National Type Evaluation Program (NTEP)
Software Sector Meeting Summary

August 23rd, 2018 / Louisville, KY
(in conjunction with the Weighing Sector)

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 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 non-retroactive are printed in ***bold faced italics***.

Table A
Table of Contents

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
NCWM	National Conference on Weights and Measures	S&T	Specifications and Tolerances Committee
NIST	National Institute of Standards and Technology	SMA	Scale Manufacturers Association
NTEP	National Type Evaluation Program	WELMEC	European Cooperation in Legal Metrology

Details of All Items
(In order by Reference Key)

WELCOME

Since the Software Sector meeting was a joint meeting with the Weighing Sector, some time was set aside to meet and greet both new and familiar faces.

STATUS REPORTS – RELATED NCWM AND INTERNATIONAL ACTIVITY

Attendees of the 2018 NCWM Interim and Annual Meetings were 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. Results related to items on our Agenda were of particular focus.

Dr. Katya Delak, NIST, Office of Weights and Measures (OWM), provided a synopsis of international activity that relates to the work of the sector. (See appendix B)

JOINT SESSION PROGRESS REPORT, ACTIVE ITEMS OF MUTUAL INTEREST

This is the second joint meeting of these Sectors. To make sure we make the most of the time a quick review of the agenda items from both Sectors will be held to identify those 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

The Software Sector Technical Advisor gave a brief presentation outlining the problems the Sector had been asked to consider and some of the consensus that has been reached to date.

CARRY-OVER ITEMS

1. Software Identification / Markings

Source:
NTEP Software Sector

Background:
See the 2017 Software Sector Meeting Summary for more background on this item.

Since its inception, the sector has wrestled with the issue of software identification and marking requirements. Numerous changes to the HB44 language were attempted and though support for the concepts was expressed, resistance to specific language made the course difficult. Finally, in 2015 in a joint meeting with the Measuring Sector, some additional fine tuning on the recommended changes to G-S.1 was done and we felt we had addressed everyone's concerns and had language ready to be voted upon for adoption. The recommended language is below.

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*~~ <u>*software*</u>*;*
  [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, 2022;**
  ~~*[Nonretroactive as of January 1, 2004]*~~
  (Added 2003) **(Amended 2017)**

  (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 2017)

   ii. ***continuously displayed or be accessible via the display. Instructions for displaying the version or revision identifier shall be described in the CC. As an alternative, permanently marking the version or revision identifier shall be acceptable providing the device does not always have an integral interface to communicate the version or revision identifier.***
   [Nonretroactive as of January 1, 2022]
   (Added 2017)

   *(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) (Amended 2017)

   (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 2017**)

The amended proposal was Accepted as a Voting item at the 2016 Interim meeting and passed at the 2016 Annual Meeting.

Since future work on this item depends on the expiration of the window for compliance (2022), the Sector agreed to table this item until 2020/2021, when we can again begin to discuss further modifications with the eventual goal of eliminating G-S.1.1 and the differentiation between built-for-purpose and not-built-for-purpose.

In July of 2016 the MDMD Work Group addressed some of these issues pertaining to software running on small devices such as phones that have very small screens. They discussed prioritization of what needed to be displayed, such as CC so that the remainder of the information can be looked up.

Discussion:
The group estimated the scope of work remaining and decided it is not necessary to start working on G-S.1 yet.

Conclusion:
This agenda item remains tabled until 2020.

## 2.	Identification of Certified Software

Source:
NTEP Software Sector

Background:
See the 2017 Software Sector Meeting Summary for more background on this item.

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 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) 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, 2022;
  (Added 2003) (Amended 2016)

> *(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 2016)*
> >
> > ii.	*continuously displayed or be accessible via the display. Instructions for displaying the version or revision identifier shall be described in the CC. As an alternative, permanently marking the version or revision identifier shall be acceptable providing the device does not always have an integral interface to communicate the version or revision identifier.*
> > *[Nonretroactive as of January 1, 2022]*
> > *(Added 2017)*
>
> *(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) (Amended 2017)
>
> **(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 recommended 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)

This original item was eventually withdrawn, and the proposal was split into two separate items. The critical need to include version/revision in the marking requirements for all software-based devices was pushed forward and passed independently.

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 previously brought up that have not really been satisfied to date are:

- 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?

Regarding field inspection and locating the required information: 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).

Tina Butcher mentioned that the Weighing Sector has a Weighing Checklist that has a similar set of approved symbols, so the examples shown in Appendix A would be in line with their current practice.

Since the recommended new G-S.1 language was voted on and adopted in 2016, we can now move forward on this item and consider adding to *NCWM Publication 14* the specifics that we have been discussing related to presenting the software identification.

Darrell Flocken asked whether it's a specification or information. That would determine whether it should belong in HB44 or only in Pub. 14. One possibility is below:

> **(3) The version or revision identifier shall be directly and inseparably linked to the software itself.**
>
> **Note: 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)**

Concern was expressed that this could cause confusion with field inspectors. Software separation isn't something that's intended to be useful in the field, it is intended to ease type approval and software maintenance release processing. - This would lend weight to the argument of keeping it in Pub. 14.

If the Sector desires to include this in Pub. 14, we would need to identify all the sections where this concept would need to be added. The Software Sector doesn't have the authority to add it to the other sectors' Pub. 14's. Darrell Flocken reported that a note regarding the concept of software separation has already been added to several of the various Pub. 14 sections.

The Chair proposed that we table Agenda Item 2 until 2021, and that we continue to pursue implementing the checklist in Pub. 14. Darrell Flocken suggested that the Software Sector recommend that the various sectors adopt this for their Pub. 14's. It would take a year or so, to make it through all the various sectors. A note could be added saying that a device can't be rejected if it doesn't meet this requirement in the checklist until 2022. It was agreed that we would table this item until the 2021 meeting, at which time we will propose the following (updated) wording for the 2022 Pub. 14:

3. Additional Marking Requirements- Software

Identification of Certified Software:
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.

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.

At the 2017 joint meeting, the MDMD Work Group discussed adding the section regarding linking of identifier to the software to their section in Pub. 14. There were no objections, so Darrell Flocken said he'd add it for next year's publication. A note shall be added that this is voluntary until 2022.

Also, we further discussed the idea of software separation, especially in how it pertains to the difference between the terms "metrologically significant" and "legally relevant". Some legal requirements have nothing to do with metrology. There is a difference in how the US regards this (since each state can have different legal requirements) vs. the philosophy in Europe. There isn't a definition of "metrologically significant" in Handbook 44, but Publication 14 has a description of all the parameters that needs to be sealed, which includes both metrologically significant and legally relevant parameters.

A definition of "metrologically significant" could be helpful, but Darrell Flocken suggested that we make sure it doesn't contradict VCAP's administrative policies.

Handbook. 44 does contain a definition for "metrological integrity".

Type evaluation is the time at which decisions are made regarding which exact parameters are sealable. According to Jim Truex, the US has never been able to come to a consensus on this subject.

Jim Pettinato suggested that we work offline to generate a description intended to provide guidance on what we mean by "metrologically significant". Jim Pettinato, Doug Bliss, Dr. Ambler Thompson, and Kevin Detert volunteered to make up a subcommittee to address this subject.

We also considered the issue of having to adopt a general software requirement to multiple sections of Publication 14 to address essentially the same requirement for each category of device separately. The idea was floated by the

Sector that perhaps a new section should be added to Publication 14 specific to software that applies to all metrologically significant software in all devices types that might contain such. Rather than formally suggesting this be done, we decided to informally run the idea past the Specifications and Tolerances committee. That way, if there was little interest or strong objection, we wouldn't waste time generating a draft.

How the Sector decides to progress on this item is dependent on the Board's decision regarding a separate section on software for Publication 14. If the decision is to grant the Sector's wishes, then we would start crafting language for our new Section. Otherwise, we can consider the suggested language put forth in the last meeting.

Discussion:
If the Software Sector gets its own section in Publication 14, we may not need to alter HB44 regarding this specific agenda item, according to Darrell Flocken. There is a general NTEP technical policy within Pub. 14, which may be the best place to address communicating the requirements for evaluation of software and software-based devices and the need to include type compliant software version/revision information on the certificate of conformance.

Conclusion:
This agenda item remains tabled until a decision on the direction for Publication 14 is made by the NTEP committee.

3. Software Protection / Security

Source:
NTEP Software Sector

Background:
See the 2017 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 NTEP 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.

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. <u>This section and section 4 are intended to be mutually exclusive. Complete this section only if you replied Yes to 1.1.</u>

   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. <u>Complete this section only if you replied No to 1.1.</u>

   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). **<u>This is a declaration or explanation by the manufacturer.</u>** □ 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). **<u>This is a declaration or explanation by the manufacturer.</u>** □ Yes □ No □ N/A

   4.7.   Check whether the manufacturer has provided a description of the software functions that are metrologically significant, meaning of the data, etc., e.g. an architecture diagram or flowchart.

   4.8.   Check that there is guidance related to the software identification (version, revision, etc.), how to view it, and how it is tied to the software.

   4.9.   Check that the manufacturer has provided an overview of the security aspects of the operating system, e.g. protection, user accounts, privileges, etc.

5.   Software Interface(s)

   5.10.   Verify the manufacturer has documented:

   5.10.1.   **<u>If software separation is employed, t</u>**he program modules of the metrologically significant software are defined and separated. □ Yes □ No □ N/A

   5.10.2.   **<u>For software that can access the operating system or if the program is accessible to the user, t</u>**he protective software interface itself is part of the metrologically significant software. □ Yes □ No □ N/A

   5.10.3.   The functions of the metrologically significant software that can be accessed ~~via the protective software interface~~. □ Yes □ No □ N/A

5.10.4. The **metrologically significant** parameters that may be exchanged ~~via the protective software interface~~ are defined. ☐ Yes ☐ No ☐ N/A

5.10.5. The description of the functions and parameters are conclusive and complete. ☐ Yes ☐ No ☐ N/A

5.10.6. There are software interface instructions for the third party (external) application programmer. ☐ Yes ☐ No ☐ N/A

This checklist was discussed during the 2017 NTEP lab meeting, and Darrell Flocken received two submissions. One response was very helpful, and the other one said that everything was N/A pertaining to their device, except for a bit regarding calculating the CRC and sealing. In general, the labs said that even when they hand the checklist out, they usually don't get it back. We're pushing the labs to be a bit more proactive.

MDMD has only one lab. All the labs have been given a copy of the checklist, but we're not sure whether their lab has found it helpful.

Again, the benefit of a separate section of Pub. 14 for software is evident for this agenda item.

Discussion:
Darrell Flocken shared (anonymously) some results from the NTEP labs. There were three checklists returned over the last year. One submission included commentary from the company responding to the checklist regarding the difference between embedded systems versus open systems. That submitter used the WELMEC guidelines.

Darrell Flocken reported that, in general, it seems that companies are starting to respond more thoughtfully to the checklist. In prior years, it seemed like they'd simply just checked everything off.

There appears to be a gap between the companies responding to the checklist and the NTEP labs perceiving use in the responses. There's a need for an explanation of what responses to the various questions mean to the NTEP lab inspectors, which should be in plain language, similar to the 2014 presentation on the general concepts of the Software Sector's work.

We also discussed the need to formalize how the checklist is distributed. Cardinal reported that they hadn't received it as part of a type approval application packet, and it seems they're not unique.

It was mentioned that Mexico now considers many things "software", including PAL's, GAL's, etc. At one time we tried to craft our own definition of software without much luck. We may be able to reference an international definition.

The VCAP program should reference the software identifier and version/revision, but until NTEP is consistent on how the software identifier and version/revision is recorded on the CC, this isn't feasible. VCAP was originally intended as an assessment whether a particular implementation meets type.

Conclusion:
Darrell Flocken will work on formalizing how the checklist is distributed. We will also need to work on crafting an explanation for the NTEP labs as to how the answers to the checklist benefit their inspectors.

4.   NTEP Application for Software and Software-based Devices

Source:
NTEP Software Sector

Background:
The purpose of initiating this item was to identify issues, requirements and processes for type approving device applications, specifically for not-built-for-purpose software since it is now explicitly allowed. It was suggested that it may be useful to the labs to devise a separate submission form for software for these applications. 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.) , how to view it, and how it is tied to the software.**
- **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.

There may be concerns with disclosure of proprietary information. Jim Truex says that the labs already protect other proprietary information. If the information provided is sufficiently high level, even theft of the data shouldn't cause too much of a concern.

While working on writing requirements for Pub. 14 from the checklist we've designed, we considered altering the second bullet point in our proposal for 9.17, so that it will require a description of how the software version or revision identifier is tied to the software itself.

At the 2016 meeting, it seemed that the goal of this agenda item has somewhat shifted back to the original purpose, which is how do we communicate to applicants the expectations related to software based devices? Diane Lee suggested we review the OIML requirements for documentation. The comment was made from the floor that OIML may go further than we are currently prepared to recommend. Jason Jordan expressed his opinion that moving forward with this item will be helpful for the labs. Darrell Flocken and Jim Truex think this should be added to the Application section. If limited to that section, it shouldn't require approval from any of the other Sectors. Doug Bliss suggested that it might be easier to provide examples that do not meet acceptable standards.

9.3 of Administrative Policy describes how to prepare for type evaluation. It might be better to add our suggested wording there instead of 9.1.7. Jim Pettinato found a page on NCWM's website that describes what's needed for a type evaluation. He suggested we could add our checklist to the list of documents there. The NTEP Committee decides what's posted on the website.

Jim Truex thinks we may need to come up with a list of software parameters and functions that are required to be protected. This will be a lot of work, but it may be the right answer, generating a separate section in Pub. 14 (and/or Hdbk. 44) pertaining specifically to software.

The group discussed whether a list of sealable parameters should include device-specific parameters as well as software-specific parameters (e.g. CRC), or only the latter. The latter should be a fairly short list, including such parameters as:

- Replacing software
- Access to critical sections of the software

Historically, requirements for software-only applications haven't been as high as requirements for software applications that include hardware. The number of software-only applications has increased dramatically over the last few years.

The topic arose once again that we propose to the NTEP Committee we add a software specific section to Pub. 14. We may not know exactly what we want to include, but we could get the ball rolling by presenting a set of examples of situations that show the need. Jim Truex thinks that the NTEP Committee will ask whether this needs a change to Hdbk. 44. We need to address that in any sort of presentation we make to them. Dick Suiter suggested that we add a requirement to HB44 that the software be sealable, which is a bit of a difference from making changes to software evident. G-S.2. appears to address this in its mention of avoiding facilitation of fraud. The philosophy of sealing and method of sealing also cover this. We want to recommend adding a separate section to Pub. 14 for software, a list of sealable parameters, explain that going to the separate sectors isn't working, and explain that manufacturers will need to address both our software section as well as application-specific portions of Pub. 14.

We provided an outline for the proposed Pub 14 section prior to the NTEP committee meeting in two weeks, to gauge their opinion as to whether this is a viable approach. No action was taken until this year's Annual meeting, where the new NTEP committee chair guaranteed he would make it a priority to make progress on the proposal.

Discussion:
Darrell Flocken is trying to get an invitation for the Software Sector to the NTEP labs meeting in April, to be able to answer any of their questions and have a discussion on how software could be addressed more formally in submissions from applicants, and how the Sector can support the labs in their evaluations.

We need to provide a recommendation for an administrative change for the NTEP Committee's approval, via Darrell Flocken and Jim Truex. Since this would be a recommendation related to the policy, not the device code, it simplifies the process.

If the Software Sector does get its own section within Publication 14, the text may gain more notice if it's within that section rather than the general administrative policy; however, if it's within the general administrative policy, it wouldn't be hard to move it to the Software Sector's section of Pub. 14.

Conclusion:
The Software Sector recommends that this text be added as part of the existing 9.1.7 in Pub. 14 Administrative Policy:

Additionally, 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.), how to view it, and how it is tied to the software.**
- **An overview of the security aspects of the software(s), e.g. protection, user accounts, privileges, platforms, etc.**

Jim Pettinato will craft the formal proposal. Darrell Flocken asked the NTEP lab evaluators in attendance what they need from the Software Sector to help them interpret the documentation they will receive from the manufacturers in response to this requirement.

5.   Training of Field Inspectors

Source:
NTEP Software Sector

Background:
During discussions at the 2009 NTEP 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.

It was suggested by Jim Truex and Darrell Flocken we make it part of our report as an attachment or an appendix of the meeting minutes. Then we can send out an email notifying the Software Sector members as to where to find it.

Alternatively, we could forward the document to the PDC Committee, tell them it was our starting point, and ask them for their suggestions.

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.

Jim Pettinato will contact Ross Anderson regarding the PDC Committee, offering the Software Sector's assistance in continuing to develop training pertaining to software.

Discussion:
Jim Pettinato is now a member of the PDC (Professional Development Committee), so he will be able to pass on any suggestions we may make. The PDC is making an effort to provide training modules/videos accessible to anyone, so everyone is on the same page. Darrell Flocken suggested that as these training modules are updated, we should provide relevant input.

There is a national EPO from NIST Office of W&M, HB112. Darrell Flocken recommended that we approach NIST regarding adding text regarding software. There are not EPO's for every equipment type. Rick said that HB112 is updated every year.

Darrell Flocken suggested that we attend the regional meetings to gain feedback on the sort of guidance the field inspectors need.

Rick said that the most value to the field inspectors would be to identify for them different means that software can be used to manipulate the metrological system. In particular, how can someone attempt to cheat using software?

Doug Bliss quickly reviewed HB112 and reported that the majority of it has to do with safety guidelines. Rick said that there are numerous references to HB44, which pertain more to the requirements for the inspections. HB112 has appendices that include step-by-step procedures. We may want to consider crafting our own procedure for a new appendix.

Adam mentioned that Mexico and Brazil (and China, to an extent) have a requirement for manufacturers to supply an auditing document when they submit for a type approval. This would be a big change for NCWM.

Conclusion:
It was suggested that perhaps a presentation on this subject at the main and regional NCWM meetings might be a good starting point. Jim Pettinato suggested an entry in the NCWM newsletter, targeted to inspectors, would also help. The newsletter is submitted quarterly. Darrell Flocken confirmed that submissions for the next newsletter are due January 15th. A helpful newsletter article could describe how to find the CC for a system that includes software. Brian Duncan volunteered to write a first draft.

Jim Pettinato suggested that members of the Software Sector download and review HB112, so that we can have a better idea regarding where we might best target additions to the text. We could have an online meeting to discuss and update the agenda prior to our next Software Sector meeting. Darrell Flocken or Jim Pettinato can set up an online meeting, which may be in late September.

Recommendations for changes to HB112 should go to Tina Butcher.

6.   Use of GPS Receivers and Mapping Software for Trade (e.g. fare determination)

Source: Software Sector

Background:
There were a few presentations at the Interim Meeting on this subject. The 2016 Annual Meeting archive (Denver 2016) includes a presentation from Lyft from that meeting.

Ambler Thompson has discussed this subject with European officials. One issue is traceability of the time stamp(s). You can also calculate velocity based upon the phase shift of the GPS signal, though it requires a high-end, survey-grade GPS receiver ($50k each). Car companies can use these devices to obtain a great deal of data.

Uber and Lyft claim that they are not billing upon GPS data, but rather a pre-negotiated contract based upon distance, time, and type of vehicle. Doug Bliss has been told that the bill is based upon the starting GPS location from the driver's phone, the ending GPS location from the same phone, and a calculation of the shortest distance from Google Maps. If the driver's phone doesn't have a great GPS receiver, or if the reception is bad so it's relying upon cell towers, etc., that's a problem. We're also not sure just how accurate Google Map's route calculation is. Also, Google Maps is a disinterested third party whose database is being used for a purpose they didn't specifically authorize.

Doug Musick reported that the Uber contract is based upon a unit price, though they do provide an estimate to the customer.

At the 2017 meeting, it was determined that at this stage there isn't much for the Software Sector to do on this subject. Jim Pettinato asked that the members of the Software Sector review the proposals in Pub. 16 pertinent to this issue.

Discussion:
Katya Delak said that OIML may attempt to address this issue as well, probably within the next few years.

The 2018 Taxi Meter code has been changed, and approvals are not being generated for GPS-based technology. OBDC-based systems have been accepted for type approval. There are no Pub. 14 guidelines.

Conclusion:
As in 2017, it doesn't seem that there is anything the Software Sector needs to address on this subject currently. After some discussion, the members of the Software Sector agree that this agenda item can be removed from future meetings.

NEW ITEMS

7.   New Publication 14 Section specific to Software

Background:
In the last few meetings, it has been recognized that there is significant difficulty aligning the various Sectors to maintain continuity and agreement in what changes go into each Sector's section of Publication 14. It also impedes the progress the Software Sector can make as we have to explain/defend our positions multiple times to different audiences. Hence, it was proposed while working on several of the carry-over items that a better process might be to segregate the software-specific requirements for type evaluation into a separate section, controlled by our Sector. Hence, the Sector agreed to forward a recommendation to the NTEP committee to grant the Sector a software-specific section of Publication 14. Accompanying this recommendation was an outline of the potential content that would be included. Full text of the recommendation is below:

Current state:

There is no single Publication 14 device category in which to place software-specific requirements, design considerations related to software or test procedures specific to software. Since most modern measurement devices contain software, to appropriately address any concerns each section of Publication 14 must include all software considerations. Further, each device section has a different governing Sector, which makes the process of change an exercise in convincing each Sector to make needed additions while keeping those additions harmonized across Sectors; an effort that has proven very difficult and time consuming.

Since the Sectors don't meet simultaneously, often our submissions are accepted into each Sector's agenda, then one will adopt and another will have comments or reject the request, leading to inconsistent treatment of software between classes of device.

Internationally, OIML and WELMEC have adopted a similar approach by segregating software recommendations/requirements into a standalone document or documents, and that approach aids both evaluators and submitters by consolidating the requirements for software into a single section that can be shared with developers.

Software Sector Proposal:

Create a Publication 14 Software category, which includes requirements, considerations and test procedures common to all software-based devices, including software-only products. Such a section might include the following:

1. *Models to be submitted for evaluation*
    a. *Determining scope of software to be approved*
        i. *Measurement and presentation*
        ii. *Calculations based on a measured value*
        iii. *Manual entry of measured value*
        iv. *Other*
    b. *Application of software may lead to additional Pub. 14 section consideration*
    c. *Minimum computing requirements statement*
2. *Software Identification*
    a. *Appropriate means of 'marking' metrologically significant software*
    b. *Software Separation and marking consequences*
    c. *Relationship between software and software identifier*
    d. *Presentation of software identifier*
        i. *Example icons and menu text*
        ii. *Exceptions*
3. *Protection against unauthorized software change*

          a. *How is software "sealed"?*
          b. *Remote software update considerations*
          c. *Audit trail (if employed) requirements for software updates*
    4. *Accuracy of data calculations*
          a. *When to stop evaluating calculations & data manipulation*
    5. *Software Evaluation Checklist*

Future Topics
    1. *Distributed software considerations*
          a. *Securing communications between metrologically significant distributed software modules or components of a system*

It seems likely that action may take place within the next year, and that means the Sector faces the task of quickly publishing the text of a new section. It is hoped that some time could be spent developing the outline further and identifying content already created/included in other sectors that would need to be migrated to the new Section.

Discussion:
James Cassidy assured Jim Pettinato at the Annual Meeting this summer that they will take this under consideration. Darrell Flocken reported that the delay was due to not receiving input from the various sectors, either for or against. Darrell Flocken and Jim Truex are urging the various members to voice their opinion.

Some of the other sections of Pub. 14 already have software requirements, and there have been some questions regarding whether this would be removed and placed in the new software section. Jim Pettinato clarified that device-specific software requirements would remain where they are. The new software section would be more generic in nature.

SMA representatives indicated that their group may possibly review this proposal and come up with a position on the subject.

In the international community, there are general guidelines for software, such as in D-31, which are then adapted and implemented in the device-specific documents.

The starting point for the new software section in Pub. 14 would be the software checklist.

The new section would not be intended for software-only applications; it would be intended for anything metrological that has software.

There should be an introduction explaining when this section applies. "This code applies to the following… This code does not apply to the following…"

    1. *Scope of application – any device of whatever type that contains software must meet the requirements herein. This includes both built-for-purpose and not-built-for-purpose software.*
    2. *Materials to be submitted for evaluation*
          a. *Determining which software modules need to be approved*
               i. *Measurement and presentation*
               ii. *Calculations based on a measured value*
               iii. *Manual entry of measured value (e.g. measurement data rather than a measurement result)*
               iv. *Other*
          b. *Application of software may lead to additional Pub. 14 section consideration*
          c. *Minimum computing requirements statement*
    3. *Software Identification*
          a. *Appropriate means of 'marking' metrologically significant software*
          b. *Software Separation and marking consequences*

    c. *Relationship between software and software identifier*
    d. *Presentation of software identifier*
      i. *Example icons and menu text*
      ii. *Exceptions*
  4. *Protection against unauthorized software change*
    a. *How is software "sealed"?*
    b. *Remote software update considerations, e.g. authentication*
    c. *Audit trail (if employed) requirements for software updates*
  5. *Accuracy of data calculations*
    a. *When to stop evaluating calculations & data manipulation*
  6. *Software Evaluation Checklist*

Gathering some of the text we've proposed all in one place:

(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)

Additionally, 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.) , how to view it, and how it is tied to the software.**
- **An overview of the security aspects of the software(s), e.g. protection, user accounts, privileges, platforms, etc.**

G-S.9. Metrologically Significant Software Updates
  A software update that changes the metrologically significant software shall be considered a sealable event.

It was suggested that we explicitly state that if something doesn't affect the metrological operation of a software-based device, we don't care about it.

It was suggested that we include a description of what information would be logged in a category 3 audit trail that pertains to software updates. What about category 2? Darrell Flocken recommended that we stay away from requiring any particular type of sealing category. For example, "When using a category 3 audit trail, the following information should be…" This would be a description of the methods to comply with the existing sealing requirements, not creating new requirements.

Mexico has a very thorough description of what is required in their audit trails. We may want to review that at some point.

We should incorporate the description of software separation from Doug Bliss' presentation.

Jim Pettinato suggested that we review some of the Software Sector meeting agendas from previous years for descriptions of exceptions and examples. Darrell Flocken will check to see if there is anything useful in the meeting agendas from the previous incarnation of the Software Sector. The D-31 document may be a good source of examples and explanations for issues to consider when performing a remote update.

Regarding the accuracy of calculations and at what point do you stop requiring evaluation, Darrell Flocken said that there's not a lot of existing documentation. The only guidance he thought HB44 includes on accuracy is regarding rounding. That's not the same thing as to when you stop the evaluation. "First final" is NTEP's standard, but the states can be different, requiring more. "First final" is in the Administrative Policy. The agreement as to where the

boundary line is drawn may come about as a result of the discussion during type evaluation, but we can hopefully provide some guidance. This can be especially confusing when data is being transmitted and calculations are being performed remote to where a measurement was originally taken. HB44 deals particularly with "first final", but how that interacts with HB130 (method of sale) can introduce complications.

Measurement Canada considers similar issues, requiring W&M regulation to the equivalent of our "first final". Anything past that point isn't metrological.

Conclusion:
The Sector concluded that we should organize and summarize the data captured in this brainstorming session on what will likely go into this new software section of Pub. 14. Teri Gulke volunteered to write a first draft for the Software Sector members to review and amend. Once the Sector has approved a draft of representative example content, we could choose to include this as an amendment to the NTEP agenda items.

8.   Review/Discussion of new WELMEC 7.3/7.4 Drafts

Background:
WELMEC has been working on additional guidance for system architecture and design of software systems based on WELMEC 7.2 and has released two new draft guides titled 'WELMEC Guide 7.3 Reference Architectures' and WELMEC Guide 7.4 Exemplary Applications of WG 7.2' for review by the wider group. These address some of the questions that have come up in our own discussions, such as cloud-based metrology, remote storage and displays, etc. Time permitting, the Sector can review this draft document and we can forward any additional comments to the Convener for consideration in their upcoming Group 7 meeting in Berlin.

Discussion:
There was some concern expressed that the text of the new draft guides may be too specific. For example, in WELMEC 7.3 there is a description of "pairing" a sensor with the software and how to accomplish it. It would be better to be more generic, and refer to "authentication" and "integrity" to establish a secure connection, rather than a particular method.

WELMEC 7.4 are oddball examples that may cause issues. Its title is "Exemplary Applications". A better title may be "Anomalous Applications". The second example is puzzling. They may be trying to describe a way to indicate that a measurement may not be accurate, but it's not coming through clearly. There didn't appear to be any authentication when a connection is established.

Discussion:
Our concerns will be relayed to the WELMEC working group via the CECOD representative.

9. Next Meeting

Background:
The sector is on a yearly schedule for NTEP Software Sector Meetings. Now that we've adopted a joint meeting system, the next Sector joint meeting will likely coincide with one of the remaining Sector meetings. The Measuring Sector would be next in the sequence if we continue in the same manner.

Discussion:
We are due to meet with the Measuring Sector next year. Their meeting will be next September in Denver. Between now and then, the conference will meet twice, so the addition of a new software section within Pub. 14 may have been addressed by that time.

We discussed whether it was still beneficial to conduct joint meetings with the other sectors. Doug Bliss is retiring, so this is his last Software Sector meeting. Darrell Flocken asked whether we intend to replace him. Jim Pettinato asked about the standard of having a NIST/NTEP technical advisor. Darrell Flocken said that there is discussion of moving away from that standard and adopting Software Sector's example of having technical advisors from industry.

The next meeting should have an agenda item for appointing a new technical advisor. If we could do that prior to the next meeting, that would be even better. Perhaps the nominations could be conducted via email.

Conclusion:
We agreed to continue with joint meetings for at least one more year (2019). After that meeting, we may want to consider conducting joint meetings with the NTEP labs.

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		

Appendix B – NIST WMD Report on International Activity

Summary of OIML D31 Revision Progress
To be presented at NCWM Software Sector Meeting, Louisville, KY
K.M. Delak 17 Aug 2018

OIML has been undertaking a revision of D-31: General Requirements for Software Controlled Measuring Instruments. This falls under Technical Committee 5, Subcommittee 2. Approval of the revision was taken at the CIML meeting in October 2016, and initial work began in spring of 2017, with the 1WD being circulated at that time for input.

September 2017: Project group met in Berlin to consolidate comments and complete a first revision. This constituted 1CD, which was subsequently circulated for vote and comment. *Circulation of a CD was chosen to ensure a maximum response from OIML member countries.* Further, two subgroups were formed: (1) Software Verification and (2) Operating Systems.

Discussions on draft language for Operating Systems were conducted largely between only the US and Germany by video-conference over the course of two months. The consensus language generated from this activity was introduced into the document in the subsequent project meeting.

Draft language for Software Verification was agreed to primarily by correspondence. This also was introduced into the document draft in the subsequent project meeting.

May 2018: Project group met in Dordrecht to consolidate comments from the 1CD. The group made rapid progress in consolidating language. The conveners initiated a third subgroup, Terminology Harmonization, to clarify the definitions on "measurement," "measurement result" and "measurement data." Current suggestions have been circulated to the USNWG for comment. The conveners ask for finalization of input to this by 24 Aug 2018.

It is expected that 2CD will be published in September. This will also be circulated to the USNWG for final comment and vote.

WELMEC WG7 has attempted to further clarify interpretation of WELMEC 7.2 with new draft documents WELMEC Guide 7.3 "Reference Architectures" and WELMEC Guide 7.4 "Exemplary Applications", meeting coming up on 8/29/18 @ PTB offices in Berlin.

US National Working Group consists of:

Dr. Katya Delak
Jim Pettinato
Doug Bliss
Teri Gulke
Jan Konijnenburg
Joe Porthouse
Shakila Xavier