HOWTO: niem-tools BOUML plugout

Tuesday, January 24, 2017 2:35 PM

1. Add niemtools-bouml.jar to the system environment property CLASSPATH
2. Copy the niemtools.properties file to your home directory and edit the target file locations to your environment
3. Create a NIEM IEPD project
    a. Copy BOUML-niem template OR
    b. Create a blank BOUML project, right click on the root package, select "Import project" and select the "niem-profile" BOUML project.
4. Model the exchange content (data)
    a. Create a UML class for each data object and set the stereotype to "niem".
    b. Create UML attributes under each class for each property of that object and set the stereotype to "niem"
    c. Create a UML class instance for each message and set the stereotype to "niem"
    d. Save the BOUML project
    e. Run "Tools->Generate NIEM IEPD" which will generate HTML documentation and blank NIEM mapping CSV and HTML files.
    f. Open tniem-mapping.csv in Excel and map each UML class and attribute to NIEM using the following columns:

Column	Description	Example 1	Example 2
NIEM XPath	A complete path from the msssage or common object (Case, Person, etc.) root , down to the specific element being modeled	nc:Case/j:CaseAugmentation/j:CaseDefendantParty/nc:EntityPerson/nc:PersonOtherIdentification/ecf:PersonIdentificationCategoryType	nc:Case/j:CaseAugmentation/j:CaseCourtEventj:CaseCourtEvent/ecf:CourtEventAugmentation/ecf:ConnectedDocument
NIEM Type	The NIEM or extension type that immediately contains the element	nc:IdentificationType	ecf:CourtEventAugmentationType
NIEM Property, @Reference, (Representation)	The NIEM or extension element(s). Multiple elements are separated by commas. Parentheses surrounding an element indicate a representation type that substitutes for the first element listed (assumed to be abstract). A "@" prefix to an element indicates that a reference to a similarly-typed element defined elsewhere is acceptable.	nc:IdentificationCategory, (ecf:PersonIdentificationCategoryCode)	@ecf:ConnectedDocument
NIEM Base Type	The NIEM or extension base type of the NIEM Property.	niem-xs:normalizedString	nc:DocumentType
NIEM Multiplicity	The mininum and maximum occurences, respectively, of the NIEM Property in the NIEM Type. A maximum occurrence of "unbounded" indicates no limit.	0,1	0,1
NIEM Mapping	Optional notes about the mapping. Notes related to extension type and	Genericode code list	

Notes	elements are included in the NIEM extension schemas as comments.	PersonIdentificationCategoryCode.gc	
Code List Code=Definition;	A list of acceptable code values and definitions of each value for a code list. A Genericode code list named <NIEM Property>.gc will be created.	DefendantNumber=Defendant Identifier;LocalAgencyID=Prosecutor Identifier;PersonID=Generic Person Identifier;PrisonerID=Jail Identifier	

- Note: You may optionally compare the mapping to a previous specification by populating the Old Xpath and Old Multiplicity columns. DO NOT modify the UML model columns, Model Class, Model Attribute, Model Multiplicity, Model Definition.
- Note: Mapping to NIEM namespace prefixes (e.g. "nc:") will be reflected in the NIEM wantlist and subset. Mapping to other namespace prefixes will generate a schema extension file with the filename <prefix>.xsd. Each message should be defined in its own extension file.

g. Save the niem-mapping.csv file
h. In BOUML, run "Tools->Import NIEM mapping" to import the mappings into the BOUML project.
i. Save the BOUML project
j. If there are existing UML packages named "NIEMSubset" and "NIEMExtension", delete these packages.
k. Run "Tools->Generate NIEM IEPD" again to regenerate the mapping CSV and HTML files, HTML documentation, the NIEM wantlist, and extension schemas.
    i. Mappings to invalid NIEM types and elements or invalid multiplicities will be reported. These errors are also shown in red in the niem-mapping.html file.
    ii. Differences between NIEM Xpath and Old Xpath and between NIEM Multiplicity and Old Multiplciity will be shown in blue in the niem-mapping.html file.
l. Repeat steps e-k until all mappings are complete and valid.

5. Model the components and interfaces
    a. Create a UML class for each component and set the stereotype to "interface".
    b. Create a UML operation under each class for each operation.
    c. Run "Tools->Generate NIEM IEPD" again to regenerate the mapping CSV and HTML files, HTML documentation, NIEM wantlist, extension schemas, and WSDL files.
6. Generate the NIEM subset
    a. Open the NIEM Subset Schema Generator Tool at https://tools.niem.gov/niemtools/ssgt/index.iepd
    b. Select "Options", "Browse", select the wantlist.xml file and "Load Wantlist".
    c. Select "Generate", "Save Subset Schema to a file" and save the ZIP file to the XSD folder that contains the XSD files created by the tool.
    d. Extract the contents of the ZIP file into a folder named "Subset".
7. Generate XML instances for each message.
    a. Using an XML editor (e.g. Altova XMLSpy), create and validate XML instances based on each schema that defines a message.
    b. If necessary, adjust mappings by repeating steps 2.e-k until all mappings are complete and valid.