National Type Evaluation Technical Committee (NTETC)
Software Sector Meeting

March 2 & 3, 2010
Red Lion Inn – Sacramento, CA

National Type Evaluation Technical Committee (NTETC)
Software Sector Meeting

March 2 & 3, 2010
Sacramento, CA

Agenda Schedule

Tuesday, March 2, 2010

8:00 AM Meeting Call to Order *(Co-Chairs)*

i. Introductions and welcome of new Sector members

ii. Reiteration of NTETC Software Sector Mission *(J. Truex)*

8:30 AM Status Reports

iii. **Report – 2009 NCWM Interim Meeting** *(J. Pettinato)*

iv. **Report – International Activity of Interest to Sector** *(A. Thompson)*

9:00 AM Work session - Carryover Items

1. **NCWM/NTEP Policies – Issuing CCs for Software**
2. **Definitions for Software Based Devices**

10:00 AM Break *(15 min.)*

10:15 AM Carryover Items (continued)

3. **Software Identification / Markings**
4. **Identification of Certified Software**

12:00 PM Lunch Break *(1 hour)*

1:00 PM – Carryover Items (cont.)

5. **Software Protection / Security**

3:00 PM – Break *(15 min.)*

3:15 PM – Carryover Items (cont.)

6. **Software Maintenance and Reconfiguration**

5:00 PM – Adjourn for the day

Wednesday March 3, 2010

8:00 AM Continue Work Session - Carryover Items

7. **Verification in the Field, By the Inspector**

10:00 AM Break *(15 min.)*

10:15 AM Carryover Items (continued)

8. **NTEP Application**

12:00 PM Lunch Break *(1 hour)*

1:00 PM Work Session – New Items

9. **Training of Field Inspectors**
10. **Next Meeting**

3:00 PM – Break *(15 min.)*

3:15 PM – Work Session

This time is reserved for revisiting items requiring additional attention and any unscheduled items brought to the Sector for consideration.

5:00 PM – Adjourn

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

i. Welcome/Introductions

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

Kevin Detert – Avery Weigh-Tronix	Andrew Gall – Foss NA

ii. NTETC Software Sector Mission

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

Source: NCWM Board of Directors

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

Mission of the Software Sector:

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

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

iii. Report of Activity at 2010 NCWM Interim Meeting

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

iv. Report of Activity from International W&M Agencies

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

CARRYOVER ITEMS

1. NCWM/NTEP Policies – Issuing CCs for Software

Source: NCWM Reports

Background: For additional background on this item, see the 2009 Software Sector Meeting summary.

Recommendation from the Sector to the NTEP Committee: (The Sector recommended the following language to be submitted to the NTEP Committee as a policy change. The Sector requests the NTEP Committee place this issue on their agenda.)

Software Requiring a Separate CC: Software, which is implemented as an add-on to other NTEP Certified main elements to create a weighing or measuring system and its metrological functions, are significant in determining the first indication of the final quantity. Such software is considered a main element of the system requiring traceability to an NTEP CC.

NOTE: OEM software ***may*** be added to an existing CC or have a stand-alone CC with applicable applications (e.g., a manufacturer adding a software upgrade to their ECR or point-of-sale system, vehicle scale weigh-in/weigh-out software added as a feature to an indicating element, automatic bulk weighing, liquid-measuring device loading racks, etc.) and minimum system requirements for "type P" devices (see proposed software definition below). It may be possible for a manufacturer to submit a single application for both hardware and software contained in the same device. A single CC would be issued.

In this instance, OEM refers to a 3rd party. The request to add software could be made by the original CC holder on behalf of the 3rd party. Alternatively, a new CC could be created that refers to the original CC and simply lists the new portions that were examined. The NTEP committee included this item in their agenda (NTEP Committee 2009 Interim Agenda Item 8); there was no discussion during the open hearing, and this became a Voting item for the 2009 Annual Meeting.

Activity since last meeting: At the 2009 NCWM Annual Meeting, this proposal was passed unanimously by the Conference.

Status: ***PASSED*** *(Question for Jim T. – was there/is there to be any actual change in any document or is this strictly a procedural change? How do the labs know they can/should handle software items differently now? Is any additional work required on this item or can it be considered complete?)*

2. Definitions for Software Based Devices

Source: 2009 Carryover Item 310-2. This item originated from the NTETC Software Sector and first appeared on the Committee's 2007 agenda as Developing Item Part 1, Item 2.

From NCWM Publication 15, 2010

310-2 Appendix D – Definition of Electronic Devices, Software-Based and Built-For-Purpose Device

Item Under Consideration:

Delete the current definition of built-for-purpose device as follows:

~~*Built-for-purpose device. Any main device or element, which was manufactured with the intent that it be used as, or part of, a weighing or measuring device or system. [1.10] (Added 2003)*~~

and, add a new definition and a cross-reference to Appendix D in HB 44 for "Electronic devices, software-based" as follows to replace the current definition of "built-for-purpose device":

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

(a) Embedded software devices (Type P), aka built-for-purpose. – A device or element with software used in a fixed hardware and software environment that cannot be modified or uploaded via any interface without breaking a security seal or other approved means for providing security and will be called a "P," or

(b) Programmable or loadable metrological software devices (Type U), aka not-built-for-purpose. – A personal computer or other device and/or element with PC components with programmable or loadable metrological software and will be called "U." A "U" is assumed if the conditions for embedded software devices are not met.

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

***Background/Discussion*:** For additional background information on this item, please reference the 2009 Software Sector Meeting summary and the 2010 NCWM Interim Meeting Agenda (Pub 15)

Activity since last meeting: At its 2009 Interim Meeting, the CWMA received comments that the proposal is sufficiently developed and recommends moving this item forward as a Voting item on the Committee's agenda.
At its 2009 Annual Technical Conference, the WWMA received comments from Mr. Straub, speaking on behalf of SMA, indicating the SMA continues to oppose this item, noting that requirements should apply equally to the two different device types described. The WWMA received no other input on this item and recommends this item should remain Informational until the Software Sector has had an opportunity to review comments from the 2009 NCWM Annual meeting and any comments made at subsequent regional weights and measures association meetings.
At its 2009 Annual Meeting, the SWMA recommended keeping the status of this proposal to delete the current definition of built-for-purpose device and add a new definition and a cross-reference to Appendix D in HB 44 for "Electronic devices, software-based" to replace the current definition of "built-for-purpose device" as an Informational item. The SWMA agreed that the Software Sector should continue to work on the proposal until it arrives at some final language.
During its 2009 Interim Meeting, NEWMA stated that it supports the Committee's decision to keep this item Informational to allow updated comments from the regional weights and measures associations and other interested parties based on information in the summary of the March 2009 meeting of the Software Sector.

Status: ***Item remains as an informational item on 2010 Annual Meeting Agenda; the S&T Committee indicated that they look forward to additional work being done on this item by the Sector.***

3. G-S.1. Identification (Software)

Source: NTETC Software Sector

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

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

The Sector developed marking information requirements and submitted a proposal to the S&T Committee for considered inclusion in NIST Handbook 44. There was much additional comment and various proposed versions of the table from NIST WMD, et al. (The complete background on this item can be seen in the '10 Interim Meeting Agenda NCWM Pub 15, 2010.)

At the 2009 Software Sector Meeting, it was agreed that the proposed table had not accomplished the intended purpose of clarifying the requirements, indeed it seemed to have generated more confusion. Hence, this item was revisited from the beginning, and it was suggested that a simpler approach be taken, namely to modify the text of G-S.1. to match our intent. The proposal as stands from our Sector is as follows:

G-S.1. Identification. – All equipment, except weights and separate parts necessary to the measurement process but not having any metrological effect **and manufactured after January 1, 201X**, 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 that is not part of a Type P (built-for-purpose) device.****;*
[Nonretroactive as of January 1, 1968]
(Amended 2003 **and 201X**)

(1) The serial number shall be prefaced by words, an abbreviation, or a symbol, that clearly identifies the number as the required serial number.
[Nonretroactive as of January 1, 1986]

(2) Abbreviations for the word "Serial" shall, as a minimum, begin with the letter "S," and abbreviations for the word "Number" shall, as a minimum, begin with the letter "N" (e.g., S/N, SN, Ser. No., and S. No.).
[Nonretroactive as of January 1, 2001]

(d) the current software version or revision identifier for ~~***not built-for-purpose***~~ ***software-based electronic*** *devices;*
[Nonretroactive as of January 1, 2004]
(Added 2003) **(Amended 201X)**

(1) The version or revision identifier shall be prefaced by words, an abbreviation, or a symbol, that clearly identifies the number as the required version or revision.
[Nonretroactive as of January 1, 2007]
(Added 2006)

(2) Abbreviations for the word "Version" shall, as a minimum, begin with the letter "V" and may be followed by the word "Number." Abbreviations for the word "Revision" shall, as a minimum, begin with the letter "R" and may be followed by the word "Number." The abbreviation for the word "Number" shall, as a minimum, begin with the letter "N" (e.g., No or No.).
[Nonretroactive as of January 1, 2007]
(Added 2006)

(e) an NTEP CC number or a corresponding CC Addendum Number for devices that have a CC. The CC Number or a corresponding CC Addendum Number shall be prefaced by the terms "NTEP CC," "CC," or "Approval." These terms may be followed by the word "Number" or an abbreviation of that word. The abbreviation for the word "Number" shall, as a minimum, begin with the letter "N" (e.g., No or No.)
[Nonretroactive as of January 1, 2003]

The required information shall be so located that it is readily observable without the necessity of the disassembly of a part requiring the use of any means separate from the device.
(Amended 1985, 1991, 1999, 2000, 2001, 2003**, and, 2006** **and 201X**)

G-S.1.1. Location Method of Marking Information for ~~Not-Built-For-Purpose~~ all Software-Based Devices. – *For* ***~~not-built-for-purpose,~~ software-based*** *devices* ***manufactured after January 1, 201X,*** *either:*

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

(b) The CC Number shall be:

(1) permanently marked on the device;

(2) continuously displayed; or

(3) accessible through an easily recognized menu and, if necessary, a submenu. Examples of menu and submenu identification include, but are not limited to, "Help," "System Identification," "G-S.1. Identification," or "Weights and Measures Identification."

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

[Nonretroactive as of January 1, 2004]
(Added 2003) (Amended 2006 **and 201X**)

The Sector noted that though currently it is allowable to display the CC number via a menu, there has been some challenges locating this information in the field due to the vagueness of the term "easily recognized." Hence, since it is left to the interpretation of the NTEP laboratory to ascertain whether a device's method for displaying the CC number meets the requirements, this vagueness has not been addressed in this new recommendation.
Mr. John Roach (California NTEP Lab) indicated that if the proposed table, or some version thereof, is not eventually included as part of G-S.1. that it may be useful to incorporate a suitable table into Publication 14. The Software Sector concluded that it does not wish to debate the merits of general marking requirements beyond that related to software identification and wishes only to address concerns related specifically to software. The Sector feels its proposed changes above better reflect the Sector's position. The Sector suggests that the following simplified version of the table may better suit the purpose if WMD, and the Committee believes a table outlining general marking requirements would clarify the intent of paragraph G-S.1.

Table G-S.1.~~a~~ Identification
for Devices Manufactured on or after January 1, 201X

Required Marking	**Full Mechanical Devices and Separable Mechanical Elements**	**Electronic Devices, Software Based**
Manufacturer or CC holder ID	**Hard Marked**	**Hard Marked, Continuously Displayed, or Via Menu (display) or by command or operator action**
Model identification	**Hard Marked**	**Hard Marked, Continuously Displayed, or Via Menu (display) or by command (operator action)**
Serial number	**Hard Marked**	**Hard Marked, Continuously Displayed**[1]
Metrologically Significant Software version	**Not Applicable**	**Continuously Displayed, Via Menu (display) or by command (operator action)**[2]
CC number	**Hard Marked**	**Hard Marked or Continuously Displayed, or Via Menu (display) or by command (operator action)**[3]

[1] **Type 'U' devices need not have a non-repetitive serial number.**

[2] **If the manufacturer declares that the primary sensing element "software" is integral, has no end user interface and no print capability, the version/revision shall be hard marked on the device. Example: Primary sensing element may be Positive Displacement (P.D.) meter with integral correction, digital load cell (only for reference, not limiting).**

[3] **If the CC number is to be displayed via menu and/or submenu, the means of access must be easily recognizable. In addition, instructions on how to obtain the remaining required information not hard-marked or continuously displayed shall be included on the NTEP CC.**

(Added 201X)

Conclusion**:** Revised submission sent to NCWM S&T Committee.

Activity since submission: Steve Patoray (CoC) commented via letter to S&T that he agrees with the submission with one modification: the NOTE in G-S.1.1 should be changed to read as follows:

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

In the "***Note***" for paragraph G-S.1.1., there remains one item that is inconsistent with all other requirements for marking. It is noted that it indicates **only** the information in G-S.1. (a), (b), and (d), which intentionally leaves out information in G-S.1. (c) Serial Number. It is the position of CoC that there should be NO limitation, which is any different from other markings, on the marking of the serial number of a device in the General Code. As written, it would require only the serial number to be permanently marked or continuously displayed. Since this is the General Code Section of HB 44, CoC sees no reason to create this limitation. CoC recommends that this can be handled in the device specific code if there are any issues that might arise with allowing the serial number to be display via menu. CoC stated that is could fully support this item incorporating the above change. At its 2009 Annual Meeting, the CWMA agreed with comments from weights and measures officials that there is a need to easily identify the software for the proposed software-based devices, especially during field inspections for "Type U" devices. They believe that a uniform or standard method for easily accessing identification information is needed to aid field inspections. The SMA stated there is no distinction between the proposed Type P and Type U devices and marking requirements should be the same for both devices. It was reported that the Software Sector had met a few weeks before the CWMA Annual meeting and that the Sector

recommendations would be submitted to the committee and its report posted on the NCWM website prior to the NCWM Annual Meeting. Consequently, the CWMA recommends this item remain Informational.
At the 2009 NEWMA Annual Meeting, the members received similar comments from SMA and the Software Sector and took no position on this item pending its member review of the Software Sector's report.

At the 2009 NCWM Annual Meeting, the S&T Committee reviewed the recommendations from:

- the 2009 meeting of the Software Sector,

- a report of the 2009 spring meeting of the SMA opposing the marking requirement differences for "Type P" and "Type U" devices, and

- comments from Mr. Patoray, CoC, supporting the Software Sector's position with his suggested changes.

NIST WMD commented that some terminology in both the Software Sector's proposed "Table G-S.1. Identification" may need to be further defined. For example, what is meant by the term "hard-marked?" WMD believes that "hard marked" is the same as "permanently marked," which is already used in other sections of HB 44. If Committee believes a table outlining general marking requirements would clarify the intent of G-S.1., WMD recommends that the words "hard marked" be replaced by "permanently marked".

Consequently, the S&T Committee agreed that this item remain Informational and that the regional weights and measures associations review the above information and provides the Committee with comments and recommendations.
At its 2009 Interim Meeting, the CWMA had lengthy discussions about providing the required identification information in a single uniform method. Some of the topics addressed were:

- A single operation or button is needed to view all software version information.

- Use a single function key to access or continuously display software version information.

- Electronic data for both Type U and Type P devices could be Hard Marked, Continuously Displayed or accessed by Command (operator action).

- The data is useless if it is not easy to access in the field.

- Concern about the cost of requiring a single designated button to access software version information.

The CWMA recommends this item remain Informational with the following changes to the Committee's recommendations in its 2009 Interim Report:

1. In proposed paragraph G-S.1.1.(a), add "or accessed by a command (operator action)" **and** delete subparagraph G-S.1.1.(b) (3) to read as follows:

G-S.1.1. Location of Marking Information for Type U (Not-Built-For-Purpose), Software-Based Devices. – *For* ***Type U not-built-for-purpose, software-based*** *devices* ***manufactured prior to January 1, 201X,*** *either**:***

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

(b) The CC Number shall be:

(1) permanently marked on the device; ***or***

*(2) continuously displayed**.***

2. Delete note 8 in "Table G-S.1. Notes on Identification."

3. Revise proposed table to delete references to "via menu display", "Print Option (8)," adding "by command (operator action)," and deleting the language at the bottom of the table as shown in following revised table.

During the open hearings at the 2009 WWMA Annual Technical Conference, Mr. Straub, speaking on behalf of SMA, indicated SMA continues to oppose this item, referring to comments made in conjunction with Item 310-2. He also noted that even if the designations of “Type U” and “Type P” were adopted, SMA would continue to oppose the proposed changes to G-S.1., noting that requirements should apply equally to the two different device types described. The WWMA also heard from Mr. Johnson, Gilbarco, who agreed with SMA’s assessment. He also indicated that it would be desirable to have the option of using a menu to provide information, citing increasingly limited space in which to provide marking information, and noted it would be virtually impossible for their company to provide a full time display.
Based on the comments received and its position on Item 310-2 relative to corresponding definitions for the device types referenced in Item 310-3, the WWMA believes this item should remain Informational until the Software Sector has had an opportunity to review comments from the 2009 NCWM Annual meeting and any comments made at subsequent regional weights and measures association meetings.

At its 2009 Annual Meeting, the SWMA agreed that the Software Sector should continue to work on the proposal until it arrives at some final language for amending paragraphs G-S.1. Identification and G-S.1.1. Location of Marking Information for Not-Built-For Purpose, Software-Based Devices. The Software Sector should work with manufacturers in its development of the requirement and any table or other tools should provide further clarity on the intent of the marking requirements.

During its 2009 Interim Meeting, NEWMA stated that is supports the Committee’s decision to keep this item Informational to allow updated comments from the regional weights and measures associations and other interested parties based on information in the summary of the March 2009 meeting of the Software Sector. Additional background information on this item can be reviewed in the Committee’s 2008 Final Report.

***Status*: This item was assigned Informational status for the 2010 Annual Meeting**

4. Identification of Certified Software

Source: NTETC Software Sector

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

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

From OIML D-31:

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

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

- CRC (cyclical redundancy check)

- Checksum
- Inextricably Linked version no.
- Encryption
- Digital Signature

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

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

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

> Separation of software parts - All software modules (programmes, subroutines, objects etc.) that perform metrologically significant functions or that contain metrologically significant data domains form the metrologically significant software part of a measuring instrument (device or sub-assembly). The conformity requirement applies to all parts and parts shall be marked according to Section G-S-X.X.
>
> If the separation of the software is not possible or needed, then the software is metrologically significant as a whole.

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

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

Identification of Certified Software:

Software-based electronic devices shall be designed such that the metrologically significant software is clearly identified. The identification of the software shall be inextricably linked to the software itself.

- Unique identifier must be displayable/printable on command or during operation, etc. (marking req't in addition)
- 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)

***Status*:** The item needs additional discussion and development by the sector.

5. Software Protection / Security

Source: NTETC Software Sector

***Background from Previous Meetings*:** The sector agreed that Handbook 44 already has audit trail and physical seal, but the question on the table is does the Handbook need to be enhanced to sufficiently discourage the facilitation of fraud, intentional or accidental, where software is concerned?

WELMEC and OIML again have addressed this issue specifically when dealing with software. From WELMEC:

> **Protection against accidental or unintentional changes**
> Metrologically significant software and measurement data shall be protected against accidental or unintentional changes.
> **Specifying Notes:**
> Possible reasons for accidental changes and faults are: unpredictable physical influences, effects caused by user functions and residual defects of the software even though state of the art of development techniques have been applied.
> This requirement includes:
> a) Physical influences: Stored measurement data shall be protected against corruption or deletion when a fault occurs or, alternatively, the fault shall be detectable.
> b) User functions: Confirmation shall be demanded before deleting or changing data.
> c) Software defects: Appropriate measures shall be taken to protect data from unintentional changes that could occur through incorrect program design or programming errors, e.g. plausibility checks.
> **Required Documentation:**
> The documentation should show the measures that have been taken to protect the software and data against unintentional changes.
> **Example of an Acceptable Solution:**
> □ The accidental modification of software and measurement data may be checked by calculating a checksum over the relevant parts, comparing it with the nominal value and stopping if anything has been modified.
> □ Measurement data are not deleted without prior authorization, e.g. a dialogue statement or window asking for confirmation of deletion.
> □ For fault detection see also Extension I.

Recommendation: The Sector derived a suitable checklist for Pub 14 from the OIML checklist, and asked the current NTEP labs to begin using this checklist on a trial basis for new type approval applications.

Devices with embedded software TYPE P (aka built-for-purpose)			
	Declaration of the manufacturer that the software is used in a fixed hardware and software environment, and		**Yes** □ **No** □ **N/A** □
	cannot be modified or uploaded by any means after securing/verification		**Yes** □ **No** □ **N/A** □
	Note: *It is acceptable to break the "seal" and load new software, audit trail is also a sufficient seal.*		
	The software documentation contains:		
		description of the (all) metrologically significant functions OIML states that there shall be no undocumented functions	**Yes** □ **No** □ **N/A** □
		description of the securing means (evidence of an intervention)	**Yes** □ **No** □ **N/A** □
		software identification	**Yes** □ **No** □ **N/A** □
		description how to check the actual software identification	**Yes** □ **No** □ **N/A** □

	The software identification is:		
		clearly assigned to the metrologically significant software and functions	**Yes** □ **No** □ **N/A** □
		provided by the device as documented	**Yes** □ **No** □ **N/A** □
Personal computers, instruments with PC components, and other instruments, devices, modules, and elements with programmable or loadable metrologically significant software TYPE U (aka not built-for-purpose)			
	The *metrologically significant* software is:		
		documented with all relevant (see below for list of documents) information	**Yes** □ **No** □ **N/A** □
		protected against accidental or intentional changes	**Yes** □ **No** □ **N/A** □
	Evidence of intervention (such as, changes, uploads, circumvention) is available until the next verification / inspection (e.g. physical seal, Checksum, CRC, audit trail, etc. means of security)		**Yes** □ **No** □ **N/A** □
Software with closed shell (no access to the operating system and/or programs possible for the user)			
	Check whether there is a complete set of commands (e.g. function keys or commands via external interfaces) supplied and accompanied by short descriptions		**Yes** □ **No** □ **N/A** □
	Check whether the manufacturer has submitted a written declaration of the completeness of the set of commands		**Yes** □ **No** □ **N/A** □
Operating system and / or program(s) accessible for the user:			
	Check whether a checksum or equivalent signature is generated over the machine code of the metrologically significant software (program module(s) subject to ~~legal contro~~l W&M jurisdiction and type-specific parameters)		**Yes** □ **No** □ **N/A** □
	Check whether the metrologically significant software will detect and act upon any unauthorized alteration of the metrologically significant software using simple software tools e.g. text editor.		**Yes** □ **No** □ **N/A** □
Software interface(s)			
	Verify the manufacturer has documented:		
		the program modules of the metrologically significant software are defined and separated	**Yes** □ **No** □ **N/A** □
		the protective software interface itself is part of the metrologically significant software	**Yes** □ **No** □ **N/A** □
		the *functions* of the metrologically significant software that can be accessed via the protective software interface	**Yes** □ **No** □ **N/A** □
		the *parameters* that may be exchanged via the protective software interface are defined	**Yes** □ **No** □ **N/A** □
		the description of the functions and parameters are conclusive and complete	**Yes** □ **No** □ **N/A** □
		there are software interface instructions for the third party (external) application programmer.	**Yes** □ **No** □ **N/A** □

Status: The Sector hopes to obtain feedback at this meeting from the NTEP labs regarding this checklist. Work is ongoing on this item.

6. Software Maintenance and Reconfiguration

Source: NTETC Software Sector

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

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

a. Verify that the update process is documented (OK)

b. For traced updates, Installed Software is authenticated and checked for integrity

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

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

Examples are not limiting or exclusive.

c. Verify that the sealing requirements are met

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

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

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

Physical Seal, software log
Category III method of sealing can contain both means of security

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

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

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

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

Verified Update

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

Traced Update

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

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

For a Traced Update, an event logger is required. The logger shall be capable of storing a minimum of the 10 most recent updates.An entry shall be generated for each software update.
Use of a Category 3 audit trail is acceptable for the software update logger. In this case the existing requirement of 1,000 entries supersedes the 10 entry requirement.A software update log entry shall include the following:

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

A Category III device may include the software update events in the Category III audit log in lieu of a separate software update log; the existing requirement for 1,000 entries supersedes the requirement for 10 entries.
The traceability means and records are part of the metrologically significant software and should be protected as such. If software separation is employed, the software used for displaying the audit trail belongs to the fixed metrologically significant software. *(Note: This needs to be discussed further due to some manufacturer's concerns about where the software that displays the audit trail information is located and who has access if this feature is provided. Manufacturers did indicate that there are methods available to encrypt the audit trail information; however, it cannot be protected from being deleted.)* (include flowchart from OIML D-31)

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

G-S.9. Metrologically Significant Software Updates
The updating of metrologically significant software shall be considered a sealable event. Metrologically significant software that does not conform to the approved type is not allowed for use.

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

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

(Note to S & T – this item assumes additional requirements in individual codes will be eventually added to address this requirement; e.g. L.M.D. code has philosophy of sealing section that could be enhanced to include processes described.)

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

7. Verification in the Field, By the W&M Inspector

Source: NTETC Software Sector

Background Question: What tools does the field inspector need?

Possible Answers:

- Have NTEP CC No. continuously displayed. (needs some type of protection) during the normal weighing or measuring operation
- Clear and simple instructions on NTEP CC to get to the other Inspection Information
- The CRC, checksum, version no. etc, needs to be easily accessible from operator console.
- Inspector needs to know how to access audit trail
- System information is easily accessible (ram, OS, etc)
- System parameters are easily accessible (AZT, motion, time outs, etc)

Discussion: Some discussion about system information requirements for the inspector took place… does the inspector really need to have access to OS, RAM information, etc? (General opinion seems to be if there is a dependency, then the NTEP lab would specifically include that requirement in the CoC.)

Audit trail info – the question was asked, does there need to be a specific requirement for providing access to this information?

Regarding the concept of First Final – There was some concern expressed as to how the inspectors are able to discern where the indication of first final be found for the SYSTEM (as opposed to the DEVICES in the system). What devices in the system are of concern to the inspector? The NTEP Administrator indicated that field inspectors need to follow the system all the way to receipt/bill generation.

Data transmission is an issue when considering systems as opposed to devices… how far does the inspector's jurisdiction extend? (Should we model future requirements on the WELMEC section concerning DTD/DSD?)
Decision: data transmission/storage is not currently being addressed by the Sector at this time.
Since part of the Sector's mission is education, do we want to assist in developing training aids for labs/inspectors related to evaluating/inspecting software-based devices? This will be a topic to be added to the Sector's agenda for the next meeting.

Status: The Sector will continue to develop this item, and initiate a new agenda item specific to inspector training in relation to evaluating/validating software-based devices.

8. NTEP Application for software requiring a separate Certificate of Conformance

Source: NTETC Software Sector

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

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

It was noted this agenda item is irrelevant if the NTEP Committee does not approve the pending item up for vote.

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

Conclusion of 2009 Sector Meeting: The Sector will continue to develop this item, contingent on the status of the related NTEP Committee agenda item after the 2009 Annual meeting.

Status: Since the NTEP committee passed the related item at the Annual, we should continue to work on this.

NEW ITEMS

9. Training of Field Inspectors

Source: NTETC Software Sector

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

10. Next meeting

The Sector is now on a yearly schedule for Sector meetings. The NTEP Administrator will determine when the next meeting is possible.

Appendix A: Report on 2009 Interim Meeting

Appendix B: Report on International W&M Activity