National Type Evaluation Program (NTEP)
Software Sector Meeting Agenda

September 16-17, 2015 / Denver, CO

INTRODUCTION

The charge of the National Type Evaluation Program (NTEP) Software Sector is important in providing appropriate type evaluation criteria for software-based weighing or measuring device based on specifications, tolerances and technical requirements of *NIST Handbook 44* Section 1.10 General Code, Section 2 for weighing devices, Section 3 for liquid and vapor measuring devices, and Section 5 for taximeters, grain analyzers, and multiple dimension measuring devices. The sector's recommendations are presented to the National Type Evaluation Program (NTEP) Committee each January for approval and inclusion in *NCWM Publication 14 Technical Policy, Checklists, and Test Procedures* for national type evaluation.

The sector is also called upon occasionally for technical expertise in addressing difficult *NIST Handbook 44* issues on the agenda of the National Conference on Weights and Measures (NCWM) Specifications and Tolerances (S&T) Committee. Sector membership includes industry, NTEP laboratory representatives, technical advisors and the NTEP Administrator. Meetings are held annually, or as needed and are open to all NCWM members and other registered parties.

Suggested revisions are shown in **bold face print** by **~~striking out~~** information to be deleted and **<u>underlining</u>** information to be added. Requirements that are proposed to be nonretroactive are printed in ***bold faced italics***.

Table A
Table of Contents

Title of Content	Page
INTRODUCTION	**1**
WELCOME / INTRODUCTIONS	**2**
STATUS REPORTS – RELATED NCWM AND INTERNATIONAL ACTIVITY	**2**
JOINT SESSION PROGRESS REPORT, ACTIVE ITEMS OF MUTUAL INTEREST	**2**
SOFTWARE SECTOR PRESENTATION	**2**
CARRY-OVER ITEMS	**3**
1. Software Identification / Markings	3
2. Identification of Certified Software	6
3. Software Protection / Security	11
4. Software Maintenance and Reconfiguration	13
5. NTEP Application for Software and Software-based Devices	16
6. Training of Field Inspectors	18
NEW ITEMS	**20**
7. Retrieval of Audit Log information	20
8. Next Meeting	20
9. 2015 NCWM Interim Meeting Report	20
10. 2014 International Report	20

Table B
Glossary of Acronyms and Terms

Acronym	Term	Acronym	Term
BIML	International Bureau of Legal Metrology	OIML	International Organization of Legal Metrology
CC	Certificate of Conformance	OWM	Office of Weights and Measures
EPO	Examination Procedure Outline	PDC	Professional Development Committee
GMMs	Grain Moisture Meters	PDC	Professional Development Committee
NCWM	National Conference on Weights and Measures	S&T	Specifications and Tolerances Committee
NTEP	National Type Evaluation Program	SMA	Scale Manufactures Association
NTETC	National Type Evaluation Technical Committee	WELMEC	European Cooperation in Legal Metrology

Details of All Items
(In order by Reference Key)

WELCOME / INTRODUCTIONS

Since the first day of this year's Sector meeting is a joint meeting with the Measuring Sector, there will be some time set aside to meet and greet both new and familiar faces. In addition, the Software Sector would like to give a brief presentation outlining the problems they've been asked to consider and some of the consensus that has been reached.

STATUS REPORTS – RELATED NCWM AND INTERNATIONAL ACTIVITY

Attendees of the 2015 NCWM Interim Meeting will be asked to share any relevant comments or discussion that took place during the open hearings or NCWM Standards and Tolerances (S&T) committee working sessions.

Dr. Ambler Thompson, NIST, Office of Weights and Measures (OWM), will provide a synopsis of international activity that relates to the work of the sector.

JOINT SESSION PROGRESS REPORT, ACTIVE ITEMS OF MUTUAL INTEREST

Since this is the first joint meeting of the Sectors, it is expected that some time will be required to review the agenda items of the Sectors that require collaboration, so all participants have a solid foundation for discussion. As part of this review, items of particular importance or interest should be allocated more time during the joint session day.

SOFTWARE SECTOR PRESENTATION

CARRY-OVER ITEMS

1. Software Identification / Markings

Source:
NTETC Software Sector

Background / Discussion:
See the 2014 Software Sector Meeting Summary and the 2015 Interim Meeting S&T Agenda Item 310-1 for more background on this item.

Since its inception the sector has wrestled with the issue of software identification and marking requirements. Attempts to modify G-S.1.1. have been controversial, and there has been little constructive feedback. Those constructive comments we have received we have attempted to address with tweaks to the language; mostly the feedback has been "We appreciate your efforts, keep it up… but we don't consider the proposed change to be ready for a vote."

At the 2014 meeting, significant work was done to make the recommendation to modify GS-1 more palatable to the Conference in general and the S&T committee(s) in particular. The new approach was a less invasive modification with effective dates set in the future for compliance to new requirements, as reflected in Pub. 15:

Amend *NIST Handbook 44:* G-S.1. Identification as follows:

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

(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***~~ ***software****;*
[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 devices~~;~~ **manufactured as of January 1, 2004 and all software-based devices or equipment manufactured**

as of January 1, 2020;
~~[Nonretroactive as of January 1, 2004]~~
(Added 2003) **(Amended 20XX)**

(1) The version or revision identifier shall be:

i. *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)

Note: If the equipment is capable of displaying the version or revision identifier but is unable to meet the formatting requirement, through the NTEP type evaluation process, other options may be deemed acceptable and described in the CC.
(Added 20XX)

ii. directly linked to the software itself; and
[Nonretroactive as of January 1, 2020]
(Added 20XX)

iii. continuously displayed or be accessible via the display. Instructions for displaying the version or revision identifier shall be described in the CC. As an exception, permanently marking the version or revision identifier shall be acceptable providing the device does not have an integral interface to communicate the version or revision identifier.
[Nonretroactive as of January 1, 2020]
(Added 20XX)

(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.). ***Prefix lettering may be initial capitals, all capitals, or all lowercase.***
[Nonretroactive as of January 1, 2007]
(Added 2006)

(e) an National Type Evaluation Program (NTEP) Certificate of Conformance (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**)

During the joint meeting of the Weighing and Software Sectors, the Chairman of the SS led a discussion on the identification of software; more specifically, the changes that have been proposed or that are needed to G-S.1. and G-S.1.1. and the reasons why these changes are important. He reviewed the SS's 2013 draft proposal to amend G-S.1. and G-S.1.1. and the comments that had been received since its distribution. Very few constructive comments had been received except for some comments provided by NIST OWM, which the Chairman reviewed one by one; requesting additional clarification from the NIST Technical advisor as needed.

Once the review of the Sector's draft proposal had been completed, it was then pointed out that the NIST Office of Weights Measures' Legal Metrology Devices Program (LMDP) staff had developed some suggested alternative changes to the SS's proposal at the request of the SS. Members of both sectors were asked to review and consider the alternative changes proposed by OWM, which were provided in a handout to members of both sectors and displayed on screen.

The NIST Technical Advisor to the WS, also a member of OWM LMDP staff, explained the reasons for OWM's proposed alternative changes to *G-S.1. - Identification*. Initial discussions of the group regarding OWM's draft changes mostly concentrated on three main issues/concerns as follows:

1. Why is it necessary to retain the term "not-built-for-purpose software-based devices" and add enforcement dates to G-S.1.(d) when it is the Sector's intention to treat built-for-purpose and not-built-for-purpose devices the same with respect to identifying software?
2. Consideration of the text that OWM had developed and was proposing for addition to G-S.1.(d) iii.
3. What would be the effective dates of any changes agreed upon by the group?

The following is a brief summary of the discussions and actions taken by the two sectors relative to these three issues/concerns:

1. With regard to the changes proposed to G-S.1.(d), the NIST Technical Advisor to the WS indicated that it was OWM's view that a separation between built-for purpose and not-built-for-purpose software-based devices needed to be maintained within the paragraph because the current requirement (i.e., G-S.1.(d)) only applies to not-built-for-purpose software-based devices. Although the SS's intention is to expand the requirement to apply to all electronic devices, it would not be appropriate to require existing built-for-purpose-equipment, which is already in service, to comply with the proposed changes to G-S.1. since this equipment has not had to do so previously. Updating existing equipment, in order to make it comply with new requirements, could be costly to both manufacturers and device owners. Additionally, it may not be possible for some built-for-purpose devices to provide an indication of the current software version or revision identifier. Although marking of the version or revision identifier using a label affixed to the device might be an option, how would officials be able to tell if the version of software installed in the device actually matched the marking on the device?
2. By adding effective dates, as proposed, the separation can be maintained and still provide a means of requiring <u>all</u> new electronic equipment to comply. The NIST Technical Advisor also acknowledged that it may be possible at some future date to remove the reference to "not built for-purpose" in the paragraph. Members of the two sectors agreed, although it was decided that the words "through December 31, 2015" in the lead-in sentence of G-S.1.(d) should be deleted because the inclusion of this date is not necessary and its removal does not in any way change the proposal. There were significant concerns raised by equipment manufacturers regarding OWM's suggested proposal to require the continuous display of the version or revision identifier on software-based equipment having a digital display. It was stated that some displays; specifically referenced were "seven-segment digital displays of simple design," do not have the capability of complying with the proposed note that had been developed by OWM. It was also stated that customer demand for these simple displays remains steady among the different scale manufacturers because of their low cost in relation to other digital displays that incorporate more current and complex technology. That is, some customers aren't willing to pay the extra money for a more complex display that can be made to comply with OWM's proposed note, such as one of the graphic types, when all that's needed is a simple basic display. Manufacturers did not see this situation changing and stated that sales of these displays are driven by their low cost. Another concern was the valuable "real estate" that the version or revision identifier would take up if it were continuously displayed.

3. In consideration of the fact that the proposed changes, if adopted, would require both built-for purpose and not-built-for-purpose software-based equipment to be able to continuously display the current software version or revision identifier or that this information be accessible via the display menus, members of the two sectors felt that the 2016 effective date proposed by OWM did not provide enough lead-in time for equipment manufacturers. Thus, the sectors agreed to extend the date to

2020 by amending OWM's proposal to reflect this new date.

A fourth issue/concern, which was raised by an equipment manufacturer somewhat later in the discussions, is that some built-for-purpose equipment have limited capability of displaying letters of the alphabet, and therefore, unable to comply with the prefacing requirements specified in G-S.1.(d)(1) and G-S.1.(d)(2). The example provided was a seven-segment display. It is not able to display a "V" or an "R," which are the current acceptable abbreviations for "version" and "revision," respectively. A "U" could be considered a symbol; however, it is not currently a symbol included in the list of acceptable abbreviations found in some *NCWM Publication 14* device checklists. Alternatively, a lower-case "r" could be displayed on such an indicator. In consideration of this concern, it was suggested that a "note" be added to G-S.1.(d) permitting the NTEP evaluators to specify a different method of indication if the device is incapable of prefacing the software version/revision with a "V" or "R." The sectors agreed to propose a "note" be added and let the S&T Committee decide whether the "note" is necessary or appropriate. An additional change agreed upon by the sectors relating to this issue/concern was to add the last sentence of G-S.1.(b) to the end of G-S.1.(d)(2). In discussing this issue/concern, it was also stated that some built-for-purpose devices only indicate the software version or revision identification during power up. That is, in order to view the software identification, it is necessary to shut off and then return power to the device. It was noted that some officials have been instructed not to power down equipment they are inspecting for liability reasons. There were no solutions to this (power down/power up) concern offered by members of either sector.

Although the SS had earlier proposed changes to G-S.1.1., it was decided during the meeting that no changes to G-S.1.1. were necessary since the sectors had agreed to retain the term "not-built-for-purpose software-based devices" in G-S.1.(d). Thus, no changes are proposed to paragraph G-S.1.1.

Conclusion:
We need to explore the reasoning behind the S&T committee's reluctance to move this item to Informational status, since we haven't heard specific criticism of the proposal.
It also appears the OWM has pointed out at least one area that needs work; to remove the reference to not-built-for purpose in G-S.1 - we need to address the non-retroactivity of existing built-for-purpose devices with software. It would be ideal to brainstorm a method by which we could eliminate the differentiation of devices in the language.

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 weights and measures inspector information that something has changed?
Yes, the Category III Audit Trail or other means of sealing.

How can the weights and measures 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 CC).

The sector believes that we should work towards language that would include a requirement similar to the International Organization of Legal Metrology (OIML) requirement in *NIST Handbook 44*. 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?)

NIST 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.**

Some of the sectors (Weighing, Measuring) have already agreed to put the below two paragraphs of text in Pub. 14. The sentence struck out was not included because Handbook 44 hadn't been yet altered.

From NCWM Publication 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 *NIST Handbook 44's* marking requirements.

In 2010, the sector recommended the following change to *NIST 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 20XX)**

(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 20XX)

Also the sector recommends the following information be added to *NCWM Publication 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)

There was some additional discussion on this item regarding where this new requirement was best located. It was suggested that the first sentence of G-S.1.d.(3) could be added as a clause to the base paragraph G-S.1(d) text, e.g. "*the current software version or revision identifier for* ~~***not built for purpose***~~ ***software-based*** *devices, which shall be directly and inseparably linked to the software itself;"* .

It also was suggested that the second sentence in G-S.1.d. (3) might be more suitable for *NCWM Publication 14*, as it describes more "how" than "what" the requirement entails.

In addition, the sector considered the following information to be added to *NCWM Publication 14* as explanation/examples:

- The current software identifier must be displayable/printable on command during operation (or made evident by other means deemed acceptable by G-S.1.)
- At a minimum, the software identifier must include a version/revision indication (1.02.09, rev 3.0 a, etc). It could also consist of / contain checksum, etc (crc32, for example).
- 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.

Other questions that are still outstanding:

- 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?

The sector recommended adding the following to *NCWM Publication 14* and forward to NTETC Weighing, Measuring, Grain Analyzer sectors for feedback:

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.

The list of acceptable menu text and symbols in Appendix A are intended to assist the labs in finding the certification number. The sector noticed no action by the sectors had been taken when this list was circulated for comment. We would like to remind them that we would like to have it reviewed. We feel that this belongs in, for example, the Weighing Device Pub. 14, page DES-22, Section 3; the Belt – Conveyor Scales, page BCS-10, Section 8.7; the Measuring Devices, page LMD-21, Section 1.6; the Grain Moisture Meter, page GMM-14, Section 1 (G.S.1); and Near Infrared Grain Analyzers, page NIR-8, Section 1 (G.S.1).

Conclusion:

If pushed, the Sectors agreed that a simple defining statement to qualify the class of devices that are to be included would be forwarded to the interested parties:

Software Based Device – Any device with metrologically significant software.

This agenda item will likely require less time during future meetings as it seems to be nearly finalized. Outstanding work remaining is to secure buy-in from the remaining sectors that have yet to adopt this recommendation to include in Pub. 14. Once those Sectors reach a decision, this item can be considered complete and removed from future Agendas.

3. Software Protection / Security

Source:
NTETC Software Sector

Background / Discussion:
See the 2014 Software Sector Summary for additional background on this item.

The Sector continued to develop a proposed checklist for *NCWM Publication 14*. The numbering will still need to be added. This is based roughly on R 76 – 2 checklist and discussions beginning as early as the October 2007 NTETC Software 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 California, Maryland and Ohio laboratories 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 February 2011, the North Carolina laboratory was also given a copy of the check list to try.

The labs using this checklist on a trial basis indicated that there was some confusion as to versions/wording. There may be more than one version in circulation. The version shown in this Summary shall be used henceforth.

During the discussion, Ed Payne (NTEP lab, MD) said that his impression is that this is at least making some of the manufacturers think about security, which they hadn't necessarily done in the past.

It was indicated that some more or better examples may be helpful to manufacturers, and that more guidance is needed. Clearer instructions could be part of the checklist, or it could be a separate document. The Sector would like additional feedback specifically regarding what portions of it are causing confusion.

Due to proprietary issues, the labs can't simply give us direct feedback from the companies they interact with. Darrell Flocken volunteered to obtain information from the labs, aggregate it, and remove any potential proprietary information issues.

The checklist as updated during the 2014 meeting:

1. Devices with Software

1.1. Declaration of the manufacturer that the software is used in a fixed hardware and software environment. **The manufacturer should indicate whether it's solely software or includes hardware in the system. Can the software be changed after the system has been shipped without breaking a seal? AND** ☐ Yes ☐ No ☐ N/A

1.2. Cannot be modified or uploaded by any means after securing/verification. **With the seal intact, can you change the software?** ☐ Yes ☐ No ☐ N/A

Note: It is acceptable to break the "seal" and load new software, audit trail is also a sufficient seal.

1.3. The software documentation contains:

1.3.1. Description of all functions, designating those that are considered metrologically significant. ☐ Yes ☐ No ☐ N/A

1.3.2. Description of the securing means (evidence of an intervention). ☐ Yes ☐ No ☐ N/A

1.3.3. Software Identification, including version/revision. **It may also include things like name, part number, CRC, etc.** ☐ Yes ☐ No ☐ N/A

1.3.4. Description how to check the actual software identification. ☐ Yes ☐ No ☐ N/A

1.4. The software identification is:

1.4.1. Clearly assigned to the metrologically significant software and functions. ☐ Yes ☐ No ☐ N/A

1.4.2. Provided by the device as documented. ☐ Yes ☐ No ☐ N/A

1.4.3. Directly linked to the software itself. **This means that you can't easily change the software without changing the software identifier. For example, the version identifier can't be in a text file that's easily editable, or in a variable that the user can edit.** ☐ Yes ☐ No ☐ N/A

2. Programmable or Loadable Metrologically Significant Software

2.1. The metrologically significant software is:

2.1.1. Documented with all relevant (see below for list of documents) information. *The list of docs referred to exists in agenda item 5.* ☐ Yes ☐ No ☐ N/A

2.1.2. Protected against accidental or intentional changes. ☐ Yes ☐ No ☐ N/A

2.2. Evidence of intervention (such as, changes, uploads, circumvention) is available until the next verification / inspection (e.g., physical seal, Checksum, Cyclical Redundancy Check (CRC), audit trail, etc. means of security). ☐ Yes ☐ No ☐ N/A

3. Software with no access to the operating system and/or programs possible for the user. This section and section 4 are intended to be mutually exclusive. Complete this section only if you replied Yes to 1.1.

3.3. 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 ☐ No ☐ N/A

3.4. Check whether the manufacturer has submitted a written declaration of the completeness of the set of commands. ☐ Yes ☐ No ☐ N/A

4. Operating System and / or Program(s) Accessible for the User. Complete this section only if you replied No to 1.1.

4.5. 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 Weights and Measures jurisdiction and type-specific parameters). **This is a declaration or explanation by the manufacturer.** ☐ Yes ☐ No ☐ N/A

4.6. 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). **This is a declaration or explanation by the manufacturer.** ☐ Yes ☐ No ☐ N/A

5. Software Interface(s)

5.7. Verify the manufacturer has documented:

5.7.1. **If software separation is employed, t**he program modules of the metrologically significant software are defined and separated. ☐ Yes ☐ No ☐ N/A

5.7.2.	**For software that can access the operating system or if the program is accessible to the user, t**he protective software interface itself is part of the metrologically significant software.	☐ Yes ☐ No ☐ N/A
5.7.3.	The functions of the metrologically significant software that can be accessed ~~**via the protective software interface**~~.	☐ Yes ☐ No ☐ N/A
5.7.4.	The **metrologically significant** parameters that may be exchanged ~~**via the protective software interface**~~ are defined.	☐ Yes ☐ No ☐ N/A
5.7.5.	The description of the functions and parameters are conclusive and complete.	☐ Yes ☐ No ☐ N/A
5.7.6.	There are software interface instructions for the third party (external) application programmer.	☐ Yes ☐ No ☐ N/A

The Sector discussed examples, such as the upgrade of application programs and how these changes would affect audit trails and version numbers. It should be clear that if the upgraded software doesn't affect anything metrologically significant, then it's irrelevant for the purposes of this checklist. On the other hand, if it does affect metrologically significant functions or parameters, it should be tracked and/or identified somehow.

Conclusion:
The revised checklist will be reviewed and sent to the labs for use. The next step will be to forward it to the four other sectors; we can report that the labs have tried using it on a trial basis and we're ready to recommend it for Pub. 14 with the modification suggested here, such as the removal of the Type P / Type U wording. We look forward to the Weighing Sector's feedback during this joint meeting.

4. Software Maintenance and Reconfiguration

Source:
NTETC Software Sector

Background / Discussion:
After the software is completed, what do the manufacturers use to secure their software? The following items were reviewed by the sector. *Note that agenda Item 3 also contains information on Verified and Traced updates and Software Log.*

1. Verify that the update process is documented (OK)
2. 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.

3. 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

4. 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 weights and measures 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.

Note: It's possible that the Philosophy of Sealing section of NCWM Publication 14 may already address the above IF the definitions of Verified and Traced Updates (and the statement below) were to be added. The contrary argument was that it may be better to be explicit).

Use of a Category 3 audit trail is required for a Traced Update. A log entry representing a traced software update shall include the software identification of the newly installed version.

The sector recommended consolidating the definitions with the above statement thus:

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**~~ Category 3 audit trail. The audit trail entry shall include the software identification of the newly installed version.

The sector recommended that as a first step, the following be added to *NCWM Publication 14*:

The updating of metrologically significant software, including software that checks the authenticity and integrity of the updates, shall be considered a sealable event.

Mr. Truex, NTEP Administrator, believes the above sentence is unnecessary since it's self-evident. It was agreed to ask the other sectors for feedback on the value of this addition.

Though the sector is currently recommending only the single sentence be incorporated into *NCWM Publication 14* for the time being, ultimately, the sector may wish to advance the remaining language of the original item submission.

At the 2013 meeting, the Sector had no information indicating that the other sectors had yet been approached for feedback on the value of the addition of the proposed sentence. This sector would still like the other sectors to evaluate this for inclusion in Pub. 14. We'd also like to include some description indicating that an existing audit trail should be protected during a software update, though that may already be a requirement. This does appear to be addressed in the Requirements for Metrological Audit Trails Appendices in Pub. 14.

Last year's Weighing Sector feedback indicated they were opposed because:

1. It would change the methods of sealing (category 1, 2, and 3 audit trails) and require a change to Handbook 44.
2. It's not clear that the requirement for authenticity and integrity of the updates is limited to metrologically significant software.

The other sectors were concerned about this as well.

Legacy equipment that's still being manufactured might need to be changed to meet this obligation since their audit trails wouldn't necessarily indicate that the software has been updated.

Reference G-S.8., which is rather loose. Pub. 14 goes into much more detail about what is metrologically significant.

Darrell Flocken referred to Handbook 44, the Scales code – the event logger category 3 – the software is not a parameter. It's not so much that the software would be tracked, as the fact that it has not been in the list of sealable parameters is the concern. It sounds like this may be a procedural issue – sections of Handbook 44 may need to be altered before the sectors can add this suggestion to Pub. 14.

In 2010 the Software Sector had considered the following:

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.

Ambler Thompson suggested that the notes under G-S.8. could be amended to include software updates as a new example. Rick Harshman recommended having it as a stand-alone item, such as discussed in 2010.

This could possibly be tied back to G-S.2.

What is the sealable parameter? Is it the software version / revision? Currently all of the parameters are user-selectable, which would make this unique.

If the general code in Handbook 44 is amended to include this in some form, it applies to everything. The various sectors don't need to add to their specific sections of Handbook 44.

Darrell Flocken suggested that we try to come up with a declaration of intent and see how the sectors respond. Doug Bliss will add it to the existing presentation. Jim Truex thought it might be valuable to obtain the opinion of the S&T Committee. The Legal Metrology group should be asked, "Is a software change that updates metrologically significant software a sealable event?" Rick Harshman can obtain an answer from them.

Ambler Thompson raised a concern about the fact that at this point none of the suggested wording requires that the software identifier be unique, i.e. a change to the metrologically significant software should require a change to the

software identifier. You could perhaps infer it from the requirement that it be inextricably linked to the software, but that isn't clear. Jim Truex thinks this will eventually need to be addressed, but not right now.

Conclusion:
After the discussion during the 2014 joint meeting, we revised the wording of the proposed G-S.9 to reflect some of the concerns heard from the other Sectors and interested parties:

G-S.9. Metrologically Significant Software Updates

A software update that changes the metrologically significant software shall be considered a sealable event.

The Sector still feels that explicitly requiring the metrologically significant software to be given at least the same level of protection as metrologically significant parameters is the best approach. We look forward to feedback from the S&T Committee and other Sectors on this proposed change. The Software Sector still would like to consider the issue of audit trail protection; there is some doubt as to whether the existing language is sufficient as it does not address the integrity of the audit trail during a software update, etc.

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. What gets submitted? What requirements and mechanisms for submission should be available? Validation in the laboratories - all required subsystems shall be included to be able to simulate the system as installed.

Mr. Roach, California Division of Measurement Standards, 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 laboratories and scale manufacturers indicated that this is not usually done for scale evaluations.

Since the NTEP Committee passed the related item at NCWM Annual Meeting we will continue to work on this. Mr. Truex, NTEP Administrator, 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.

Comments given at the meeting indicate that current practice does not require anything different for software / software based devices compared to any other type approval. It was also noted that for international applications, OIML D-31.6.5 states, "The approval applicant is responsible for the provision of all the required equipment and components." This would likely also be the policy of NTEP.

Since the checklist is still being tried out by some of the laboratories, the sector is not quite ready to develop this fully. Some documentation that eventually might be required by applicants could include (from WELMEC doc. 7-2 Issue 4): This is the list of documents referred to in the checklist.

- A description of the software functions that are metrologically significant, meaning of the data, etc., e.g. an architecture diagram or flowchart.

- The software identification (version, revision, etc.) and how to view it.
- An overview of the security aspects of the operating system, e.g. protection, user accounts, privileges, etc.

Darrell Flocken and Jim Truex reviewed existing documentation required for obtaining certification in Pub. 14, administrative policy, and the application, to see what is already required. Administrative policy 9.1.7 was where this was found:

- Engineering specification
- Operating descriptions that characterize the type

NTEP evaluators already have the authority to request whatever documentation they need. We can provide them with a list of documents that we think would assist the evaluator in his job and also give the manufacturer a good idea of what they should be capable of providing.

Darrell Flocken suggested that this list could be added to administrative policy 9.1.7 in Pub. 14. Jim Truex suggested it could also be added to the application.

If we combine the two lists, it might appear as something like this:

- A description of the software functions that are metrologically significant, meaning of the data, etc., e.g. an architecture diagram or flowchart.
- A description of the user interface, communication interface, menus, and dialogs.
- The software identification (version, revision, etc.) and how to view it.
- An overview of the system hardware, e.g. topology block diagram, type of computer(s), type of network, etc, if not described in the operating manual.
- An overview of the security aspects of the operating system, e.g. protection, user accounts, privileges, etc.
- The operating manual.
- Engineering specification.
- Operating descriptions that characterize the type.

A statement could be made along the lines of, "If not included in the operating manual, provide the following, as applicable."

After the last sentence in 9.1.7, this could be added:
As part of the type evaluation submission, the following information should be provided for software-based devices:

- **A description of the software functions that are metrologically significant, meaning of the data, etc., e.g. an architecture diagram or flowchart.**
- **The software identification (version, revision, etc.) and how to view it.**
- **An overview of the security aspects of the operating system, e.g. protection, user accounts, privileges, etc.**

These documentation requirements will be considered as input for requirements that will eventually appear in *NCWM Publication 14* and the application paperwork. Further work by the sector to develop the *NCWM Publication 14* requirements is needed, after more input from the labs is gathered. The Sector recommends including the above bulleted list as an introduction to the checklist as part of our recommendation to include the checklist from agenda item 3 in Pub. 14. As a description of the accuracy of the measuring algorithms, simply declaring the type and class being aimed for may be sufficient. This list should reflect the needs of the labs for an evaluation. The bulleted list and the paragraph before it should be brought to the labs for an initial review and their input.

Conclusion:
The Sector needs to discuss any input from the labs and finalize this list, prior to submitting the list to the other Sectors for incorporation into Pub. 14.

6. Training of Field Inspectors

Source:
NTETC Software Sector

Background:
During discussions at the 2009 NTETC Software Sector 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.

California has an Examination Procedure Outline (EPO) that begins to address this. Use *California 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 *NIST Handbook 44*.
- Reference materials / information sources

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

9. Motion detection.
   9.1. 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)
10. Over capacity.
    10.1. 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.

Mr. Jordan, California Division of Measurement Standards, is already doing something similar, and he may be able to assist. Mr. Roach, California Division of Measurement Standards, will talk to him to see whether they're available. In addition Mr. Parks, California Division of Measurement Standards, is based in Sacramento and a potential resource. If the meeting is held in Sacramento next year, they may be able to attend.

Mr. Truex, NTEP Administrator, pointed out that the PDC would also be a valuable resource on this subject. Mr. Pettinato, Co-Chair, will contact them.

**NIST Handbook 112- Examination Procedure Outline for Commercial Weighing and Measuring Devices.*

The PDC is focused on training sessions at the moment, so it's unsure how much time they'd have to review this currently.

Discussion:
California has some direction for inspectors regarding third party software. Mike Wedman is currently tasked with revising and expanding some of California's documentation on the subject, and we asked him to share it with us when it is complete.

Is it California's Handbook 112? Mike Wedman said they don't have such a handbook; it's in their device enforcement documentation.

NIST Handbook 112 doesn't have anything specific to software, and Jim Truex says that this handbook has actually been out of production for years. Its last edition was in 2002. There's an online copy of it that was searched to verify if there was anything software-specific in it, and nothing was found.

Jim Pettinato proposed that we put together a group to begin writing something ourselves, and Jim Truex stressed that it needed to be written at a level that the field inspectors would find useful.

Conclusion:
We'll wait until Mike Wedman has completed his work on the California EPO. Jim Pettinato, Teri Gulke, and Mike Wedman volunteered to work on this offline.

The Sector would like to continue exploring means by which it can be of assistance in training of field inspectors as software and electronic systems become more and more prevalent in their daily tasks.

It was also suggested we contact Ross Anderson, a paid consultant working with the PDC committee, to ask his opinion on how the Software Sector could best proceed to assist in the training of field inspectors. The Sector chair, Jim Pettinato, will act as primary point of contact for this communication.

Conclusion:
The Sector would like to continue exploring means by which it can be of assistance in training of field inspectors as software and electronic systems become more and more prevalent in their daily tasks.

NEW ITEMS

7. Retrieval of Audit Log information

Source:
Adam Oldham, Gilbarco

Background/Discussion:
The current requirements for a Category III audit trail include printing of log on demand. However, many devices are approved standalone and can be connected to systems that are approved standalone. How could Category 3 audit trail mechanisms be approved in situations where multiple devices need to work together to attain it? How can a device maintain Category 2 and 3 approvals in this scenario? What alternatives to printing can be considered as potentially valid solutions? (files, laptop, flash drive, etc).

8. Next Meeting

Background:
The sector is on a yearly schedule for NTETC Software Sector Meetings. Now that we've adopted a joint meeting system, the next Sector joint meeting will coincide with one of the remaining Sector meetings.

9. 2015 NCWM Interim Meeting Report

There was one item on the NCWM S&T Committee Agenda for the 2013 NCWM Interim Meeting related to work done by the NTETC Software Sector. *2013 Publication 15* S&T Item 360-2 relates to the 2013 NTETC Software Sector Agenda Item 1: Marking Requirements.

10. 2014 International Report

Dr. Ambler Thompson, NIST, Office of Weights and Measures (OWM), will provide a synopsis of international activity that relates to the work of the Sector. Software Sector Co-Chair Mr. Jim Pettinato will summarize the discussions that took place at the European Cooperation in Legal Metrology (WELMEC) WG7 meeting in Dec. 2013.

Highlights of interest to the NTETC Software Sector:

- New WELMEC 7.2 draft document circulated for comment by WG7
- R-117 working group

Appendix A – Acceptable Menu Text/Icons for Weights Measures information

Permitted Menu Text examples	*Permitted Icon shape examples*	*Essential characteristics*
Information Info	  	Top level menu text or icon • 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.
Help **?**	 	Top level menu text or icon • 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.
Metrology Metrological Information	 	Top or second level menu text or icon • 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 circular, rectangular, or rounded rectangle border. • If present, the activation of this menu text/icon must recall at a minimum the NTEP CC number.
NTEP Data N.T.E.P. Certificate		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?
Weights & Measures Info