RIXML Research v2.5 Release Notes

September 21, 2017

IndexEnum: EURO STOXX 50

EURO STOXX 50 [SX5E]

Description from the Stoxx site: “The EURO STOXX 50 Index, a Blue-chip index for the Eurozone, provides a Blue-chip representation of supersector leaders in the Eurozone. The index covers 50 stocks from 11 Eurozone countries: Austria, Belgium, Finland, France, Germany, Ireland, Italy, Luxembourg, the Netherlands, Portugal and Spain.”

Adding “EUROSTOXX50” to the **IndexEnum** enumeration that guides the **Index** element.

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name=“IndexEnum”>

  <xs:restriction base=“xs:string”>

    …

    <xs:enumeration value=“EUROSTOXX50” />

    …

  </xs:restriction>

</xs:simpleType>
```

IndexEnum: Canadian Indices

CSE Composite

Description from the CSE site: “The CSE Composite Index is a broad indicator of market activity for the Canadian Securities Exchange (CSE). With approximately 75% coverage of all equities listed on the CSE it is a uniquely positioned gauge of the Canadian small cap market. The index provides a distinctly different risk/return profile than the broad Canadian equity market.”

Adding “CSE-COMPOSITE” to the **IndexEnum** enumeration that guides the **Index** element.

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name=“IndexEnum”>
  <xs:restriction base=“xs:string”>
    …
    <xs:enumeration value=“EUROSTOXX50” />
    <xs:enumeration value=“CSE-COMPOSITE” />
    …
  </xs:restriction>
</xs:simpleType>
```

IndexEnum: Canadian Indices

S&P/TSX Composite [SPTSX]

Description from the S&P site: “The S&P/TSX Composite is the headline index for the Canadian equity market. It is the broadest in the S&P/TSX family and is the basis for multiple sub-indices including but not limited to equity indices, Income Trust Indices, Capped Indices, GICS Indices and market cap based indices. The Toronto Stock Exchange (TSX) serves as the distributor of both real-time and historical data for this index.”

Adding “SPTSX-COMPOSITE” to the **IndexEnum** enumeration that guides the **Index** element.

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name=“IndexEnum”>
  <xs:restriction base=“xs:string”>
    …
    <xs:enumeration value=“EUROSTOXX50” />
    <xs:enumeration value=“CSE-COMPOSITE” />
    <xs:enumeration value=“SPTSX-COMPOSITE” />
    …
  </xs:restriction>
</xs:simpleType>
```

StatusTypeEnum: Back-Filled

Re-send content to back-fill without jumping to the top

The **StatusTypeEnum** enumeration that guides the **StatusInfo.statusType** attritube already includes a value for “Revised”, but it does not explicitly cover the case where a publisher is back-filling (or re-transmitting) a report that has already been published. The intent is to back-fill the report without having it jump to the top of any time line. It should be “inserted” into the time line at its original publication date. For example, re-transmitting a report on January 19 that was originally published on the prior November 17 should NOT cause it to appear at the top of any list of reports on January 19 on any platform.

Adding “BackFilled” to the **StatusTypeEnum** enumeration that guides the **StatusInfo.statusType** attribute.

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name=“StatusTypeEnum”>
  <xs:restriction base=“xs:string”>
    …
    <xs:enumeration value=“BackFilled” />
    …
  </xs:restriction>
</xs:simpleType>
```

StatusTypeEnum: Tags-Updated

When only the tags are changed, not the payload

The **StatusTypeEnum** enumeration that guides the **StatusInfo.statusType** attritube already includes a value for “Revised”, but it does not explicitly cover the case where a publisher is only updating the indexing information (tags) in the RIXML Research file without updating any payload file (e.g. PDF). This would improve transmission efficiency by eliminating unnecessary re-sending of potentially large payload file(s) in these cases. And it explicitly informs the receiver not to expect the payload file(s), so false failures are also avoided.

Adding “TagsUpdated” to the **StatusTypeEnum** enumeration that guides the **StatusInfo.statusType** attribute.

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name=“StatusTypeEnum”>
  <xs:restriction base=“xs:string”>
    …
    <xs:enumeration value=“BackFilled” />
    <xs:enumeration value=“TagsUpdated” />
    …
  </xs:restriction>
</xs:simpleType>
```

StatusTypeEnum: Revised w/ Material Change

Need to distinguish revisions with and without material changes

The **StatusTypeEnum** enumeration that guides the **StatusInfo.statusType** attritube already includes a value for “Revised”, but it does not distinguish between revisions that contain material changes and those that do not. We want to add two new values so publishers can make that distinction explicit. The intent is to give platforms enough information to make meaningful decisions about jumping the revision to the top of the time line.

- Revised: Existing value does not specify whether or not material change was made in this revision.

Adding the below values to the **StatusTypeEnum** enumeration that guides the **StatusInfo.statusType** attribute.

- RevisedMaterialChange: New value that indicates at least one material change was made in this revision.
- RevisedNoMaterialChange: New value that indicates no material changes were made in this revision.

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name=“StatusTypeEnum”>
  <xs:restriction base=“xs:string”>
    …
    <xs:enumeration value=“BackFilled” />
    <xs:enumeration value=“TagsUpdated” />
    <xs:enumeration value=“RevisedMaterialChange” />
    <xs:enumeration value=“RevisedNoMaterialChange” />
    …
  </xs:restriction>
</xs:simpleType>
```

Entitlement: New sub-elements

Broadening support for different types of entitlements

The RIXML Research Schema already offers several **Entitlement** types (sub-elements).

Adding the following new optional **Entitlement** types:

- **ProductCategoryEntitlement**: Limit content to subset of Comment, Report, Model, Compilation, Event, Primer, etc..
- **AssetClassEntitlement**: Limit content to subset of Equity, FixedIncome, Currency, Commodity, RealEstate, etc..
- **OrganizationTypeEntitlement**: Limit audience to organizations of certain types only – e.g. BuySideFirm, Government, etc..
- **ActionEntitlement**: Limit audience actions to subset of View, Save, Print, etc..

```
File: RIXML-common-2_5.xsd


<xs:complexType name="EntitlementType">
  <xs:sequence>
   ...
   <xs:element ref="ProductCategoryEntitlement” minOccurs=”0"/>
   <xs:element ref=”AssetClassEntitlement” minOccurs=”0"/>
   <xs:element ref=”OrganizationTypeEntitlement” minOccurs=”0"/>
   <xs:element ref=”ActionEntitlement” minOccurs=”0"/>
   ...
  </xs:sequence>
</xs:complexType>
```

Entitlement: New sub-elements

Broadening support for different types of entitlements

The RIXML Research Schema already offers several **Entitlement** types (sub-elements).

Adding the following new optional **Entitlement** types:

- **ProductCategoryEntitlement**: Limit content to subset of Comment, Report, Model, Compilation, Event, Primer, etc..
- **AssetClassEntitlement**: Limit content to subset of Equity, FixedIncome, Currency, Commodity, RealEstate, etc..
- **OrganizationTypeEntitlement**: Limit audience to organizations of certain types only – e.g. BuySideFirm, Government, etc..
- **ActionEntitlement**: Limit audience actions to subset of View, Save, Print, etc..

```
File: RIXML-common-2_5.xsd


<xs:element name="ProductCategoryEntitlement">
  <xs:annotation>
    <xs:documentation>An entitlement based on the
product category of the research.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ProductCategory”
         maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Entitlement: New sub-elements

Broadening support for different types of entitlements

The RIXML Research Schema already offers several **Entitlement** types (sub-elements).

Adding the following new optional **Entitlement** types:

- **ProductCategoryEntitlement**: Limit content to subset of Comment, Report, Model, Compilation, Event, Primer, etc..
- **AssetClassEntitlement**: Limit content to subset of Equity, FixedIncome, Currency, Commodity, RealEstate, etc..
- **OrganizationTypeEntitlement**: Limit audience to organizations of certain types only – e.g. BuySideFirm, Government, etc..
- **ActionEntitlement**: Limit audience actions to subset of View, Save, Print, etc..

```
File: RIXML-common-2_5.xsd


<xs:element name=”AssetClassEntitlement">
  <xs:annotation>
    <xs:documentation>An entitlement based on the
asset class of the research.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element ref=”AssetClass”
         maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Entitlement: New sub-elements

Broadening support for different types of entitlements

The RIXML Research Schema already offers several **Entitlement** types (sub-elements).

Adding the following new optional **Entitlement** types:

- **ProductCategoryEntitlement**: Limit content to subset of Comment, Report, Model, Compilation, Event, Primer, etc..
- **AssetClassEntitlement**: Limit content to subset of Equity, FixedIncome, Currency, Commodity, RealEstate, etc..
- **OrganizationTypeEntitlement**: Limit audience to organizations of certain types only – e.g. BuySideFirm, Government, etc..
- **ActionEntitlement**: Limit audience actions to subset of View, Save, Print, etc..

```
File: RIXML-common-2_5.xsd


<xs:element name=”OrganizationTypeEntitlement">
  <xs:annotation>
    <xs:documentation>An entitlement based on the organization type of the research
consumer.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="organizationType" type="rixmldt:OrganizationTypeEnum"
use="required">
          <xs:annotation>
            <xs:documentation>Indicates the organization type that the author
intended to address with the product.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
```

Entitlement: New sub-elements

Broadening support for different types of entitlements

The RIXML Research Schema already offers several **Entitlement** types (sub-elements).

Adding the following new optional **Entitlement** types:

- **ProductCategoryEntitlement**: Limit content to subset of Comment, Report, Model, Compilation, Event, Primer, etc..
- **AssetClassEntitlement**: Limit content to subset of Equity, FixedIncome, Currency, Commodity, RealEstate, etc..
- **OrganizationTypeEntitlement**: Limit audience to organizations of certain types only – e.g. BuySideFirm, Government, etc..
- **ActionEntitlement**: Limit audience actions to subset of View, Save, Print, etc..

```
File: RIXML-common-2_5.xsd


<xs:element name=”ActionEntitlement">
  <xs:annotation>
    <xs:documentation>An entitlement based on the actions of the research consumer
permitted by the research publisher.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name=”action" type="rixmldt:ActionEnum" use="required">
          <xs:annotation>
            <xs:documentation>Indicates the specific action to be performed by the
research consumer on the research product.</xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
```

Entitlement: New sub-elements

Broadening support for different types of entitlements

The RIXML Research Schema already offers several **Entitlement** types (sub-elements).

Adding the following new optional **Entitlement** types:

- **ProductCategoryEntitlement**: Limit content to subset of Comment, Report, Model, Compilation, Event, Primer, etc..
- **AssetClassEntitlement**: Limit content to subset of Equity, FixedIncome, Currency, Commodity, RealEstate, etc..
- **OrganizationTypeEntitlement**: Limit audience to organizations of certain types only – e.g. BuySideFirm, Government, etc..
- **ActionEntitlement**: Limit audience actions to subset of View, Save, Print, etc..

```
File: RIXML-datatypes-2_5.xsd


<xs:simpleType name="ActionEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="View"/>
    <xs:enumeration value="ViewOffline"/>
    <xs:enumeration value="Save"/>
    <xs:enumeration value="Download"/>
    <xs:enumeration value="BatchDownload"/>
    <xs:enumeration value="Print"/>
    <xs:enumeration value="BatchPrint"/>
  </xs:restriction>
</xs:simpleType>
```

Names in Multiple Languages

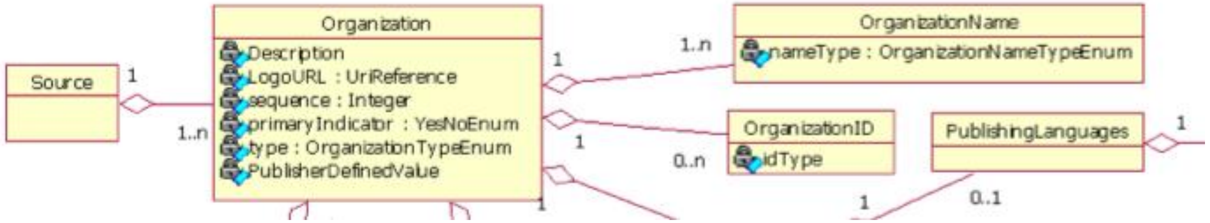
Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract



The OrganizationName element is already one-to-many under Organization. All we need to do is add an optional "language" attribute and we can include additional language translations of the organization name.

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:element name="OrganizationName">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="nameType"
                type="rixmldt:OrganizationNameTypeEnum" use="required” />
        <xs:attribute name="language"
                type="rixmldt:ISO639-2LanguageType" use="optional">
          <xs:annotation>
            <xs:documentation>Indicates the language of the name value given.
Overrides the language attribute on the Research element. </xs:documentation>
          </xs:annotation>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

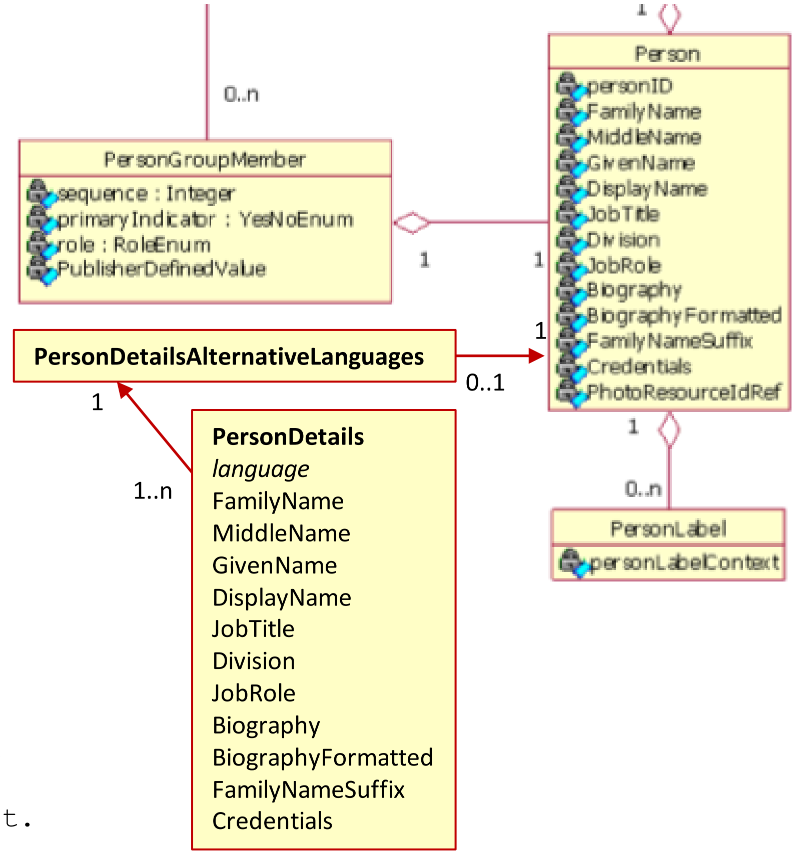
- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

In order to avoid breaking compatibility with v2.4, we must throw normalization to the wind and add a duplicative construct beneath Person to house the name data in other languages.

We do this via the new PersonDetails construct.



Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:complexType name="PersonDetailsType">

  <xs:sequence>

    <xs:element name=”JobRole" type="xs:string” minOccurs="0”/>

    <xs:element name="FamilyName" type="xs:string”/>

    <xs:element name="GivenName" type="xs:string" minOccurs="0”/>

    <xs:element name="DisplayName" type="xs:string" minOccurs="0”/>

    …

  </xs:sequence>

  <xs:attribute name="language"

        type="rixmldt:ISO639-2LanguageType" use="required”>

    <xs:annotation>

      <xs:documentation>Indicates the language of the Person Details. If not
specified, default to value of language attribute of top Research element. If
language attribute of Research element is not specified, default to English.
Described using the ISO 639-2/T Code. </xs:documentation>

    </xs:annotation>

  </xs:attribute>

</xs:complexType>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:element name="PersonDetails" type="PersonDetailsType">

  <xs:annotation>

    <xs:documentation>Includes name and job information for a person in a
specific language. </xs:documentation>

  </xs:annotation>

</xs:element>

<xs:element name="PersonDetailsAlternativeLanguages">

  <xs:annotation>

    <xs:documentation>Container element for one or more individual
PersonDetails elements meant to offer name and job information in a
specific, non-default language. </xs:documentation>

  </xs:annotation>

  <xs:complexType>

    <xs:sequence>

      <xs:element ref="PersonDetails" minOccurs="1" maxOccurs="unbounded"/>

    </xs:sequence>

  </xs:complexType>

</xs:element>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:complexType name="PersonType">
  <xs:annotation>
    <xs:documentation>ComplexType for Person</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    …
    <xs:element ref="ContactInfo" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="PersonLabel" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref="PersonDetailsAlternativeLanguages" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="personID" type="xs:string" use="required”/>
</xs:complexType>
```

Names in Multiple Languages

Toward improved globalization

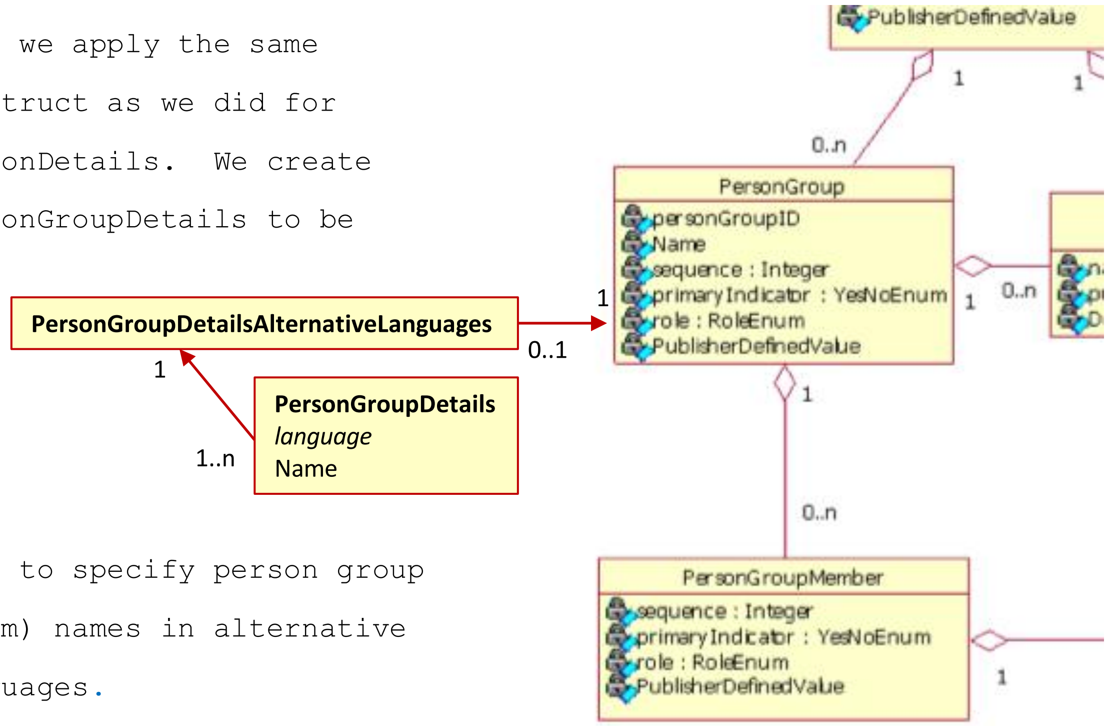
The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

Here we apply the same construct as we did for PersonDetails. We create PersonGroupDetails to be able to specify person group (team) names in alternative languages.



Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:complexType name="PersonGroupDetailsType">

  <xs:sequence>

    <xs:element name=”Name" type="xs:string"/>

  </xs:sequence>

  <xs:attribute name="language"

        type="rixmldt:ISO639-2LanguageType" use="required”>

    <xs:annotation>

      <</xs:documentation>xs:documentation>Indicates the language of
the Person Group Details. If not specified, default to value of
language attribute of top Research element. If language attribute of
Research element is not specified, default to English. Described
using the ISO 639-2/T Code.

    </xs:annotation>

  </xs:attribute>

</xs:complexType>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:element name="PersonGroupDetails" type="PersonGroupDetailsType">

  <xs:annotation>

    <xs:documentation>Includes name information for a person group in a specific
language. </xs:documentation>

  </xs:annotation>

</xs:element>

<xs:element name="PersonGroupDetailsAlternativeLanguages">

  <xs:annotation>

    <xs:documentation>Container element for one or more individual
PersonGroupDetails elements meant to offer name information in a specific, non-
default language. </xs:documentation>

  </xs:annotation>

  <xs:complexType>

    <xs:sequence>

      <xs:element ref="PersonGroupDetails" minOccurs="1" maxOccurs="unbounded"/>

    </xs:sequence>

  </xs:complexType>

</xs:element>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:element name="PersonGroup">
  <xs:complexType>
    <xs:sequence>
      …
      <xs:element ref= "PersonGroupMember" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="ContactInfo" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="PersonGroupDetailsAlternativeLanguages" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="personGroupID" type="xs:string" use= "optional" />
    …
  </xs:complexType>
</xs:element>
```

Names in Multiple Languages

Toward improved globalization

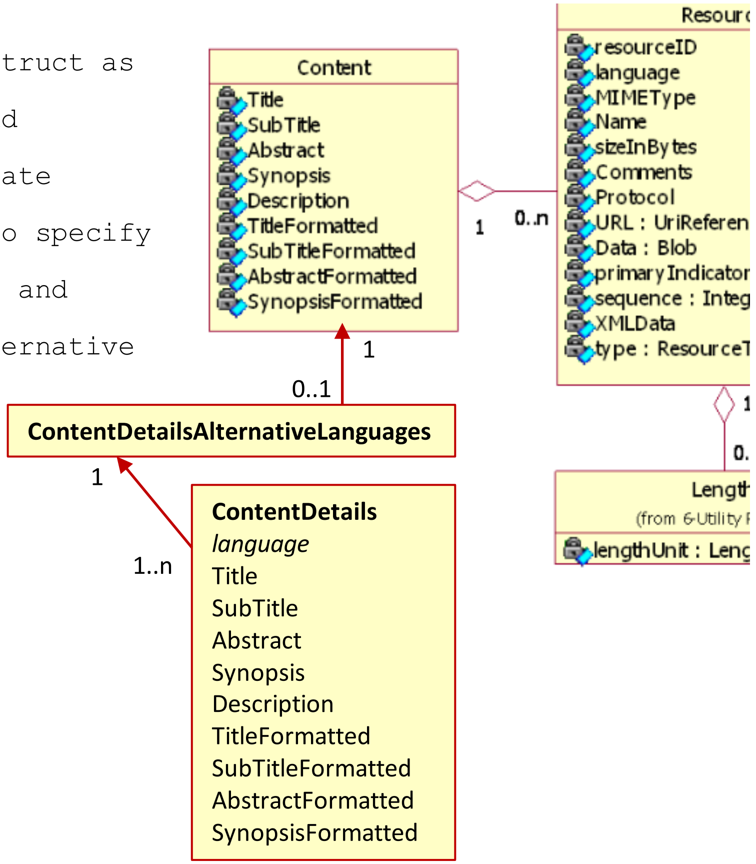
The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

Here we apply the same construct as we did for PersonDetails and PersonGroupDetails. We create ContentDetails to be able to specify title, sub-title, synopsis, and abstract information in alternative languages beneath the Content element.



Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:complexType name="ContentDetailsType">
  <xs:sequence>
    <xs:element name="Description" type="xs:string" minOccurs="0"/>
    <xs:element ref="Title”/>
    <xs:element ref="TitleFormatted” minOccurs="0”/>
    <xs:element ref="SubTitle" minOccurs="0”/>
    <xs:element ref="SubTitleFormatted" minOccurs="0”/>
    <xs:element ref="Abstract" minOccurs="0”/>
    <xs:element ref="AbstractFormatted" minOccurs="0”/>
    <xs:element ref="Synopsis" minOccurs="0”/>
    <xs:element ref="SynopsisFormatted" minOccurs="0">
  </xs:sequence>
  <xs:attribute name="language" type="rixmldt:ISO639-2LanguageType”
                use="optional”/>
</xs:complexType>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:element name="ContentDetails" type="ContentDetailsType">

  <xs:annotation>

    <xs:documentation>Includes title, sub-title, synopsis, and abstract information
for a content element in a specific language. </xs:documentation>

  </xs:annotation>

</xs:element>

<xs:element name="ContentDetailsAlternativeLanguages">

  <xs:annotation>

    <xs:documentation>Container element for one or more individual ContentDetails
elements meant to offer title, sub-title, synopsis, and abstract information in a
specific, non-default language. </xs:documentation>

  </xs:annotation>

  <xs:complexType>

    <xs:sequence>

      <xs:element ref="ContentDetails" minOccurs="1" maxOccurs="unbounded"/>

    </xs:sequence>

  </xs:complexType>

</xs:element>
```

Names in Multiple Languages

Toward improved globalization

The RIXML Research Schema already supports multiple languages in a few ways.

- Each payload document (**Resource**) is tagged with the primary language in which the document is written.
- Tag values in a RIXML Research instance document can be in any language indicated in an attribute on the top **Research** element.
- Languages can be specified as forms of organizational **Expertise**.

We now improve our support for multiple languages by providing publishers with a method for offering names in more than one language. Larger publishers often do business in many countries with different native languages.

- Organization Name
- Person Name, PersonGroup (Team) Name
- Title, Sub-Title, Synopsis, Abstract

```
File: RIXML-Common-2_5.xsd


<xs:complexType name="ContentType">
  <xs:sequence>
    …
    <xs:element ref="Synopsis" minOccurs="0”/>
    <xs:element ref="SynopsisFormatted" minOccurs="0">
    <xs:element ref="Resource" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element ref=”ContentDetailsAlternativeLanguages" minOccurs="0”/>
  </xs:sequence>
</xs:complexType>
```

Event: Announcement, Invitation, Summary

Need to distinguish

The RIXML Research Schema already supports “events”. But we need to better support the distinction between a pre-event announcement, an invitation to an event, and a post-event summary.

The existing “Event” value in the **ProductCategoryEnum** enumeration is described as a “meeting notification” and will continue to serve as the tag for pre-event announcements.

Adding “EventInvitation” and “PostEventSummary” to the **ProductCategoryEnum** enumeration that guides the **ProductCategory.productCategory** attribute.

```
File: RIXML-datatype-2_5.xsd


<xs:simpleType name=”ProductCategoryEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value=”Comment"/>
    <xs:enumeration value=”Report"/>
    …
    <xs:enumeration value=”Overview"/>
    <xs:enumeration value=”EventInvitation"/>
    <xs:enumeration value=”PostEventSummary"/>
    <xs:enumeration value=”PublisherDefined"/>
  </xs:restriction>
</xs:simpleType>
```

Specialty in CoverageUpdates Side-Car

Support Specialties as Covered Entities

Adding **Specialty** to the **CoveredEntitiesType** complex type that guides the **CoveredEntity** element.

```
File: RIXML-CoverageUpdates-2_5.xsd


<xs:complexType name="CoveredEntitiesType">
  <xs:choice maxOccurs="unbounded">
    <xs:element ref="AssetClass"/>
    <xs:element ref="AssetType"/>
    <xs:element ref="SecurityType"/>
    <xs:element ref="Issuer"/>
    <xs:element ref="Security"/>
    <xs:element ref="SectorIndustry"/>
    <xs:element ref="Region"/>
    <xs:element ref="Country"/>
    <xs:element ref="Index"/>
    <xs:element ref="Discipline"/>
    <xs:element ref="Specialty"/>
  </xs:choice>
</xs:complexType>
```

TitleFormatted: Cardinality Bug in v2.4!

Should be optional

The **TitleFormatted** element in the **Content** package is meant to be optional, with a cardinality of 0..1, as described in the Data Dictionary. But in the schema definition, the “minOccurs=0” attribute is missing. This is a bug.

Making **TitleFormatted** optional under **Content**. The XSD specification for **TitleFormatted** needs the “minOccurs=0” attribute added.

```
File: RIXML-Common-2_5.xsd


<xs:complexType name="ContentType">

  <xs:sequence>

    <xs:element name="Description" type="xs:string" minOccurs="0"/>

    <xs:element ref="Title”/>

    <xs:element ref="TitleFormatted" minOccurs="0”/>

    …
```

IssuerSecurityIDTypeEnum: DUNS Number

Identifying private companies

Description from the Dun & Bradstreet site: “The D&B D-U-N-S® Number is a unique nine-digit identifier for businesses. It is used to establish a business credit file, which is often referenced by lenders and potential business partners to help predict the reliability and/or financial stability of the company in question. D-U-N-S, which stands for data universal number system, is used to and maintain accurate and timely information on +265M global businesses.”

Our Interactions work stream wants to be able to tag private companies and the DUNS Number is a common identifier type used for that purpose.

Adding “DUNS” to the **IssuerSecurityIDTypeEnum** enumeration that guides the **IssuerID.idType** and **SecurityID.idType** attributes.

```
File: RIXML-datatype-2_5.xsd


<xs:simpleType name="IssuerSecurityIDTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="CUSIP"/>
    <xs:enumeration value="SEDOL"/>
    …
    <xs:enumeration value="ExchangeTicker"/>
    <xs:enumeration value="DUNS"/>
  </xs:restriction>
</xs:simpleType>
```