National Type Evaluation Program (NTEP)
Software Sector Meeting Summary

May 3rd, 2017 / Columbus, OH
(in conjunction with the Multiple Dimension Measuring Device work group)

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 **underlining** information to be added. Requirements that are proposed to be non-retroactive are printed in ***bold faced italics***.

Table A
Table of Contents

Title of Content **Page**

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)

STATUS REPORTS – RELATED NCWM AND INTERNATIONAL ACTIVITY

NCWM Activity

The Software Sector didn't have any agenda items for the 2017 NCWM Interim Meeting. The Grain Analyzer Sector had an item regarding removable devices which did pertain to software. There were some suggested wording changes, to ensure it only covered metrologically-significant software, not all software changes. That item is Developing.

International Activity

Two weeks ago, we received a markup of D-31, which is being revised. Our concern is to ensure that the requirements are workable in the field. Originally the meeting was scheduled for June, but they've moved it back to mid-September in Berlin. Dr. Thompson intends to ask them to add a specific section for field inspectors. R129 was reviewed by the MDMD Work Group.

SOFTWARE SECTOR PRESENTATION

Technical Advisor Doug Bliss gave a presentation from the Software Sector for the benefit of those MDMD Work Group members who may not have been familiar with the Software Sector agenda items and the background behind them. The presentation can be found on the NCWM.net web site for those interested in reviewing the background.

CARRY-OVER ITEMS

1. Software Identification / Markings

Source:
NTEP Software Sector

Background:
See the 2016 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***~~ ***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, 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 the 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.

Discussion:
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.

Conclusion:
This agenda item remains tabled until 2020.

2. Identification of Certified Software

Source:
NTEP Software Sector

Background:
See the 2016 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 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 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 3.0a, etc.). Could also consist of / contain checksum, etc. (crc32, for example)

This item was eventually withdrawn. Darrell Flocken shared his recollection of why the S&T Committee objected to this wording back in 2010. Basically, it went too deep for Handbook 44 and would be better placed in Pub. 14.

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?

The possibility of creating a separate Publication 14 section specific to software was debated. There are pros and cons in terms of the chances of adoption with that approach. It might be beneficial to manufacturers, due to keeping the requirements in one place. This becomes a philosophical question – is the content of Handbook 44 intended to be a guide to manufacturers, or is it intended as direction to field inspectors? This discussion was tabled for present.

Historically, CC's have been written in terms of "version X and higher". It is not our intention to change that "policy", but it isn't documented anywhere. Perhaps that should be addressed by the Software Sector. Jim Truex reviewed the administrative policy text, which includes the requirement to report changes to NTEP, based on whether they're metrologically significant.

California indicated that their NTEP lab only puts the software version on the certificate if it's not-built-for-purpose, but it seems that the other labs do so for all software-based devices.

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.*

The Software Sector decided that we'd leave the previously withdrawn recommendation as-is, in the hopes that the other changes to G-S.1 will be adopted and then this can be revisited. Several Measuring Sector members and all the labs indicated their support for the language as written.

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.

It was also noted that the checklist being developed for the labs currently includes (1.4.3) the requirement that the software version or revision be linked to the software itself.

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.

Discussion:
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.

Conclusion:

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.

3. Software Protection / Security

Source:
NTEP Software Sector

Background:
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 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 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

Jim Pettinato reiterated the Software Sector’s request that the labs continue (or begin) to ask manufacturers whether they’re willing to participate in the use of this checklist (on a voluntary basis), and to send their feedback to Darrell Flocken. Teri Gulke will clean up the checklist and put it in a separate document that can be posted on the NCWM website under the Software Sector’s documents.

The contents of the checklist should tie back to requirements in Pub. 14. We originally crafted our checklist from the contents of D-31, so we went back to it to see if we could use it as a starting point for writing our own requirements for Pub. 14.

Though they need to be reworded, of course, the most useful portion of D-31 for our current purposes are probably sections 5.1.1., 5.1.3.2.a., 5.1.3.2.d, and 5.2.6.1. which state, respectively:

> *5.1.1 Software identification*
> *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.*
>
> *5.1.3.2.a The legally relevant software shall be secured against unauthorized modification, loading, or changes by swapping the memory device. In addition to mechanical sealing, technical means may be necessary to secure measuring instruments having an operating system or an option to load software.*
> *5.1.3.2.d Software protection comprises appropriate sealing by mechanical, electronic and/or cryptographic means, making an unauthorized intervention impossible or evident.*
>
> *5.2.6.1 Only versions of legally relevant software that conform to the approved type are allowed for use (see 5.2.5). Applicability of the following requirements depends on the kind of instrument and is to be worked out in the relevant OIML Recommendation. It may differ also depending on the kind of instrument under consideration.*

The question was again asked, do these new requirements need to go into a new appendix specific to software in Pub. 14? Do we need to document new requirements at all if the checklist is put into Pub. 14? It could be considered that the checklist itself constitutes the new requirements. Darrell Flocken and Jim Truex supported that interpretation.

At the 2016 meeting, we learned that the Grain Analyzer Sector’s labs have not had the opportunity to try using the checklist because they didn’t meet in 2015. Tom Buck from Ohio reported that they’ve been giving the checklist to manufacturers but haven’t been getting them back. Darrell Flocken has two examples, one for built-for-purpose and one for a not-built-for-purpose device. Jason Jordan from GIPSA said that they’d try it out. Doug Bliss and Jim Pettinato have volunteered to answer any questions that might arise as the labs attempt to use the checklist.

The Sector asked that the revised checklist continue to be used by the labs.

Discussion:
As we meet with each Sector jointly, we can get an updated report on the trial and decide if we’re ready to recommend it for Pub. 14. We can also look at the language from D-31 in more detail in an effort to craft guidance in line with NCWM/NTEP philosophy.

This checklist was discussed during the 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.

Conclusion:
Darrell Flocken will continue to be a point of contact if businesses or the labs have questions, but no one has yet contacted him in that regard.

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

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 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.) , 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.

As we began discussing the training of field inspectors, Darrell Flocken asked that we also provide further training for lab inspectors. There's an annual lab meeting typically around April, in 2017 it will be in Annapolis, MD.

Discussion:
The Software Sector's recommendation is to add the requirements to the Application section. The Software Sector agreed to provide support for any desired training of lab personnel at the April meeting.

Jim Pettinato suggested that this agenda section has become largely redundant to the previous agenda section (the checklist). As time has passed, we've begun to address software the same regardless of its platform. Built-for-Purpose and Not-Built-for-Purpose differentiation seems less relevant. Doug Bliss pointed out that we still need to address how to communicate these issues to manufacturers. For now, we will continue with two different agenda items since the contents of the checklist are a separate issue from how we want to address / communicate the requirements.

As previously stated in earlier meetings, the labs can ask for any documentation they like, but it would be good to give manufacturers advance notice. Part of the Technical Policy (NTEP or individual codes) could include a requirement to fill out the checklist. Jim Truex suggested our best path forward may be to take the checklist (once we're sure it's mature) to the NTEP committee and ask them to add it to their policy.

Though we haven't thoroughly considered adding this to Hdbk. 44, Darrell Flocken pointed out that there is a portion of the handbook that pertains only to type evaluation.

Jim Truex's suggestion is probably the option with the best chance of success, but it will require some convincing. Doug Bliss suggested that we may need to put together a presentation like what we did for the adoption of our G-S.1 wording.

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.

Darrell Flocken suggested we create a new agenda item for addressing the NTEP Committee. They meet 4 times a year. In fact, they meet 2 weeks from now (after the NUMA meeting) in Saratoga Springs, NY. Thereafter they meet in July.

Jim Truex said that he doesn't think that the software security concept has progressed far enough for it to be adopted in any formal manner.

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.

Conclusion:
We will provide an outline for the proposed Pub 14 section prior to the meeting in two weeks, to run past the NTEP Committee to get their feedback. We want to make sure this is a viable approach, in their opinion.

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.
   1.3 Software version/revision.(added at the 2017 Software Sector meeting)
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.

For the Grain Analyzer Sector, Diane Lee thought it would take some time to put together some training material, as they do not currently have anything in place for software requirements.

Examples from completed checklists would be very helpful when putting together field inspector training. A lot of training videos have been recently generated. Doug Musick suggested that we recommend adding this to the agenda for the PDC Committee. Certification exams could be updated more easily, on a state-by-state level. It might be better to make software a separate exam.

Diane Lee suggested we look at developing a basic course for software, incorporating specific guidelines for specific device types.

Amanda Dubin was concerned about having the field inspectors know all the different existing software, which is a monumental task. Instead, the training should focus on how to find the pertinent CoC and look up information from it on the website. Ideally, down the road there could be some sort of database or software tool disseminated to field inspectors to assist in the look up of certificate numbers and the approved version number(s) for the software for a particular device, and even instructions on how to view/print the audit trail.

Jim Truex holds a meeting once a year for the lab evaluators. Darrell Flocken suggested that we also focus on training them on software. Diane Lee mentioned that NIST has been having manufacturers coming in to provide training on, for example, how to access the audit trail.

Discussion:

A video explaining the different sealing requirements was developed several years ago. It was intended for inspectors. NIST has given this video out at several training sessions.

The very first thing a field inspector needs to do is determine whether the software/system is metrological. Jim Truex said that they need guidance in figuring this out.

Inspectors are trained to look for a CC and look it up. A lot of the time this occurs during initial implementation of new equipment.

Not all devices are evaluated by NTEP, so they won’t have a CC. That might be because NTEP hasn’t established an evaluation for that type of device yet.

There are only 4 states that don’t participate in NTEP. Two of those do participate informally (not legally required).

Conclusion:

Jim Pettinato suggested that at least for the short term, we work with California on an EPO.

6. 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.).

This was discussed during the Measuring Sector's meeting on 9/15. The wording suggested was not agreed upon. Adam Oldham would like to have the Software Sector's suggestions, so he can put together a proposal for next year.

The US has rather unique requirements for printing the Category 3 audit trail, which are quite unwieldy – both in terms of the actual printing process (and results), as well as the needed approvals (the example provided by Adam Oldham required an approval for each and every POS system that might be connected to their system). The most similar is from Mexico, but they require an electronic copy.

Darrell Flocken reported that there has been a little movement forward – alternative methods are now allowable, to some degree, but it's dependent on what the states are going to allow, and it still requires the ability to print it. The change will be in LMD Code S.2.2., not in Handbook 44 G-S.2.2.

We discussed the difficulty of requiring that the electronic data be printable on-site, given that some sites don't have any printers, and other sites may have printers attached to computers that are restricted in what can be used to attach to them.

In Mexico, Gilbarco relies upon laptops being present, supplied by the auditing company.

LMD Pub. 14 has a section in Appendix B Requirements for Metrological Audit Trails on the event logger, and that information doesn't seem to be in Handbook 44. In fact, it may even contradict what's in the LMD Pub. 14. In practice, what's in Pub. 14 tends to be more influential with evaluators.

Adam Oldham volunteered to work on the wording for a proposal to present at the 2016 Software Sector meeting for review, but was not in attendance at the meeting, so the item was tabled.

The chair has anecdotal evidence that other parties have also expressed interest in specifying alternate methods for distributing audit trail information aside from the current 'printing' requirements. This discussion should be continued at the 2017 meeting.

Discussion:
In 2016 the Conference worked on some changes, but some states don't care for them. Previously only printouts were allowed, but now an "alternate method" is potentially allowed if all parties agree. Jim Truex thinks this issue has moved as far forward as it probably will for the time being.

Darrell Flocken suggested that if/when we get a separate software section added to Pub. 14, we ensure that our wording match the other similar sections in Pub. 14.

Conclusion:
This agenda item was closed by the Sector.

NEW ITEMS

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

Source: Software Sector

Background:

See the 2016 Software Sector Meeting Summary for additional 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.

Discussion:
Both Uber and Lyft provided presentations at the 2017 Interim Meeting to address some of the concerns that have been raised.

There is a US working group devoted to this subject now. There are three commercial parties – the driver, the rider, and the company itself. The driver is the one providing the phone that is relied upon for measuring the time and distance. There is an option for an up-front fare, which doesn't fall under W&M jurisdiction. The driver's compensation is based upon time and distance, so that is pertinent for W&M, as is the rider's cost if a destination isn't provided. Google Maps isn't being evaluated, just the Uber/Lyft software. The focus for testing is inputs and outputs. GPS data is traceable by NIST, which accounts for a different approach between the US and Europe. Google Maps is not traceable.

There are two proposals before the S&T Committee pertaining to these systems. One is for transportation management systems (i.e. Uber and Lyft), and if approved it would be on a test basis; systems wouldn't be red-tagged based upon it yet. The other one is for amending the existing taxi meter code to address these systems. Both will be voted upon in July. There's a third item, but it isn't as extensive of the other two items.

Conclusion:
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.

8. 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 coincide with one of the remaining Sector meetings.

Everyone thinks that the joint sessions have been productive; however, Darrell Flocken pointed out that if we do get a separate software section in Pub. 14 we may want to have a separate working session to work out wording/terminology before meeting with the other sectors for feedback.

If we stick with the same rotation, the Weighing Sector would be the next meeting, though that would a year and a half from now. The MDMD Work Group will meet again in May 2018, and they're asking that we meet with them again since they anticipate a lot of changes in their functionality in that time. There is less pressure to meet right before regional meetings because we aren't currently proposing changes to Hdbk. 44 though our proposal to the NTEP Committee might have to go through the regionals.

Conclusion:
After some consideration, we decided we would meet with the Weighing Sector in August 2018.