NIEM Internationalization

An NTAC perspective on options for an internationalization strategy for NIEM.

NIEM Technical Architecture Committee (NTAC)

5/12/20

Table of Contents

1. Introduction

Internationalization of NIEM has been a topic of discussion for long time, but the term “Internationalization” can mean different things to different people. Internationalization focuses on creating independence from a specific locale, to encourage wider adoption. Within that broad objective, there can be many specific goals of an internationalization effort, and many routes to achieve those goals.

This document was written to clarify the NTAC’s perspective on preferred goals and preferred routes to achieving those goals.

2. Definitions

These definitions are from Wikipedia:[1]

Localization is the process of adapting internationalized software for a specific region or language by translating text and adding locale-specific components.

Internationalization is the process of designing a software application so that it can be adapted to various languages and regions without engineering changes.

A **locale** is a set of parameters that defines the user's language, region and any special variant preferences that the user wants to see in their user interface. Usually a locale identifier consists of at least a language code and a country/region code.

For NIEM, localization is the process of developing technical artifacts (e.g., translations, mappings, and data definitions) that support the use of NIEM in a different locale.

Localization is not the role of the main NIEM governance bodies or staff; localization is conducted by experts on a specific locale, for use with that locale. We expect that localization will be conducted by experts on a specific locale, and that implementations will use various localizations as needed to support their users and data requirements.

Internationalization is the process that ensures localization efforts can be successful. This may consist of ensuring that technical rules for NIEM (e.g., the NDR rules) support multiple languages, or ensuring that NIEM data definitions include places for localized content to be included (e.g., substitution groups and augmentation points). A NIEM internationalization strategy may enable users to define schema components, names, and definitions in any language.

3. Principles

An internationalization strategy for NIEM should maintain the principles that make NIEM work.

---

[1] Wikipedia, Internationalization and localization. https://en.wikipedia.org/wiki/Internationalization_and_localization

3.1. Ensure governance bodies control their own artifacts

A factor that contributes to NIEM's success is that each NIEM governance body controls its own artifacts. For example, the NBAC controls of the NIEM Core namespace and related schema documents; the XSTF controls the Justice domain namespace and related schema documents; the NTAC controls NIEM technical specifications. This distributed governance ensures that NIEM can progress in a controlled way, and that its participants get value from the process and products.

Local control implies that domain and core data model governance and localization efforts should not interfere with each other. Internationalization efforts should help ensure that domains and core produce data models that are easily localizable. Localization efforts should be able to produce language mappings, and localized components as needed, without altering domain and core data models or their governance.

An internationalization strategy should ensure that the local control provided by the NIEM governance model is maintained. Localization efforts should be allocated their own governance mechanisms, to ensure that all aspects of the NIEM governance and release process work together smoothly.

Counterexample: Require NIEM Core to include Esperanto definitions for each NIEM component, alongside the US English definitions. This would require a translation process that would interfere with the production and release of the NIEM Core schemas, which would take time and delay the release process, forcing it to wait on an outside body. Any *official* locale-specific translations should be carried in separate artifacts, providing translation of components from NIEM Core.

3.2. Maintain a single source of truth

The NIEM data model maintains a single source of truth:

Definition: a **single source of truth** is the practice of structuring information models and associated data schema such that every data element is mastered (or edited) in only one place. [2]

In the NIEM model, there is exactly one authoritative definition for each data component. For example, the definition of a person is defined only in the definition of complex type `nc:PersonType` within the NIEM Core schema. Other definitions

---

[2] "Single Source of Truth." Wikipedia. Wikimedia Foundation, March 30, 2020. https://en.wikipedia.org/wiki/Single_source_of_truth.

may derived, specialized, or translated from that type, but there is only one authoritative definition.

An internationalization process should ensure that localized definitions do not compete for authority with the single authoritative definition of a component. It should ensure that different localized expressions of a concept are built from the same authoritative definitions.

Counterexample: Define an Esperanto version of `nc:PersonType`, in competition with `nc:PersonType`. There's no real-world difference between the two concepts of person; the only difference is the name that we use to denote a person. NIEM is based in the community building off of the same set of components.

Additional counterexample: Define a copy of NIEM, translated to Esperanto. This would entail creating new components, competing with existing components. It would be best to maintain NIEM as a single governed set of components, with localization methods so that they can work across locales, rather than duplicating NIEM components.

3.3. Maintain a single unique identifer for each component

A principle that helps NIEM to work is that there is a single authoritative unique identifier for each component. NIEM expresses these identifiers as URIs, which are used to define namespaces and local names, which then help define qualified component names.

An internationalization strategy should maintain exactly one authoritative unique identifier for each NIEM component.

Counterexample: Define a new element for "Person" in Esperanto ("Persono"), and assign it a unique identifier matching the element name. An internationalization strategy should not create new unique identifiers for language translations of NIEM elements.

3.4. Maintain conformance to standards

A factor in NIEM's success has been its conformance to existing standards. NIEM's use of XML, XML Schema, and RDF in traditional ways has ensure that NIEM can work with free, open source, and off the shelf tools without NIEM-specific customization.

An internationalization strategy for NIEM should ensure that artifacts and their use conforms to the rules and intent of standards.

Counterexample: Define an alternate vocabulary for XML Schema in Esperanto, so that “complex type” elements can be expressed as “kompleksa tipo”. This would break conformance with XML Schema, requiring custom processes to use produced artifacts.

We should not create new URIs as authoritative identifiers for localized names for existing NIEM components.

4. Candidate goals

An internationalization strategy for NIEM can have a number of different intended outcomes. The effort should focus on providing specific capabilities that are needed by the community to support interoperability and participation.

4.1. Support governance of locale-specific schema components

At the time of this writing, the NIEM program has no explicit support for governance of locale-specific data components. Data components live under any of:

- Namespaces governed by the NBAC: the NIEM core namespace and supporting code list namespaces
- Namespaces governed by individual domains
- Namespaces that are a part of an IEPD or organizational effort to produce exchange models

No governance bodies have been stood up to specifically manage localized NIEM components. There has been discussion of creating explicit support for governance bodies other than domains, to manage portions of the NIEM data model. For example, there is a committee proposing components to support controlled unclassified information (CUI). This body is not a domain, yet it has the responsibility of managing a set of components for CUI metadata. A similar strategy could be applied to stand up bodies to support locale-specific data components.

4.2. Enable creation of components in any language

As of NIEM 4, NIEM schemas and components are defined in US English. A NIEM internationalization strategy could enable users to define schema components, names, and definitions in any language.

For example, the developer of an IEPD supporting a culturally-integrated community may wish to define components in various languages, all within the

same namespace. These components should be built with names, definitions, and structure that reflect their locale's languages and data requirements.

4.3. Enable creation of entire XML Schema documents in any language

An internationalization strategy should support the creation of XML Schema documents supporting any language. For example, an Esperanto-speaking community may wish to create a set of NIEM-conformant data components. It would be best for that community if those components were, for them, intuitive and easy to understand, and so they should use Esperanto for component names and text definitions.

4.4. Provide for an alternate-language definition of an existing component

Localization efforts can and should produce translations of NIEM component names and definitions, so that they can be understood and used by more people. It is important that these efforts do not interfere with the NIEM release process, governance of content, and NIEM's single source of truth.

A proven strategy is to enable developers to provide localizations, separately from the authoritative NIEM schemas. To enable this strategy, we could create an XML or JSON file format by which developers can provide translations of NIEM components. This may be a simple list of source component names, along with translations of each component's name and definition to a target language.

5. Impacts

Decisions on internationalization will have impacts on many pieces of the NIEM ecosystem:

- The NIEM Naming and Design Rules: See below; NIEM rules are built with a requirement for the use of English.
- The NIEM Model Package Description (MPD) Specification / IEPD syntax: Text and content describing an IEPD may need to be localized.
- The NIEM Code Lists Specification: Codes, definitions, and other fields have no explicit support for language descriptions.
- The NIEM metamodel: As NIEM concepts may be defined in different languages, that will put different demands on the metamodel, requiring language identifiers on component descriptions, and may require attributes to cover information currently put into component names (e.g., `MetadataType`)
- NIEM tools, including the Schema Subset Generation Tool (SSGT) and the Conformance Test Assistant (ConTesA) may need to be localized, and may

need to accommodate localization information to produce localized schemas or to support localized data instances.

- The NIEM release / NIEM data model: new namespaces, components, and updates may be required to accommodate localization.

5.1. NIEM NDR v4.0 rules on language

The NIEM NDR version 4.0 includes language that requires the use of English.[3]

Rule 10-44 requires words from a standard English-language dictionary: "The name of any XML Schema component defined by the schema SHOULD be composed of words from the English language, using the prevalent U.S. spelling, as provided by [OED]."

Rule 10-45 limits characters in component names in a way that prohibits many foreign-language names: "The name of an XML Schema component defined by the schema must be composed of only the characters uppercase 'A' through 'Z', lowercase 'a' through 'z', numbers '0' through '9', underscore, hyphen, and period."

The rules for the structure of names, with object term, property term, representation term, and qualifiers may be difficult or impossible in some languages. In addition, the representation terms are entirely English, and no accommodation is made for use of other terms.

The use of required representation terms for Metadata, Association, Type, and others will not be viable for languages other than English.

The rules for names with upper-case, lower-case, and camel-case parts will not be applicable to some languages.

The use of standard opening phrases for NIEM component definitions will not be applicable to languages other than English.

---

[3] Webb Roberts. "National Information Exchange Model Naming and Design Rules, Version 4.0." NIEM Technical Architecture Committee (NTAC), November 7, 2017.< https://reference.niem.gov/niem/specification/naming-and-design-rules/4.0/niem-ndr-4.0.html>.

6. Available methods

There are several techniques that NIEM may adopt to achieve its goals toward internationalization.

6.1. Localized NIEM components

Existing NIEM governance, and the international tiger team, should assess whether existing components for NIEM content are sufficient for international use. NIEM should develop components that are usable within the US and other countries and cultures.

Examples include:

- NIEM `nc:PersonNameType` is targeted towards Western names (prefix, given name, middle name, surname). Consider if it should be refactored for international names.
- NIEM dates are targeted toward the Gregorian calendar, which is sufficient for identifying a point in time, but is not sufficient for identifying a day or month of the Chinese, Islamic, or Hebrew calendars.
- NIEM addresses target US addresses, and may need different information for locations in other countries.

6.2. Additional namespaces and governance bodies

As described above, NIEM's governance has no explicit support for localization efforts for specific locales. NIEM should embrace the extension of the domain concept to support governance of sets of components and localization information on existing components, for specific locales.

6.3. The `xml:lang` attribute

XML defines the attribute `xml:lang`, which indicates the language of text contained within the element to which it applies. Attribute `xml:lang` is defined by the XML Specification. [4]

---

[4] "Extensible Markup Language (XML) 1.0 (Fifth Edition)". W3C Recommendation. 26 November 2008. https://www.w3.org/TR/REC-xml/#sec-lang-tag

The attribute xml:lang has lexical scope. The language identified by xml:lang applies to attribute values and simple content that are within the bounds of the element on which xml:lang appears. [5]

The values of xml:lang are defined by IETF BCP 47. [6]

6.3.1. xml:lang in data

The attribute xml:lang in data defines the language of text within the element where it appears. Any effort to enable multiple languages within NIEM data should ensure that xml:lang may appear within NIEM data to identify the language used within that data.

The attribute xml:lang appears in NIEM 4.2 only on type nc:TextType and its derived types. It may be introduced by developers in extension schemas, however its use in NIEM data is currently very limited.

An internationalization strategy should use xml:lang whenever it may facilitate better localization of data. The NBAC and domains should consider if xml:lang should appear in additional places within the NIEM release.

The xml:lang attribute may be added to XML Schema's xs:schema element, indicating a default language for the entire document.

Attribute xml:lang, when used within an XML Schema document, will only identify the language of text used within the schema itself. It will not affect the language of messages defined by the schema. To identify the language of messages defined by a schema, the schema should ensure that xml:lang can appear in instance data, to make the language of messages explicit.

The following is an example of xml:lang appearing on an XML Schema document, identifying the language of text within the schema itself.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="http://release.niem.gov/niem/niem-core/4.0/"
```

---

[5] "xml:lang in XML document schemas". https://www.w3.org/International/questions/qa-when-xmllang

[6] Phillips, A., Ed., and M. Davis, Ed., "Tags for Identifying Languages", BCP 47, September 2009. https://tools.ietf.org/html/bcp47

```
  version="1"
  xml:lang="en-US"
  xmlns:appinfo="http://release.niem.gov/niem/appinfo/4.0/"
  … >
  <xs:annotation>
    <xs:documentation>NIEM Core.</xs:documentation>
  </xs:annotation>
  …
```

The `xml:lang` attribute may be added to any XML Schema component definition, which indicates a language for the name, and a default language for component definitions.

The following is an example of `xml:lang` appearing on an element declaration. This language declaration identifies that the text of the element itself is Spanish. The lexical scope of the `xml:lang` attribute would affect the content and attributes of the element declaration; specifically, it would identify that the name and `xs:documentation` element contain Spanish-language text.

```
<xs:element name="Fecha" xml:lang="es"
    type="niem-xs:date"
    substitutionGroup="nc:DateRepresentation"
    nillable="true">
  <xs:annotation>
    <xs:documentation>Una fecha completa, con año, mes y
día.</xs:documentation>
  </xs:annotation>
</xs:element>
```

The `xml:lang` attribute may be applied to the definitions of XML Schema components, enabling alternate-language definitions for a schema component.

The following example shows an element declaration, with additional definitions, providing definitions in Spanish and French.

```
<xs:element name="Date" type="niem-xs:date"
  substitutionGroup="nc:DateRepresentation" nillable="true">
  <xs:annotation>
    <xs:documentation>A full date, with year, month, and
day.</xs:documentation>
    <xs:documentation xml:lang="es">Una fecha completa, con año,
mes y día.</xs:documentation>
```

```
    <xs:documentation xml:lang="fr">Une date complète, avec
l'année, le mois et le jour.</xs:documentation>
  </xs:annotation>
</xs:element>
```

6.4. Provide translations

To enable users and developers of NIEM exchanges to provide translations of NIEM schemas and messages.

In alignment with the goal of creating components and schemas in any language, while not interfering with the principles described above, a developer may provide a file that contains translations of NIEM component names and definitions into a target language.

Providing translations in a file separate from the original source schema allows translations to be performed by parties other than the originator of the schema. For example, an XML Schema document for the NIEM Core namespace may be accompanied by an XML or JSON document that identifies specific components and provides for each component a translated component name, along with a translated definition / documentation. These localized names and definitions may be used for presentation to humans with the correct locale.

6.4.1. XLIFF

The XML Localisation Interchange File Format (XLIFF) format is an example of a format developed to carry localizations for internationalized software. XLIFF is an OASIS standard for localization data, providing a standard format for carrying localized text between tools. [7]

Using XLIFF tools, a person can extract source documents into (1) a skeleton, containing the structure of the source document, with localizable text extracted and replaced by identifiers, and (2) an XLIFF document, carrying the extracted localizable text, with identifiers for each text location.

The separated XLIFF document, containing translatable text portions and identifiers, is translated into target languages. These translated files are carried as separate localizations that may be selected at some point, to be merged with the original source document, for presentation to the user in the appropriate localized form.

---

[7] https://www.oasis-open.org/standards#xliffv2.1

The XLIFF standard is fit for its process wherein a source document has translatable text removed, translating to localized forms, and then merging into the source document for localiz It is not evident that this usage pattern matches NIEM's requirements for either (1) localization of schema component names and definitions, (2) localization of schema component structure and relationships, or (3) localization of text within a NIEM instance document.

6.5. Incorporate localizations in tools

Although translations should be managed separately from NIEM Core, they may be accommodated by tools, to provide simple language-specific artifacts.

For example, if NIEM has been provided French and Spanish localizations for a set of NIEM components, those French and Spanish component names and definitions may be integrated into a schema set provided for development of an IEPD, to ease the use and understanding of those schemas within a locale.

7. Conclusion

An effort to internationalize NIEM should focus on:

1. Governance that supports localization efforts.
2. NIEM data components that support international use.
3. Technical specifications that support defining and understanding data models internationally.

The NTAC has considered internationalization and localization of NIEM data and data definitions, and welcomes concrete requirements from the NBAC and NIEM community to ensure efficient international use of NIEM.