National Type Evaluation Technical Committee (NTETC)
Software Sector Meeting
May 20 & 21, 2008
Reynoldsburg, Ohio

Agenda Items

CARRYOVER ITEMS

1.a. NTETC Software Sector Mission

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.

Comments from October meeting:
Jim Truex noted there would 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.

1.b. NCWM/NTEP Policies – Issuing CCs for Software

Source: NCWM Reports

Background: Excerpts of reports from the 1995-1998 Executive Committee were provided to NTETC Software Sector members at their April 2006 meeting. The chair asked the sector to review the following NTEP policy decision adopted by the NCWM in 1998 relative to the issuance of a separate Certificate of Conformance (CC) for software.

The NCWM has struggled with software issues for many years. Prior to 1995, NTEP had evaluated stand-alone software (e.g. weigh-in / weigh-out, POS, and batch controller software) and, in some cases, had issued CCs for stand-alone software. The Board established a software work group to study the issues and make recommendations.

The work group discussed many issues, including: first indication of the final quantity, metrologically significant software, definitions, software marking, software checklist evaluation, a software EPO for the field inspector, user programmable software, and third party software. According to conference reports, it seems in 1997 some concerns were raised about the direction of the work group. In 1997, after the annual meeting, the NCWM chair appointed a new Software Work Group.

During the 1998 NCWM, the following recommendation was adopted as NTEP policy:

- **"Software, regardless of its form, shall not be subject to evaluation for the purpose of receiving a separate, software Certificate of Conformance from the National Type Evaluation Program."**
- **"Remove all of the software categories from the index of NCWM Publication 5, NTEP Index of Device Evaluations."**
- **"Reclassify all existing software CCs according to their applicable device categories."**

The policy is still in effect today.

Also noteworthy is a statement in Section C of NCWM Publication 14, Administrative Policy. It states: "In general, type evaluations will be conducted on all equipment that affect the measurement process or the validity of the transaction (e.g. electronic cash registers interfaced with scales and service station consoles interfaced with retail fuel dispensers); and all equipment to the point of the first indicated or recorded representation of the final quantity on which the transaction will be based."

October Meeting Discussion:

Some concerns were raised by the California laboratory regarding this recommendation. During the course of the discussion, these concerns were addressed and resolved.
Don Onwiler indicated that this may be a technical policy that needs to be inserted into each different volume or chapter of NCWM Publication 14 or it may need to be placed in the Administrative Policy volume.
It was agreed that overall, there would be no change to what is currently being done by NTEP and the labs to certify devices, however; the device type or name of the device certified would be changed.

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.

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.

TO DATE: The NTEP Committee has taken no action on this item.

1.c. Definitions for Software-Based Devices

Source: NTETC Software Sector

Background: Discussed was marking and G-S.1.1. It was initially suggested that "not built-for-purpose" be removed from the wording in NIST HB 44 G-S.1.1. However, after further discussion this may not be the correct or final decision. There is no definition for a not built-for-purpose device in HB 44. The current HB 44 definition for a built-for-purpose device reads:

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)

There was also the suggestion to use the definitions from the WELMEC document for Type P and Type U instruments. They were modified by the sector. It was also suggested that a list of examples be provided.

Draft definitions for consideration:

Built-for-purpose weighing or measuring instrument (device) (type P): A weighing or *measuring Instrument (device)* designed and built specially for the task in-hand. Accordingly, the embedded software is assumed to be designed for the specific task. It may contain many components also used in PCs, e.g. motherboard, memory card, etc.

A weighing or measuring instrument (device) using a universal Computer (type U): *A weighing or measuring Instrument (device)* that uses a general-purpose computer, usually a PC-based system, for performing metrologically significant functions.

Examples:
Type U
Weigh-in Weigh-out
Open Architecture

October Meeting Discussion:

After some discussion on this item, the Sector agreed to forward the recommendation to the S&T Committee.

Recommendation from the Sector to the S&T Committee:

The Sector recommended that the following definitions be submitted to the S&T Committee as a developing item and be considered for inclusion in NIST Handbook 44.

NEW DEFITION:

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.

From NCWM Publication 16, 2008

310-2 D Appendix D – Definition of Electronic Devices, Software-Based

Source: National Type Evaluation Technical Committee (NTETC) – Software Sector (This item was assigned developing status and moved to 360-2 Part 1, Item 2.)

Appendix A. Part 1, Item 2 Appendix D – Definition of Electronic Devices, Software-Based

(This item first appeared on the 2008 S&T Committee Interim Agenda as Item 310-2)

Source: National Type Evaluation Technical Committee (NTETC) – Software Sector

Recommendation: Add a new definition and cross-reference term to Appendix D in HB 44 for "Electronic devices, software-based" as follows:

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: During the NTETC Software Sector discussion on marking requirements and G-S.1.1. Location of Identification Information, it was initially suggested that the term "not-built-for-purpose" be removed from the wording in NIST HB 44 paragraph G-S.1.1. since there is no definition for a not-built-for-purpose device in HB 44. After a lengthy discussion related to the terms "built-for-purpose" and "not-built-for-purpose," the Sector agreed these terms were not clear and should be replaced with the terminology proposed above. The proposed definitions are base on the revision of OIML R 76 Non-automatic weighing instruments Subsections 5.5.1. (Type P) and 5.5.2. (Type U).

At the 2008 Interim Meeting, the SMA supported the intent of the item, but stated that it is premature to place these definitions in HB 44. The SMA recommended that the status of the item be changed to Developing on the S&T Committee Agenda. The Committee agreed to move Item 310-2 of the 2008 S&T Committee Interim Agenda and assign Developing status as 360-2 Part 1, Item 2.

1.d. Software Identification / Markings

Source: NTETC Software Sector

Background: At the last meeting, there was discussion on specific sections of the WELMEC document that deal with TYPE P and TYPE U requirements. The comments and recommendations under consideration are contained in the following.

October Meeting Comments:

This section needs to be completed with the actual changes to HB 44 sections
There is some concern with the note that is contained below Type P device.

There may be the need to have a delineation of devices with "firmware".
An exception may need to be made for a device that is "integral and blind"
It is possible that NTEP needs to determine if the "software" is integral and does not need to be identified.
Need to know the rules up front.

Metrologically significant software shall be clearly identified with the software version. The identification may consist of more than one part but one part shall be only dedicated for the metrologically significant portion.

Measurement Canada commented on "primary sensing elements": exemption from certain requirements (digital load cells and devices with correction methods) this is needed to prevent a "black box" could be added in between other main elements and then be exempt from certain requirements.

Difference may be that the Digital Load Cell has been evaluated integral, while the digital J-Box can be modified or built with various components and characterized in the field

One manufacturer still has a problem with the exemption, (footnote 3 below) and as an example used a smart J-box.

The “Via Menu (display) or Print option” may be supplemental for devices that use the hard-marked or continuously displayed identification method for the NTEP CC Make/Model, Serial No. information.

Metrologically Significant software shall be clearly identified with the software version. The identification may consist of more than one part but one part shall be only dedicated for the metrologically significant portion.

Currently there is no specification for permanence of the marking for software. (The CC No. on the screen) This will need to be addressed by the sector.

Developing Recommendation from the Sector to the S&T Committee:

The Sector recommended that the following marking information be submitted to the S&T Committee as a developing and be considered for inclusion in NIST Handbook 44.

TYPE P Shall meet at least one of the methods in each column:

Method	NTEP CC No.	Make/Model/Serial No.	Software Version/Revision[3]
Hard-Marked	X	X	Not Acceptable
Continuously Displayed	X	X	X
By command or operator action	Not Acceptable	Not Acceptable	X[4]

TYPE U Shall meet at least one of the methods in each column:

Method	NTEP CC No.	Make/Model	Software Version/Revision
Hard-Marked	X[1]	X	Not Acceptable
Continuously Displayed	X	X	X
Via Menu (display) or Print Option	Not Acceptable	X[2]	X[2]

[1] – Only if no means of displaying this information is available

[2] – Information on how to obtain Make/Model, Version/Revision shall be included on the NTEP CC.

[3] – If the manufacture declares that the primary sensing element "software" is integral, has no end user interface and no print capability, the element may be considered exempt from the marking requirement for version/revision.

Example: primary sensing element may be P.D. meter with correction, digital load cell. (only for reference, not limiting)

[4] - Information on how to obtain the Version/Revision shall be included on the NTEP CC.

From NCWM Publication 16, 2008

Appendix A. Part 1, Item 1 General Code: G-S.1. Identification – (Software)

Source: National Type Evaluation Technical Committee – Software Sector

Recommendation: Amend G-S.1. and/or G-S.1.1. to include the following:

Method	NTEP CC No.	Make/Model/Serial No.	Software Version/Revision[1]
TYPE P electronic devices shall meet at least one of the methods in each column:			
Hard-Marked	X	X	Not Acceptable
Continuously Displayed	X	X	X
By command or operator action	Not Acceptable	Not Acceptable	X[2]
TYPE U electronic devices shall meet at least one of the methods in each column:			
Hard-Marked	X[3]	X	Not Acceptable
Continuously Displayed	X	X	X
Via Menu (display) or Print Option	Not Acceptable	X[4]	X[4]

[1] If the manufacturer declares that the primary sensing element "software" is integral, has no end user interface and no print capability, the element may be considered exempt from the marking requirement for version/revision. Example: Primary sensing element may be Positive Displacement (P.D.) meter with integral correction, digital load cell (only for reference, not limiting).
[2] Information on how to obtain the Version/Revision shall be included on the NTEP CC.
[3] Only if no means of displaying this information is available.
[4] Information on how to obtain Make/Model, Version/Revision shall be included on the NTEP CC.

Metrologically significant software shall be clearly identified with the software version. The identification may consist of more than one part but one part shall be only dedicated for the metrologically significant portion.

Background/Discussion: In 2005, the Board of Directors established a NTETC Software Sector. The tasks of the Sector are to:

- Develop a clear understanding of the use of software in today's weighing and measuring instruments.
- Develop NIST HB 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 an NTEP Certificate. Training aids to educate manufacturers, designers, service technicians and end users may also be considered.

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 recommended that the recommendation to amend G-S.1. and/or G-S.1.1. be given Developmental status since additional work is needed to develop the appropriate language to amend paragraphs G-S.1. and G-S.1.1. The Sector is also interested in receiving input from the weights and measures community about this item. Working with input from the weights and measures community, the Sector plans to introduce proposed modifications to current requirements through the regional weights and measures associations and other technical committees. In the meantime, the Sector welcomes opportunities to discuss this item at regional weights and measures associations to ensure the item is adequately addressed.

To comment on this proposal, contact Steve Patoray spatoray@mgmtsol.com (e-mail), or by telephone at (828) 859-6178 or by mail at NCWM, Inc., 15245 Shady Grove Road, Suite 130, Rockville, MD 20850.

2. Identification of Certified Software

Source: NTETC Software Sector

October Meeting Discussion:

The sector agreed that the title of this item needs changed to "Identification of Certified Software."
Currently, use version no., ID no., Serial No., however, there is no physical tie to the actual software.
Some international documents, like Welmec document tell how to do tie the ID to the software; these include:

Possible methods: (not limited to)
CRC (cyclical redundancy check)
Checksum
Inextricably Linked version no.
Encryption

The question remains is there some method to give the W&M inspector information that something has changed?
How can the W&M inspector easily identify an NTEP Certified version?

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.

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

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)

Conclusion from the October Meeting: The sector will continue to develop this item.

3. Software Protection / Security

October Meeting Discussion:

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

From Welmec Document:

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

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

This requirement includes:
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.

Proposed checklist for Pub 14 numbering will still need to be added. This is based roughly on R 76 – 2 checklist and discussion at October Sector Meeting

Devices with embedded software TYPE P (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 metrologically significant 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 (not built-for-purpose)			
	The *metrologically significant* software is:		
	documented with all relevant 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 (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 control 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 ☐**

From previous notes this may be part of another section in the Pub.

Software identification		
	The metrologically significant software is identified by a software identification	**Yes ☐ No ☐ N/A ☐**
	The software identification:	
	covers all program modules of the metrologically significant software and the type-specific parameters at runtime of the instrument	**Yes ☐ No ☐ N/A ☐**
	is easily provided by the instrument	**Yes ☐ No ☐ N/A ☐**
	can be compared with the reference identification fixed at type approval	**Yes ☐ No ☐ N/A ☐**
	Spot checks whether the checksums (signatures) are generated and work as documented	**Yes ☐ No ☐ N/A ☐**
	There exists an effective audit trail	**Yes ☐ No ☐ N/A ☐**

Data storage devices (DSD) **From the previous meeting, this was tabled (This checklist was not reworked at this time)**		
5.5.3	**G.3.1**	DSD realised with embedded software (examine software acc. to G.1) Yes ☐ No ☐
		DSD realised with programmable/loadable software (examine software acc. to G.1) Yes ☐ No ☐

		documentation with all relevant information			
5.5.3.1	**G.3.2**	sufficient storage capacity for the intended purpose			
		data are stored and given back correctly			
		sufficient description of measures to prevent data loss			
5.5.3.2	**G.3.3**	storage of all relevant information necessary to reconstruct an earlier weighing, i.e. gross, net, tare values, decimal signs, units, identifications of the data set, instrument number, load receptor, (if applicable), checksum / signature of the data set stored.			
5.5.3.3	**G.3.4**	protection of the stored metrologically significant data against accidental or intentional changes			
		protection of the stored metrologically significant data at least with a parity check during transmission to the storage device			
		protection of the stored metrologically significant data at least with a parity check of a storage device with embedded software (5.5.1)			
		protection of the stored metrologically significant data by an adequate checksum or of a storage device with programmable or loadable software (5.5.2)			
5.5.3.4	**G.3.5**	identification and indication of the stored metrologically significant data with an identification number			
		record of the identification number on the official transaction medium, i.e. on the print-out			
5.5.3.5	**G.3.6**	automatic storage of the metrologically significant data			
5.5.3.6	**G.3.7**	a device subject to legal control prints or displays the stored metrologically significant data for verifying			

4. Software Maintenance and Reconfiguration

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

Source: NTETC Software Sector

October Meeting discussion:

This section taken from Document OIML D-SW Working Draft 1 WD and provided as background.

Maintenance and re-configuration

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

Verified update

The software to be updated can be loaded locally (e.g. directly) on the weighing or measuring device or remotely via a network. Loading and installation may be two different steps (as shown in Fig. above) or combined to one, depending on the needs of the technical solution. After update of the metrologically significant software of a weighing or measuring device (exchange with another approved version or re-installation) the weighing or measuring device is not allowed to be used for legal purposes before a (subsequent) verification of the instrument has been performed and the securing means have been renewed A person responsible for verification must be at place. (NOTE: This may need to be in the HB under user requirement.)

Traced update

The software is implemented into the instrument according to the requirements for traced update. Traced update is the procedure of changing software in a verified instrument or device after which the subsequent verification by a responsible person at place is not necessary. The software to be updated can be loaded locally (e.g. directly) on the weighing or measuring device or remotely via a network. The software update is recorded in an audit trail. The procedure of a traced update comprises several steps: loading, integrity checking, checking of the origin (authentication), installation, logging and activation.

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

The target measuring instrument (device, sub-assembly) shall have a fixed metrologically significant software that cannot be updated and that contains all of the checking functions necessary for fulfilling traced update requirements.

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 use the previous version of the software **or become inoperative**.

Technical means shall be employed to guarantee the integrity of the loaded software i.e. that it has not been inadmissibly changed before loading. This can be accomplished 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 use the previous version of the software **or become inoperative.**

It shall be guaranteed by technical means that software may only be updated with the explicit consent of the user or owner of the measuring instrument.

If the requirements above cannot be fulfilled, it is still possible to update the legally non-relevant software part. In this case, the following requirements shall be met:

- There is a distinct separation between the metrologically significant and non-relevant software.
- The whole metrologically significant software part cannot be updated without breaking a seal.
- It is stated in the type approval certificate that updating of the legally non-relevant part is acceptable.

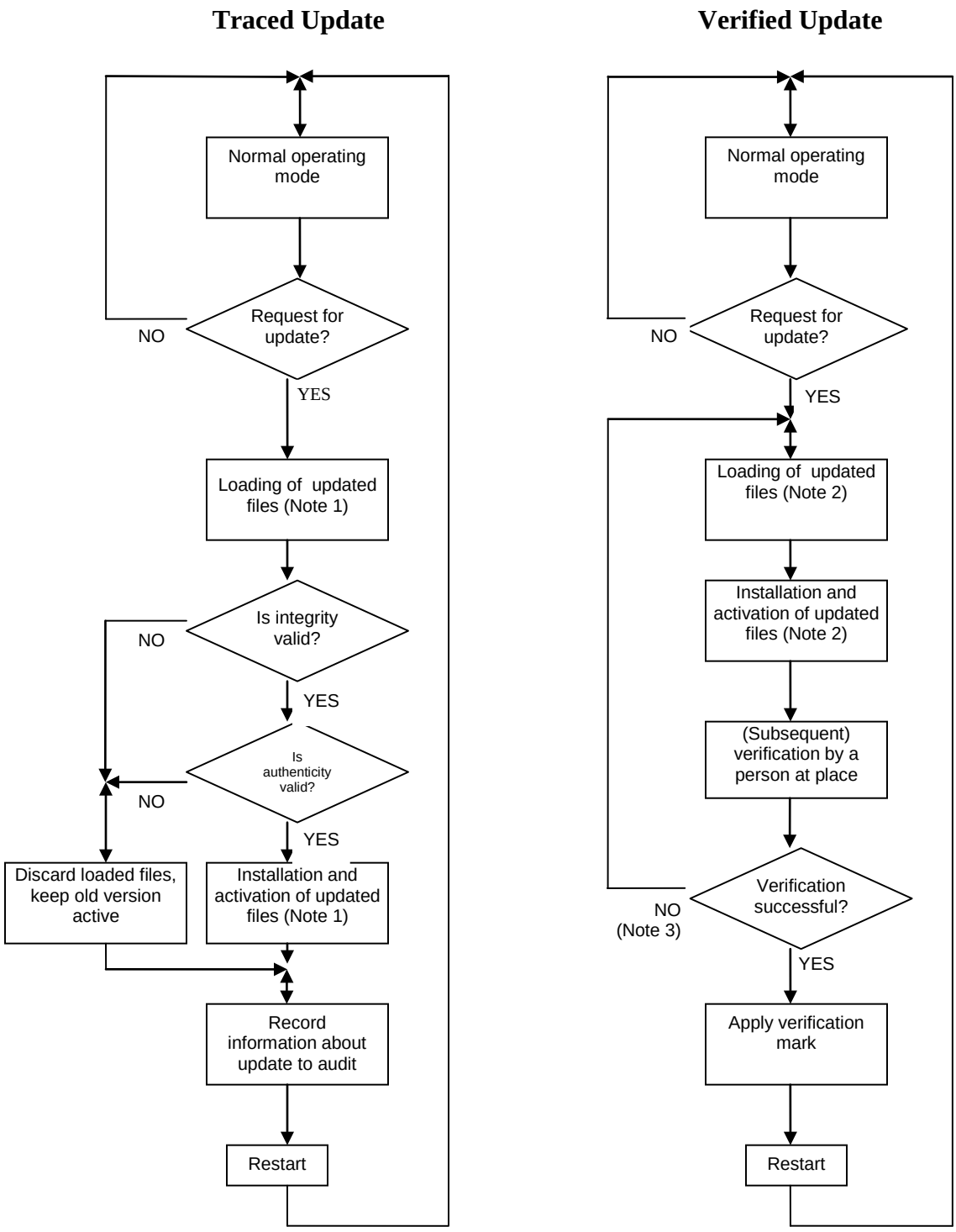

Figure 5-1: Software update procedures

Notes to

Figure 5-1**:**

1) In case of *Traced update* updating is separated into the steps: “loading” and “installing/activating”. This implies that the software is temporarily stored after loading without being activated because it must be

possible to discard the loaded software and fall back to the old version, if the checks fail **or become inoperative.**

2) In case of *Verified update*, the software may also be loaded and temporarily stored before installation but depending on the technical solution, loading and installation may also be accomplished in one-step.
3) Here, only failing of the verification because of the software update is considered. Failing because of other reasons doesn't require re-loading and re-installing of the software, symbolised by the NO-branch.

End of background information

Conclusions from October meeting discussion:

These four items are the accepted checklist questions:

1. Verify that the update process is documented
2. Software to be installed is authenticated and checked for integrity
3. Verify that the sealing requirements are met
4. Verify that if the upgrade process fails, the device is inoperable or the original software is restored

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).

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

- notification of the update procedure,
- software identification of the installed version,
- time stamp of the event,
- identification of the downloading party.

The traceability means and records are part of the metrologically significant software and should be protected as such. 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.***

The sector will continue to develop this item.

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

Source: NTETC Software Sector

October Meeting Comments:

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.
How to access audit trail
System information is easily accessible (ram, OS, etc)
System parameters are easily accessible (AZT, motion, time outs, etc)

Conclusion from the October meeting: The sector will continue to develop this item.

6. NTEP Application

Source: NTETC Software Sector

Conclusion from the October meeting: No direct discussion on this item took place at the October 2007 meeting.

7. Recommendations by the Sector on Sector Chair and Technical Advisor

Source: NTEP Director

With the changes to the management structure of NCWM, the Sector will need to discuss and make recommendations regarding persons to fill the roles of (NTETC) Sector Chair, and Technical Advisor to the Sector. Refer to NCWM Publication 14 Administrative Policy Section B. Administration, Subsection B.3. Paragraph 2. Page AP-4.

8. Next Meeting

The Sector is now on a yearly schedule for meetings.