**P21 e3 DIS Testing**

Minutes of May 20, 2104 Telecon

## Attendees

Martin Hardwick [<hardwick@steptools.com>](mailto:hardwick@steptools.com)   
Robert Lipman [<robert.lipman@nist.gov>](mailto:robert.lipman@nist.gov)   
Jacques Heinisch [<jacques.heinisch@3ds.com>](mailto:jacques.heinisch@3ds.com)   
Jochen Fritz [<jfritz@steptools.com>](mailto:jfritz@steptools.com)   
Michael Benda [<mjbenda@rockwellcollins.com>](mailto:mjbenda@rockwellcollins.com)   
Hedlind Mikael [<mikael.hedlind@scania.com>](mailto:mikael.hedlind@scania.com)   
Nicolas Figay [<nicolas.figay@eads.net>](mailto:nicolas.figay@eads.net)   
Dave Loffredo [<loffredo@steptools.com>](mailto:loffredo@steptools.com)   
Ed Paff  [<ejp@transcendata.com>](mailto:ejp@transcendata.com)   
Tom Thurman [<thomas.r.thurman@imonmail.com>](mailto:thomas.r.thurman@imonmail.com)

## PMI Splitting

ITI has split the sp3\_boxy file produced by NIST into a geometry file and a pmi file. STEP Tools has viewed the geometry file and is working on a viewer for the separated pmi file. The two files and the original NIST data can be found at the following URL:

<ftp://www.steptools.com/private/P21e3_DIS_testing/PMI/>

* We decided the names of the anchors should be made more meaningful by including the type name of the destination entity in each anchor name.
* We are developing open-source software to merge and split the files. The software is to be called the ***Thunder Code*** because two files enter and one file leaves.
* The Thunder Code will use mappings to determine where to split files.

## ZIP Assemblies

STEP Tools has developed a ZIP file P21 e3 equivalent for the as1\_ac.stp data set. The example is an outline. Except for the step files in the component sub-directory none of the data files are correct. The ZIP and the original data can be found at the following URL:

<ftp://www.steptools.com/private/P21e3_DIS_testing/ZIP_Assembly/>

* We decided the sub-tree directory names should include the NUAO identifier(s) for their corresponding node in the assembly tree.
* We learned that the LZMA algorithm has been shown to make STEP files three times more compressed than the more commonly used deflate algorithm, but at the cost of an increase in the compression time.
* We learned that each compressed file is required to document the algorithm used for its compression in its header and that different components in the same ZIP can used different compression algorithms.
* We recommended that the choice of the best compression algorithm should be left to the end user and that the standard should be silent on the matter.

## Unit Definitions

Tom Thurman has developed a set of files to show how the unit definitions of a STEP model can be centralized to one location. A set of sample files showing how the new units might be defined can be found at the following URL:

<ftp://www.steptools.com/private/P21e3_DIS_testing/Units/>

* If implementors are required to pick the right file then some will pick the wrong file name by mistake so we recommend that all of the unit definitions be put into one file.
* We should consider using URN’s for the unit definitions. A Universal Resource Name (URN) is a standard definition for a concept that reading algorithms are required to understand.
* The following URL and URN definitions are roughly equivalent except the URL identifies a “real” file, and the URN requires the reading software to implicitly know the referenced definitions

<http://standards.iso.org/iso/10303/tech/reference\_data/41/si\_base\_units.stp#METRE>;

<urn:iso:std:iso:10303:-41-:tech:unit:metre>;

Explanation of URN:

urn Indicates this URI is a URN, instead of the more common URL (http)

iso   URN namespace  (other examples are oid, usbn)

std ISO standard

iso originating organization (other examples are iec, iso-ies, iso-cie)

10303 STEP standard

-41- part of multipart standard (hyphens required)

Tech associated or embedded resource defined by committee that created the standard

<the rest> unspecified -- controlled by committee.

The iso namespace is defined by RFC 5141 (<http://tools.ietf.org/html/rfc5141>)

* The unit definition files need not be signed because their trustworthiness will be implied by their being posted on the ISO web site. However, they may be signed so that reading software can determine if they have changed since they were last read.

## Digital Signatures

We plan to add a section to the files for digital signatures. The new capability will be modeled after the tools used to sign the authenticity of DLL’s. The new capability will allow the file reader to verify its source, and validate that it has not been tampered with since it was written by the source.

* The digital signature section will be the last one in the file
* The signature will include a hash of all the contents of the file before the signature section.
* There may be multiple signatures when multiple organizations are required to authorize the file
* Any organization with the public key for the signature will be able to read and verify the hash value.
* Only the organization with the private key will be able to write the signature.
* The signature content will be as defined in the Cryptographic Message Syntax (CMS) defined by RFC 5652, clause 5 (<http://tools.ietf.org/html/rfc5652>).
* The CMS structure will be encoded as Base-64 within the P21 signature section.
* We will create open source examples to read and write the signatures.
* We will create a server to create signatures and certificates for testing purposes.

## Other

* We will continue to use the Wg11 exploder for broadcasting. Other exploders may be used occasionally for sending finished items such as these minutes, but anyone who wants to participate in the technical discussions must join this exploder which can be found at the following URL

<http://lists.steptools.com/mailman/listinfo/wg11>

* Please invite anyone who is interested in Part 21 to join the exploder.
* Tom Thurman will post these minutes to the appropriate place on the ISO web site.
* The next conference call will be held on Thursday June 5 at 4PM Paris, 3PM London, 10AM New York and 7AM Seattle.

## Action Items

1. Make a New ZIP assembly example with the NUAO names.
2. Consolidated all the unit definitions into one file.
3. Create an open source repository for the codes to split and merge files (thunder codes), and the codes to read and write digital signatures.
4. Investigate how the recommended practices of the EADS TDP project can be applied to the digital signatures section.

As recorded by Martin Hardwick [<hardwick@steptools.com>](mailto:hardwick@steptools.com)