

National Type Evaluation Technical Committee (NTETC)
Software Sector Meeting
March 15 & 16, 2011
Doubletree Hotel – Annapolis, MD

MEETING AGENDA

	<u>Page</u>
i. Welcome/Introductions.....	3
ii. NTETC Software Sector Mission.....	3
iii. Report of Activity at 2011 NCWM Interim Meeting.....	3
iv. Report of Activity from International W&M Agencies.....	3

Carryover Items

1. Software Identification/Markings.....	4
2. Identification Of Certified Software.....	13
3. Software Protection/Security.....	16
4. Software Maintenance and Reconfiguration.....	20
5. NTEP Application for Software and Software-Based Devices.....	24
6. Training of Field Inspectors.....	25

New Items

7. Remote or Distributed Metrologically Significant Functionality.....	27
8. Next Meeting.....	27
<i>Appendix A: Report on 2011 Interim Meeting.....</i>	<i>28</i>
<i>Appendix B: Report on International W&M Activity.....</i>	<i>29</i>

National Type Evaluation Technical Committee (NTETC)
Software Sector Meeting
March 15 & 16, 2011
Doubletree Hotel – Annapolis, MD

Agenda Schedule

Tuesday, March 15, 2010

- 8:00 AM *Meeting Call to Order* (Co-Chairs)
i. Introductions and welcome to new Sector members
ii. Reiteration of NTETC Software Sector Mission (J. Truex)
8:30 AM *Status Reports*
iii. Report – 2011 NCWM Interim Meeting (Interim Attendees)
iv. Report – International Activity of Interest to Sector (A. Thompson – NIST WMD)
- 9:00 AM *Work session - Carryover Items*
1. Software Identification / Markings
10:00 AM *Break* (15 min.)
10:15 AM *Carryover Items (continued)*
Software Identification / Markings (cont.)
12:00 PM *Lunch Break* (1 hour)
1:00 PM – *Carryover Items (cont.)*
2. Identification of Certified Software
3:00 PM – *Break* (15 min.)
3:15 PM – *Carryover Items (cont.)*
3. Software Maintenance and Reconfiguration
5:00 PM – *Adjourn*

Wednesday March `6, 2010

- 8:00 AM *Continue Work Session - Carryover Items*
4. Software Protection / Security
10:00 AM *Break* (15 min.)
10:15 AM *Carryover Items (continued)*
5. NTEP Application for Software and Software-Based Devices
6. Training Of Field Inspectors
12:00 PM *Lunch Break* (1 hour)
1:00 PM *Work Session – New Items*
7. Next Meeting
3:00 PM – *Break* (15 min.)
3:15 PM – *Work Session*
**This time is reserved for revisiting items requiring additional attention
and any unscheduled items brought to the Sector for consideration.**
5:00 PM – *Adjourn*

Note: topic times are approximate and merely included as a rough guideline to aid in maintaining meeting pace; some issues will invariably involve more detailed discussion than others.

i. Welcome/Introductions

The Chair would like to welcome new individuals that have joined our Sector since the last meeting. Please welcome:

Rick Lydon, *Sick, Inc.* Chris Scott, *Gilbarco* Michael Kelley, *Ohio*

ii. National Type Evaluation Technical Committee / Software Sector - Mission

Mr. Jim Truex, NTEP Administrator, will reiterate/convey the mission of the Sector.

Source: NCWM Board of Directors

Background: In 2005, the Board of Directors established a National Type Evaluation Technical Committee (NTETC) Software Sector. A mission statement for the sector was developed at that time.

Mission of the Software Sector:

- Develop a clear understanding of the use of software in today's weighing and measuring instruments.
- Develop NIST Handbook 44 specifications and requirements, as needed, for software incorporated into weighing and measuring devices. This may include tools for field verification, security requirements, identification, etc.
- Develop NCWM Publication 14 checklist criteria, as needed, for the evaluation of software incorporated into weighing and measuring devices, including marking, security, metrologically significant functions, etc.
- Assist in the development of training guidelines for W&M officials in verifying software as compliant to applicable requirements and traceable to a NTEP Certificate. Training aids to educate manufacturers, designers, service technicians and end users may also be considered.

Recommendation: There should be an attempt to follow the four bullet items above in order from the top down when discussing agenda items. Focus should begin with any possible impact on NIST Handbook 44.

iii. Report of Activity at 2011 NCWM Interim Meeting

Attendees of the Interim Meeting will provide a synopsis of highlights from this year's deliberations, touching on items of general interest and particularly focusing on items of interest to the Sector specifically.

iv. Report of Activity from International W&M Agencies

Dr. Ambler Thompson of NIST will provide a synopsis of international activity that relates to the work of the Sector.

CARRYOVER ITEMS

1. Software Identification / Markings

Source: NTETC Software Sector

Background: During their October 2007 meeting, the Sector discussed the value and merits of required markings for software. This included the possible differences in some types of devices and marking requirements. After hearing several proposals, the Sector agreed to the following technical requirements applicable to the marking of software.

1. The NTEP CC Number must be continuously displayed or hard marked,
2. The version must be software-generated and shall not be hard marked,
3. The version is required for embedded (Type P) software,
4. Printing the required identification information can be an option,
5. Command or operator action can be considered as an option in lieu of a continuous display of the required information, and
6. Devices with Type P (embedded) software must display or hard mark make, model, S.N. to comply with G-S.1. Identification.

The Sector developed marking information requirements and submitted a proposal to the S&T Committee for considered inclusion in NIST Handbook 44. It remained an Informational item for the Annual Meeting. For several years the Sector has responded to comments and modified the proposed language to address concerns and limit the scope of the changes.

It seems that at each meeting of the Sector, state weights and measures officials reiterate the problems they have in the field locating the basic information required when the CC number is marked via the rather general current HB 44 requirement of 'accessible through an easily recognizable menu, and if necessary a sub-menu [G-S.1.1. (b)(3)]. States have indicated that this is too vague and field inspectors often cannot find the certificate number on unfamiliar devices. Hence, at the 2010 Software Sector meeting additional changes were proposed to the language to address both the previously voiced concerns and also address the field issues related to locating the required marking information.

(See the 2010 Software Sector meeting summary and the 2011 Interim Meeting S&T agenda item 310-2 for more background on this item.)

The proposed change as it appeared in the 2011 Interim Agenda (Publication 15) was as follows:

G-S.1. Identification. – All equipment, except weights, ~~and~~ separate parts necessary to the measurement process but not having any metrological effect, and software-based devices covered in G-S.1.1. Location of Marking Information*, shall be clearly and permanently marked for the purposes of identification with the following information:

*[*Nonretroactive as of January 1, 201X]*

- (a) the name, initials, or trademark of the manufacturer or distributor;
- (b) a model identifier that positively identifies the pattern or design of the device;
 - (1) The model identifier shall be prefaced by the word “Model,” “Type,” or “Pattern.” These terms may be followed by the word “Number” or an abbreviation of that word. The abbreviation for the word “Number” shall, as a minimum, begin with the letter “N” (e.g., No or No.). The abbreviation for the word “Model” shall be “Mod” or “Mod.” Prefix lettering may be initial capitals, all capitals, or all lowercase.*
 - [Nonretroactive as of January 1, 2003]*
 - (Added 2000) (Amended 2001)*
- (c) a nonrepetitive serial number, except for equipment with no moving or electronic component parts ~~and not built for purpose software based software devices~~;
 - [Nonretroactive as of January 1, 1968]*
 - (Amended 2003)*
 - (1) The serial number shall be prefaced by words, an abbreviation, or a symbol, that clearly identifies the number as the required serial number.*
 - [Nonretroactive as of January 1, 1986]*
 - (2) Abbreviations for the word “Serial” shall, as a minimum, begin with the letter “S,” and abbreviations for the word “Number” shall, as a minimum, begin with the letter “N” (e.g., S/N, SN, Ser. No., and S. No.).*
 - [Nonretroactive as of January 1, 2001]*
- (d) the current software version or revision identifier for ~~not built for purpose software based electronic~~ devices;
 - [Nonretroactive as of January 1, 2004]*
 - (Added 2003) (Amended 201X)*
 - (1) The version or revision identifier shall be prefaced by words, an abbreviation, or a symbol, that clearly identifies the number as the required version or revision.*
 - [Nonretroactive as of January 1, 2007]*
 - (Added 2006)*
 - (2) Abbreviations for the word “Version” shall, as a minimum, begin with the letter “V” and may be followed by the word “Number.” Abbreviations for the word “Revision” shall, as a minimum, begin with the letter “R” and may be followed by the word “Number.” The abbreviation for the word “Number” shall, as a minimum, begin with the letter “N” (e.g., No or No.).*
 - [Nonretroactive as of January 1, 2007]*
 - (Added 2006)*
- (e) an NTEP CC number or a corresponding CC Addendum Number for devices that have a CC.
 - (1) The CC Number or a corresponding CC Addendum Number shall be prefaced by the terms “NTEP CC,” “CC,” or “Approval.” These terms may be followed by the word “Number” or an abbreviation of that word. The abbreviation for the word “Number” shall, as a minimum, begin with the letter “N” (e.g., No or No.)*
 - [Nonretroactive as of January 1, 2003]*

The required information shall be so located that it is readily observable without the necessity of the disassembly of a part requiring the use of any means separate from the device.

(Amended 1985, 1991, 1999, 2000, 2001, 2003, ~~and~~ 2006 and 201X)

**added by S&T committee based on SMA comments, and not in original SS submission*

G-S.1.1. Location of Marking Information for ~~Not-Built-For-Purpose~~ all Software-Based Devices. – For ~~not-built-for-purpose~~, software-based devices, either:

(a) The required information in G-S.1. Identification. ~~(a), (b), (d), and (e)~~ shall be permanently marked or continuously displayed on the device; or

(b) The CC Number shall be:

(1) permanently marked on the device;

(2) continuously displayed; or

(3) accessible through one or, at most, two levels of access. an easily recognized menu and, if necessary, a submenu. Examples of menu and submenu identification include, but are not limited to, “Help,” “System Identification,” “G-S.1. Identification,” or “Weights and Measures Identification.”

(i) For menu based systems, “Metrology,” “System Identification,” or “Help.”

(ii) For systems using icons, a metrology symbol “(M),” “(SI),” or a help symbol (“?”, “i,” or an “i” within a magnifying glass).

Note: For (b), clear instructions for accessing the information required in G-S.1. (a), (b), and (d) shall be listed on the CC, including information necessary to identify that the software in the device is the same type that was evaluated.

[Nonretroactive as of January 1, 2004]

(Added 2003) (Amended 2006 **and 201X**)

In addition to the proposal above, in 2010 the Software Sector also initiated discussion on two new concepts, which may eventually result in additional recommendations to amend G-S.1.

First, the Sector sees merit to requiring some “connection” between the software identifier (i.e., version/revision) and the software itself. The proposal was to add a new sub-subparagraph (3) to G-S.1.(d) to read as follows (with the expectation that examples of acceptable means of implementing such a link would be included in Pub 14).

“The version or revision identifier shall be directly and inseparably linked to the software itself. The version or revision identifier may consist of more than one part, but at least one part shall be dedicated to the metrologically significant software.”

Second, rather than the limited options shown in the proposed G-S.1.1 (b)(3) (i) && (ii) above, a more extensive list of accepted menu text and symbols/icons was generated that could be used as a reference by manufacturers, NTEP labs during approval, and by field inspectors. The initial list was generated and circulated for comment soon after the 2011 Software Sector meeting (see Table 1 on page 9). Note that this is not suggested to be the final list of valid options for locating the point of access for the CC number; the Software Sector would like to have feedback specifically on other acceptable menu text/icon images that identify how to access the CC number on software-based systems. The Software Sector agreed that a reasonable list of acceptable options is not as much of an issue as the fact that the list is finite. The sector realizes this may affect manufacturers so feedback from associate members and representative groups is also appreciated.

At its spring 2010 meeting, NEWMA recommended leaving this item informational to allow review of the software Sector's newly proposed language from its March 2010 meeting.

During the 2010 Annual Meeting, the SMA stated that the proposal from the Software Sector addresses one of the SMA's concerns dealing with the use of the term "not built for purpose;" however, it still has concerns with the requirement in G-S.1. stating that the software version or revision identifier must be clearly and permanently marked. The SMA recommends that the Software Sector and the S&T Committee review and correct what appears to be conflicting requirements as stated in G-S.1. and G-S.1.1. dealing with the marking requirement.

The Committee also received a summary of the 2010 meeting of the NTETC laboratories where some of the NTEP evaluators were concerned that the revised language could be interpreted such that no markings are required on a device. These evaluators expressed concern that an inspector would have to guess which of the eight methods recommended in the Software Sector Summary is to be used to find the CC number and questioned whether this would mean that a weighing or measuring device might not be marked with any identifier markings including the manufacturer.

The Committee amended the item under consideration based on the recommendations of the Software Sector at its March 2010 meeting. The Committee agreed to clarify and document the SMA concerns with the requirements in G-S.1. where it states that "all equipment . . . shall be permanently marked . . ." and G-S.1.1. that allows alternate methods, other than "permanently marked," to identify software-based devices. Consequently, the Committee revised the first paragraph of G-S.1. to read as shown in the "Item Under Consideration" in the 2011 NCWM Interim Agenda.

At its 2010 fall Interim Meeting, the CWMA S&T Committee stated that it believes that this item should be moved to a Vote and suggested an editorial change on G.S.1.1. (b) (3), to read "***no more than two levels of access***" instead of "***one or, at most, two levels of access.***"

During the fall 2010 WWMA Annual Technical Conference, Mr. Cook, NIST Technical Advisor to the WS, provided an update to the Committee. Mr. Cook also discussed the conflicting language between G-S.1. and G-S.1.1. identified by the SMA and the NCWM S&T Committee's solution to eliminate the conflict. The WS reviewed the list of acceptable abbreviations and icons as requested by the Software Sector and agreed that the abbreviation "SI" should not be included in the list since "SI" is also the abbreviation to the International System of Measurement.

The WS also noted that the icon "M" with the green fill should not be used since it is used by the European Union as a metrology mark for all devices, not just for metrological software identification. Mr. Flocken, speaking on behalf of the SMA, restated its April 2010 position based on the conflicting language in paragraphs G-S.1. and G-S.1.1. He added that the revised language for G-S.1. in the S&T Agenda should also be reviewed by the Software Sector. Mr. Johnson, Gilbarco, added that their current Retail Motor-Fuel Dispenser (RMFD) software cannot display alpha characters in or for software version identification which is problematic since the latest version of the proposal includes software identification for all software based devices. Mr. Johnson added that a possible solution would be to allow the software version to be reported on the NTEP CC.

The WWMA recommended the following amendment to G-S.1. (d)(2) that addresses Gilbarco's comments on devices with limited character sets such as RMFD without alpha displays and/or annunciators to read as follows: S&T Committee 2010 Interim Agenda S&T - 20

(d) the current software version or revision identifier for software-based devices;
[Nonretroactive as of January 1, 2004]

(Added 2003)

(1) . . .

(2) *Except for devices with limited character sets (e.g., primary indications without alpha characters or annunciators*) the version or revision identifier shall be prefaced by words, an abbreviation, or a symbol, that clearly identifies the number as the required version or revision.*

(WMD Technical Advisor Note: After the WWMA meeting, WMD noted that it believes there is a need to address the exception by adding language to address the method for identifying the version or revision number for devices with limited character sets. For example: Add a new sentence at the end of (2) such as *“For devices with limited character sets, the instructions to identify the version or revision identifier shall be listed on the NTEP CC.”*)

[Nonretroactive as of January 1, 2007]

**[Nonretroactive as of January 1, 201X]*

The WWMA believes that the above changes to the proposal sufficiently address all issues identified during the open hearings and that this should remain an Information item to allow the Software Sector an opportunity to comment on the revisions proposed by the NCWM and WWMA S&T Committees. At its 2010 Annual Meeting open hearing, the SWMA heard from Mr. Johnson, Gilbarco, restating his concern about how this proposal would apply to some simpler devices that may have a limited display capability; while these devices may be able to display a software version number, they aren't able to display a designation that defines it as a “version number.” Mr. Johnson also noted that the WWMA modified the proposed language to provide an exception for devices with limited character sets and encouraged the Committee to review this language. Mr. Straub, Fairbanks Scales, speaking on behalf of SMA stated that SMA, at its 2010 spring meeting, opposed this item. Mr. Straub also pointed out that there appears to be a conflict with regard to the required permanence of the marking, noting that G-S.1. refers to “permanently marked,” whereas G-S.2. makes reference to “continuously displayed” markings.

The SWMA S&T Committee considered whether or not the proposal is ready to be adopted. Based on the variety of comments heard, comments opposing the item, and the alternatives presented, the Committee did not feel it could make a recommendation at this time. The Committee felt that the Software Sector should be given the opportunity to review the input and comments made on this issue since the last meeting of the Sector. Consequently, the Committee felt that the item should remain as an Information item on the NCWM S&T Committee's agenda.

At its fall 2011 Interim Meeting, NEWMA states that the WWMA proposed revision to the “item under consideration” and questions raised have merit. NEWMA recommends this remain an Information item to give the Weighing Sector and the NCWM S&T Committee time to evaluate the new language.

Software Sector proposed identifier text/icons - circulation for comments.

The Software Sector developed the following brief initial list of ideas of allowable/acceptable menu text and icons as a starting point for developing the complete list of acceptable options for the readily identifiable menu. A request for comments was made to all NCWM Software Sector mailing list recipients. Returned comments from external parties follow after the table.

Table 1 - Software Sector Proposed Menu Text /Icons

<i>Permitted Menu Text examples</i>	<i>Permitted Icon shape examples</i>	<i>Essential characteristics</i>
<p>Information</p> <p>Info</p>		<p>Top level menu text or icon</p> <ul style="list-style-type: none"> • Icon text is a lower case “i” with block serifs • Text color may be light or dark but must contrast with the background color • Icon may have a circular border • Activation of this menu text/icon may invoke a second level menu text/icon that recalls metrology information.
<p>Help</p> <p>?</p>		<p>Top level menu text or icon</p> <ul style="list-style-type: none"> • Icon text is a question mark • Text color may be light or dark but must contrast with the background color • Icon may have a circular border • Activation of this menu text/icon may invoke a second level menu text/icon that recalls metrology information.
<p>Metrology</p> <p>Metrological Information</p>		<p>Top or second level menu text or icon</p> <ul style="list-style-type: none"> • Icon text is an upper case “M” • Text color may be light or dark but must contrast with the background color • Icon may have a rectangle or rounded rectangle border • If present, the activation of this menu text/icon must recall at a minimum the NTEP CC number.
<p>SI</p> <p>S.I.</p>		<p>Top or second level menu text or icon</p> <ul style="list-style-type: none"> • Icon text is upper case “SI” • Text color may be light or dark but must contrast with the background color • Icon may have a rectangle or rounded rectangle border • If present, the activation of this menu item/icon must recall at a minimum the NTEP CC number.
<p>NTEP Data</p> <p>N.T.E.P. Certificate</p>		<p>This one is debatable – what if the certificate is revoked? Does NTEP grant holders of CCs the right to display the logo on the device, or just in documentation?</p>

Acceptable examples of where the text or icon may be displayed:

1. The “M” icon is available on the home screen. Activation of the icon displays a new screen containing the CC number and some additional metrology information including the software version/revision number(s).
2. The “SI” icon is available on the home screen. Touch screen activation of the icon displays a pop-up containing the CC number. Releasing the icon erases the pop-up.
3. The main screen contains the “i” icon (information). Activating this icon displays a screen of other icons including the “M” icon. Activating the “M” icon displays the NTEP CC.
4. The main menu includes a “Help” selection which in turn contains a “Metrology” selection. Activation of the Metrology selection displays a pop-up screen containing all global metrological approvals, including the NTEP CC number. The user manually dismisses the pop-up screen by pressing the [X] button.
5. The main menu includes an “Info” selection which in turn contains a “SI” selection. Activation of the SI selection displays a pop-up screen containing all global metrological approvals, including the NTEP CC number. The user manually dismisses the pop-up screen by pressing the [OK] button.

Feedback from various external groups from which we requested info:

Measuring Sector:

This item was included on the Measuring Sector’s agenda as an information item to keep Sector members informed of the progress of this NCWM S&T Issue and to ask for input from Sector members on this issue.

Discussion at measuring sector meeting: The S&T Committee has been considering changes to G-S.1. to better address identification requirements for metrologically significant software in software-based systems. The Committee has considered multiple proposals under this item from the NTETC Software Sector and the weights and measures community. At the July 2010 NCWM Annual Meeting, the S&T Committee agreed to maintain this as an Information item on its agenda to allow for additional review and input. As noted above, the Software Sector is looking for specific feedback on proposed modifications to paragraph G-S.1. so that it can develop a revised proposal for consideration by the S&T Committee. Should time permit the Measuring Sector to discuss this item, the NCWM S&T Committee and the Software Sector would appreciate the Sector’s input.

Discussion: NTEP Director and past Software Sector Chairman, Mr. Jim Truex, provided a history of how this issue evolved. He noted that there were multiple attempts to address software in not-built-for purpose devices. The Software Sector has attempted to further simplify the identification requirements that apply to software-based systems and has made multiple suggestions that were not accepted. The Sector has taken a step back and is trying to get the point across that the marking requirements are not for the manufacturer, but to assist the inspector in the inspection process and in assessing whether or not a specific device, including software, is covered under an NTEP CC. The Sector realizes that this information is not going to be physically marked on the device and is looking for alternatives in which this information can be provided electronically to inspectors in an easily accessible manner. It will likely be provided on the device’s display screen and there is limited space for this information to be displayed. The SW Sector is looking for input on the general direction it should take in developing/updating NIST Handbook 44 requirements. If the direction seems

reasonable, the SW Sector will further develop the idea; if not, the Sector will consider an alternative direction.

The Sector discussed some of the symbols in the proposed list of icons and discussed differences between built-for-purpose and not-built-for-purpose devices. Some Sector members also acknowledged that sometimes changes to software will affect the metrological functions of the device, even though the change was not intended to have that effect and was supposed to be a “non-metrologically significant” change. Some members, particularly the regulators, supported the idea of a “Weights and Measures” key that would be standardized and, thus, readily recognized by the field official. Mr. Truex acknowledged that the regulatory community has, in his opinion, indicated that the options need to be limited. Mr. Rich Tucker, RL Tucker Consulting LLC, and Mr. Michael Keilty (Endress Hauser Flowtec AG USA), expressed support for labeling the key that would enable display of the required information as “help.”

Measuring Sector recommendation: *The Measuring Sector had no additional technical guidance to offer to the S&T Committee on this issue. However, based on comments from Sector members present, the Sector expressed general support for trying to refine the marking requirements and limit the number of options for marking keys that enable the inspector to view the required marking information.*

Weighing Sector:

The WS reviewed the initial list of menu text and icons and provided the following comments:

- Darrell Flocken indicated that the green M is an EU metrology mark and for that reason should not be considered an acceptable icon.
- There was general consensus amongst WS members that the SI should not be considered acceptable since it is also used to identify the International System of Units.

NTETC Grain Analyzer (GA) Sector:

Discussion: It should be noted that these new ideas are in the developmental stage, and are included here at the request of the Software Sector, which is seeking comments from interested parties. The GA Sector is asked to comment on the proposed changes to G-S.1. and G-S.1.1. shown above, specifically those that will most affect Grain Analyzers.

1. G-S.1.(d) and its sub paragraphs will require a software version or revision identifier that is directly and inseparably linked to the software itself; and
2. G-S.1.1. and its sub paragraphs will allow the identifiers required in G-S.1. to be either permanently marked or continuously displayed for software-based electronic devices. This includes the software version or revision identifier. It also allows display of the CC number to be accessible by menu or icon (as opposed to continuously displayed.)
3. If not either permanently marked or continuously displayed, the CC Number will have to be accessible through one or two levels of access identified by the labels, “Metrology”, “System Identification”, or “Help” in menu based systems, or for systems using icons, a metrology symbol (“M” or “SI”), or a help symbol (“?”, “I,” or an “i” within a magnifying glass). Note that this is not suggested to be the final list of valid options; the Software Sector would like to have feedback specifically on additional menu text/icon images that

should be considered acceptable. The Software Sector feels that the number of acceptable options is less of an issue (within reason) than the fact that the list is finite.

GA Sector Comments: The GA Sector found the wording of G-S.1.1. confusing. It seemed to say that the markings spelled out in G-S.1. were to be ***EITHER** permanently marked or continuously displayed on the device **OR** the Certificate of Conformance (CC) Number shall be either: permanently marked or continuously displayed, or accessible through menu or icon.* To some, this implied that the software version identifier did NOT have to be displayed. Others believed that the “**OR**” phrase meant that only the CC had three options for marking (permanent, continuously displayed, or accessible via menu or icon), and that the software/firmware version/revision number must be either permanently marked or continuously displayed.

Regardless of how the wording is interpreted, the GA Sector agreed that it was not practical to permanently mark or continuously display the software/firmware version/revision identifier for GMMs. The GA Sector recommends that G-S.1.1.(b) be amended to include accessing the software version or revision identifier by menu or icon. At present all NTEP GMMs are built-for-purpose. They all have permanently marked CC numbers. Software version/revision identifiers, however, are accessible by menu or icon. GMM displays are of limited size. Some existing devices don’t have room to display the software version/revision identifier on every “screen”. Hard marking of that identifier is not practical, because it precludes updating software without also replacing the hard-marked label.

Scale Manufacturer’s Association Comments: The SMA supports the requirement to access a version number for software based devices. The SMA looks forward to the Software Sector's definition of the term "software based device".

SMA adamantly opposed the definition we provided previously. From the 2009 Software Sector Meeting Summary and 2010 Pub 15 310-2:

Electronic devices, software-based. Weighing and measuring devices or systems that use metrological software to facilitate compliance with Handbook 44. This includes:

- (a) Embedded software devices (Type P), aka built-for-purpose. A device or element with software used in a fixed hardware and software environment that cannot be modified or uploaded via any interface without breaking a security seal or other approved means for providing security, and will be called a “P,” or**
- (b) Programmable or loadable metrological software devices (Type U), aka not-built-for-purpose. A personal computer or other device and/or element with PC components with programmable or loadable metrological software, and will be called “U.” A “U” is assumed if the conditions for embedded software devices are not met.**

Software-based devices – See Electronic devices, software-based.

2. Identification of Certified Software

Source: NTETC Software Sector

Background/Discussion: This item originated as an attempt to answer the question “How does the field inspector know that the software running in the device is the same software evaluated and approved by the lab?” In previous meetings it was shown that the international community has addressed this issue (both WELMEC and OIML). From WELMEC 7.2:

Required Documentation:

The documentation shall list the software identifications and describe how the software identification is created, how it is inextricably linked to the software itself, how it may be accessed for viewing and how it is structured in order to differentiate between version changes with and without requiring a type approval.

From OIML D-31:

The executable file “**tt100_12.exe**” is protected against modification by a checksum. The value of checksum as determined by algorithm **XYZ** is **1A2B3C**.

Previous discussions have included a listing of some additional examples of possible valid methods (not limiting):

- CRC (cyclical redundancy check)
- Checksum
- Inextricably Linked version no.
- Encryption
- Digital Signature

Is there some method to give the W&M inspector information that something has changed? (Yes, the Category III audit trail or other means of sealing). How can the W&M inspector identify an NTEP Certified version? (They can’t, without adding additional requirements like what is described here, in conjunction with including the identifier on the CoC).

The Sector believes that we should work towards language that would include a requirement similar to the OIML requirement in HB44. It is also the opinion of the Sector that a specific method should not be defined; rather the manufacturer should utilize a method and demonstrate the selected identification mechanism is suitable for the purpose. It is not clear from the discussion where such proposed language might belong.

NTEP strongly recommends that metrological software be separated from non-metrological software for ease of identification and evaluation. From OIML:

Separation of software parts - All software modules (programmes, subroutines, objects etc.) that perform metrologically significant functions or that contain metrologically significant data domains form the metrologically significant software part of a measuring instrument (device or sub-assembly). The conformity requirement applies to all parts and parts shall be marked according to Section G-S-X.X.

If the separation of the software is not possible or needed, then the software is metrologically significant as a whole.

(Segregation of *parameters* is currently allowed - see table of sealable parameters)

Initial draft proposed language: (G-S.1.1?)

Handbook 44 (This has been written into G-S.1.d.3):
Identification of Certified Software:

Software-based electronic devices shall be designed such that the metrologically significant software is clearly identified by the version or revision number. The identification, and this identification of the software shall be inextricably directly and inseparably linked to the software itself. The version or revision number may consist of more than one part, but at least one part shall be dedicated to the metrologically significant software.

Pub. 14:
Identification of Certified Software:

Note: Manufacturers may choose to separate metrologically significant software from non-metrologically significant software. Separation would allow the revision of the non-metrological portion without the need for further evaluation. In addition, non-metrologically significant software may be updated on devices without breaking a seal, if so designed. Separation of software requires that all software modules (programs, subroutines, objects etc.) that perform metrologically significant functions or that contain metrologically significant data domains form the metrologically significant software part of a measuring instrument (device or sub-assembly). If the separation of the software is not possible or needed, then the software is metrologically significant as a whole. ~~The conformity requirement applies to all parts and parts shall be marked according to Section G-S-X-X.~~

The manufacturer must describe and possibly demonstrate how the version or revision identifier is directly and inseparably linked to the metrologically significant software. Where the version revision identifier is comprised of more than one part, the manufacturer shall describe which portion represents the metrological significant software and which does not.

From OIML D-31:

Legally relevant software of a measuring instrument / electronic device / sub-assembly shall be clearly identified with the software version or another token. The identification may consist of more than one part but at least one part shall be dedicated to the legal purpose.

The identification shall be inextricably linked to the software itself and shall be presented or printed on command or displayed during operation or at start up for a measuring instrument that can be turned off and on again. If a sub-assembly/an electronic device has neither display nor printer, the identification shall be sent via a communication interface in order to be displayed/printed on another sub-assembly/electronic device.

The first sentence of the first paragraph above is already addressed in Handbook 44's marking requirements.

Recommendation: Recommend the following change to Handbook 44, General Code: G-S.1(d) to add a new subsection (3):

(d) the current software version or revision identifier for ~~not-built-for-purpose~~ software-based electronic devices;

[Nonretroactive as of January 1, 2004]
(Added 2003) **(Amended 201X)**

(1) The version or revision identifier shall be prefaced by words, an abbreviation, or a symbol, that clearly identifies the number as the required version or revision.

[Nonretroactive as of January 1, 2007]
(Added 2006)

(2) Abbreviations for the word “Version” shall, as a minimum, begin with the letter “V” and may be followed by the word “Number.” Abbreviations for the word “Revision” shall, as a minimum, begin with the letter “R” and may be followed by the word “Number.” The abbreviation for the word “Number” shall, as a minimum, begin with the letter “N” (e.g., No or No.).

[Nonretroactive as of January 1, 2007]
(Added 2006)

(3) The version or revision identifier shall be directly and inseparably linked to the software itself. The version or revision identifier may consist of more than one part, but at least one part shall be dedicated to the metrologically significant software.

[Nonretroactive as of January 1, 201X]
(Added 201X)

Also the Sector recommends the following information be added to Pub. 14 as explanation/examples:

- Unique identifier must be displayable/printable on command or during operation, etc.
- At a minimum, a version/revision indication (1.02.09, rev 3.0 a, etc). Could also consist of/ contain checksum, etc (crc32, for example)

Conclusions: The item needs additional discussion and development by the sector. Outstanding questions: If we allow hard-marking of the software identifier (the Sector has wavered on this in the past), does the above wording then imply that some mechanical means is required (i.e. physical seal) to ‘inseparably link’ the identifier to the software? If a device is capable of doing so, does it still have to be able to display, print or communicate the identifier somehow, even if it is hard-marked?

3. Software Protection / Security

Source: NTETC Software Sector

Background: The sector agreed that Handbook 44 already has audit trail and physical seal, but these may need to be enhanced.

From the WELMEC Document:

Protection against accidental or unintentional changes

Metrologically significant software and measurement data shall be protected against accidental or unintentional changes.

Specifying Notes:

Possible reasons for accidental changes and faults are: unpredictable physical influences, effects caused by user functions and residual defects of the software even though state of the art of development techniques have been applied.

This requirement includes consideration of:

- a) Physical influences: Stored measurement data shall be protected against corruption or deletion when a fault occurs or, alternatively, the fault shall be detectable.
- b) User functions: Confirmation shall be demanded before deleting or changing data.
- c) Software defects: Appropriate measures shall be taken to protect data from unintentional changes that could occur through incorrect program design or programming errors, e.g. plausibility checks.

Required Documentation:

The documentation should show the measures that have been taken to protect the software and data against unintentional changes.

Example of an Acceptable Solution:

The accidental modification of software and measurement data may be checked by calculating a checksum over the relevant parts, comparing it with the nominal value and stopping if anything has been modified.

Measurement data are not deleted without prior authorization, e.g. a dialogue statement or window asking for confirmation of deletion.

For fault detection see also Extension I.

The Sector continued to develop a proposed checklist for Pub 14. The numbering will still need to be added. This is based roughly on R 76 – 2 checklist and discussion at October 2007 Sector Meeting

The information requested by this checklist is currently voluntary, however, it is recommended that applicants comply with these requests or provide specific information as to why they may not be able to comply. Based on this information, the checklist may be amended to better fit with NTEP's need for information and the applicant's ability to comply.

The CA, MD and OH labs agreed to use this check list on one of the next devices they have in the lab and report back to the Sector on what the problems may be. In Feb. 2011, the S. Carolina lab was also given a copy of the check list to try.

Devices with embedded software TYPE P (aka built-for-purpose)		
	Declaration of the manufacturer that the software is used in a fixed hardware and software environment, and	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>

	cannot be modified or uploaded by any means after securing/verification	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	<i>Note: It is acceptable to break the "seal" and load new software, audit trail is also a sufficient seal.</i>	
	The software documentation contains:	
	description of all the metrologically significant functions, designating those that are considered metrologically significant <i>OIML states that there shall be no undocumented functions</i>	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	description of the securing means (evidence of an intervention)	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	software identification	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	description how to check the actual software identification	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	The software identification is:	
	clearly assigned to the metrologically significant software and functions	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	provided by the device as documented	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
Personal computers, instruments with PC components, and other instruments, devices, modules, and elements with programmable or loadable metrologically significant software TYPE U (aka not built-for-purpose)		
	The <i>metrologically significant</i> software is:	
	documented with all relevant (see below for list of documents) information	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	protected against accidental or intentional changes	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	Evidence of intervention (such as, changes, uploads, circumvention) is available until the next verification / inspection (e.g. physical seal, Checksum, CRC, audit trail, etc. means of security)	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
Software with closed shell (no access to the operating system and/or programs possible for the user)		
	Check whether there is a complete set of commands (e.g. function keys or commands via external interfaces) supplied and accompanied by short descriptions	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	Check whether the manufacturer has submitted a written declaration of the completeness of the set of commands	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
Operating system and / or program(s) accessible for the user:		
	Check whether a checksum or equivalent signature is generated over the machine code of the metrologically significant software (program module(s) subject to legal control W&M jurisdiction and type-specific parameters)	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	Check whether the metrologically significant software will detect and act upon any unauthorized alteration of the metrologically significant software using simple software tools e.g. text editor.	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
Software interface(s)		
	Verify the manufacturer has documented:	
	the program modules of the metrologically significant software are defined and separated	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	the protective software interface itself is part of the metrologically significant software	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
	the <i>functions</i> of the metrologically significant software that can be accessed via the protective software interface	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>

		the <i>parameters</i> that may be exchanged via the protective software interface are defined	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
		the description of the functions and parameters are conclusive and complete	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
		there are software interface instructions for the third party (external) application programmer.	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>

Discussion: The labs again indicated they had not had a chance to utilize the checklist. The list was reviewed and some minor modifications to the checklist text were incorporated as shown in this excerpt:

	The software documentation contains:		
		description of all the metrologically significant functions, designating those that are considered metrologically significant <i>OIML states that there shall be no undocumented functions</i>	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
		description of the securing means (evidence of an intervention)	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>
		software identification	Yes <input type="checkbox"/> No <input type="checkbox"/> N/A <input type="checkbox"/>

Conclusion: Work is ongoing on this item with the intent that it eventually be incorporated as a checklist in Pub 14; again the labs are requested to try utilizing this checklist for any evaluations on software-based electronic devices.

This information may need to be included in HB 44. It may be possible to add this to the general code section.

May need to define what a software update log is.

G-S.9. Verification of Software Update

Only versions of metrologically significant software that conform to the approved type are allowed for use.

Updates to software shall be either manually verified (Verified Update) or automatically performed and traced (Traced Update).

For a Traced Update, an event logger is required. An entry shall be generated for each software update and must include the following:

- an event logger (with a minimum of 10 updates),
- the parameter ID, which indicates the software update
- the date and time of the change, and
- the new value of the parameter, which is the software identification of the installed version.

~~An entry is generated for each software update.
The software log/audit trail shall contain the following information:~~

~~parameter ID; software update, etc,
new value; software identification of the installed version,
date and time of the change,
identification of the downloading party. (considered this~~

~~The device shall clearly indicate that it is in the remote configuration mode and record such message if capable of printing in this mode or shall not operate while in this mode.~~

If the device continues to operate during a software update, then the metrological performance shall not be affected.

The MD lab wanted it on record that they disagree with this statement and striking the first sentence based on discussions within the weighing sectors and the measuring sector and the NTEP lab meetings on the subject of calibration and configuration while in the normal weighing measuring mode. The sentence that has been struck out was placed in the DES checklist years ago to address field concerns.

It was noted there is a statement in the WELMEC document that concurs with the statement above as stricken.

Use of a Category 3 audit trail is acceptable for the software update logger.

Definitions Recommendation:

Verified Update

A verified update is the process of installing new software where the security is broken and the device must be re-verified. Checking for authenticity and integrity is the responsibility of the owner/user.

Traced Update

A traced update is the process of installing new software where the software is automatically checked for authenticity and integrity, and the update is recorded in a software update log or audit trail.

Comment: The **sector agreed** that the two definitions directly above for Verified update and Traced update were acceptable.

Question, do we need the definitions below any longer?

Comment: There is text in these definitions that don't belong in the definition, but may be applicable for other purposes, primarily the bit about the software protection environment being at the same level after upgrade when doing traced update. The Sector has not addressed that yet and it is important.

Previous definitions:

Verified update

The software to be updated can be loaded locally (e.g. directly) on the weighing or measuring device or remotely via a network. Loading and installation may be two different steps (as shown in Fig. above) or combined to one, depending on the needs of the technical solution. After update of the metrologically significant software of a weighing or measuring device (exchange with another

approved version or re-installation) the weighing or measuring device is not allowed to be used for legal purposes before a (subsequent) verification of the instrument has been performed and the securing means have been renewed. A person responsible for verification must be at place. (NOTE: This may need to be in the HB under user requirement.)

Traced update

Traced update is the procedure of changing software in a weighing or measuring device after which the subsequent verification by a responsible person at place is not necessary. The software to be updated can be loaded locally (e.g. directly) on the weighing or measuring device or remotely via a network. The software update is recorded in a software log or audit trail.

Traced update of software shall be automatic. On completion of the update procedure, the software protection environment shall be at the same level as required by the type approval.

4. **Software Maintenance and Reconfiguration**

Source: NTETC Software Sector

Background: After the software is completed, what do the manufacturers use to secure their software?

Discussion: The Following Items were reviewed by the Sector. Note that agenda item 3 also contains information on Verified and Traced updates and Software Log.

- a. Verify that the update process is documented (OK)
- b. For traced updates, Installed Software is authenticated and checked for integrity

Technical means shall be employed to guarantee the authenticity of the loaded software i.e. that it originates from the owner of the type approval certificate. This can be accomplished e.g. by cryptographic means like signing. The signature is checked during loading. If the loaded software fails this test, the instrument shall discard it and either use the previous version of the software or become inoperative.

Technical means shall be employed to guarantee the integrity of the loaded software i.e. that it has not been inadmissibly changed before loading. This can be accomplished e.g. by adding a checksum or hash code of the loaded software and verifying it during the loading procedure. If the loaded software fails this test, the instrument shall discard it and either use the previous version of the software or become inoperative.

Examples are not limiting or exclusive.

- c. Verify that the sealing requirements are met

The Sector asked, what sealing requirements are we talking about?

This item is only addressing the software update, it can be either verified or traced. It is possible that there are two different security means, one for protecting software updates (software log) and one for protecting the other metrological parameters (Category I II or III method of sealing).

Some examples provided by the Sector members include but are not limited to.

Physical Seal, software log

Category III method of sealing can contain both means of security

d. Verify that if the upgrade process fails, the device is inoperable or the original software is restored

The question before the group is can this be made mandatory?

The manufacturer shall ensure by appropriate technical means (e.g. an audit trail) that traced updates of metrologically significant software are adequately traceable within the instrument for subsequent verification and surveillance or inspection. *This requirement enables inspection authorities, which are responsible for the metrological surveillance of legally controlled instruments, to back-trace traced updates of metrologically significant software over an adequate period of time (that depends on national legislation).* The statement in italics will need to be reworded to comply with US W&M requirements.

The Sector **agreed** that the two definitions below for Verified update and Traced update were acceptable.

Verified Update

A verified update is the process of installing new software where the security is broken and the device must be re-verified. Checking for authenticity and integrity is the responsibility of the owner/user.

Traced Update

A traced update is the process of installing new software where the software is automatically checked for authenticity and integrity, and the update is recorded in a software update log or audit trail.

The Sector also worked towards language proposed for defining the requirements for a Traced Update (currently considered as relevant for Pub 14):

For a Traced Update, an event logger is required. The logger shall be capable of storing a minimum of the 10 most recent updates. An entry shall be generated for each software update.

Use of a Category 3 audit trail is acceptable required for the software update logger Traced Update. In this case the existing requirement of 1,000 entries supersedes the 10 entry requirement. If software update is the only loggable event, then the Category 3 audit trail can be limited to only 10 entries. A software update log entry representing a software update shall include the following: the software identification of the newly installed version.

- An event counter;
- the date and time of the change; and
- the event type/parameter ID, which indicates a software update event (if not using a dedicated update log);
- the new value of the parameter, which is the *software identification* of the newly installed version.

A Category III device may include the software update events in the Category III audit log in lieu of a separate software update log; the existing requirement for 1,000 entries supersedes the requirement for 10 entries.

The traceability means and records are part of the metrologically significant software and should be protected as such. If software separation is employed, the software used for displaying the audit trail belongs to the fixed metrologically significant software. (Note: This needs to be discussed further due to some manufacturer's concerns about where the software that displays the audit trail information is located and who has access if this feature is provided. Manufacturers did indicate that there are methods available to encrypt the audit trail information; however, it cannot be protected from being deleted.) (include flowchart from OIML D-31)

The Sector discussed how to best move this item forward, and there was also some discussion as to whether new language for the General Code was required. The following new text was proposed:

G-S.9. Metrologically Significant Software Updates

The updating of metrologically significant software shall be considered a sealable event. Metrologically significant software that does not conform to the approved type is not allowed for use.

The NTEP Administrator indicated that the current requirements in G-S.8 already make the statement that any changes that affect metrological function are sealable, hence software updates may be covered and the proposed G-S.9 unnecessary. Todd Lucas suggested we go ahead and submit the proposed G-S.9 to the Committee and request a clarification/interpretation of G-S.8

At the 2009 meeting, the Sector opined that the explicit language proposed for G-S.9 is clearer than any implied requirement in G-S.8. The Sector would like a clarification/interpretation of G-S.8 as it relates to software updates from the S&T Committee (with their response preferably to be included in Pub 16). The Sector will also continue to develop the proposed text (and flow chart) targeted for inclusion in Pub 14.

Discussion: The Sector reviewed the proposal and reconsidered allowing a separate 'update log'. It was decided that this would probably generate confusion and is not likely to be adopted by manufacturers anyway. Hence, the previously proposed text was modified to require a category III audit trail for 'traced updates':

~~For a Traced Update, an event logger is required. The logger shall be capable of storing a minimum of the 10 most recent updates. An entry shall be generated for each software update.~~

Use of a Category 3 audit trail is acceptable required for the software update logger Traced Update. In this case the existing requirement of 1,000 entries supersedes the 10 entry requirement. If software update is the only loggable event, then the Category 3 audit trail can be limited to only 10 entries. A software update log entry representing a software update shall include the following: the software identification of the newly installed version.

- ~~An event counter;~~
- ~~the date and time of the change; and~~
- ~~the event type/parameter ID, which indicates a software update event (if not using a dedicated update log);~~
- ~~the new value of the parameter, which is the software identification of the newly installed version.~~

~~A Category III device may include the software update events in the Category III audit log in lieu of a separate software update log; the existing requirement for 1,000 entries supersedes the requirement for 10 entries.~~

Conclusions: The general consensus of the group after considering feedback from external interested parties is that a new G-S.9 with explicit requirements is not necessary (nor likely to be adopted by the Conference) and that this requirement belongs in the Pub. 14 lists of sealable parameters rather than in Handbook 44; i.e.

The updating of metrologically significant software shall be considered a sealable event.

Additional work is to be done to further develop the proposed text toward inclusion in Pub 14.

*Since the 2010 meeting, the Grain Analyzer sector remitted the following:

Discussion: At its August 2009 meeting the GA Sector questioned the need for a definition of “Traced Update”. The traced update was initially intended to cover cases in Europe where the National Body controls a network of devices and wants to update all the devices simultaneously from a central location. Denmark and France do this with NIR Grain Analyzers. Even though individual states may still require that a device updated via a “Traced Update” must be “returned to service” by a registered serviceperson before it can be used, the Sector may want to consider adopting “Traced Update” requirements for all Category 3 Grain Analyzers. The device is still subject to later inspection by state Weights and Measures personnel. By designing to the requirements for “traced update”, states might be encouraged to allow devices updated to those requirements to be returned to service without requiring a visit by a registered serviceperson.

[No formal comments or recommendations were made by the Grain Analyzer sector.]

5. NTEP Application for Software and Software-based Devices

Source: NTETC Software Sector

Background/Discussion: The purpose of initiating this item was to identify issues, requirements and processes for type approving type U device applications. It was suggested that it may be useful to the labs to devise a separate submission form for software for Type U devices. Question: what gets submitted? What requirements and mechanisms for submission should be available?

Validation in the lab - all required subsystems shall be included to be able to simulate the system as installed.

John Roach (CA NTEP Lab) stated that if the software package being evaluated supports platforms/subsystems from multiple manufacturers, testing should be done using at least two platforms/subsystems. Scale labs and scale manufacturers indicated that this is not usually done for scale evaluations.

Discussion: Since the NTEP committee passed the related item at the Annual we will continue to work on this. The NTEP director indicated that we can move in this direction, but felt that it was somewhat premature to develop this thoroughly now. At the point where the sector has developed checklist requirements, then we could move to perhaps add a subsection to current NTEP applications for applicable software. Refer to D-31.6.1. It was also agreed that there seems to be no reason for limiting the scope of this item to software-only applications, and hence all software/software-based devices could benefit from an enhanced application process. Hence the description of this agenda item was modified as shown in the marked up heading.

Conclusion: The item will be revisited at the 2011 Meeting and it will be decided whether to begin further development of this item at that time.

6. Training of Field Inspectors

Source: NTETC Software Sector

Background: During discussions at the 2009 meeting, the Sector concluded that a new agenda item should be initiated specific to the training of field inspectors in relation to evaluating/validating software-based devices.

Discussion: CA has an EPO (Examination Procedure Outline) that begins to address this. Use Handbook 112 as a pattern template for how it could read.

Items to be addressed:

- Certificate of Conformance
- Terminology (as related to software) beyond what is in HB 44.
- Reference materials / information sources
- Safety

System Verification Tests

NOTE: Item numbers 1 through 5 apply to both weighing and measuring devices. Numbers 6 and 7 are specific to weighing devices; while numbers 9 and 10 apply to measuring devices.

1. Identification. The identification (ID) tag may be on the back room computer server and could be viewed on an identification screen on the computer monitor. The ID information may be displayed on a menu or identification screen. Though currently discouraged, some systems may be designed so the system must be shut down and reset to view the ID information. **G-S.1 (1.10)**

1.1. Manufacturer.

1.2. Model designation.

2. Provisions for sealing. **G-S.8 [1.10]; S.1.11 [2.20]; S.2.2 [3.30]**

2.1. Verify sealing category of device (refer to Certificate of Approval for that system).

2.2. Verify compliance with certificate.

3. Units of measure.

3.1. A computer and printer interfaced to a digital indicator shall print all metrological values, intended to be the same, identically. **G-S.5.2.2(a); G-S.5.1 [1.10]**

3.2. The unit of measure, such as lb, kg, oz, gal, qts, liters, or whatever is used, must agree.

4. Operational controls, indications and features (buttons and switches). Verify that application criteria and performance criteria are met (refer to Certificate of Approval).

4.1. Any indication, operation, function or condition must not be represented in a manner that interferes with the interpretation of the indicated or printed values.

5. Indications and displays.

5.1. Attempt to print a ticket. The recorded information must be accurate or the software must not process and print a ticket with erroneous data interpreted as a measured amount.

Weighing Devices

6. Motion detection.

6.1. For railway track, livestock, and vehicle scales apply or remove a test load of at least 15d while simultaneously operating a print button, push-button tare or push-button zero. A good way to do this is to try to print a ticket while pulling the weight truck or another vehicle onto the scale. Recorded values shall not differ from the static display by more than 3d. Perform the test at 10%, 50% and 100% of the maximum applied test load.

S.2.5.1(a) [2.20]; EPO NO. 2-3, 2.4

6.2. For all other scales, apply or remove at least 5d. Printed weight values must agree with the static weight within 1d and must exactly agree with other indications.

S.2.5.4(b) [2.20]; EPO NO. 2-3, 2.4

7. Behind zero indication.

7.1 Apply a load in excess of the automatic zero setting mechanism (AZSM) and zero the scale. **S.2.1.3 [2.20]; EPO NO. 2-3, 2.4, 2.5.2**

Example: On a vehicle scale have someone stand on the scale, then zero them off (AZSM is 3d). Remove the weight (person) and note the behind zero display (usually a minus

weight value) or error condition.

7.2. Attempt to print a ticket. With a behind zero condition, (manually or mechanically operated) a negative number must not be printed as a positive value.

8. Over capacity.

8.1. Manually enter a gross weight if permissible or apply a test load in excess of 105% of the scale's capacity. **S.1.7 [2.20]; S.1.12, UR.3.9 [2.20]**

8.2. Attempt to print a weight ticket. A system must not print a ticket if the manually entered weight or load exceeds 105% of the scale capacity.

Measuring Devices

10. Motion detection.

10.1. Initiate flow through the measuring element. Attempt to print a ticket while the product is flowing through the measuring chamber. The device must not print while the indication is not stable. **S.2.4.1. (3.30)**

11. Over capacity.

11.1. Attempt to print a ticket in excess of the indicated capacity. A system must not print a ticket if the device is manually or mechanically operated in excess of the indicated value.

NOTE: Be aware of error codes on the indicator which may be interrupted as measured values.

Conclusion: This item is in the early stages; work will continue on the item working toward materials to aid in the training of field inspectors. It was indicated that working in conjunction with the Professional Development Committee to develop training materials, etc. would be a logical path of progress once we have developed the information content to include.

New Items

7. Remote or Distributed Metrologically Significant Functionality

Source: CA NTEP Lab

Background: A database on a remote server contains metrological data for a commercial transaction. The server storage containing the database is leased and access is granted for analysis manipulation, viewing, and/or printing the transaction data.

Previously the Sector has discussed situations where data that is used as part of a transaction (e.g. tare values) are being retrieved from a remote server, but examples can be given that extends the boundaries. Is it acceptable to allow situations where data printed on the transaction report is not locally available or cannot be reproduced without server access? What about situations where actual metrologically significant software routines are executed on a remote server?

Does this relate to what the WELMEC working group on software terms 'Data Transmission'?

8. Next meeting

The Sector is on a yearly schedule for Sector meetings. The NTEP Administrator will determine when the next meeting is possible. Our normal rotation would have us in Columbus in 2012.

Appendix A: Report on 2011 Interim Meeting

There was one item on the NCWM S & T Committee agenda for the 2011 Interim meeting related to work done by the Sector. 2011 Pub 15 S&T Item 310-2 relates to our 2011 agenda item 1 (Marking Requirements). After some discussion, mostly supportive but tentative, the Chair had the impression that the bulk of the feedback seemed to indicate that the goals of the proposal are worthwhile but the language is still not satisfactory or sufficiently clear to some.

Appendix B: Report on International W&M Activity

Highlights:

CIML meeting in Orlando FL

MAA updates

Steve Patoray appointed BIML director

New draft WELMEC 7.2 circulated in February of this year for comment.

Workshop on Operating Systems in Legal Metrology hosted by PTB Dec 2010