</i>	
Each message should be associated with a name (ASCII string). And a new activation code should be defined to allow message activation by calling the message name. This will allow a common way to activate the same message in multiple signs regardless of whether it is stored in volatile or changeable memory.	
<i>Response</i>	
The WG has not recognized a justification for adding this object as a standard feature. It can readily be handled within a central software system and/or handled as a proprietary object (but use of name without CRC is discouraged to activate a message).	

<i>Page</i>	<i>Paragraph</i>
	WYSIWIG
<i>Comment</i>	
<p>The standards should support 'simpler' objects to define the WYSIWYG function. Since most of the signs we are dealing with are still based on 5x7 pixel characters, we should develop an object that can provide this function. I calculated that we would need about 541 bytes where each bit represents a pixels starting in the upper left hand corner moving right and then down. This byte number equals in a full matrix sign (2 pixels on each side, 2 pixels in between the up-to 21 characters and 2 pixels on the bottom and top plus 3 lines with 2 pixels between the lines = $(2+21*5+20*2+2)*(2+3*7+2*2+2) = 149 * 29 = 4321$ pixels). Since this is too large for an SNMP object, I could see developing objects that allow the transmission by display line (and I recognize the limitations. However, I think we should discuss how strong an issue this is).</p>	
<i>Response</i>	
A new WYSIWYG mechanism will be considered for version 2.	

<i>Page</i>	<i>Paragraph</i>
	Graphics
<i>Comment</i>	
The standard should support graphics	
<i>Response</i>	
Graphics will be a major consideration for the development of version 2 of the standard.	

<i>Page</i>	<i>Paragraph</i>
	Tables
<i>Comment</i>	
In general, I would like to see a better definition of the intent for tables. Particularly whether they are meant to be sparsely populated or not, and the proper method to add and delete rows. Also if sparsely populated and a row is not in use, should it return a default value for that row or noSuchName?	
<i>Response</i>	
The handling of sparsely populated tables will be clarified in version 2 with a warning provided in the amendment that these items may change.	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
There should be better description of the intended use of objects. For example, clarification of the different uses of pixel test, pixel service, and lamp test.	
<i>Response</i>	
The terms 'pixel,' 'pixel service,' and 'lamp' have been clarified.	

<i>Page</i>	<i>Paragraph</i>
	General
<i>Comment</i>	
How should sign respond if a MIB object is not supported, genErr or noSuchName? For example, if a permanent sign has no fuelLevel, how should it respond? Probable answer is noSuchName, but if spec called for it but sign doesn't logically do it, would genErr make sense?	
<i>Response</i>	
If an object is absent from the manufacturer's supplied MIB and it is requested anyway, the device must respond with noSuchName, according to NTCIP 1101, section 5.1.1.4. If the device claims support for the object, the device must support at least one value for the object; in this case, the object would presumably support the value of zero indicating that there is no fuel.	

<i>Page</i>	<i>Paragraph</i>
	messageDisplay
<i>Comment</i>	
The messageDisplay pixel error specification has no way to tell what page of a message contains the error.	
<i>Response</i>	
This issue is being further discussed within the WG. For now, the working group has concluded that the messageDisplay pixel error type is a cumulative account of failed pixels in the display. This will be further clarified in the amendment and the next version.	

Page	Paragraph
	Fan Failures
Comment	
The fanFailures object does not specify ordering of bits.	
Response	
The relationship between individual bits and individual fans is manufacturer specific and should be explained in the manufacturer's documentation.	

Page	Paragraph
	General
Comment	
<p>I can't find any specifications on objects that represent PICTOGRAMs (bitmap image) messages!</p> <p>There is an object that represent font bitmap : characterBitmap but this is for character representation ... The message itself is represent by the dmsMessageMultiString. dmsMessageMultiString must be written in MULTI-language (text and message attributes) ... The MULTI-language does not support pictograms.</p> <p>How can I implement the remote transfer/management of pictograms for our portable full matrix VMS ?</p> <p>---> TS3.6 MESSAGE TABLE CONFORMANCE GROUP (no support for pictogram mesages)</p> <p>With our current proprietary protocol, our client can change text and pictogram messages ... We would like to keep that functionality ...</p>	
Response	
Graphics will be a major item for consideration in version 2.	

Page	Paragraph
	2.1
Comment	
<p>In the MIB : vmsCfg OBJECT IDENTIFIER ::= { dms 2 }</p> <p>fontDefinition OBJECT IDENTIFIER ::= { dms 3}</p> <p>The index for those objects must be "swap" according to < 1.5 DMS OBJECT TREE > TS3.6-1997 Page 1-13 or Page 12 Sec 3 20/85 of VMSOB14.DOC (MS Word document)</p>	
Response	
The graphic in section 1.5 will be corrected.	

<i>Page</i>	<i>Paragraph</i>
1-4	Central Remote Control Mode
<i>Comment</i>	
Shouldn't this term be "Central Control Mode"? Also, it would be nice to improve these definitions as non-DMS experts have a difficult time understanding what is meant.	
<i>Response</i>	
Yes, the 'central remote control mode' choice for controlMode will be changed to "Central Control Mode." In addition, the definitions of the control modes will be clarified in the standard, in the definition section.	

<i>Page</i>	<i>Paragraph</i>
2-12	Colors
<i>Comment</i>	
Do we want to support more than the current list of 10 colors? If so how do we wish to support this feature?	
<i>Response</i>	
We will consider this issue in the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
2-12	New
<i>Comment</i>	
There should be an object to define how many pages a sign controller can allow in a single message.	
<i>Response</i>	
We will add a read-only object in version 2 to define the maximum number of pages supported by the sign.	

<i>Page</i>	<i>Paragraph</i>
2-15	Message Table
<i>Comment</i>	
The message table should include a name for each message.	
<i>Response</i>	
The WG has not recognized a justification for adding this object as a standard feature. It can readily be handled within a central software system and/or handled as a proprietary object (but use of name without CRC is discouraged to activate a message).	

<i>Page</i>	<i>Paragraph</i>
2-17	Multi-Message Displays
<i>Comment</i>	
There should be a way to allow multiple messages to be displayed "simultaneously". For example, while Message 1 is displayed, someone requests Message 2 and wants the two pages alternated.	
<i>Response</i>	
The WG concluded that if a central system wishes to display multiple messages simultaneously, it should create a new multi-page message. This design will allow the central/system and sign to check for display conflicts (e.g., maximum number of pages, confusing interpretations of text between 2 messages, etc.).	

<i>Page</i>	<i>Paragraph</i>
2-17	2.6.1.1.1.8.5 Message CRC
<i>Comment</i>	
It is unclear how to calculate the Message CRC when the optional objects for pixel service and beacons are not present.	
<i>Response</i>	
The amendment modified the wording such that these objects shall be disabled/off by default (perhaps in the future we will make the objects mandatory, but an implementation may sub-range to off). If these objects are not present, the CRC shall be calculated with the default values.	

<i>Page</i>	<i>Paragraph</i>
2-18	Run Time Priority
<i>Comment</i>	
It is unclear how prioritization works. Which priority (activation or run time) takes priority over the other? This should be clarified in the standard.	
<i>Response</i>	
The operation of the prioritization scheme was partially clarified in the amendment and will be further clarified in version 2.	

<i>Page</i>	<i>Paragraph</i>
2-18	Pixel Service
<i>Comment</i>	
<p>The pixel service objects need further clarification. For instance what happens if I power up a sign with a PixelServiceFrequency of every hour but the PixelServiceTime is 6 hours from the power up time? Also how does one turn the pixel service off? A duration of 0?</p>	
<i>Response</i>	
<p>The pixelServiceTime specifies the time after midnight of each day at which the FRIST pixel service will occur. Only after that the value of the pixelServiceFrequency becomes relevant.</p> <p>Additionally, the WG added wording for the vmsPixelServiceFrequency: "A value of zero indicates continuous pixel service from the value indicated in vmsPixelServiceTime to the epoch of midnight. A value of 1440 indicates one pixel service in a 24-hour period."</p> <p>The Description of the vmsPixelServiceDuration was amended to read:" The number of seconds to perform pixel service on a complete sign. If the vmsPixelServiceDuration expires a pixel service routine, that routine shall be completed before stopping or restarting. A value of zero disables the pixel service."</p>	

<i>Page</i>	<i>Paragraph</i>
2-18	dmsMessageRunTimePriority
<i>Comment</i>	
<p>What is the dmsMessageRuntimePriority of the current message (currentBuffer)? Is it the priority of the stored message that was the table source of the current message?</p>	
<i>Response</i>	
<p>The run time priority value will be copied from the dmsMsgRunTimePriority object from the row identified by dmsMsgTableSource. The amendment clarifies that the row referenced by the dmsMsgTableSource shall be the schedule row when the schedule is active. For example, if the sign is currently displaying the EndDurationMessage, the run time priority of the currentBuffer shall be the same as the run time priority defined for the message referenced by the End Duration Message.</p>	

<i>Page</i>	<i>Paragraph</i>
2-19	2.6.1.1.2
<i>Comment</i>	
<p>- a 16-bit bitmap might be better than an enumeration for the dmsValidateMessageError object</p>	
<i>Response</i>	
<p>All of these failures in the dmsValidateMessageError object can be true at the same time but the first value discovered will be the only error that is set. This is partially due to the complex logic of fully verifying a message (and thus being able to set various bits) when the message contains errors.</p>	

Page	Paragraph
2-19	Validate Message
Comment	
The dmsValidateMessageError items beacons and pixelService should be better defined. What is a pixelService error in terms of validating a message? Is the error the fact that pixelService was set but the sign does not do pixel servicing?	
Response	
The WG concluded that the 'pixelService' and 'beacons' enumerations of dmsValidateMessageError don't really make sense; thus, a statement will be added to the DESCRIPTION field indicating that the use of this value is strongly discouraged.	

Page	Paragraph
2-2	MessageIDCode and MessageActivationCode
Comment	
In the description for MessageIDCode and MessageActivationCode, it says "In both cases, bit 0 is the most significant bit (msb) of the most significant byte (MSB)"	
<p>This statement refers to the different fields in the MessageIDCode and MessageActivationCode structures. At first glance, I interpreted this to mean all the bits in all the fields are in reverse order. A single byte field would have bit0 in the highest position so all 8 bits must be reversed back to the proper order before any type of operation.</p> <p>I read it again and came up with a different interpretation that the bit numbers only refer to the field position in the octet string. For example, bit 0-15 is the "Duration" field of MessageActivationCode and all this means is that this field occupies the first 2 bytes (two MSB) of the 12-byte octet string. The value of the field is NOT actually stored with all the bits in reversed order. The two bytes are of course in Big-Endian format.</p> <p>Am I correct in this second interpretation? If there is a future amendment to 3.6, could this be explained more clearly?</p>	
Response	
These definitions will be changed to ASN.1 structures that are encoded according to OER. This change will include a new reference to OER (NTCIP 1102). We will also add an example in the informative annex.	

Page	Paragraph
2-2	MessageActivationCode
Comment	
The definition of Duration should be modified to indicate that the unit is minutes.	
Response	
The duration shall be in minutes. We will consider a seconds based activation object in the future.	

<i>Page</i>	<i>Paragraph</i>
2-2	2.1 Message Activation Code
<i>Comment</i>	
- The MessageActivationCode should indicate the units for duration	
<i>Response</i>	
The units for the MessageActivationCode shall be minutes.	

<i>Page</i>	<i>Paragraph</i>
2-2, 2-5	Sign Access, Technology
<i>Comment</i>	
- SignAccess and SignTechnology should explicitly state the meaning of a '0' and '1'	
<i>Response</i>	
A clarification regarding the meaning of setting/not setting a bit will be added to these and other object definitions within the standard.	

<i>Page</i>	<i>Paragraph</i>
2-20	dmsMessageTimeRemaining
<i>Comment</i>	
Should setting this object to zero '0' blank the display? If so, should we clarify the definition to more clearly indicate this.	
<i>Response</i>	
A system can shorten or lengthen the remaining duration of the currently displayed message by setting the dmsMessageTimeRemaining parameter; writing to zero shall cause the sign to immedietly display the EndDurationMessage.	

<i>Page</i>	<i>Paragraph</i>
2-20	2.7.1.1.1.4
<i>Comment</i>	
- There is a question as to what the read-write meaning of messageTimeRemaining means.	
<i>Response</i>	
A system can shorten or lengthen the remaining duration of the currently displayed message; writing to zero shall cause the sign to immedietly display the EndDurationMessage.	

<i>Page</i>	<i>Paragraph</i>
2-21	2.7.1.1.1.3
<i>Comment</i>	
There should be a standard way to blank the sign (and turn off lamps).	
<i>Response</i>	
The Amendment to the standard has added a new dmsMessageMemoryType 'blank (7)' in order to provide a standard mechanism to blank the sign. The dmsMessageNumber for this memory type shall be reflective of the RunTime Priority and shall be between 1 and 255, inclusive. The CRC for this memory type shall be 0x00 00. The activate priority for any MessageActivationCode using this type of memory shall be used as the actual activation priority. The dmsMessageMultiString shall be an octet string of length 0.	

<i>Page</i>	<i>Paragraph</i>
2-21	2.7.1.1.1.3
<i>Comment</i>	
We need a standard way to blank the sign	
<i>Response</i>	
The Amendment to the standard has added a new dmsMessageMemoryType 'blank (7)' in order to provide a standard mechanism to blank the sign. The dmsMessageNumber for this memory type shall be reflective of the RunTime Priority and shall be between 1 and 255, inclusive. The CRC for this memory type shall be 0x00 00. The activate priority for any MessageActivationCode using this type of memory shall be used as the actual activation priority. The dmsMessageMultiString shall be an octet string of length 0.	

<i>Page</i>	<i>Paragraph</i>
2-21	dmsMessageActivationCode
<i>Comment</i>	
What happens to the sign display if the controller receives an invalid dmsActivationCode? Does it continue to display the same message or does it blank?	
<i>Response</i>	
If the sign receives an invalid dmsActivationCode, the message active prior to activation should remain displayed, according to all of the other rules such as message duration. This has been clarified in the standard.	

<i>Page</i>	<i>Paragraph</i>
2-21	dmsActivateMessage
<i>Comment</i>	
What should one get upon reading the dmsActivateMessage?	
<i>Response</i>	
A get on dmsActivateMessage should return the value for the last message that was successfully implemented; if this message was activated by a message ID code (such as EndDurationMessage), the duration will indicate 65535 (infinite), Activate priority of the associated code (e.g., 200 until we get a new object), and Source address of the sign.	

<i>Page</i>	<i>Paragraph</i>
2-21	dmsActivateMessage
<i>Comment</i>	
Is it OK to put dmsActivateMessage in same SNMP packet as the sending of the message table data and validateReq command?	
<i>Response</i>	
No. SNMP requires all objects set in the same data packet to be implemented as if they were set simultaneously; thus the suggested coupling of a activate command with table data and verification would result in unpredictable operation on signs and should most properly be rejected by signs.	

<i>Page</i>	<i>Paragraph</i>
2-21	dmsActivateMessage
<i>Comment</i>	
What should happen if a message is currently up and an activate message fails, clear the sign or leave the existing untouched?	
<i>Response</i>	
If a message activation fails due to an error in the command, the previous message should continue running as if the request was never received. This has been clarified in the standard.	

<i>Page</i>	<i>Paragraph</i>
2-21	2.7.1.1.1.3
<i>Comment</i>	
What is the preferred way to blank a sign? Should permanent message #1 be required to be blank?	
<i>Response</i>	
The Amendment to the standard has added a new dmsMessageMemoryType 'blank (7)' in order to provide a standard mechanism to blank the sign. The dmsMessageNumber for this memory type shall be reflective of the RunTime Priority and shall be between 1 and 255, inclusive. The CRC for this memory type shall be 0x00 00. The activate priority for any MessageActivationCode using this type of memory shall be used as the actual activation priority. The dmsMessageMultiString shall be an octet string of length 0.	

<i>Page</i>	<i>Paragraph</i>
2-21	2.7.1.1.1.3
<i>Comment</i>	
What should the activate priority be when a message is activated from the controller front panel or from a schedule? How about the default messages, such as CommunicationsLoss, PowerLossRecovery, etc?	
<i>Response</i>	
The Amendment to the standard has clarified the intended operation. The schedule is activated by calling dmsMessageMemoryType schedule(6) and message number 1 and dmsMessageCRC is 0x00 00. The dmsRunTimePriority.6.1 shall be set by central. The dmsMessageMultiString.6.1 dmsMessageBeacon and dmsMessagePixelService are copied from the message called by the most recently called action. The activation priority of these items shall be 255.	

<i>Page</i>	<i>Paragraph</i>
2-21	Priority
<i>Comment</i>	
There is an ambiguity between the definition of how the activate priority works with current message priority. In the definitions at the front of the standard it says the activation priority must be greater than or equal to the current message runtime priority for the new message to be allowed. Under the definition of the MessageActivationCode, it says the activation priority must be greater than the current message runtime priority.	
<i>Response</i>	
The amendment has changed the wording of the prioritization code to Greater than or equal to more clearly state the intent.	

<i>Page</i>	<i>Paragraph</i>
2-21	2.7.1.1.1.3
<i>Comment</i>	
We need to clarify what happens if an activation code is received with a duration of zero (0).	
<i>Response</i>	
This will be clarified in version 2; the comment was received after the completion of the amendment.	

<i>Page</i>	<i>Paragraph</i>
2-22	Default Messages
<i>Comment</i>	
<p>How can default messages (e.g., dmsPowerLossMessage) be configured to indicate no change to the display? Presumably this is achieved by setting the message number to the "currentBuffer"? But there is no statement of this within the standard. Is this allowed? My suggestion is to Amend the standard to clearly indicate that such objects are allowed to reference the currentBuffer message.</p> <p>Also, how do you revert to the previous message after conditions are restored (e.g., after power restoration)?</p>	
<i>Response</i>	
The near-term solution is that when you write the activate message object, you should also write to the subject default message. You can also set the default message object to the currentBuffer with a CRC of 0x0000, but this does not necessarily solve all issues, depending upon desired operation. We may revisit this issue in version 2.	

<i>Page</i>	<i>Paragraph</i>
2-22	Default Messages
<i>Comment</i>	
Clarify descriptions. For example, describe the messages displayed after power recovery, and explain how to point to the previous message displayed if another message is displayed when power is lost.	
<i>Response</i>	
The near-term solution is that when you write the activate message object, you should also write to the subject default message. We may revisit this issue in version 2 work.	

<i>Page</i>	<i>Paragraph</i>
2-22	2.7.1.1.1.8
<i>Comment</i>	
<p>I have seen several specifications recently that require the sign to run schedules as the default condition. If no message has been commanded to display on the sign, the sign will display messages based on its schedules. When a message is commanded to be displayed on the sign, the schedules will stop running so the commanded message will not be superseded. Once the commanded message is removed from the sign, schedules will once again be run on the sign. This seems to imply a need for a "reconciliation" feature: having the sign determine what message it would have been running had it not been interrupted by the commanded message display.</p>	
<i>Response</i>	
<p>The Amendment to the standard has clarified the intended operation. The schedule is activated by calling dmsMessageMemoryType schedule(6) and message number 1 and dmsMessageCRC is 0x00 00. The dmsRunTimePriority.6.1 shall be set by central. The dmsMessageMultiString.6.1 dmsMessageBeacon and dmsMessagePixelService are copied from the message called by the most recently called action. The activation priority of these items shall be 255.</p>	

<i>Page</i>	<i>Paragraph</i>
2-22	2.7.1.1.1.8
<i>Comment</i>	
<p>It appears that one of the goals of various default message parameters (like dmsShortPowerRecoveryMessage) is to allow the current message (current at time of power loss) to be put back up. This would be done by specifying the currentBuffer message #1. At the time of specifying this MessageIDCode, however, we do not know the CRC of the message.</p>	
<i>Response</i>	
<p>In order to achieve this operation, the default message(s) should be set to the currentBuffer message with a CRC of 0x0000 when you write the activate message object. This operation will be further reviewed in version 2.</p>	

<i>Page</i>	<i>Paragraph</i>
2-23	dmsActivateMsgError
<i>Comment</i>	
<p>If a message's status is "error", should an attempt to activate the message result in a dmsActivateMsgError code of "underValidation"? Seems like there should be a code for the error case (validation completed but failed).</p>	
<i>Response</i>	
<p>The amendment changes the value name from 'underValidation' to 'messageStatus' to account for this.</p>	

<i>Page</i>	<i>Paragraph</i>
2-25	2-8 Illumination
<i>Comment</i>	
<p>The DMS standard only provides support for a single photocell, yet most sites require multiple sensors. One vendor used proprietary objects, the other used the AuxIO ports for this information. The result is this feature is not interoperable.</p> <p>It is recommended that the standard support multiple sensors.</p>	
<i>Response</i>	
<p>The amendment clarifies that the photocell level objects deal with a virtual photocell level, i.e., it is a level that is algorithmically determined from one or more photocells and is the value used for calculations dealing with the brightness table. The algorithm used to derive the virtual level from the actual photocell readings is manufacturer specific to accommodate various hardware needs.</p> <p>We may consider adding additional objects in the future if there is demonstrated need.</p>	

<i>Page</i>	<i>Paragraph</i>
2-25	2.8 Brightness
<i>Comment</i>	
<p>The design of the brightness table and levels do not ensure an appropriate degree of interoperability. For example, one vendor may have 20 brightness levels while another uses 255. A central system might try to set to max brightness (20) and only get 10% illumination on the other manufacturers sign.</p> <p>In addition, it is unclear how the brightness table should be used.</p>	
<i>Response</i>	
<p>One cannot send common values to every sign and sign type expecting the same brightness. However, the user interface (GUI) could address this issue. The WG will develop an informational annex to explain the 'brightness/illumination' functionality.</p> <p>The WG will consider a percentage-based object for manual control for the next version (version 2) of the standard.</p>	

<i>Page</i>	<i>Paragraph</i>
2-29	Action Table
<i>Comment</i>	
<p>within the actionTable, the dmsActionMsgCode object doesn't have an end of duration associated with the SYNTAX of "MessageIDCode".</p> <p>=> meaning that a message displayed for peak-hour traffic might be displayed the entire night, if no BLANK message is sent to replace the scheduler-activated message.</p> <p>=> possible solution: change SYNTAX of dmsActionMsgCode object to "messageActivationCode"</p>	
<i>Response</i>	
<p>The design of the scheduler requires the user to explicitly configure a blanking action at the desired time.</p>	

<i>Page</i>	<i>Paragraph</i>
2-29	2.9
<i>Comment</i>	
<p>One question which has come up in our implementations involves scheduling. We have been unable to find a mechanism in NTCIP to enable and disable scheduled events. The upshot of this is that the user must download the complete schedule to start it, and then delete the schedule to run an individual message. I would suggest an enable/disable scheduled events object would simplify this greatly. I am particularly looking at this problem from the DMS perspective, so there may be additional ramifications. I would at least like to place this one on the table.</p>	
<i>Response</i>	
<p>The amendment clarifies this issue. You can activate the schedule for a defined duration by setting the duration in the activation code.</p>	

<i>Page</i>	<i>Paragraph</i>
2-29	2.10
<i>Comment</i>	
<p>- consider renaming the maxAuxIODigital and maxAuxIOAnalog objects</p>	
<i>Response</i>	
<p>The WG considered this, but rejected the suggestion of renaming the objects. Additional information would be needed from the commentor for the WG to reconsider this decision. In addition, these objects will be moved to the Global Objects standard in the next version of NTCIP.</p>	

<i>Page</i>	<i>Paragraph</i>
2-29	Scheduling
<i>Comment</i>	
<p>One item that a customer brought up with me concerning schedules was the necessity to run multiple schedule entries at the same time. The scenario we ran into was in running a weekday/weekend kind of schedule, but being required to start it and stop it on week boundaries in the middle of a month. The best I could work out, it would require six schedule entries: two for the first month, two for the complete months, and two for the last month. This got me to thinking about enabling/disabling the schedule entries. A method of providing enable/disable would be to do it by schedule number. In the example given, all of the entries would be given the same schedule number, and could be enabled and disabled together. It would also be possible to mix and match schedule entries.</p>	
<i>Response</i>	
<p>The DMS WG agreed that this feature would perhaps be nice, but it is the topic for the GO WG and has been forwarded to them for consideration.</p>	

<i>Page</i>	<i>Paragraph</i>
2-29	Action Table
<i>Comment</i>	
Action Table is not well defined. The definitions imply that the table cannot be sparse—is that true? If so, there is no way to determine how many entries can be stored (numActionTableEntries is the “number of rows that are stored” and there is no max is specified in the spec). Also, how does one delete entries from the table (and therefore reduce numActionTableEntries)?	
<i>Response</i>	
The action table has a fixed number of rows that does not change during the operation of the sign. The number (which is also the maximum) is recorded in the numActionTableEntries object. We will indicate that the DEFVAL of dmsActionMsgCode shall be 0x00 00 00 00 00. And you can also disable a row by setting this object to this value.	

<i>Page</i>	<i>Paragraph</i>
2-29	2.9
<i>Comment</i>	
<p>Discussion: Both vendors expressed great displeasure with the Scheduler object. They stated that there is a problem with the override of a scheduler task without clearing the scheduler table. There is no global mechanism to enable or disable the scheduler. Both vendors created custom objects to overcome this issue.</p> <p>Upon analysis of the core functions captured data, the standard had deviations related to DMS scheduler functionality. There were 138 discrepancies out of a total of 3,049 data packets that were analyzed.</p> <p>Discussion: As shown above, the scheduling action object is addressed under standard 1203 for some objects, and the rest are addressed under 1201 for global objects. During the interview process, both vendors identified that the scheduler related portions of the NTCIP - Object Definitions for Dynamic Message Signs (1203) standard were deficient. Both vendors sought additional guidance from NEMA related to this issue. The standards, though addressing most of the objects, do not define an object for enabling or disabling the scheduler. The solution to address the lack of this object and remain compliant with NTCIP standards was to create a custom object. See IC-2 for interview comments.</p> <p>Recommendations: The standards (both 1201 & 1203) need to be enhanced to include an object to enable and disable the scheduler.</p> <ol style="list-style-type: none"> 1. A companion document that could serve as a users guide could be developed to assist the vendors in implementing the scheduler objects. 	
<i>Response</i>	
The amendment clarifies the intended operation. You can activate (deactivate) the schedule by setting the activation code to point (not point) to the schedule row of the message table. The Global Object Working Group considered comments regarding the simplification of the scheduler table but has been unable to identify a simplified scheme that meets the base requirements. This issue will be further clarified in a future version of the standard once reformatting work is complete.	

<i>Page</i>	<i>Paragraph</i>
2-30	Auxiliary IO
<i>Comment</i>	
Does not make sense to me that the auxiliary IO parameters, other than auxIOValue, should always be required to be read-write (read-only should be sufficient in most implementations).	
<i>Response</i>	
The WG discussed this suggestion and determined that the standard already allows an implementation to return a GenErr in those cases where write operations are not appropriate. The objects need to remain read-write in order to allow for all possible implementations rather than just most.	

<i>Page</i>	<i>Paragraph</i>
2-31	2.11
<i>Comment</i>	
- The purpose of the statMultiField objects is unclear	
<i>Response</i>	
The Multi Field Table includes field values of currently displayed fields in sequential order. Order is based from beginning of MULTI text to end corresponding to table entry 1 through the last entry. This will be clarified in the standard.	

<i>Page</i>	<i>Paragraph</i>
2-33	2.11.1.1.1.5
<i>Comment</i>	
- The purpose of the watchdog failure count object is unclear	
<i>Response</i>	
watchdogFailureCount indicates a count of sign controller resets. Note: Some vendors may or may not be able to differentiate between a power failure and a watchdog failure. This will be clarified in the standard.	

<i>Page</i>	<i>Paragraph</i>
2-33	2.11.2
<i>Comment</i>	
- The pixel failure table should be cleared when setting the pixelTestActivation object to either test or clearTable	
<i>Response</i>	
The requirement to clear the pixel failure table under these conditions is already defined like this in the standard. However, wording is being added to the pixelTestActivation to specifically state this.	

<i>Page</i>	<i>Paragraph</i>
2-34	Status
<i>Comment</i>	
The dmsStatDoorOpen value is a set of bits but the spec does not specify ordering.	
<i>Response</i>	
The ordering of bits in the dmsStatDoorOpen object is manufacturer specific since the layout of doors is not standardized within the standards.	

<i>Page</i>	<i>Paragraph</i>
2-35	2.11.2.1.1.1 ShortErrorStatus & 2.11.2.1.1.10
<i>Comment</i>	
I found no provision in the TS3.6 to clear the error bits in these two objects. Am I wrong in assuming that the bits should be cleared as soon as the errors are cleared. But if this is the case, I will never be able to catch some of the errors such as communication errors (unless I configure the event log to capture these errors). I am resorting to use a private object to clear these error bits.	
<i>Response</i>	
The error status bits should be cleared as soon as the errors are cleared (i.e., no longer exist). Short error status is cleared by other objects. Utilizing a proprietary object to clear this object is actually a violation of the standard.	
Errors that cannot be caught could be logged in the reporting mechanism of the Global Objects.	
Further commentor information needs to be provided, if this response is not sufficient.	

<i>Page</i>	<i>Paragraph</i>
2-35	Pixel Test
<i>Comment</i>	
The pixelFailureTableNumRows is not very convenient since the pixel failure table has a major key of pixelFailureDetectionType. It would make more sense to have a count for each detection type.	
<i>Response</i>	
It does seem to make sense to have a count of pixel failures by detection type. The WG will consider this in the next version of this standard.	

<i>Page</i>	<i>Paragraph</i>
2-35	Pixel Errors
<i>Comment</i>	
Clarify difference between pixel errors and lamp errors. I assume a bulb sign would report errors under pixel errors and that lamp errors is for signs that use lamps to illuminate multiple pixels or to illuminate an entire line (ie. illuminated drum) or to illuminate a CMS or blankout sign.	
<i>Response</i>	
Yes, a bulb sign would report errors under pixel errors and the lamp error is used for signs using a single lamp for multiple pixels/lines. However, the issue relates to sign technology, not sign type; thus, it may or may not apply to a CMS or blank-out sign, depending upon the technology used.	
We have modified the lamp, pixel and bulb terms in the Terms section to clarify this point.	

<i>Page</i>	<i>Paragraph</i>
2-35	2.11.2.1.1.3.5
<i>Comment</i>	
how to differentiate between mechanical and electrical errors? How to express that both coil of the pixel are broken, meaning that the sign is neither electrically stuck on and stuck off?	
<i>Response</i>	
The ability to distinguish between mechanical and electrical errors will be discussed as a part of the version 2 work.	

<i>Page</i>	<i>Paragraph</i>
2-36	Pixel Test
<i>Comment</i>	
Pixel error is whole-pixel only—no support for half-pixel errors.	
<i>Response</i>	
The WG discussed this issue, but decided that this issue will not be addressed due to the following reasons: - having higher resolution with smaller and smaller pixels will not severely affect the legibility - not many clients will go out to the field to replace a pixel that has half-pixel error.	
However, an implementation may want to report a half-pixel error as a pixel failure.	

<i>Page</i>	<i>Paragraph</i>
2-36	Pixel Test
<i>Comment</i>	
I assume the pixelFailureIndex parameter should be the index within a particular detection type. The spec should make it clear. It is somewhat ambiguous since there is only one count of rows, pixelFailureTableNumRows.	
<i>Response</i>	
Clarification regarding the pixelFailureIndex parameter has been added.	

<i>Page</i>	<i>Paragraph</i>
2-36	Pixel Test
<i>Comment</i>	
There should be a means to limit max number of pixel errors (per detection type).	
<i>Response</i>	
The WG decided to address the ability to place a maximum on the number of pixel errors per detection type in a future version.	

<i>Page</i>	<i>Paragraph</i>
2-36	Pixel Test
<i>Comment</i>	
Actually I think the entire pixel error reporting system is deficient. The method of a table row for each pixel creates tremendous communication times (ie. unacceptably long) if a large portion of the sign has errors (say an entire line goes out). We need a more compact way to report. The messageDisplay errors are also limited in that the page is not known (as mentioned before) and communication times could also become very long. In addition there should be a way to provide a positive feedback of all pixel states in a message rather than just the errors to be overlaid on a computed message display.	
<i>Response</i>	
The Working Group will work on a more compact way to report errors. The resolution will be performed in conjunction with the graphics transmission in version 2.	

<i>Page</i>	<i>Paragraph</i>
2-37	2.11.3 Power Supplies
<i>Comment</i>	
<p>The standard provides for a single power supply on a sign. DMSs have multiple power supplies and these are not addressed. Solutions by different vendors vary (e.g., proprietary objects or using the AuxIO inputs). It also seems that the types of power supplies are not the most common ones and limited monitoring capabilities exist.</p> <p>The standard should be modified to include coverage of one or more power supplies.</p>	
<i>Response</i>	
<p>There are currently no objects for power supplies (i.e., converters of voltage) only objects for power source for the sign (i.e., how the sign as a whole receives power). We will consider adding objects for multiple ~power supplies~ in version 2.</p>	

<i>Page</i>	<i>Paragraph</i>
2-37	2.11.3.1.1.2
<i>Comment</i>	
<p>The low fuel threshold object should have a range of 0..100 instead of 0..255.</p>	
<i>Response</i>	
<p>Yes, the range for the fuelThreshold object will be changed in the amendment in order to reflect the intended range.</p>	

<i>Page</i>	<i>Paragraph</i>
2-38	Fan Failure
<i>Comment</i>	
<p>The definition of "fan failure" should be reworded to include any ventilation error (ie. airflow restricted due to clogged filters).</p>	
<i>Response</i>	
<p>The description of fanFailures has been clarified to clearly include ventilation errors.</p>	

<i>Page</i>	<i>Paragraph</i>
2-40	2.11.4.1.1.6 maxSignHousingTemp
<i>Comment</i>	
<p>The range should probably be -128..127 instead of 0..255</p>	
<i>Response</i>	
<p>Yes, the range for maxSignHousingTemp will be changed in the amendment to the correct value.</p>	

<i>Page</i>	<i>Paragraph</i>
2-7	Font Definition
<i>Comment</i>	
When a font is being modified, the entire font should be disabled to ensure that a message is not displayed with a partial font. Thus, a status column is needed.	
<i>Response</i>	
Wording has been added to the fontTable object indicating that modifying a font while it is in use 'will yield unpredictable results. For predictable results, do not change characters of a font while that font is being displayed.'	

<i>Page</i>	<i>Paragraph</i>
2-8	2.4.1.1.1.2.4
<i>Comment</i>	
- There is some ambiguity as to what invalidating a row means within the font table	
<i>Response</i>	
Attempting to utilize an invalid font will generate a "fontNotDefined" error for dmsMultiSyntaxError object. Compare CRC's to verify the font currently stored on the display is correct. This will be clarified in the standard.	

<i>Page</i>	<i>Paragraph</i>
2-9	fontVersionId
<i>Comment</i>	
There is a concern about the amount of processing required in the DMS if an entire font is downloaded (e.g., a new CRC is calculated over the entire font each time a new character is downloaded).	
<i>Response</i>	
The processing requirements for the font download process will be considered in the development of the version 2 document.	

<i>Page</i>	<i>Paragraph</i>
2-9	Font Version ID
<i>Comment</i>	
Replace PER with OER to reflect change in base standard. Also due a search through rest of document to ensure proper referencing, esp. in the Reference and Acronym sections.	
<i>Response</i>	
Yes, the amendment updates the 'PER' text to 'OER' in order to reflect the changes in the base standards.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI
<i>Comment</i>	
Is a null character (0x00) allowed in a multi string? I think not, but want to be sure.	
<i>Response</i>	
The null character is not allowed in a MULTI String because the octet is supposed to reference an entry in the character table, and the character table starts with row one. This is reflected in the amendment through the addition of text.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI - field
<i>Comment</i>	
The field tag does not specify how to determine the field width in pixels when using a proportional font. For example, how should the colon in the time field be counted width-wise (assuming it is a proportional font where the colon is smaller width than most digits).	
<i>Response</i>	
The amendment refines the field table to clarify the intent for each field type.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI - field
<i>Comment</i>	
If proportional characters used in a field do not fill the field width, how should they be horizontally aligned?	
<i>Response</i>	
The amendment refines the field table to clarify the intent for each field type.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI - field
<i>Comment</i>	
If a specified field width is too small, what should be done? Asterisk fill? Blank fill? Error (none is currently defined for this)?	
<i>Response</i>	
The amendment refines the field table to clarify the intent for each field type.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI - field
<i>Comment</i>	
Should fields use zero-fill or not?	
<i>Response</i>	
The amendment refines the field table to clarify the intent for each field type.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI
<i>Comment</i>	
When spec says something does not have to be implemented in the sign, but that the “Controller must recognize the tag and take appropriate action”, what is the appropriate action? Does it depend on the tag, ie. OK to ignore a color that can't be done but an error if a conflicting justification?	
<i>Response</i>	
If the controller is responsible for 'recognizing a tag and taking appropriate action,' but is not required to implement the functionality; the 'action' shall be either to implement the defined functionality or to generate a dmsMultiSyntaxError with a value of unsupportedTag. This will be clarified in the amendment.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI - field
<i>Comment</i>	
Field #7 is day of week and is 3 characters wide by default. Is this to be “SUN”, “MON”, etc. in all upper case?	
<i>Response</i>	
The amendment refines the field table to clarify the intent for each field type.	

<i>Page</i>	<i>Paragraph</i>
3-1	MULTI
<i>Comment</i>	
The pixel failure status byte forces you to say the pixel is stuck on or off (bit 0). The color error bit does not fit well with that. For example what if a color fails to work, but the pixel is neither stuck on or stuck off?	
<i>Response</i>	
We will review the design of the pixel error object with a broader review of the entire pixel test operation in version 2. This review should also explain what conformance groups are appropriate for what technologies.	

<i>Page</i>	<i>Paragraph</i>
3-12	MULTI - spacing
<i>Comment</i>	
What character or line spacing should be used if averaging and the result is not a whole number of pixel columns?	
<i>Response</i>	
The spacing should be rounded to the next full number (rounding UP), if this condition occurs. This will be clarified in the standard.	

<i>Page</i>	<i>Paragraph</i>
3-2	Table
<i>Comment</i>	
The SC tag should indicate the companion closing flag [/sc]	
<i>Response</i>	
The amendment adds the intended '/sc' closing flag to the table.	

<i>Page</i>	<i>Paragraph</i>
3-3	MULTI - color
<i>Comment</i>	
Lots of questions about background color. a. Does it apply to inter-character spacing on line and full matrix? b. Does it apply to area above or below character when multiple height characters are on the same line? c. Does it apply to inter-line spacing on a full-matrix? d. Does it apply to blanked areas by a flash tag? e. Does it apply to empty areas inside a moving text window (ie. if bitmap is smaller than window)? f. Does it apply to areas on the sign that do not have characters (ie. leading and trailing area on a center justified line)?	
<i>Response</i>	
a. Yes, background color does apply to inter-character spacing on line and full matrix signs. b. Yes, background color does apply to above and below character space on lines with multiple size characters. c. Yes, background color does apply to inter-line spacing. d. Yes, background color does apply to blanked areas due to a flashing tag. e. Yes, background color does apply to the empty areas inside a moving text box. f. Yes, background color does apply to areas on the sign that do not have characters. These issues are addressd in the amendment.	

<i>Page</i>	<i>Paragraph</i>
3-4	3.4.3
<i>Comment</i>	
- The temperature field in MULTI should explicitly reference ambient temperature	
<i>Response</i>	
Yes, the intent is 'ambient' temperature. A reference regarding this issue will be added to the FIELD table within Section 3 of the Standard.	

<i>Page</i>	<i>Paragraph</i>
3-6	3.4.5 Font
<i>Comment</i>	
References to fontDefinition should be replaced by font	
<i>Response</i>	
Yes, the amendment changes this erroneous reference to fontDefinition.	

<i>Page</i>	<i>Paragraph</i>
3-6	MULTI fonts
<i>Comment</i>	
How should various height fonts on a single line be vertically aligned?	
<i>Response</i>	
<p>The WG decided that fonts with various heights on one line should be 'bottom alligned'. While it would be more difficult from a software view, the legibility will be much better this way.</p> <p>For the next version of this standard (version 2), another tag and another default object (with a default of 'bottom alligned') will be added to the standard to allow a choice.</p>	

<i>Page</i>	<i>Paragraph</i>
3-7	MULTI - line justification
<i>Comment</i>	
There probably should be more specifics on how to handle line justifications that seem to conflict (ie. a left justify region followed by a right justified followed by another left justify).	
<i>Response</i>	
<p>The line justification tag can only be used in logical order (left, center, right) on a line; otherwise, a 'tagConflict' error (see dmsMultiSyntaxError object) should be issued.</p> <p>If an overlap of text occurs due to different line justification tags, an 'textTooBig' error will be issued.</p> <p>No other line justification value can be used on the same line, if the 'full' value is used. This clarification shall be added to the standard.</p>	

<i>Page</i>	<i>Paragraph</i>
3-8	MULTI - page justification
<i>Comment</i>	
How should page justification be interpreted if there is more than one in a page. For example, should it be possible to top justify the first line and then bottom justify the second line?	
<i>Response</i>	
<p>The page justification tag can only be used in logical order (top, middle, bottom); otherwise a 'tagConflict' error will be issued.</p> <p>If an overlap of the text occurs due to different page justification tags, a 'textTooBig' error will be issued. This will be clarified in the standard.</p>	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
After a field or move tag, should character spacing be applied before the following character?	
<i>Response</i>	
<p>The amendment clarifies that 'character spacing' will be applied automatically for two MULTI tags: 'Field', 'Moving Text':</p> <ol style="list-style-type: none"> a.) before the FIELD or MOVING TEXT, if another character is displayed before this field, and/or b.) after the FIELD or MOVING TEXT, if another character is displayed after this field. 	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
What should be done if a move tag width is not a multiple of a module width on a character matrix sign?	
<i>Response</i>	
It is recommended that the sign should return an unsupportedTagValue error if a move tag width is not a multiple of a character width on a character matrix sign. This has been clarified in the amendment.	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
In a circular move, should character spacing be applied when repeating the bitmap?	
<i>Response</i>	
Yes, character spacing should be added when repeating the bitmap on a circular move. Clarification was added in the amendment.	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
If the last shift of a linear shift would move the last column of the bitmap beyond the last column of the window, what should be filled in the gap? Or should the last shift be different so that the last column of the bitmap exactly occupies the last column of the window?	
<i>Response</i>	
The following clarification applies to 'linear' only, and appropriate wording is added in the amendment: The last jump (indicated by the S-subtag within the MOVING TEXT tag) shall put the last pixel row into the last display column of the field. This shall apply to both right and left scrolling.	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
What should be done in moving text tag if the bitmap produced by the character string is smaller than the window?	
<i>Response</i>	
<p>The resolution of the problem of text being too small for a window needs to be separated by 'circular' and 'linear' scrolling.</p> <p>Two examples will be added to the standard explaining how smaller text is to be handled.</p> <p>'Circular': text will just be repeated as stated in the standard.</p> <p>'Linear': don't scroll at all, just display the text. If 'linear' scrolling is desired for smaller text, an appropriate number of spaces need to be added before and after the text.</p>	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
What should it look like if a moving tag on a character matrix specifies a pixel shift that is not a multiple of a module width?	
<i>Response</i>	
It is recommended that the sign should return an unsupportedTagValue error if a move tag width is not a multiple of a character width on a character matrix sign. This has been clarified in the amendment.	

<i>Page</i>	<i>Paragraph</i>
3-9	MULTI - moving text
<i>Comment</i>	
Would be nice to allow the hex character tag inside of the text in a moving text tag. This would allow graphic symbols to be inserted in moving text.	
<i>Response</i>	
<p>This comment actually targets a larger issue. The WG discussed the issue and concluded that 'Nesting' is NOT prohibited except for within the same tags, i.e., having a FIELD tag within a FIELD tag is prohibited.</p> <p>Appropriate modifications were made to Section 3.3 Rules within the standard (added another rule and modified statement in Section 3.2)</p>	

<i>Page</i>	<i>Paragraph</i>
4-3	4.6
<i>Comment</i>	
dmsMessageMsgStatus should be dmsMessageStatus	
<i>Response</i>	
Yes, the amendment changes the reference for dmsMessageMsgStatus.	

<i>Page</i>	<i>Paragraph</i>
4-5	4.11
<i>Comment</i>	
Check object names	
<i>Response</i>	
Yes, the amendment has corrected these items. (BrightLevelStatus, BrightnessValuesError)	

<i>Page</i>	<i>Paragraph</i>
4-6	4.13 Auxiliary
<i>Comment</i>	
change aux to auxIO	
<i>Response</i>	
Yes, the amendment changes this erroneous reference to aux.	

<i>Page</i>	<i>Paragraph</i>
General	General
<i>Comment</i>	
External references should be used when appropriate and clearly explained. For example, data structure encodings should be clearly spelled out.	
<i>Response</i>	
Encodings will not be spelled out in the standard based on the fact that encoding is an application profile issue. In other words, the correct encoding is dependent upon which application protocol is being used; defining encoding within this standard would likely confuse the reader.	

<i>Page</i>	<i>Paragraph</i>
General	General
<i>Comment</i>	
Check for errors in ACCESS fields	
<i>Response</i>	
The ACCESS field values were already corrected in the final publication.	

<i>Page</i>	<i>Paragraph</i>
General	General
<i>Comment</i>	
- Consider adding objects for beacon service	
<i>Response</i>	
Beacon testing and status reporting will be considered for version 2.	

<i>Page</i>	<i>Paragraph</i>
General	General
<i>Comment</i>	
- Consider the addition of objects to support multiple fans, power supplies, and lamps as well as objects to indicate the number of objects, tables describing objects, and test objects	
<i>Response</i>	
These features will be considered for version 2 (including lamps and fans in excess of 32).	

<i>Page</i>	<i>Paragraph</i>
General	General
<i>Comment</i>	
Identify the core functions of the sign	
<i>Response</i>	
The standard only defines the mandatory/optional objects for a sign; the importance ('coreness') of objects is subjective based on project needs.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
How do we control gates?	
<i>Response</i>	
This feature will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
How do we control multiple displays with one controller? Especially lane control signs and or lane signs with message boards and/or gates. Also blank out signs	
<i>Response</i>	
This feature will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
Add capability to monitor status of multiple photocells	
<i>Response</i>	
This feature will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
Are there objects to deal with the operation of 'drum' signs which have been widely deployed. Their characteristic are different in that one selects a combination of 'presentations' but those are not variable. there are also some interesting feedback options for drum sign errors. Another issue is if the sign should include internal consistency checks - i.e. it may restrict the combinations possible?	
<i>Response</i>	
Features to support special operations of drum signs will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
Are any traps defined? Should signs support the standard SNMP traps? How should they be handled?	
<i>Response</i>	
No, there are no DMS-specific traps defined in the current standard; we will address the need for traps in version 2. It is currently manufacturer specific as to whether a device supports the SNMP standard traps, this issue will also be revisited for version 2 and may be also addressed by AP-STMF.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
Is anyone working on developing graphics for the NTCIP DMS standard?	
<i>Response</i>	
The ability to support graphics will be a major consideration in the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	Security Conformance Group
<i>Comment</i>	
The Security Conformance Group of NTCIP 1201 Amendment 1 should be mandatory for signs	
<i>Response</i>	
The requirement for Security objects will be addressed in the Version 2 update.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
There should be a conformance group for testing sign drivers.	
<i>Response</i>	
This feature will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
There need to be parameters to allow the identification of under/over current conditions on pixels.	
<i>Response</i>	
This feature will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
need an ability to establish a legibility threshold which is compared to the ratio of the number of pixels that actually achieve the commanded state divided by the number of pixels commanded to the requested state. If the ratio is less than the legibility threshold, the controller should be able to blank the sign and report the error condition. The test would only include those pixels that are contained in the character positions of the message text.	
<i>Response</i>	
This feature will be considered during the development of version 2.	

<i>Page</i>	<i>Paragraph</i>
New Feature	
<i>Comment</i>	
It would be nice to have some sort of monitoring capability to detect when a sign is about to exceed the manufacturer-specific temperature thresholds that will shut the sign off or dim the display in order to protect the display technology devices (I.e., read, LED). Also to know when the display has shut down / dimmed due to heat.	
<i>Response</i>	
This feature will be considered during the development of version 2.	