National Type Evaluation Technical Committee (NTETC)
Software Sector Meeting
May 7-8, 2007
Sacramento, CA

Agenda Items

Carryover Items **Page**

New Items

<u>CARRYOVER ITEMS</u>

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.

Discussion:

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. It may be helpful for the sector to review the NTEP policy decision adopted by the NCWM 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.

Many issues were discussed by the work group, 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, a new Software Work Group was appointed by the NCWM chair.

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.”

Discussion:

Recommendation

2. 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 group. 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 is likely to contain many of the 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 legally relevant functions.

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

Discussion:

Recommendation:

3.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.

P1: Documentation

In addition to the specific documentation required in each of the following requirements, the documentation shall basically include:

a. A description of the legally relevant software.

b. A description of the accuracy of the measuring algorithms (e.g. price calculation and rounding algorithms).

c. A description of the user interface, menus and dialogues.

d. The unambiguous software identification.

e. 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.

f. The operating manual.

This should not be an issue,

P2: Software identification

The legally relevant software shall be clearly identified. An identification of the software shall be inextricably linked to the software itself. It shall be presented on command or during operation.

We should have addressed this in previous notes. Action item 1

P3: Influence via user interface
Commands entered via the user interface shall not inadmissibly influence the legally relevant software and measurement data.

There shall be a means to prevent changes from the user interface to any metrologically significant portions of the software and measurement data without authorization.(?) May need to define authorization per HB 44.

P4: Influence via communication interface
Commands inputted via communication interfaces of the instrument shall not inadmissibly influence the legally relevant software and measurement data.

There shall be a means to prevent changes from the communication interface to any metrologically significant portions of the software and measurement data without authorization. (?) May need to define authorization per HB 44.

P5: Protection against accidental or unintentional changes
~~Legally relevant~~ (metrologically significant, [find and replace]) software and measurement data shall be protected against accidental or unintentional changes.

P6: Protection against intentional changes
~~Legally relevant~~ (metrologically significant, [find and replace]) software shall be secured against the ~~inadmissible~~ unauthorized modification, loading or swapping of hardware memory.

P7: Parameter protection
~~Parameters that fix legally relevant characteristics~~ Metrologically Significant Parameters of the measuring instrument shall be secured against unauthorized modification.

U1: Documentation
In addition to the specific documentation required in each requirement below, the documentation shall basically include:

a. *A description of the legally relevant software functions, meaning of the data, etc.*
b. *A description of the accuracy of the measuring algorithms (e.g. price calculation and rounding algorithms).*
c. *A description of the user interface, menus and dialogues.*
d. *A legal software identification.*
e. *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.*
f. *An overview of the security aspects of the operating system, e.g. protection, user accounts, privileges, etc.*
g. *The operating manual.*

This one should be acceptable.

U2: Software identification
The ~~legally relevant~~ (metrologically significant, [find and replace]) software shall be clearly identified. An identification of the software shall be inextricably linked to the software itself. It shall be determined and presented on command or during operation.

Inextricably (cannot be separated)
AB: put a note in the checklist for the lab that they cannot "change" the ID?
This should be covered in permanence of marking

U3: Influence via user interfaces
Commands entered via the user interface shall not inadmissibly influence legally relevant software and measurement data.

Use words from P3

U4: Influence via communication interface

Commands or other inputs via ~~non-sealed~~ communication interfaces of the device shall not inadmissibly influence the legally relevant software and measurement data.

There are question on "sealed" this may not be a physical seal.
Being a U, the person selling, may not know about all of the interfaces

There shall be a means to prevent changes from any communication interface to ~~any~~ metrologically significant portions of the software and measurement data without authorization. (?) May need to define authorization per HB 44.

Means to prevent??
The word Commands, may limit what will need to be evaluated. That may be the intent

Done for now.

U5: Protection against accidental or unintentional changes

~~Legally relevant~~ (metrologically significant, [find and replace]) software and measurement data shall be protected against accidental or unintentional changes.

OK

U6: Protection against intentional changes

~~Legally relevant~~ (metrologically significant, [find and replace]) software and measurement data shall be secured against ~~inadmissible~~ unauthorized modification.

OK

U7: Parameter protection

~~Legally relevant~~ (metrologically significant, [find and replace]) parameters shall be secured against unauthorized modification.

Specifying Notes:

1. Type specific parameters are identical for each specimen of the type and are in general part of the program code i.e. part of the legally relevant software. Therefore requirement U6 applies to them.

2. Device specific parameters:

“Secured” parameters may be changed using an on-board keypad or switches or via interfaces but only *before* the action of securing. Because device specific parameters could be manipulated using simple tools *on universal computers they shall not be stored in standard storages of a universal computer*. Storing of these parameters is acceptable only in additional hardware.

Settable device specific parameters may be changed after securing.

OK

U8: Software authenticity and presentation of results

Means shall be employed to ensure the authenticity of the ~~legally relevant~~ (metrologically significant, [find and replace]) software. The authenticity of the results that are presented shall be guaranteed.

Had discussion on this on Weds,
RM, there is a method to ID that this is the actual software, trace update,

It shall not be possible to fraudulently simulate approved ~~legally (MS) relevant~~ *(metrologically significant, [find and replace])* software using simple software tools.

Definition for simple software tools, e.g. text editor, notepad, office tools, and other commonly available software tools.

U9: Influence of other software
The ~~legally relevant~~ (metrologically significant, [find and replace]) software shall be designed in such a way that other software does not inadmissibly (??) influence it.

This is DOOM!

A suggestion to consider a metrological device table was presented to the group. After modifications were made, the following table was discussed.

General Marking of Metrological Devices

	Software Only (this is U)	Software + Hardware (this is P)	Hardware Only (this is neither P nor U, mechanical)
Make	X	X	X
Model	X	X	X
Revision/Version	X	X	
COC	X	X	X
Serial Number		X	X

Discussion:

Recommendation:

4. Identification of Unapproved/Unauthorized Software

Source: NTETC Software Sector

Background: During the last meeting much discussion was generated. Many comments were addressed.

Segregation of parameters is currently allowed. (see table of sealable parameters)

Right now there are two methods, physical seal, audit trail, does the group believe that there needs to be some other category?

Currently, industry does protect software, but it is not audit trail.

There is an issue of audit trail, if the software is not running, or have a software service, the changes could be made and not tracked by audit trail.

There is no way to tell someone how to do sealing, you can say what needs to be accomplished.

Examples of methods of sealing.
authentication
access control
X509 Certificates,
PCATS certifies vendors
Version Number, application (checksum) There is a challenge response with different certifications. They validate who they are, there may also be limits set.
receive data verification

Does not believe that HB 44 does not need to be changed?

W&M needs to know that software is not being manipulated,

SW: X509 Certification, it is something like version, electronic signature and verification.

Scale System Controller
The scale system controller has approval certifications for USA and the European Union. In this case, a Commercial Off The Shelf (COTS) PC is used in conjunction with a scale system (terminal and weigh platform). The scale system provides the PC with approved gross weight and accepts commands to zero the weight indication. The PC application program

- stores and recalls weights
- computes net weight using a stored weight or manually entered weight
- provides the user display of net weight
- may compute price based on the net weight and a selected commodity code
- may print a weigh ticket

Protection of configuration and price parameters
Metrologically significant parameters are maintained within the scale terminal and are controlled there. Other parameters are stored in a password protected database. The user controls password protection access and distribution.

Separation of software
Separation of metrological and application software as described in the WELMEC documents is maintained.

Protection of software
Metrologically significant software is supplied only as binary code. Each such module is protected by a CRC32 checksum. The expected checksums, revision levels, and dates are kept in an encrypted configuration file. If run-time values differ from expected values the system will not operate. The configuration information can be recalled by an inspector using the Help/About menu in the application program.

Protection of active data
Data from the scale terminal is wholly owned by the scale server metrological interface. No other agent can acquire that data when the scale server is running, and the application program will not accept data except from the scale server.
Transactional information is stored in an encrypted Alibi Memory log. No access is permitted to this data except via the supplied application program. Data can be exported via the application program for external use, but no user modifications are permitted to the original transaction data.

Protection of operating system user interface
There are no special restrictions to the operating system. The application program runs as any other on the PC and can be started, stopped, or minimized.

In Europe, there are things like, safety, highest level security etc. First
modification there would be a limit to the risk classes.

P5: Protection against accidental or unintentional changes
Legally relevant 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.

Validation Guidance: Typical Examples
Checks based on documentation:

- Check that a checksum of the program code and the relevant parameters is generated and verified automatically.
- Check that overwriting of data cannot occur before the end of the data storage period that is foreseen and documented by the manufacturer.
- Check that a warning is issued to the user if he is about to delete measurement data files.

Functional checks:

- Check by practical spot checks that before deleting measurement data a warning is given, if deleting is possible at all.

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.

Discussion:

Recommendation:

5. Software Protection / Security

Source: NTETC Software Sector

JT note: The discussion from the last meeting on this issue is mingled in item 4. Appropriate sections need to be pulled out by the group.

Background:

Discussion:

Recommendation:

6. Software Maintenance and Reconfiguration

Source: NTETC Software Sector

Background: After discussion during the 10/06 meeting, it appeared these issues may go beyond the scope of current NTEP procedures, and possibly NTEP resources. The question was asked, does the sector need to address this issue? There was a split vote, no consensus, so it remains on the agenda.

OIML D-SW 5.2.6. was discussed. Comments included:

Only versions of legally relevant software that conform with 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 on the kind of instrument under consideration. The following options 5.2.6.1 and 5.2.6.2 are equivalent alternatives. This issue concerns verification in the field. Refer to chapter 7 for additional constraints.

This follows the traced update, the verified update is still an option.

This appears to be covered by Cat 3 and enforcement.
This may appear to be covered by other sections or security.
This section should not include eproms.
Is there a security key?
Does it download correctly?
OIML says that the audit trail needs to be updated.

The following flow chart, developed to assist the manufacturer/designer was discussed in depth.

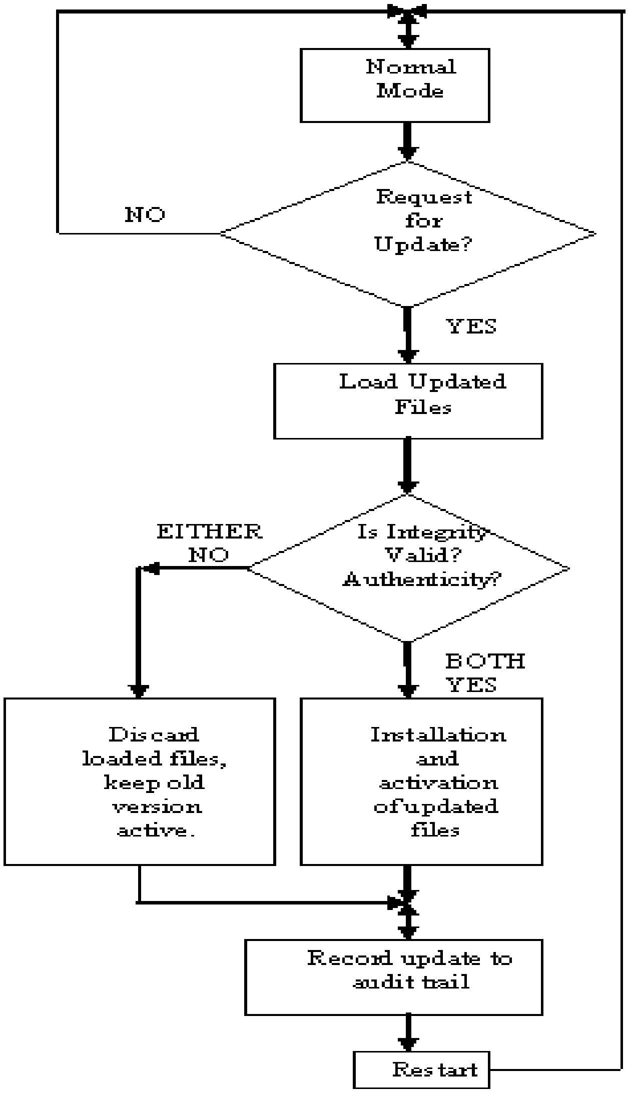
Normal Mode
Request for Update?
NO
YES
Load Updated Files
Is Integrity Valid? Authenticity?
EITHER NO
BOTH YES
Discard loaded files, keep old version active.
Installation and activation of updated files
Record update to audit trail
Restart

Figure 1.0 Traced Update Requirements

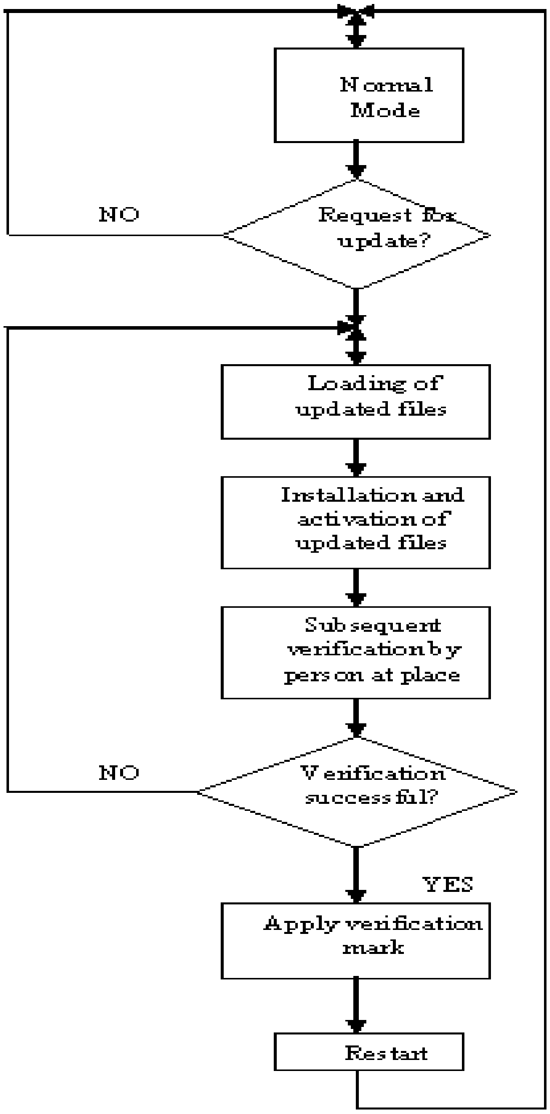

Figure 2.0 Verified Update Model

10/06 Conclusion: *It is apparent a lot more study and understanding of these complex issues are necessary. More discussion will need to take place during the next meeting. Sector members are encouraged to submit specific proposals for consideration.*

Discussion:

Recommendation:

7. Verification in the Field, By the Inspector

Source: NTETC Software Sector

***10/06 Recommendation**:* Cover this at another time. Ohio has developed a field checklist that may be used as a starting point.

8. NTEP Application – [mfg documentation to be submitted]

Source: NTETC Software Sector

***10/06 Recommendation*:** Cover this at a later time. Paul Lewis, Rice lake Weighing Systems, submitted info based on the OIML Document and information on what is now being requested by the laboratories.

NEW ITEMS

9. S&T Item 310-1 / G-S.2 Facilitation of Fraud

Source: NCWM S&T Committee

The S&T Committee has Item 310-1 on their agenda as a voting item. They have requested a position, pro or con, form the NTETC Software Sector. The following is item 310-1 as it appears in NCWM Pub. 16.

Recommendation: Amend Handbook 44, Section 1.10. General Code paragraph G-S.2. as follows:

> **G-S.2. Facilitation of Fraud** - All equipment, and all mechanisms, and devices **~~attached thereto or used in connection therewith~~**, **without limitation**, shall be so **designed**, constructed, assembled, and installed for use such that they do not facilitate the perpetration of fraud.
> **(Amended 2007)**

Background/Discussion: This proposal modifies the language in paragraph G-S.2. to clarify that the prohibition against facilitating fraud applies to the electronically programmed and coded components of weighing and measuring devices to address electronic manipulation or alteration. Some argue the existing language in Section 1.10. General Code. Paragraph G-S.2. Facilitation of Fraud is intended to address only hardware components of weighing and measuring devices. That is, “equipment, mechanisms, and devices” and the mechanics of how they are “constructed, assembled, and installed” appear to deal with tangible components. Fraud issues in the past ten years involved: (1) altering, manipulating, or interfering with software interfaced or installed in equipment; (2) microprocessor issues such as additional pulser units hidden in gas pumps and taximeters; and (3) software programs permitting manipulation of motor truck scale data used to generate weighmaster certificates.

The CWMA, the SWMA, and the WWMA recommended this item move forward for a vote.

The NEWMA recommended this item be referred to the NTETC Software Sector for review and input.

At the 2007 NCWM Interim Meeting, the Committee considered the WWMA proposal and an alternate proposal developed by the SMA. The Committee acknowledged that neither proposal was reviewed by the NTETC Software Sector. The Committee agreed that updating the requirement could be accomplished by adding general terms to address the types of electronic and software-based technology being fraudulently used today. The WWMA proposed language naming specific software applications that should not facilitate fraud. Whereas, the SMA alternate proposal included broader language that is intended to prohibit fraudulent use of software, wireless connections, and all future technology "without limitation." The Committee agreed that the SMA proposal encompasses all possible equipment configurations and more appropriately addresses the problem at hand. Therefore the Committee agreed to present the SMA proposal for a vote at the 2007 NCWM Annual Meeting.

Sector Discussion:

Sector Position:

10. Next Meeting

The NCWM Board agreed to fund a May 2007 meeting of the NTETC Software Sector. This is the third meeting of the sector in a thirteen month span. The meeting is being scheduled leading into a meeting of NTEP laboratory representatives. The scheduling was intentional, as the decision has been made that it is the "best fit", in an attempt to have as much NTEP lab(s) representation as possible. Piggybacking meetings also saves travel costs. Therefore, the next planned meeting of the Software Sector will be for the spring of 2008 in adjacent to the NTEP labs meeting.