

Award Number
SB1341-13-SE-0287

Procedure for building the STEP Module and Resource Library
ISO TC 184/SC 4 WG 12 N8496

Thomas Thurman

Acknowledgment of Sponsorship:

This report was prepared by Thomas Thurman using Federal funds under Award No. SB1341-13-SE-0287 from the National Institute of Standards and Technology (NIST), U.S. Department of Commerce. The statements, findings, conclusions, and recommendations are those of the author and do not necessarily reflect the views of NIST or the U.S. Department of Commerce.

March 24, 2014

Procedure for updating the STEP module and resource library.

1 Revision Page

Version: 0.9

Status: Draft for Review

Date: 3/24/14

2 Background

The purpose of this document is to provide a repeatable process for creating a new edition of the STEP Module and Resource Library (SMRL) for publication by ISO. An SMRL edition consists of html, PDF, and EXPRESS files and a set of indices. The file content is cumulative from one edition to the next but the indices are founded on the requirements for the Application Protocols (APs) to be created from the specific edition of the SMRL. Therefore ISO retains individual SMRL editions for use in creating AP documents for publication. For example, SMRLv6 will be used in the publication of the second edition of AP 209, the third edition of AP 210 and the first edition of AP 242. If a new edition of e.g., AP 233 were to be desired, an update to the SMRL would be required so that the indices were consistent with that AP.

Certain files on stepmod are used as SMRL control data files and are updated as part of the SMRL publication process. The SMRL control data files are in directory stepmod/data/library. They are

- part.xml,
- docs.xml,
- arm_index.xml and
- mim_ir_index.xml.

Procedure for updating the STEP module and resource library.

Source data for some information for the SMRL version is
the set of SMRL Change Request files

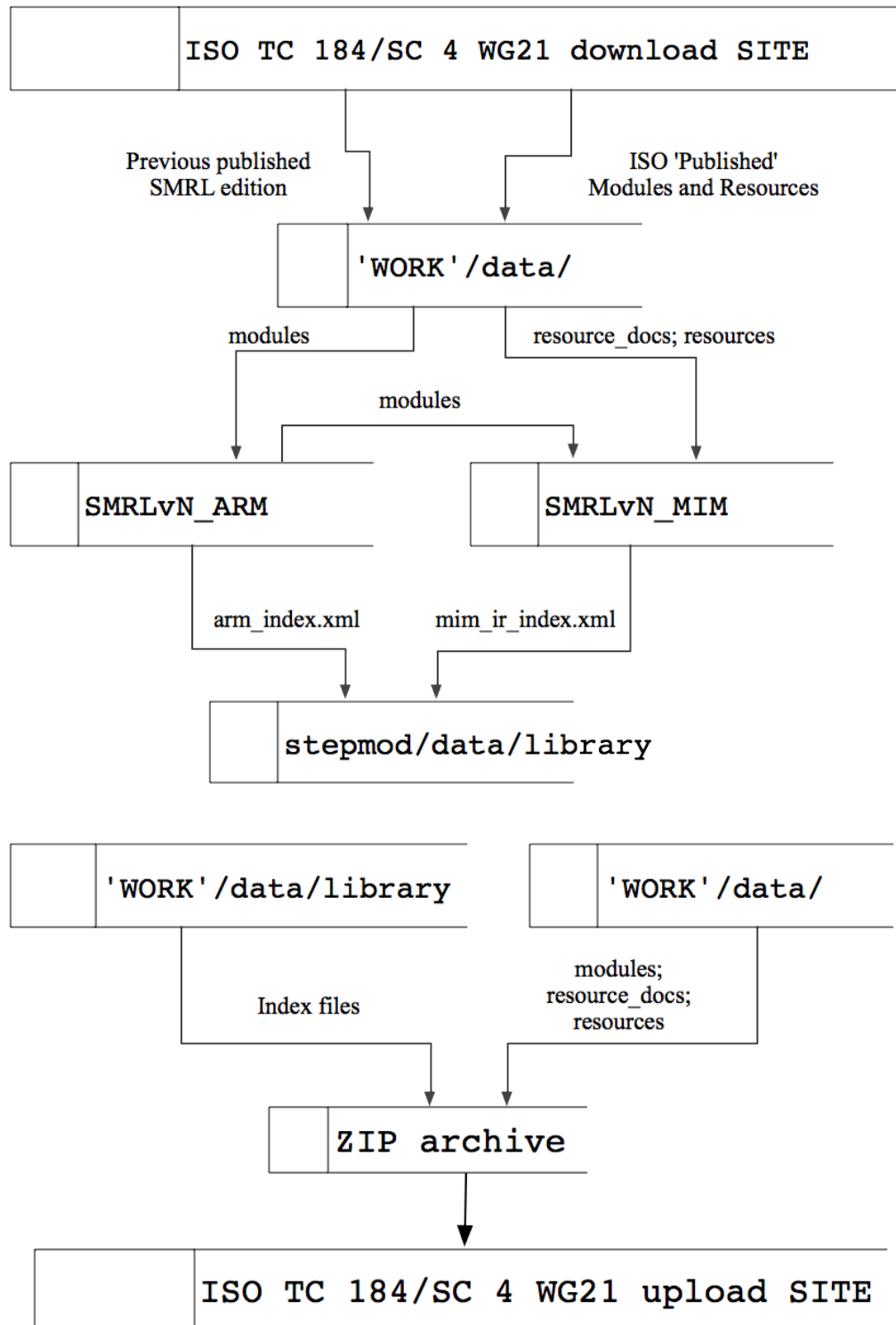
stepmod/publication/part1000/CR0000<N>/publication_index.xml.¹

¹ AP, IR, and AIC control data is not formally represented in publication_index.xml and must be obtained from WG12 convener.

3 Workflow

The figure illustrates the primary data flow that occurs during one iteration of the SMRL update process. Note in particular that there is no transport of documents from the local stepmod environment to the 'WORK' directory nor to the WG21 upload site.

Procedure for updating the STEP module and resource library.



Procedure for updating the STEP module and resource library.

Processing location

All processing occurs on local machine in the eclipse workspace². Some applications are executed from within eclipse Integrated Development Environment; some are command line applications executed within Windows XP(™).

pre-conditions

1. User has 'developer' status on stepmod.cvs.sourceforge.net
2. User is a member of TC 184/SC 4 WG 12 and WG 21 and has authorization to read and write to relevant pages at ISO.
3. User is familiar with eclipse client software GUI (www.eclipse.org)
 - Caution! If you forget to select apply! on any eclipse input screen the values won't be saved to disk.
4. An eclipse client is installed. The eclipse client is 4.2.2 or later and includes all updates.
 - Eclipse standard/sdk is sufficient for this use case.
 - By implication an eclipse workspace is also operational.
 - By implication a compatible version of Java JRE is installed and eclipse is configured for that Java installation. See eclipse help for details.
 - Set java memory settings as needed (64, 1024 typical).
5. User has an operational connection to ISO site for e committees for SC4. <http://isotc.iso.org/>
6. The LKSoft JSDAI eclipse EXPRESS toolkit is installed and operational
7. The manifest of Change Requests to include in SMRL version exists
8. The manifest of IRs/AICs to include exists
9. All documents specified in manifests are available from WG21 web page for validation ballot Change requests. Usually the same zip file exists on the evaluation ballot and the validation ballot subdirectories. Use the CR from the validation ballot directory only. e.g., [ISO 10303 SMRL CR9 Validation Ballot](#).

² The eclipse workspace will contain at least three projects: stepmod, SMRLvN_arm, and SMRLvN_mim

Procedure for updating the STEP module and resource library.

10. All resource documents to be included in SMRL that are not included in a CR and that are not available from the previous edition of the SMRL are available from WG12 web page. Example: ISO 13584-20 TC1 is not included in CR9 and was not included in SMRLv4 but is to be included in SMRLv5. Therefore the document is available from the WG12 web page.

11. The stepmod project from stepmod.cvs.sourceforge.net is checked out and on local machine.

12. Apache-ant version 1.8 or later is installed in e.g., C:\apache-ant\apache-ant-1.9.0. This is a separate install from eclipse.

13. Saxon version 9-1-0-8j is installed in e.g., C:\saxon9\saxonb9-1-0-8j. This is a separate install from eclipse.

14. Strawberry perl is installed in e.g., C:\strawberry; We used version: 5.10.1.1. This is a separate install from eclipse.

15. Stepmod/utis/part1000/smrl_setup.bat is modified to local environment.

16. Stepmod/utis/part1000/build.xml is modified to local environment if needed

17. A local target directory “WORK” (e.g., /Volumes/raid1/Users/tom/Documents/2013/SMRL_v5_rc3) exists. [This directory is NOT in local checked out copy of stepmod.] This directory will contain the final work product to be zipped and delivered. The name of this directory will evolve during iterations of the publication process unless a local version control repository is used. Instruction is correct application of a local version control repository is beyond the scope of this document.

18. A temp directory (e.g., /tmp) exists. The contents of this directory will be discarded.

19. An ARM working directory (e.g., workspace/SMRLv<N>_arm) exists³.

20. A MIM working directory (e.g., workspace/SMRLv<N>_mim) exists⁴.

Process

1. Setup JSDAI
2. Update stepmod/data/library/part.xml
3. Update stepmod/data/library/docs.xml
4. Create target SMRL directory
- Download all needed content.

Revise any non-html resource file names to agree with SMRL control data requirements.

³ This directory is at the same level as stepmod in the local workspace.

⁴ This directory is at the same level as stepmod in the local workspace.

Procedure for updating the STEP module and resource library.

5. Create ARM directory; add content to ARM directory
6. Create MIM directory; add content to MIM directory
7. Update arm_index.xml
8. Update mim_index.xml
9. Create cover page.
10. Upload SMRL to WG12 drop box for convener and ISO CS review.
11. Iterate over 1-10 until ISO publishes SMRL.
12. Remove local work directories and temp directory contents.

4 Setup JSDAI

JSDAI is available from <http://www.jsdai.net>;

To setup JSDAI, Set Global EXPRESS Preferences in eclipse

Caution! If you forget to select apply on any screen the values won't be saved to disk.

- select from top menu: Eclipse/Preferences/JSDAI/Express

- select radio button (leave the current perspective)

- select apply

- select from top menu: Eclipse/Preferences/JSDAI/Express compiler

- input of express files (the Express directory)

- select check box "Using (use excl..)"

- creating the library jar

- select radion button (no)

- select apply

- select from top menu: Eclipse/Preferences/JSDAI/Express compiler/more settings

- select checkbox 'enable expressions'

- select checkbox 'treat as REFERENCE FROM'

- select checkbox 'compile stepmod schemas'

- select checkbox 'compile arm files only'

- select checkbox 'compile stepmod mim files'

- select checkbox 'show advanced settings'

- select 'run the compiler in a separate process'

- select 'use custom memory size...'

- set initial value to 64

- set maximum value to 1024 (may need to increase if JAVA crashes due to memory 'heap'

issue)

- select apply

- select from top menu: Eclipse/Preferences/JSDAI/Express projects

- select checkbox 'Create template .exl file...'

- select checkbox 'Create template ..._excluded.exl file'

- select checkbox 'temp in Eclipse workspace'

- select apply

- select ok.

Procedure for updating the STEP module and resource library.

5 Update part.xml

2. Update part.xml to contain the correct header information for this SMRL version based on WG12 convener provided information. In particular, update

- ☐ part.version.number,
- ☐ part.publication.date, and
- ☐ part.publication.year.

6 Update docs.xml

1 In the case of existing modules, update elements:

- docs/doc.edition :=
part1000.publication_index/modules/module.version,
- docs/doc.pub_year_mo :=
part1000.publication_index/modules/date.iso_publicatio
n, or data provided by WG12 convener after CR is
balloted.⁵

2 In the case of new modules, add element instances:

- docs.xml/docs/doc.name :=
repository_index.xml/repository_index/modules/module.name;
- docs.xml/docs/doc.part :=
repository_index.xml/repository_index/modules/module.part;
- docs.xml/docs/doc.series := "AM";
- docs.xml/docs/doc.pub_type := "TS";
- docs.xml/docs/doc.edition := "1";
- docs.xml/docs/doc.format := "html-stepmod";
- docs.xml/docs/doc.pub_year_mo :=
part1000.publication_index/modules/date.iso_publication, or
data provided by WG12 convener after CR is balloted.

⁵ WG12 does not revise publication_index.xml file for date changes after a CR is balloted.

Procedure for updating the STEP module and resource library.

- 3 In the case of resource docs update resource elements1 based on data provided by WG12 convener.
- 4 In the case of withdrawn parts update file docs.xml by changing the element name from "doc" to "withdrawn_doc".
- 5 Ensure there are no duplicates in the combination of docs.xml/docs/doc and docs.xml/docs/withdrawn_doc.

7 Create WORK SMRL directory

Copy the baseline SMRL from previous version contents to the WORK directory.

```
{data/*.*/..., images, index.htm}
```

8 Create ARM directory

8.1 Create ARM subdirectories

1. Create following subdirectories in SMRLv<N>_arm⁶:

Complex entities,
express_arm_documentation,
express_files,
express_indices,
generated_arm_xml_and_exg,
Generated express,
Short names.

2. Create file SMRLv<N>_arm/SMRLv<N>.exl.

File SMRLv<N>.exl format:

SMRLv<N>_arm/express_files/data/modules/<module_name_1>/arm
.exp
SMRLv<N>_arm/express_files/data/modules/<module_name_2>/arm
.exp.

The source data for initialization is the docs.xml file updated earlier in this procedure. There is an XSLT transform included in clause [14.5 - Extract module names from docs.xml](#) that extracts the list of module directory names from a docs.xml file.

⁶Directories must be created prior to actually executing JSDAI in order to better control output results in case an unused option is selected in JSDAI.

8.2 Create JSDAI EXPRESS project FOR ARM

Create a new EXPRESS project in eclipse workspace for ARM

data (i.e., SMRLv<N>_arm).

1. set eclipse perspective to 'resource'
2. go to top menu and select File/New Project
3. select JSDAI/Express Project
4. select next
5. set Project name: SMRLv<N>_arm
6. use default location
7. select next
8. Input settings screen
 - select radio button "list"⁷
 - Directory value: <absolute path>/ SMRLv<N>_arm/express_files
 - example:: /Volumes/Users/tom/workspace/SMRLv5_arm/express_files⁸
 - otherwise just use defaults
 - select next
9. Output settings screen
 - Creating library jar
 - select "no, just check for express errors and, if success, create exd only"
 - Creation of empty inclusion and exclusion list files
 - input list
 - select "create"
 - Output list
 - select "create"
 - select next
10. Express compiler settings

⁷ If JSDAI is properly initialized the filename will be SMRLv<N>_arm.exl

⁸ Input directory is target for data insertion from ISO e-committee page

Procedure for updating the STEP module and resource library.

- Enable Express expressions
 - leave at default(yes)
 - Treat (*Reference from*) USE FROM as REFERENCE FROM
 - leave at default(no)
 - show advanced settings radio button
 - run compiler in separate process
 - select radio button (no)
 - compile stepmod schemas (yes)
 - compile arm files only (yes)
 - compile mim files only (yes)
 - select next
11. Temporary files
- Location of temporary files
 - leave at default (use the setting in the global express project preferences)
 - select finish

Procedure for updating the STEP module and resource library.

9 Create MIM directory

Create new EXPRESS project in eclipse workspace for MIM data (e.g., SMRLvN_mim) by repeating steps 1-11 from clause [8.1](#) – [Create ARM subdirectories](#) except use SMRLv<N>_mim in step 5.

9.1 Create MIM subdirectories

1. Create following subdirectories in SMRLv<N>_mim:

Complex entities,
express_mim_documentation,
express_files,
express_indices,
generated_mim_xml_and_exg,
Generated express,
Short names,
resource_doc_list_dir.

2. Create empty file SMRLv<N>_mim/SMRLv<N>.exl

Copy the ARM contents from SMRLv<N>_arm/SMRLv<N>.exl to SMRLv<N>_mim/SMRLv<N>.exl and change each occurrence of the string 'arm.exp' to 'mim.exp' and each occurrence of 'ARM' to 'MIM'.

Extract the resource directory names from the repository_index.xml file for each resource document in the docs.xml file. The part numbers are extracted from docs.xml and are then used to extract the names from repository_index.xml. Add the extracted names to SMRLv<N>_mim/SMRLv<N>.exl.

Procedure for updating the STEP module and resource library.

There is an XSLT transform included in [14.2 - Extract resource doc part numbers from docs.xml](#) that extracts the list of resource doc part numbers from docs.xml.

There is an XSLT transform included in [14.4 - Extract current resource names from repository_index.xml](#) that extracts the list of resources from repository_index.xml that was current as of 2014-01-31.

9.2 Create JSDAI EXPRESS project for MIM

Create a new EXPRESS project in eclipse workspace for MIM data (i.e., SMRLv<N>_mim).

1. Duplicate the process in clause [8.2 - Create JSDAI EXPRESS project FOR ARM](#) except all postfixes are _mim.

10 Update arm_index.xml

10.1 Populate ARM directory

Steps to follow to populate ARM subdirectories:

1. Initialize
 - a. Copy the previous SMRL data/modules directory contents to SMRLv<N>_arm/express_files/data/modules.
 - b. For each change request: Download the module change request CRxxx contents from the WG21 dropbox into a local directory WORK/CRxxx.
2. For each change request (applied sequentially) operate on SMRLv<N>_arm/express_files/data/modules⁹:
 - a. remove directories for deleted modules (specified in the
stepmod/publication/part1000/CR<N>/publication_index.xml/deleted.modules element),
 - b. remove directories for modified modules (specified in the
stepmod/publication/part1000/CR<N>/publication_index.xml/modules element),
 - c. add directories and contents for modified/new modules (specified in the

⁹There is an XSLT transform included in [14.1 - Extract module names from publication_index.xml](#) that will generate a text file containing the new and modified modules included in a publication_index.xml file.

Procedure for updating the STEP module and resource library.

stepmod/publication/part1000/CR<N>/publication_index.xml/modules element) from the local directory in (1b) above to SMRLv<N>_arm/express_files/data/modules using the file generated in (2b) as an index file.

10.2 Create ARM index

Right click on the file SMRLv<N>_arm/SMRLv<N>.exl and execute the JSDAI "Compile express schemas according to the list" command; check results in console.

If there are missing schemas in SMRLv<N>_arm/SMRLv<N>.exl, determine root cause and add to latest publication_index.xml file if those modules are in scope for this version of the SMRL.¹⁰ In the case that the modules are out of scope for this version of the SMRL, comment out the reference in the SMRLv<N>_arm/SMRLv<N>.exl.

Create a concatenated ARM EXPRESS file for each 4xx module to be supported by this version of the SMRL. Execute the shtolo function in EDMS from Jotne (<http://epmtech.jotne.com/products/express-data-manager-edm>) to create a long form for each 4xx module to be supported by this version of the SMRL. Execute the pretty printer (exppp) from stepcode (<https://github.com/stepcode/stepcode>) for each 4xx

¹⁰ There may be compile errors related to modules that have not been updated because they are out of scope of the current SMRL. Because SMRL versions are in support of specific APs the files from prior SMRLs are included but the index is not updated because of broken links cause by the legacy modules.

Procedure for updating the STEP module and resource library.
module. Check each long form in EDMS. Verify that each long form created from the SMRL is consistent with the long form included in the relevant change requests. E.g., for SMRL v6 verify that 409, 410, 442 long forms are unchanged from CR00010.

Upon success, generate "XML express index for part 1000" from JSDAI by:

right clicking on SMRLv<N>_arm.exd and selecting "export";

setting "export directory" to SMRLv<N>_arm/express_indices;

setting "File name created" to express_index.xml.

Rename "express_index.xml" to "arm_index.xml".

Check out stepmod/data/library/arm_index.xml.

Replace the checked out arm_index.xml with arm_index.xml created in previous operation using JSDAI;

Update the element <express_index
part.version.number="xxxx" > in arm_index.xml to actual number
(for publication this will be an INTEGER and for drafts "xxxx"
will be e.g., "Draft N for version NN") and commit the file to
stepmod.

11 Update mim_index.xml

11.1 Populate MIM directory

Steps to follow to populate MIM subdirectories:

1. Initialize
 - a. Copy the contents of
SMRLv<N>_arm/express_files/data/modules directory to
SMRLv<N>_mim/express_files/data/modules.
 - b. Copy the contents of the previous SMRL data/resources
directory contents to
SMRLv<N>mim/express_files/data/resources.
 - c. Copy the contents of the previous SMRL
data/resource_docs directory contents to
SMRLv<N>mim/express_files/data/resource_docs.
2. For each change request (applied sequentially):
 - a. remove directories for modified resource_docs based on
convener input,
 - b. remove directories for modified resources based on
resource_doc removal in (a),
 - c. add directories and contents for modified or new
resource_docs based on convener input.
 - d. add directories for resources based on added
resource_docs directories.

Procedure for updating the STEP module and resource library.

A part number from the convener may be converted to the resource_docs name and the resource contents determined using included xslt snippets.

There is an XSLT transform included in [14.4 - Extract current resource names from repository_index.xml](#) that extracts the list of resources that is current as of 2014-01-31.

There is an XSLT transform included in [14.6 - Extract a resource name from docs.xml given a part number](#) that extracts a resource_doc name from docs.xml.

There is an XSLT transform included in [14.3 - Extract a resource name for a specific part from repository_index.xml](#) that extracts a list of resources from repository_index.xml for a specific part.

Examine the results for inconsistencies.

Example: The file docs.xml lists 10303-59 as quality_of_product_shape. The file stepmod/repository_index.xml lists 10303-59 as quality_of_product_shape_and_related_data. The directory stepmod/resource_docs lists quality_of_product_shape_and_related_data.

11.2 Create MIM index

Right click on the file SMRLv<N>_mim/SMRLv<N>.exl and execute the JSDAI "Compile express schemas according to the

Procedure for updating the STEP module and resource library.

list" command; check results in console. If there are problems, correct the root cause and repeat¹¹.

Example: SMRLv4 included ISO 10303-110 pdf document but did not include basis_schema, conditions_schema, domain_schema, equations_schema, hierarchy_schema, or results_schema in resources directory. In this case, the schemas were in stepmod.

Example: SMRLv4 included ISO 10303-112 pdf document but did not include procedural_sketch_schema in resources directory. In this case the schema was not in stepmod.

Create a concatenated MIM EXPRESS file for each 4xx module to be supported by this version of the SMRL. Execute the shtolo function in EDMS from Jotne (<http://epmtech.jotne.com/products/express-data-manager-edm>) to create a long form for each 4xx module to be supported by this version of the SMRL. Execute the pretty printer (exppp) from stepcode (<https://github.com/stepcode/stepcode>) for each 4xx module. Check each long form in EDMS. Verify that each long form created from the SMRL is consistent with the long form included in the relevant change requests. E.g., for SMRL v6 verify that 409, 410, 442 long forms are unchanged from CR00010.

Upon success, generate "XML express index for part 1000" from JSDAI:

¹¹ The modules for ARM and MIM are required to be identical so any missing schema in the MIM and not in the ARM is indicative of an error in the source MIM EXPRESS

Procedure for updating the STEP module and resource library.

by right clicking on SMRLv<N>_mim.exd and selecting
"export";

exporting directory is SMRLv<N>_mim/express_indices;

setting "File name created" to express_index.xml.

Rename file name to mim_ir_index.xml.

Check out stepmod/data/library/mim_ir_index.xml.

Replace the stepmod/data/library/mim_ir_index.xml with
mim_index.xml created in previous operation.

Update the element <express_index
part.version.number="xxxx" > in mim_ir_index.xml to actual
number (for publication this will be an INTEGER and for drafts
"xxxx" will be e.g., "Draft 1 for version 5") and commit.

Procedure for updating the STEP module and resource library.

12 Create cover page

Execute

```
Cd ../stepmod/utils/part1000
```

```
smrl_build_setup.bat
```

```
Ant smrl-greadme
```

Output is written to directory setup in
smrl_build_setup.bat that is 'WORK/data/library'.

12.1 Add entry for withdrawn modules in index

To indicate a document is withdrawn, manually change html
page 'published' column value to 'withdrawn'.

e.g., from:

Part	Name	Edition	Published
1047	activity	2	2010-07

to:

Part	Name	Edition	Published
1047	activity	2	withdrawn

13 Upload SMRL to ISO

Once all issues have been resolved, zip the 'WORK'
directory and upload it to the ISO TC 184/SC 4 WG 21 upload
site.

14 Scripts

14.1 Extract module names from publication_index.xml

This is an XSLT transform¹² to extract the list of module names from a change request publication_index.xml. It ignores deleted modules.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
  <xsl:output method="text"/>
  <xsl:template match="/">
    <xsl:apply-templates
select="/part1000.publication_index/modules/module"/>
  </xsl:template>
  <xsl:template match="module">
    <xsl:value-of select="@name" xml:space="default"/>
    <xsl:text> &#xA; </xsl:text>
  </xsl:template>
</xsl:stylesheet>
```

14.2 Extract resource doc part numbers from docs.xml

This is an XSLT transform to extract the complete list of resource doc part numbers from docs.xml.

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xalan="http://xml.apache.org/xalan" xmlns:date="IntDate" exclude-result-
prefixes="xalan date" extension-element-prefixes="date" version="1.0">
  <xsl:output method="text"/>
  <xsl:template match="/">
    <xsl:apply-templates select="/docs/doc[@series='IGR' or @series='IAR' or
@series='AIC' or @series='PLIB']">
    </xsl:apply-templates>
  </xsl:template>
```

¹² filename:
stepmod/xsl/extract_module_name_from_publication_index.xsl

Procedure for updating the STEP module and resource library.

```
<xsl:template match="doc">
  <xsl:value-of select="@name"/>
  <xsl:text>&#xA;</xsl:text>
</xsl:template>
</xsl:stylesheet>
```

14.3 Extract a resource name for a specific part from repository_index.xml

The following XSLT may be used to extract a list of resources from repository_index.xml for a specific part (where NNNN is the part number) :

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xalan="http://xml.apache.org/xalan" xmlns:date="IntDate" exclude-result-
  prefixes="xalan date" extension-element-prefixes="date" version="1.0">
  <xsl:output method="text"/>
  <xsl:template match="/">
    <xsl:apply-templates select="/repository_index/resources/resource[
      @part='NNNN']">
    </xsl:apply-templates>
  </xsl:template>
  <xsl:template match="resource">
    <xsl:value-of select="@name"/>
    <xsl:text>&#xA;</xsl:text>
  </xsl:template>
</xsl:stylesheet>
```

14.4 Extract current resource names from repository_index.xml

This is an XSLT transform to extract the list of resources from repository_index.xml that is current as of 2014-01-31.

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xalan="http://xml.apache.org/xalan" xmlns:date="IntDate" exclude-result-
  prefixes="xalan date" extension-element-prefixes="date" version="1.0">

  <xsl:output method="text"/>
  <xsl:template match="/">

    <xsl:apply-templates select="/repository_index/resources/resource[
```

Procedure for updating the STEP module and resource library.

@part='41' or
@part='42' or
@part='43' or
@part='44' or
@part='45' or
@part='46' or
@part='47' or
@part='49' or
@part='50' or
@part='51' or
@part='52' or
@part='53' or
@part='54' or
@part='55' or
@part='56' or
@part='58' or
@part='59' or
@part='61' or
@part='101' or
@part='104' or
@part='105' or
@part='107' or
@part='108' or
@part='109' or
@part='110' or
@part='111' or
@part='112' or
@part='501' or
@part='502' or
@part='503' or
@part='504' or
@part='505' or
@part='506' or
@part='507' or
@part='508' or
@part='509' or
@part='510' or
@part='511' or
@part='512' or

Procedure for updating the STEP module and resource library.

```
@part='513' or
@part='514' or
@part='515' or
@part='517' or
@part='518' or
@part='520' or
@part='521' or
@part='522' or
@part='523' or
@part='13584-20']">
</xsl:apply-templates>
</xsl:template>
<xsl:template match="resource">
  <xsl:text>SMRLv5_mim/express_files/data/resources/</xsl:text>
  <xsl:value-of select="@name"/>
  <xsl:text>.exp</xsl:text>
  <xsl:text>&#xA;</xsl:text>
</xsl:template>
</xsl:stylesheet>
```

14.5 Extract module names from docs.xml

The following XSLT may be used to extract all module names from docs.xml and embed each module name in a string in accordance with the .exl file syntax requirements:

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xalan="http://xml.apache.org/xalan" xmlns:date="IntDate" exclude-result-
prefixes="xalan date" extension-element-prefixes="date" version="1.0">
  <xsl:output method="text"/>
  <xsl:template match="/">
    <xsl:apply-templates select="/docs/doc[@series='AM']">
    </xsl:apply-templates>
  </xsl:template>
  <xsl:template match="doc">
    <xsl:text>SMRLv5_arm/express_files/data/modules/</xsl:text>
    <xsl:value-of select="@name"/>
    <xsl:text>/arm.exp</xsl:text>
    <xsl:text>&#xA;</xsl:text>
  </xsl:template>
</xsl:stylesheet>
```

14.6 Extract a resource name from docs.xml given a part number

Procedure for updating the STEP module and resource library.

The following XSLT may be used to extract a resource_doc name from docs.xml given a part number:

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xalan="http://xml.apache.org/xalan" xmlns:date="IntDate" exclude-result-
prefixes="xalan date" extension-element-prefixes="date" version="1.0">
  <xsl:output method="text"/>
  <!-- source data file is docs.xml -->
  <!-- extract resource doc names from docs.xml -->
  <xsl:template match="/">
    <xsl:apply-templates select="/docs/doc[@part='NNNN']">
      </xsl:apply-templates>
    </xsl:template>
    <xsl:template match="doc">
      <xsl:text>data/resource_docs/</xsl:text>
      <xsl:value-of select="@name"/>
      <xsl:text>&#xA;</xsl:text>
    </xsl:template>
  </xsl:stylesheet>
```

Procedure for updating the STEP module and resource library.

15 resource builds

Target for CR build of resource part is now:

all

The 'all' target includes publish_isoresdocs but does not generate dependent resources nor does it generate publication_records for individual schemas.

a. Update contents of resource_doc_list_dir directory based on direction from convener (i.e., a list of part numbers).

- Each file in resource_doc_list_dir is the name of a resource doc (e.g., material_and_other_engineering_properties).
- contents of each file is the set of resources for that resource doc (e.g., material_property_definition_schema material_property_representation_schema qualified_measure_schema.)

□

16 Future work

The build script for pdf style resource part for CR remains to be documented.

The build script for part 1000 CR needs to be extended to support multiple document types in one publication_index.xml.

Procedure for updating the STEP module and resource library.

17 Works Cited

ISO TC 184/SC 4 N2538 Procedure for the maintenance of the
STEP Module and Resource Library 2009-12-03