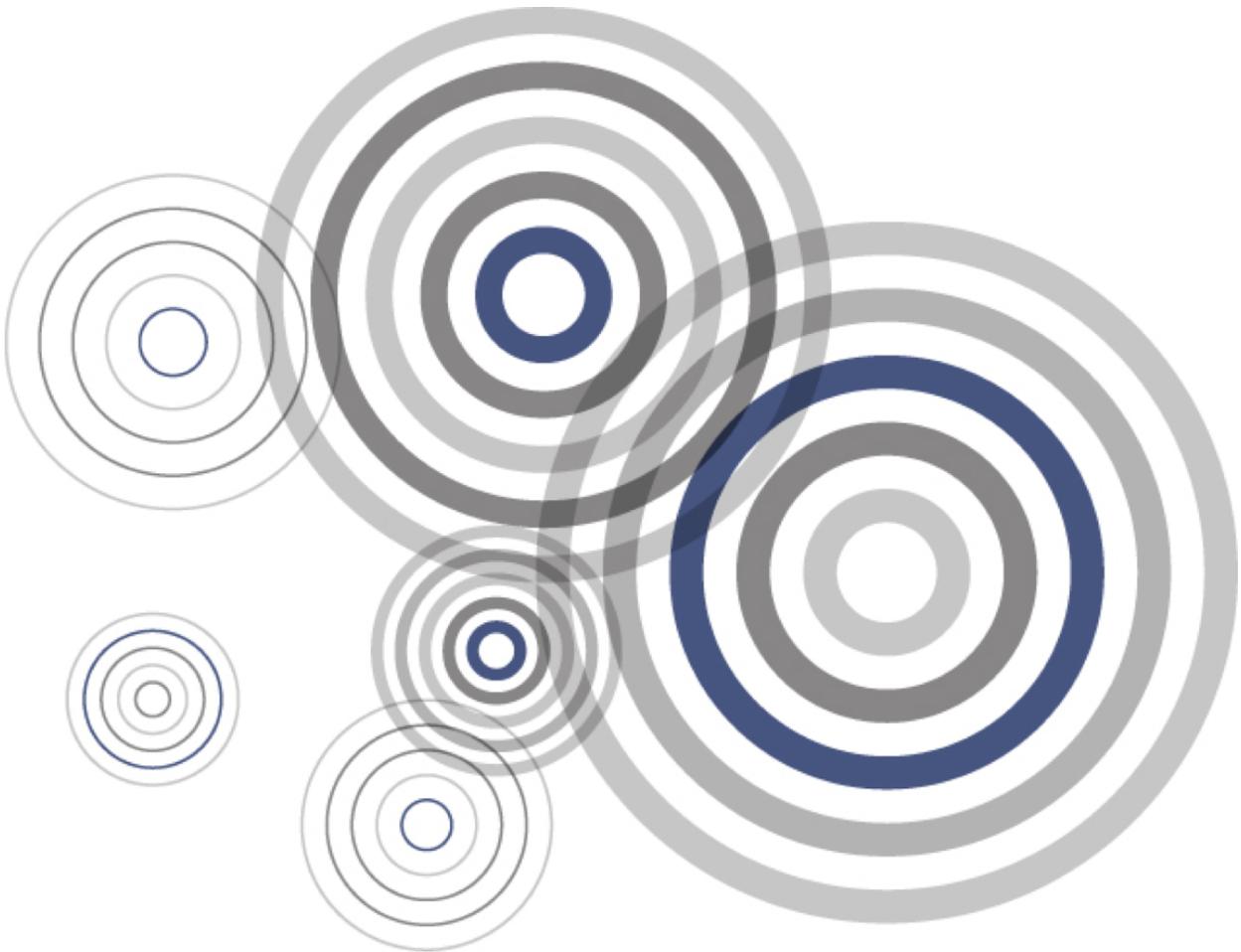


Public

# Relationship Record Common Data File format V1.1

Final V1.1 2017-03-16



## Contents

1. Introduction .....	5
1.1. Status of this document .....	5
1.2. Terminology and Typographical Conventions.....	5
1.3. Cardinalities .....	6
1.4. Business Rules.....	6
1.5. XML Syntax.....	6
1.5.1. XML Design Rules.....	7
1.5.2. XML Schema .....	7
1.6. Release Notes .....	15
1.6.1. Version 1.1 .....	15
1.6.2. Version 1.0.....	16
1.7. Change Management.....	16
1.7.1. Errata Version .....	16
1.7.2. Minor Version .....	16
1.7.3. Major Version .....	16
1.7.4. Minor Version Changes to the XML Schema .....	17
1.7.5. Major Version Changes to the XML Schema.....	18
2. Abstract Data Content.....	18
2.1. Relationship Data File Header.....	19
2.2. Relationship Record .....	20
2.2.1. Relationship .....	20
2.2.2. Registration .....	22
2.2.3. Extension .....	23
2.3. Data Element Reference .....	24
2.3.1. Element rr:RelationshipData .....	25
2.3.2. Element rr:RelationshipDataType / Header .....	26
2.3.3. Element rr:RRHeaderType / ContentDate .....	27
2.3.4. Element rr:RRHeaderType / Originator .....	28
2.3.5. Element rr:RRHeaderType / FileContent .....	29
2.3.6. Element rr:RRHeaderType / DeltaStart .....	31
2.3.7. Element rr:RRHeaderType / RecordCount.....	32

2.3.8. Element rr:RRHeaderType / NextVersion ..... 33

2.3.9. Element rr:RRHeaderType / Extension ..... 34

2.3.10. Element rr:RelationshipDataType / RelationshipRecords ..... 35

2.3.11. Element rr:RelationshipRecordsType / RelationshipRecord..... 36

2.3.12. Element rr:RelationshipRecordType / Relationship ..... 37

2.3.13. Element rr:RelationshipContainerType / StartNode..... 38

2.3.14. Element rr:NodeType / NodeID ..... 39

2.3.15. Element rr:NodeType / NodeIDType ..... 40

2.3.16. Element rr:RelationshipContainerType / EndNode ..... 41

2.3.17. Element rr:RelationshipContainerType / RelationshipType ..... 42

2.3.18. Element rr:RelationshipContainerType / RelationshipPeriods ..... 43

2.3.19. Element rr:RelationshipPeriodsType / RelationshipPeriod ..... 44

2.3.20. Element rr:RelationshipPeriodType / StartDate ..... 45

2.3.21. Element rr:RelationshipPeriodType / EndDate..... 46

2.3.22. Element rr:RelationshipPeriodType / PeriodType ..... 47

2.3.23. Element rr:RelationshipContainerType / RelationshipStatus ..... 48

2.3.24. Element rr:RelationshipContainerType / RelationshipQualifiers..... 49

2.3.25. Element rr:RelationshipQualifiersType / RelationshipQualifier ..... 50

2.3.26. Element rr:RelationshipQualifierType / QualifierDimension ..... 51

2.3.27. Element rr:RelationshipQualifierType / QualifierCategory..... 52

2.3.28. Element rr:RelationshipContainerType / RelationshipQuantifiers ..... 53

2.3.29. Element rr:RelationshipQuantifiersType / RelationshipQuantifier ..... 54

2.3.30. Element rr:RelationshipQuantifierType / MeasurementMethod..... 55

2.3.31. Element rr:RelationshipQuantifierType / QuantifierAmount ..... 56

2.3.32. Element rr:RelationshipQuantifierType / QuantifierUnits..... 57

2.3.33. Element rr:RelationshipContainerType / NextVersion ..... 58

2.3.34. Element rr:RelationshipContainerType / Extension ..... 59

2.3.35. Element rr:RelationshipRecordType / Registration ..... 60

2.3.36. Element rr:RegistrationContainerType / InitialRegistrationDate ..... 61

2.3.37. Element rr:RegistrationContainerType / LastUpdateDate ..... 62

2.3.38. Element rr:RegistrationContainerType / RegistrationStatus ..... 63

2.3.39. Element rr:RegistrationContainerType / NextRenewalDate ..... 65

2.3.40. Element rr:RegistrationContainerType / ManagingLOU..... 66

2.3.41. Element rr:RegistrationContainerType / ValidationSources..... 67

2.3.42. Element rr:RegistrationContainerType / ValidationDocuments..... 68

2.3.43. Element rr:RegistrationContainerType / ValidationReference..... 69

2.3.44. Element rr:RegistrationContainerType / NextVersion..... 70

2.3.45. Element rr:RegistrationContainerType / Extension..... 71

2.3.46. Element rr:RelationshipRecordType / NextVersion..... 72

2.3.47. Element rr:RelationshipRecordType / Extension..... 73

2.3.48. Simple Type rr:LEIDateTimeProfile ..... 74

2.3.49. Simple Type rr:LEIType ..... 75

2.3.50. Simple Type rr:FileContentEnum ..... 76

2.3.51. Simple Type rr:NodeIDTypeEnum..... 78

2.3.52. Simple Type rr:RelationshipTypeEnum..... 79

2.3.53. Simple Type rr:PeriodTypeEnum ..... 80

2.3.54. Simple Type rr:RelationshipStatusEnum ..... 81

2.3.55. Simple Type rr:QualifierDimensionEnum ..... 82

2.3.56. Simple Type rr:QualifierCategoryTypeEnum ..... 83

2.3.57. Simple Type rr:MeasurementMethodTypeEnum ..... 84

2.3.58. Simple Type rr:QuantifierUnitsTypeEnum ..... 85

2.3.59. Simple Type rr:RegistrationStatusEnum ..... 86

2.3.60. Simple Type rr:ValidationSourcesTypeEnum..... 88

2.3.61. Simple Type rr:ValidationDocumentsTypeEnum ..... 89

2.3.62. Simple Type rr:Tokenized500Type ..... 90

## 1. Introduction

Documentation updated: 2017-03-16

This XML schema defines a reporting format for relationship records for Global Legal Entity Identifier System (GLEIS) Local Operating Units (LOUs) to report relationships between two legal entities (one relationship per relationship record).

Types of relationship supported:

- LEI to LEI relationships.
- LEI to (GLEIS-internal) Provisional Node Identifier (PNI) for reporting parent entities which do not yet have an LEI.

### 1.1. Status of this document

This section describes the status of this document at the time of its publication. Later versions may supersede this document. The most up to date version will always be available from [www.gleif.org](http://www.gleif.org).

The file format references and honors the policy document published by the LEI ROC entitled "Collecting data on direct and ultimate parents of legal entities in the Global LEI System – Phase 1" (10 March 2016; available from [www.leiroc.org](http://www.leiroc.org)).

### 1.2. Terminology and Typographical Conventions

The following typographical conventions are used throughout the document:

- ALL CAPS type is used for the special terms enumerated above.
- `Monospace type` is used to denote programming language, UML, and XML identifiers, as well as for the text of XML documents.

### 1.3. Cardinalities

- The cardinality of each element (the number of times it MUST or MAY appear in an XML data file conforming to this schema) is expressed as a number range in the format {minimum occurrences, maximum occurrences} in the XML examples shown below the notes of its containing element. This notation is equivalent to the following explanations in words:
  - Mandatory, unique: {1, 1} - the element MUST appear, exactly once.
  - Mandatory, repeatable: {1, unbounded} - the element MUST appear at least once. It may be repeated any number of times.
  - Optional, unique: {0, 1} - the element NEED NOT appear; it MAY appear once at most.
  - Optional, repeatable: {0, unbounded} - the element NEED NOT appear. It MAY be repeated any number of times.

Please note:

- The default cardinality is {1,1} (mandatory, unique). This document highlights when an element differs from this either by its `minOccurs` (minimum occurrences) or `maxOccurs` (maximum occurrences) value, or both.
- XML cardinalities apply in the context of any containing elements. This means that a contained element may have a cardinality of one or more even if its containing element may be omitted, because the contained element is mandatory **given** the presence of the container.
- XML cardinalities enforce a minimum data quality and standards conformance. Other business rules (as explained below) and data quality checks applied by GLEIF may encourage stricter cardinalities in live implementations.

### 1.4. Business Rules

The accompanying documentation in addition to this Technical Specification specifies business rules where applicable for each element. These are rules that are not enforced by validating against the XML schema, but are still mandatory for all Common Data File (CDF) format files.

### 1.5. XML Syntax

This section specifies the XML schema for an LEI data file conforming to this standard.

### 1.5.1. XML Design Rules

- The XSD schema conforms to W3C's XML Schema specification, version 1.0.
- The XML namespace is "http://www.gleif.org/data/schema/rr/2016".
- All interior XML elements are namespace-qualified (element form = qualified).
- All XML attributes are in the null namespace (attribute form = unqualified), with the exception of `xml:lang`.
- Element names are upper camel case.
- Attribute name are lower camel case.
- XSD type names are upper camel case.
- Enumeration code list values are all caps with underscores.
- Elements are used in preference to attributes except for language and type qualifiers.
- For a data element specified as having unbounded cardinality, the XML includes a single container element whose subelements are one or more instances of the data element whose cardinality is unbounded. The name of the container element is formed as the plural of the name of the contained elements.

### 1.5.2. XML Schema

An XML file conforming to this standard is valid according to the following XSD 1.0 schema.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:rr="http://www.gleif.org/data/schema/rr/2016"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  targetNamespace="http://www.gleif.org/data/schema/rr/2016">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>
  <xs:element name="RelationshipData"
    type="rr:RelationshipDataType"/>
  <xs:complexType name="RelationshipDataType">
    <xs:sequence>
      <xs:element name="Header" type="rr:RRHeaderType"/>
```

```

        <xs:element name="RelationshipRecords"
type="rr:RelationshipRecordsType"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="RRHeaderType">
    <xs:sequence>
        <xs:element name="ContentDate"
type="rr:LEIDateTimeProfile"/>
        <xs:element name="Originator" type="rr:LEIType"
minOccurs="0"/>
        <xs:element name="FileContent"
type="rr:FileContentEnum"/>
        <xs:element name="DeltaStart"
type="rr:LEIDateTimeProfile" minOccurs="0"/>
        <xs:element name="RecordCount"
type="xs:nonNegativeInteger"/>
        <xs:element name="NextVersion"
type="rr:HeaderNextVersionType" minOccurs="0"/>
        <xs:element name="Extension" type="rr:ExtensionType"
minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="LEIType">
    <xs:restriction base="xs:string">
        <xs:minLength value="20"/>
        <xs:maxLength value="20"/>
        <xs:pattern value="([0-9A-Z]{18}[0-9]{2})"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Tokenized500Type">
    <xs:restriction base="xs:string">
        <xs:maxLength value="500"/>
        <xs:minLength value="1"/>
        <xs:pattern value="\S+(\ \S+)*"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="FileContentEnum">
    <xs:restriction base="xs:string">
        <xs:enumeration value="LOU_FULL_INTERNAL"/>

```

```

        <xs:enumeration value="LOU_DELTA_INTERNAL"/>
        <xs:enumeration value="LOU_FULL_PUBLISHED"/>
        <xs:enumeration value="LOU_DELTA_PUBLISHED"/>
        <xs:enumeration value="GLEIF_FULL_INTERNAL"/>
        <xs:enumeration value="GLEIF_DELTA_INTERNAL"/>
        <xs:enumeration value="GLEIF_FULL_PUBLISHED"/>
        <xs:enumeration value="GLEIF_DELTA_PUBLISHED"/>
        <xs:enumeration value="QUERY_RESPONSE"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="ExtensionType">
    <xs:sequence>
        <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax" namespace="##other"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="RelationshipRecordsType">
    <xs:sequence>
        <xs:element name="RelationshipRecord"
type="rr:RelationshipRecordType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="RelationshipRecordType">
    <xs:sequence>
        <xs:element name="Relationship"
type="rr:RelationshipContainerType"/>
        <xs:element name="Registration"
type="rr:RegistrationContainerType"/>
        <xs:element name="NextVersion"
type="rr:RelationshipRecordNextVersionType" minOccurs="0"/>
        <xs:element name="Extension" type="rr:ExtensionType"
minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

    <xs:complexType name="RelationshipContainerType">
        <xs:sequence>

```

```

        <xs:element name="StartNode" type="rr:NodeType"/>
        <xs:element name="EndNode" type="rr:NodeType"/>
        <xs:element name="RelationshipType"
type="rr:RelationshipTypeEnum"/>
        <xs:element name="RelationshipPeriods"
type="rr:RelationshipPeriodsType" minOccurs="0"/>
        <xs:element name="RelationshipStatus"
type="rr:RelationshipStatusEnum"/>
        <xs:element name="RelationshipQualifiers"
type="rr:RelationshipQualifiersType"
            minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="RelationshipQuantifiers"
type="rr:RelationshipQuantifiersType"
            minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="NextVersion"
type="rr:RelationshipNextVersionType" minOccurs="0"/>
        <xs:element name="Extension" type="rr:ExtensionType"
minOccurs="0"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="NodeType">
    <xs:sequence>
        <xs:element name="NodeID" type="rr:LEIType"/>
        <xs:element name="NodeIDType" type="rr:NodeIDTypeEnum"/>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="NodeIDTypeEnum">
    <xs:restriction base="xs:string">
        <xs:enumeration value="LEI"/>
        <xs:enumeration value="ISO_17442_COMPATIBLE"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="RelationshipTypeEnum">
    <xs:restriction base="xs:string">
        <xs:enumeration value="IS_DIRECTLY_CONSOLIDATED_BY"/>
        <xs:enumeration value="IS_ULTIMATELY_CONSOLIDATED_BY"/>
        <xs:enumeration value="IS_INTERNATIONAL_BRANCH_OF"/>
    </xs:restriction>
</xs:simpleType>

```

```
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="RelationshipPeriodsType">
        <xs:sequence>
            <xs:element name="RelationshipPeriod"
type="rr:RelationshipPeriodType"
                maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="RelationshipPeriodType">
        <xs:sequence>
            <xs:element name="StartDate" type="rr:LEIDateTimeProfile"
minOccurs="0"/>
            <xs:element name="EndDate" type="rr:LEIDateTimeProfile"
minOccurs="0"/>
            <xs:element name="PeriodType" type="rr:PeriodTypeEnum"/>
        </xs:sequence>
    </xs:complexType>
    <xs:simpleType name="PeriodTypeEnum">
        <xs:restriction base="xs:string">
            <xs:enumeration value="ACCOUNTING_PERIOD"/>
            <xs:enumeration value="RELATIONSHIP_PERIOD"/>
            <xs:enumeration value="DOCUMENT_FILING_PERIOD"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RelationshipStatusEnum">
        <xs:restriction base="xs:string">
            <xs:enumeration value="ACTIVE"/>
            <xs:enumeration value="INACTIVE"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="RelationshipQualifiersType">
        <xs:sequence>
            <xs:element name="RelationshipQualifier"
type="rr:RelationshipQualifierType"
                minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

```
</xs:complexType>
<xs:complexType name="RelationshipQualifierType">
  <xs:sequence>
    <xs:element name="QualifierDimension"
type="rr:QualifierDimensionEnum"/>
    <xs:element name="QualifierCategory"
type="rr:QualifierCategoryTypeEnum" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="QualifierDimensionEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ACCOUNTING_STANDARD"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="QualifierCategoryTypeEnum">
  <xs:restriction base="xs:string">
    <xs:enumeration value="US_GAAP"/>
    <xs:enumeration value="IFRS"/>
    <xs:enumeration value="OTHER_ACCOUNTING_STANDARD"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="RelationshipQuantifiersType">
  <xs:sequence>
    <xs:element name="RelationshipQuantifier"
type="rr:RelationshipQuantifierType"
minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="RelationshipQuantifierType">
  <xs:sequence>
    <xs:element name="MeasurementMethod"
type="rr:MeasurementMethodTypeEnum"/>
    <xs:element name="QuantifierAmount" type="xs:decimal"/>
    <xs:element name="QuantifierUnits"
type="rr:QuantifierUnitsTypeEnum" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="MeasurementMethodTypeEnum">
```

```

        <xs:restriction base="xs:string">
            <xs:enumeration value="ACCOUNTING_CONSOLIDATION"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="QuantifierUnitsTypeEnum">
        <xs:restriction base="xs:string">
            <xs:enumeration value="PERCENTAGE"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="RegistrationContainerType">
        <xs:sequence>
            <xs:element name="InitialRegistrationDate"
type="rr:LEIDateTimeProfile"/>
            <xs:element name="LastUpdateDate"
type="rr:LEIDateTimeProfile"/>
            <xs:element name="RegistrationStatus"
type="rr:RegistrationStatusEnum"/>
            <xs:element name="NextRenewalDate"
type="rr:LEIDateTimeProfile" minOccurs="0"/>
            <xs:element name="ManagingLOU" type="rr:LEIType"/>
            <xs:element name="ValidationSources"
type="rr:ValidationSourcesTypeEnum"/>
            <xs:element name="ValidationDocuments"
type="rr:ValidationDocumentsTypeEnum"/>
            <xs:element name="ValidationReference"
type="rr:Tokenized500Type" minOccurs="0"/>
            <xs:element name="NextVersion"
type="rr:RegistrationNextVersionType" minOccurs="0"/>
            <xs:element name="Extension" type="rr:ExtensionType"
minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:simpleType name="RegistrationStatusEnum">
        <xs:restriction base="xs:string">
            <xs:enumeration value="PENDING_VALIDATION"/>
            <xs:enumeration value="PUBLISHED"/>
            <xs:enumeration value="DUPLICATE"/>
            <xs:enumeration value="LAPSED"/>
            <xs:enumeration value="RETIRED"/>
        </xs:restriction>
    </xs:simpleType>

```

```

        <xs:enumeration value="ANNULLED"/>
        <xs:enumeration value="TRANSFERRED"/>
        <xs:enumeration value="PENDING_TRANSFER"/>
        <xs:enumeration value="PENDING_ARCHIVAL"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ValidationDocumentsTypeEnum">
    <xs:restriction base="xs:string">
        <xs:enumeration value="ACCOUNTS_FILING"/>
        <xs:enumeration value="REGULATORY_FILING"/>
        <xs:enumeration value="SUPPORTING_DOCUMENTS"/>
        <xs:enumeration value="CONTRACTS"/>
        <xs:enumeration value="OTHER_OFFICIAL_DOCUMENTS"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ValidationSourcesTypeEnum">
    <xs:restriction base="xs:string">
        <xs:enumeration value="PENDING"/>
        <xs:enumeration value="ENTITY_SUPPLIED_ONLY"/>
        <xs:enumeration value="PARTIALLY_CORROBORATED"/>
        <xs:enumeration value="FULLY_CORROBORATED"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="LEIDateTimeProfile">
    <xs:restriction base="xs:dateTime">
        <xs:pattern
value="([^\.\.]*|([^\.\.]*\.\.((\d){1,3}))\{0,1\})\{0,1\}(Z|\+([01][0-9]|2[0-3])\:([0-5][0-9])|-\:([01][0-9]|2[0-3])\:([0-5][0-9]))"
        />
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="HeaderNextVersionType">
    <xs:sequence>
        <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
        namespace="##targetNamespace"/>

```

```
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="RelationshipRecordNextVersionType">
        <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
                namespace="##targetNamespace"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="RelationshipNextVersionType">
        <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
                namespace="##targetNamespace"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="RegistrationNextVersionType">
        <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
                namespace="##targetNamespace"/>
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

## 1.6. Release Notes

### 1.6.1. Version 1.1

#### Corrections:

- Corrected `elementFormDefault="unqualified"` to `elementFormDefault="qualified"`.

### **1.6.2. Version 1.0**

The first release.

## **1.7. Change Management**

Changes to this standard that affect the data schema SHALL be made by approval and publication of a new version of this document. A new version SHALL be one of the following:

### **1.7.1. Errata Version**

An errata version makes corrections to the normative content of the standard (excluding corrections which would change the data schema) and/or makes changes to non-normative content such as explanatory material. An errata version does not change the XML schema definitions, only the documentation parts, and so does not affect the interoperability of systems implementing the standard. An errata version is indicated by incrementing the third version number; e.g., 1.0 to 1.0.1, or 1.0.1 to 1.0.2.

### **1.7.2. Minor Version**

A minor version may include all changes permitted in an errata version, and in addition adds one or more data elements and/or adds one or more codes to a code list (“enum” data type). A minor version changes the XML schema. Minor version changes to schema MUST provide for forward and backward compatibility. This allows existing implementations to continue to interoperate even if they are using different minor versions. A minor version is indicated by incrementing the second version number; e.g., 1.0 to 1.1 or 1.1.3 to 1.2.

### **1.7.3. Major Version**

A major version may make any change at all, including incompatible changes to the XML schema. Major version changes to schema require that the new version uses a different XML namespace. This requires existing implementations to separately understand both the old and new versions during a period of transition. A major version is indicated by incrementing the first version number; e.g., 1.1 to 2.0.

The release of a new minor or major version shall always be accompanied by a transition plan for LOUs and GLEIF, to ensure a smooth and time-bounded migration to the new

version.

#### 1.7.4. Minor Version Changes to the XML Schema

A minor version may introduce new XML elements and/or adds one or more codes to a code list ("enum" data type). Minor version changes to schema SHALL be made as specified below, in order to achieve forward and backward compatibility.

Forward compatibility means that an LEI Data File which is valid according to the older version's schema is also valid according to the newer version's schema.

Backward compatibility means that an LEI Data File which is valid according to the newer version's schema is also valid according to the older version's schema.

New data elements may be added at pre-defined extension points within the schema, each with an optional XML element `NextVersion`. New data elements are always added within a `NextVersion` element. When a minor version adds a new data element to a `NextVersion` element, a new `NextVersion` element is also added inside the previously added `NextVersion` element, to accommodate additional data elements in subsequent minor versions. Each successive `NextVersion` element set is contained directly within the previous minor version's `NextVersion` set.

As can be seen from the full XML schema presented here, the following rules SHALL be observed to ensure forward and backward compatibility:

- The initial XSD declaration for a `NextVersion` element SHALL use the element name "NextVersion", XML data type "lei:NextVersion1Type" and cardinality optional, unique {0,1}. The XML data type allows a sequence of any elements, each of cardinality optional, repeatable (unbounded) and with lax content processing, but in the target namespace.
- The `minOccurs` declaration on the `NextVersion` element allows it to be omitted in files conforming to the first minor version. The schema wildcard `xsd:any` allows for forward compatibility: a file conforming to a new minor version still validates in the old version because the wildcard matches any new elements introduced in the new minor version.
- New elements SHALL be introduced in a subsequent minor version by modifying the declaration for the above type declaration as follows:
  - A sequence of the new elements introduced in the previous version.
  - A subsequent `NextVersionN` element where `N` is an index number starting at 1 and incremented by 1 with each minor version.

- Each new element SHALL be declared `minOccurs="0"`, to ensure backward compatibility: a file conforming to the old version still validates in the new version because the new schema does not require the presence of elements not defined in the old version. If a new element is mandatory for conformance to the new version, this MUST be enforced outside schema validation.
- The new definition of the `NextVersion` element SHALL include a declaration of an inner `NextVersion` element, as illustrated above, to provide for additional elements in subsequent minor versions. The nesting of `NextVersion` elements is required to satisfy the “unique particle attribution” constraint of XSD 1.0.
- Each code list (Enum types) is implemented in the XML schema simply as the XSD string data type. This provides for forward compatibility because the schema for an older minor version will validate any string, including codes defined in newer minor versions. The schema for each minor version includes the list of valid codes for that minor version as a documentation annotation to the type declaration for each Enum type.

### 1.7.5. Major Version Changes to the XML Schema

A major version may make any change to the XML schema whatsoever, including incompatible changes.

A schema introduced in a new major version SHALL use an XML namespace URI that is different from the XML namespace URI defined in any other major version of this standard. The namespace URI for a new major version SHOULD be the same as the namespace URI specified in this standard, with the year at the end changed to the year in which the new major version is introduced. If more than one major version is introduced in the same year, a letter “a”, “b”, “c”, etc., may be appended to the year as needed.

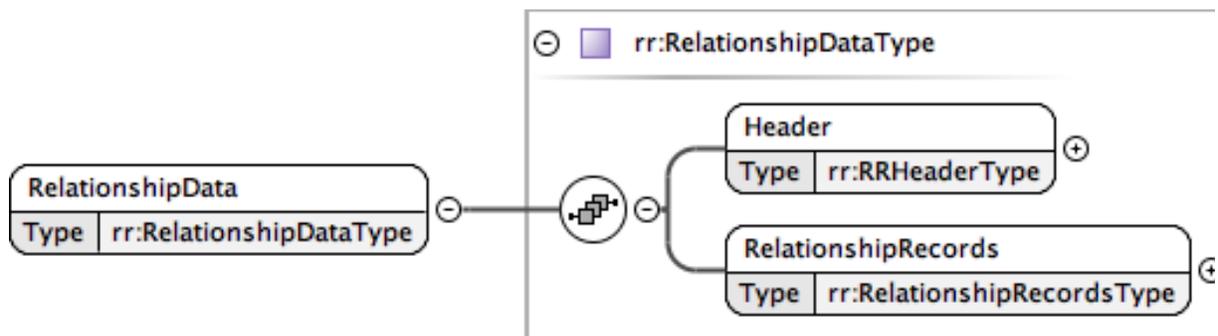
A new major version MUST be accompanied by an implementation plan which explains how implementations will make the transition from the old major version to the new major version. Generally speaking, such a plan typically provides for a period of transition in which an implementation capable of receiving the new major version is required to also receive the old major version.

## 2. Abstract Data Content

This section specifies the abstract data content of a data file conforming to this standard. A data file conforming to this standard SHALL consist of:

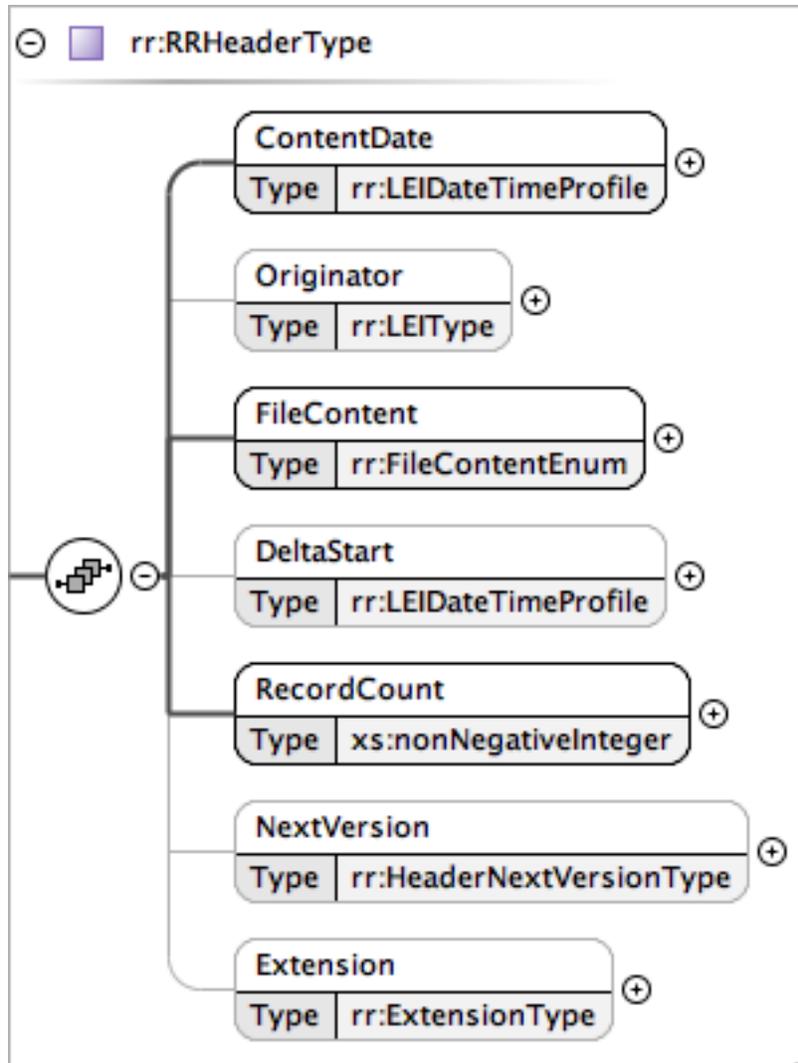
- A Header.

- Zero or more Relationship Records.



## 2.1. Relationship Data File Header

The Relationship Data File Header describes the context for the Relationship Records contained in the main body of the file. The header exists to answer such questions as where the data came from, when it was collected into this file, etc. The content of the header SHALL NOT be required to interpret the data content of any Relationship Record; each Relationship Record is self-contained.

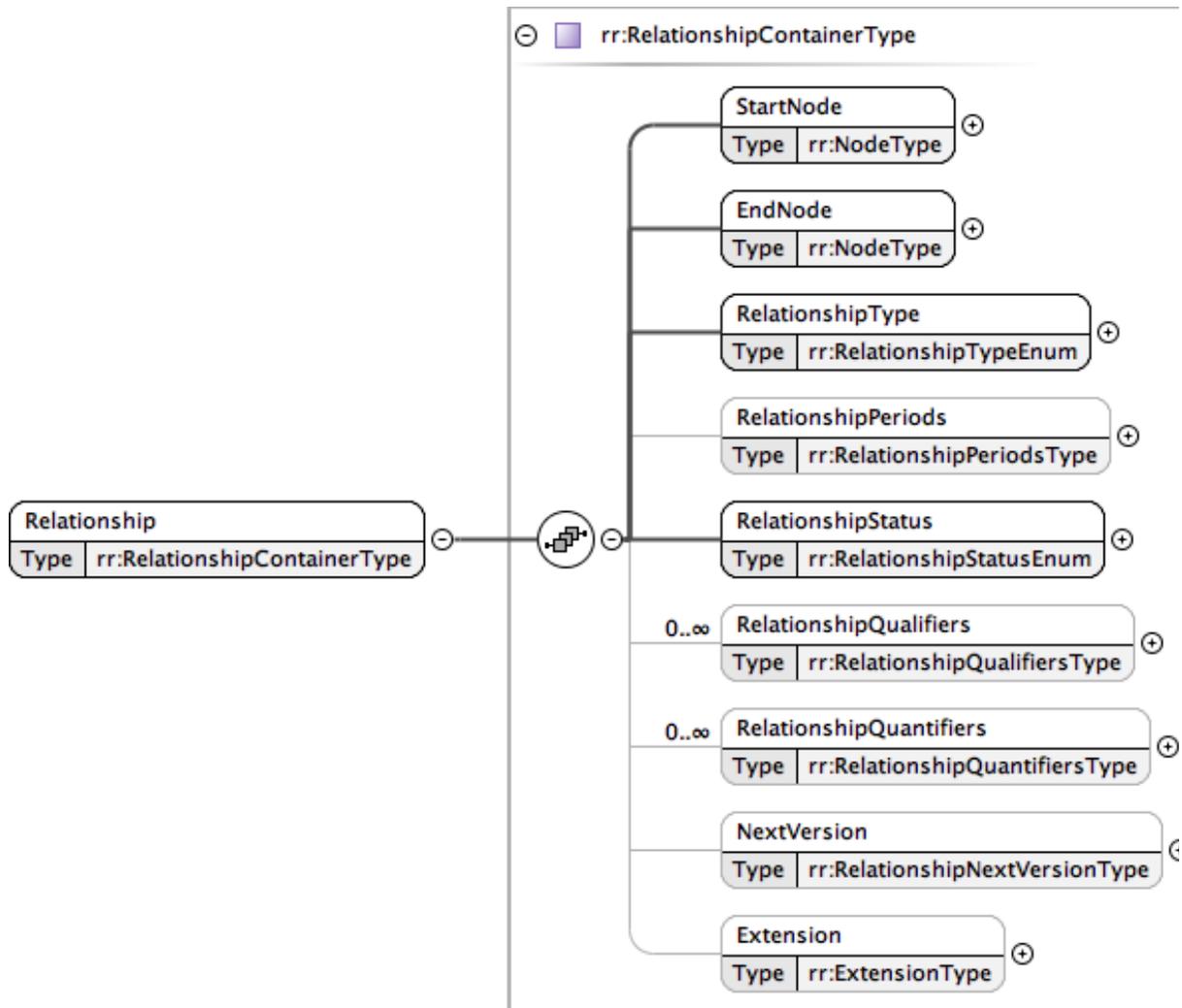


## 2.2. Relationship Record

An Relationship Record describes a single LEI registration. Each Relationship Record in a file conforming to this standard SHALL include data elements as described below:

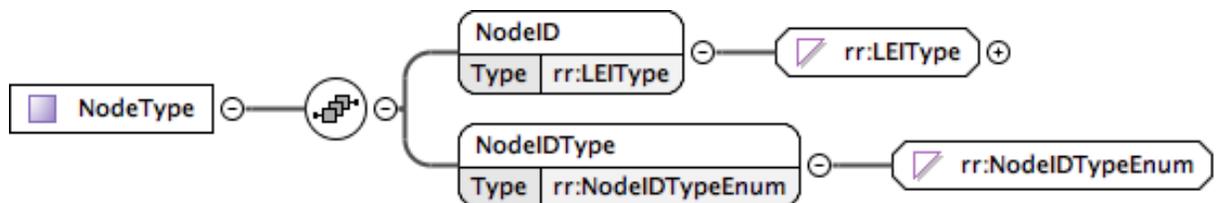
### 2.2.1. Relationship

The Relationship container element identifies the two related entities and their relationship.



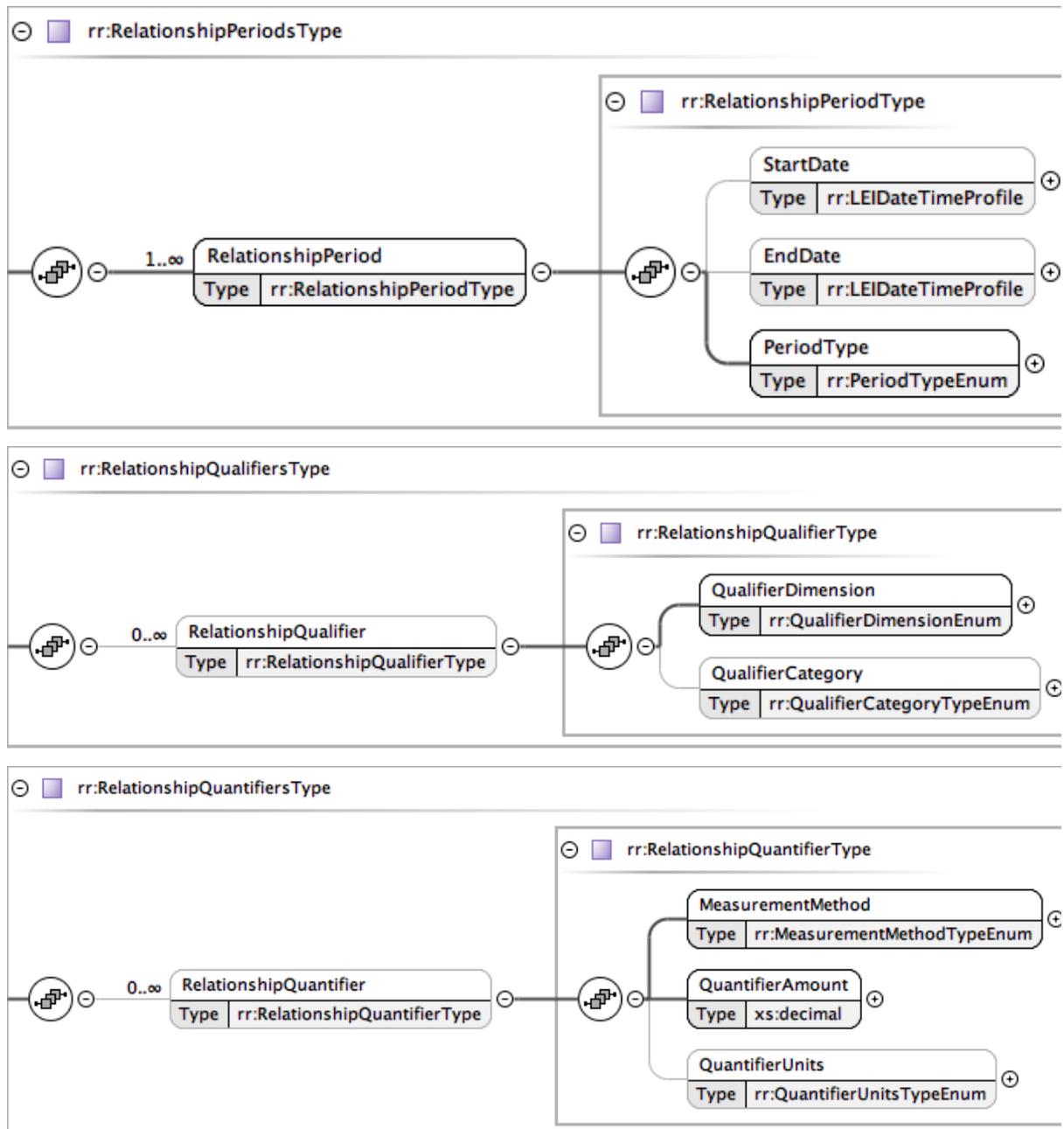
### 2.2.1.1. Related Entities

The ISO 17442-compliant identifiers of the legal entities related by this Relationship Record.



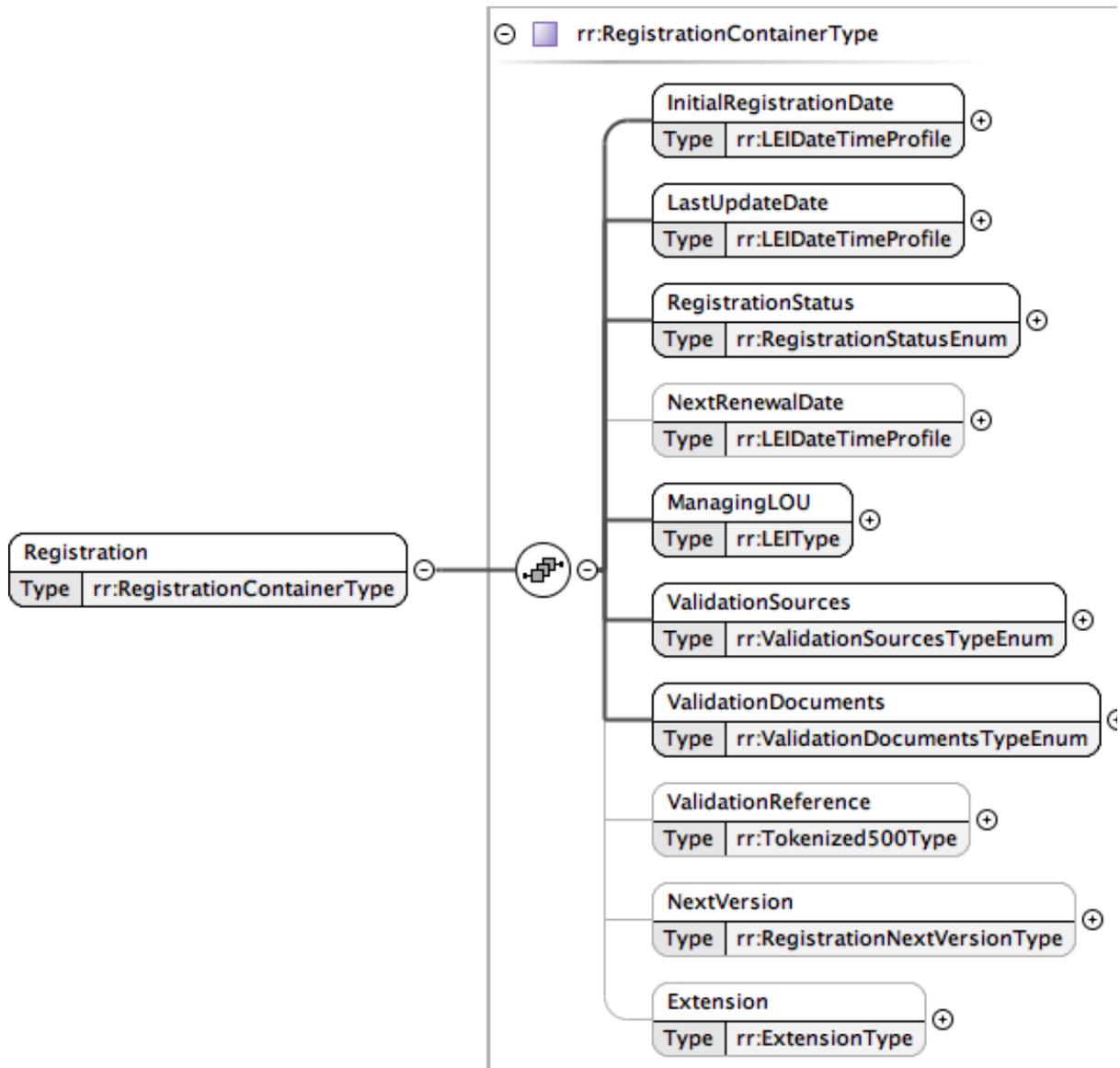
### 2.2.1.2. Relationship Attributes

Attributes describing the dates, type, qualitative and quantitative aspects of the relationship itself, as required. The data is supplied by the legal entity, and recorded and published by the LOU.



### 2.2.2. Registration

Attributes describing the registration of this relationship information with an LOU. The `Registration` data is maintained by the LOU.



### 2.2.3. Extension

The optional `Extension` section of a Relationship Record may be used to include additional data not defined in this standard. This may include data specific to an LOU, data specific to a publisher of LEI data, and so on.

For example, an LOU may use `Extension` to publish additional data elements it collects as part of registration.

The following rules MUST be observed:

- Each XML element included in the content of the `Extension` element SHALL be in an XML namespace that is not null and not equal to the XML namespace of the LEI Data File as specified in this standard.

- The XML namespace for an `Extension` element SHALL be a namespace which the creator of the extension element exclusively or jointly controls, or from which the creator re-uses existing elements and their definitions, e.g. a namespace derived from the Internet Domain Name of the creator, a namespace agreed upon by a group of trading partners, etc.
- An `Extension` element SHALL NOT be defined in such a way as to require the recipient of the file to recognize the `Extension` element in order to interpret the data elements specified in this standard. A recipient of the file MUST be able to ignore all `Extension` elements and still interpret the standard content correctly.
- A recipient of a data file conforming to this standard SHALL NOT reject a file solely because it contains extensions not understood by the recipient. A recipient MUST be prepared to accept a file containing extensions and ignore any it does not understand, provided that the file complies to this standard.

### 2.3. Data Element Reference

The following pages describe all individual data elements, attributes and datatypes, one per page.

### 2.3.1. Element rr:RelationshipData

Contains the file structure for the whole relationship records file as specified in the XML datatypes below.

#### **rr:RelationshipDataType**

content **complex**

```
<rr:RelationshipData xmlns:rr="http://www.gleif.org/data/schema/rr/2016">  
  <Header>{1,1}</Header>  
  <RelationshipRecords>{1,1}</RelationshipRecords>  
</rr:RelationshipData>
```

### 2.3.2. Element rr:RelationshipDataType / Header

Contains the file upload information for this RelationshipData file.

#### **rr:RRHeaderType**

content **complex**

```
<Header>  
  <ContentDate>{1,1}</ContentDate>  
  <Originator>{0,1}</Originator>  
  <FileContent>{1,1}</FileContent>  
  <DeltaStart>{0,1}</DeltaStart>  
  <RecordCount>{1,1}</RecordCount>  
  <NextVersion>{0,1}</NextVersion>  
  <Extension>{0,1}</Extension>  
</Header>
```

### 2.3.3. Element rr:RRHeaderType / ContentDate

The date and time as of which the data contained in the file is valid.

**rr:LEIDateTimeProfile**

content **simple**

pattern

```
((^\.]*|(^\.]*(\.(\d){1,3})))(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-([01][0-9]|2[0-3]):([0-5][0-9]))
```

### 2.3.4. Element rr:RRHeaderType / Originator

The LEI of the entity that created the content of this file.

#### rr:LEIType

content **simple**

minOccurs **0**

minLength **20**

maxLength **20**

pattern **{[0-9A-Z]{18}[0-9]{2}}**

### 2.3.5. Element rr:RRHeaderType / FileContent

A code describing the content of this relationship record file.

#### rr:FileContentEnum

content **simple**

enumeration <b>LOU_FULL_INTERNAL</b>	The file contains all relationship records created for internal use by an LOU (all internal relationship records for which the LOU is the <i>ManagingLOU</i> ) as of the date/time the file is created.
enumeration <b>LOU_DELTA_INTERNAL</b>	The file contains those relationship records created by an LOU for internal use (all internal relationship records for which the LOU is the <i>ManagingLOU</i> ) which are new or changed since the <i>DeltaStart</i> specified in the <i>Header</i> , as of the date/time the file is created.
enumeration <b>LOU_FULL_PUBLISHED</b>	The file contains all relationship records published by an LOU (all relationship records for which the LOU is the <i>ManagingLOU</i> ) as of the date/time the file is created.
enumeration <b>LOU_DELTA_PUBLISHED</b>	The file contains those relationship records published by an LOU (all relationship records for which the LOU is the <i>ManagingLOU</i> ) which are new or changed since the <i>DeltaStart</i> specified in the <i>Header</i> , as of the date/time the file is created.
enumeration <b>GLEIF_FULL_INTERNAL</b>	The file contains all relationship records GLEIF manages internally to the GLEIS (including all internal relationship records from all LOUs) as of the date/time the file is created.
enumeration <b>GLEIF_DELTA_INTERNAL</b>	The file contains those relationship records GLEIF manages internally to the GLEIS (including all

	internal relationship records from all LOUs) which are new or changed since the <code>DeltaStart</code> date specified in the <code>Header</code> , as of the date/time the file is created.
enumeration <b>GLEIF_FULL_PUBLISHED</b>	The file contains all relationship records published by GLEIF (including all relationship records from all LOUs) as of the date/time the file is created.
enumeration <b>GLEIF_DELTA_PUBLISHED</b>	The file contains those relationship records published by GLEIF (including all relationship records from all LOUs) which are new or changed since the <code>DeltaStart</code> date specified in the <code>Header</code> , as of the date/time the file is created.
enumeration <b>QUERY_RESPONSE</b>	The file contains records matching criteria specified in a query.

### 2.3.6. Element rr:RRHeaderType / DeltaStart

The date and time of the baseline relative to which this file contains new or changed Relationship Records.

#### rr:LEIDateTimeProfile

content **simple**

minOccurs **0**

pattern `((^[^\.]*|([^\.\.]*\.(\\d){1,3}))){0,1})(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-[01][0-9]|2[0-3]):([0-5][0-9]))`

### 2.3.7. Element rr:RRHeaderType / RecordCount

The number of relationship records in the file.

**xs:nonNegativeInteger**

content **simple**

### 2.3.8. Element `rr:RRHeaderType / NextVersion`

A structure for adding further elements in to the LEI data file header in anticipation of a new version, by nesting a series of XML elements with this content model within the `NextVersion` element, one for each new minor version of the schema, postpending a serial number (1,2,3...) to the element name upon each iteration.

#### `rr:HeaderNextVersionType`

content **complex**

minOccurs **0**

### 2.3.9. Element rr:RRHeaderType / Extension

This `Extension` element may contain any additional elements required to extend the Header container element.

#### **rr:ExtensionType**

content **complex**

minOccurs **0**

### 2.3.10. Element `rr:RelationshipDataType` / `RelationshipRecords`

Container for all of the `RelationshipRecord` container elements submitted with this file.

#### **`rr:RelationshipRecordsType`**

content **complex**

```
<RelationshipRecords>  
  <RelationshipRecord>{0,unbounded}</RelationshipRecord>  
</RelationshipRecords>
```

### 2.3.11. Element `rr:RelationshipRecordsType` / `RelationshipRecord`

Contains all relationship information including identifiers referring to the related entities, the specific type and other attributes of the relationship itself, and details of the relationship's registration with the `ManagingLOU`.

#### `rr:RelationshipRecordType`

content **complex**

minOccurs **0**

maxOccurs **unbounded**

```
<RelationshipRecord>
  <Relationship>{1,1}</Relationship>
  <Registration>{1,1}</Registration>
  <NextVersion>{0,1}</NextVersion>
  <Extension>{0,1}</Extension>
</RelationshipRecord>
```

### 2.3.12. Element `rr:RelationshipRecordType` / `Relationship`

The `Relationship` container element contains the identifiers of the two entities related by the reported relationship, as well as the type of relationship, dates related to the relationship and other relationship quantifiers and qualifiers.

#### **`rr:RelationshipContainerType`**

content **complex**

```
<Relationship>
  <StartNode>{1,1}</StartNode>
  <EndNode>{1,1}</EndNode>
  <RelationshipType>{1,1}</RelationshipType>
  <RelationshipPeriods>{0,1}</RelationshipPeriods>
  <RelationshipStatus>{1,1}</RelationshipStatus>
  <RelationshipQualifiers>{0,unbounded}</RelationshipQualifiers>
  <RelationshipQuantifiers>{0,unbounded}</RelationshipQuantifiers>
  <NextVersion>{0,1}</NextVersion>
  <Extension>{0,1}</Extension>
</Relationship>
```

### 2.3.13. Element `rr:RelationshipContainerType / StartNode`

An LEI or ISO 17442-compatible ID for the entity at the "start" of a directional relationship.

#### **rr:NodeType**

content **complex**

```
<StartNode>  
  <NodeID>{1,1}</NodeID>  
  <NodeIDType>{1,1}</NodeIDType>  
</StartNode>
```

### 2.3.14. Element rr:NodeType / NodeID

The identifier for the entity designated by this node.

#### **rr:LEIType**

content **simple**

minLength **20**

maxLength **20**

pattern **([0-9A-Z]{18}[0-9]{2})**

### 2.3.15. Element **rr:NodeType / NodeIDType**

The type of identifier used to designate this node's entity.

#### **rr:NodeIDTypeEnum**

content **simple**

- |   |   |
|---|---|
| enumeration <b>LEI</b>                  | An LEI code taken from the LEI issuing prefix namespace of a GLEIS LOU.                       |
| enumeration <b>ISO_17442_COMPATIBLE</b> | An ISO 17442-compatible code, not taken from the LEI issuing prefix namespace of a GLEIS LOU. |

### 2.3.16. Element `rr:RelationshipContainerType / EndNode`

An LEI or ISO 17442-compatible ID for the entity at the "end" of a directional relationship.

#### **rr:NodeType**

content **complex**

```
<EndNode>  
  <NodeID>{1,1}</NodeID>  
  <NodeIDType>{1,1}</NodeIDType>  
</EndNode>
```

### 2.3.17. Element rr:RelationshipContainerType / RelationshipType

A unique code designating the specific category of a directional relationship between two legal entities.

#### rr:RelationshipTypeEnum

content **simple**

enumeration **IS\_DIRECTLY\_CONSOLIDATED\_BY**

*StartNode* is directly consolidated by *EndNode*. The *StartNode* "child" entity has its accounts fully consolidated by the *EndNode* "parent" entity, in the sense given by the accounting standard(s) specified in *RelationshipQualifiers*; the *EndNode* entity is the closest fully consolidating parent to the *StartNode* entity in any applicable hierarchical ownership structure.

enumeration **IS\_ULTIMATELY\_CONSOLIDATED\_BY**

*StartNode* is ultimately consolidated by *EndNode*. The *StartNode* "child" entity has its accounts fully consolidated by the *EndNode* "parent" entity, in the sense given by the accounting standard(s) specified in *RelationshipQualifiers*; the *EndNode* entity is the most distant fully consolidating parent from the *StartNode* entity in any applicable hierarchical ownership structure.

enumeration **IS\_INTERNATIONAL\_BRANCH\_OF**

*StartNode* is an international branch of the legal entity designated by *EndNode* (in jurisdiction country of *StartNode*). The *EndNode* is the Head Office and MUST be an LEI.

### 2.3.18. Element `rr:RelationshipContainerType / RelationshipPeriods`

A collection of paired beginning and end dates relating to: the relationship itself, periods (e.g. accounting cycles) covered by documents demonstrating the relationship, or the filing date(s) of those documents.

#### **`rr:RelationshipPeriodsType`**

content **complex**

minOccurs **0**

```
<RelationshipPeriods>  
  <RelationshipPeriod>{1,unbounded}</RelationshipPeriod>  
</RelationshipPeriods>
```

### 2.3.19. Element rr:RelationshipPeriodType / RelationshipPeriod

Contains one set of start and end dates for a particular type of period, for example, the duration of the relationship itself, the filing or validity period of any documents demonstrating the relationship, or the accounting period they refer to.

#### rr:RelationshipPeriodType

content **complex**

maxOccurs **unbounded**

```
<RelationshipPeriod>
  <StartDate>{0,1}</StartDate>
  <EndDate>{0,1}</EndDate>
  <PeriodType>{1,1}</PeriodType>
</RelationshipPeriod>
```

### 2.3.20. Element rr:RelationshipPeriodType / StartDate

The start date for a particular period relevant to the relationship.

#### rr:LEIDateTimeProfile

content **simple**

minOccurs **0**

pattern

```
(([^\.]*|([^\.]*(\.(\\d){1,3}))){0,1})(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-[01][0-9]|2[0-3]):([0-5][0-9]))
```

### 2.3.21. Element rr:RelationshipPeriodType / EndDate

The end date for a particular period relevant to the relationship.

#### rr:LEIDateTimeProfile

content **simple**

minOccurs **0**

pattern

```
(([^\.]*|([^\.]*(\.(\\d){1,3}))?{0,1}})(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-[01][0-9]|2[0-3]):([0-5][0-9]))
```

### 2.3.22. Element `rr:RelationshipPeriodType / PeriodType`

The particular type of period, for example, the duration of the relationship itself, the filing or validity period of any documents demonstrating the relationship, or the accounting period they refer to.

#### `rr:PeriodTypeEnum`

content **simple**

enumeration **ACCOUNTING\_PERIOD**

The dates in this instance of `RelationshipPeriod` indicate the accounting period covered by the most recent validation documents for this relationship.

enumeration **RELATIONSHIP\_PERIOD**

The dates in this instance of `RelationshipPeriod` indicate the duration of validity of the relationship itself, as distinct from any administrative or reporting aspects.

enumeration **DOCUMENT\_FILING\_PERIOD**

The dates in this instance of `RelationshipPeriod` indicate the validity period of a regulatory filing, accounting document, or other document demonstrating the relationship's validity.

### 2.3.23. Element rr:RelationshipContainerType / RelationshipStatus

The status of the legal entities' relationship itself: `ACTIVE` or `INACTIVE`.

#### **rr:RelationshipStatusEnum**

content **simple**

enumeration **ACTIVE**

As of the last report or update, the reporting legal entity reported that it is legally registered and/or operating, AND that the relationship detailed in this `RelationshipRecord` is still valid.

enumeration **INACTIVE**

It has been determined that the relationship ended, e.g. because entity that reported this relationship is no longer legally registered and/or operating; or the relationship is no longer valid for other reasons.

### 2.3.24. Element `rr:RelationshipContainerType` / `RelationshipQualifiers`

Any additional qualitative attributes that help to categorize the relationship.

#### `rr:RelationshipQualifiersType`

content **complex**

minOccurs **0**

maxOccurs **unbounded**

```
<RelationshipQualifiers>  
  <RelationshipQualifier>{1,unbounded}</RelationshipQualifier>  
</RelationshipQualifiers>
```

### 2.3.25. Element `rr:RelationshipQualifiersType` / `RelationshipQualifier`

Container for all sets of relationship qualifier information.

#### **`rr:RelationshipQualifierType`**

content **complex**

minOccurs **1**

maxOccurs **unbounded**

```
<RelationshipQualifier>  
  <QualifierDimension>{1,1}</QualifierDimension>  
  <QualifierCategory>{0,1}</QualifierCategory>  
</RelationshipQualifier>
```

### 2.3.26. Element `rr:RelationshipQualifierType / QualifierDimension`

Designates the optional list of additional qualitative attributes that help to categorize the relationship.

**`rr:QualifierDimensionEnum`**

content **simple**

enumeration **ACCOUNTING\_STANDARD**

The accounting standard applied to determine the definition of e.g. ultimate or direct accounting consolidating parent for the relationship detailed in this `RelationshipRecord`. The relevant accounting standard is that applicable to the `EndNode` (the "parent" entity).

### 2.3.27. Element `rr:RelationshipQualifierType / QualifierCategory`

Specifies the additional qualitative attributes that help to categorize the relationship.

#### `rr:QualifierCategoryTypeEnum`

content **simple**

minOccurs **0**

enumeration **US\_GAAP**

United States-Generally Accepted Accounting Principles.

enumeration **IFRS**

International Financial Reporting Standard (developed by the International Accounting Standards Board – IASB see <http://www.ifrs.org>).

enumeration **OTHER\_ACCOUNTING\_STANDARD**

A financial reporting (accounting) standard not otherwise listed in the latest version of the relationship data file format.

### 2.3.28. Element `rr:RelationshipContainerType` / `RelationshipQuantifiers`

Any additional quantitative attributes that help to categorize the relationship.

#### `rr:RelationshipQuantifiersType`

content **complex**

minOccurs **0**

maxOccurs **unbounded**

```
<RelationshipQuantifiers>  
  <RelationshipQuantifier>{1,unbounded}</RelationshipQuantifier>  
</RelationshipQuantifiers>
```

### 2.3.29. Element `rr:RelationshipQuantifiersType` / `RelationshipQuantifier`

Specifies one additional quantitative attribute of the relationship, according to a particular measurement method.

#### `rr:RelationshipQuantifierType`

content **complex**

minOccurs **1**

maxOccurs **unbounded**

```
<RelationshipQuantifier>
  <MeasurementMethod>{1,1}</MeasurementMethod>
  <QuantifierAmount>{1,1}</QuantifierAmount>
  <QuantifierUnits>{0,1}</QuantifierUnits>
</RelationshipQuantifier>
```

### 2.3.30. Element rr:RelationshipQuantifierType / MeasurementMethod

Specifies the method of measurement (or set of rules) used to quantitatively categorize the relationship.

#### rr:MeasurementMethodTypeEnum

content **simple**

enumeration **ACCOUNTING\_CONSOLIDATION**  
n **N**

Accounting consolidation holds when "[in the] financial statements of a group [...] the assets, liabilities, equity, income, expenses and cash flows of the parent and its subsidiaries are presented as those of a single economic entity (please see <http://www.iasplus.com/en/standards/ias/ias-27-2011>).

### 2.3.31. Element `rr:RelationshipQuantifierType / QuantifierAmount`

Specifies the quantity measured as a decimal (positive or negative) number, using a . as the decimal point, with no spaces, and without thousand delimiters (e.g. , ).

**xs:decimal**

content **simple**

### 2.3.32. Element rr:RelationshipQuantifierType / QuantifierUnits

Specifies the units, where applicable, of a measurement made on a relationship.

**rr:QuantifierUnitsTypeEnum**

content **simple**

minOccurs **0**

enumeration **PERCENTAGE**

### 2.3.33. Element `rr:RelationshipContainerType / NextVersion`

A structure for adding further elements in to the `Registration` section of the Relationship Record in anticipation of a new version, by nesting a series of XML elements with this content model within the `NextVersion` element, one for each new minor version of the schema, postpending a serial number (1,2,3...) to the element name upon each iteration.

#### `rr:RelationshipNextVersionType`

content **complex**

minOccurs **0**

### 2.3.34. Element `rr:RelationshipContainerType` / Extension

This `Extension` element may contain any additional elements required to extend the `Relationship` container element.

#### `rr:ExtensionType`

content **complex**

minOccurs **0**

### 2.3.35. Element rr:RelationshipRecordType / Registration

The `Registration` container element contains information specifying the LOU's administration of the relationship report.

#### **rr:RegistrationContainerType**

content **complex**

```
<Registration>
  <InitialRegistrationDate>{1,1}</InitialRegistrationDate>
  <LastUpdateDate>{1,1}</LastUpdateDate>
  <RegistrationStatus>{1,1}</RegistrationStatus>
  <NextRenewalDate>{0,1}</NextRenewalDate>
  <ManagingLOU>{1,1}</ManagingLOU>
  <ValidationSources>{1,1}</ValidationSources>
  <ValidationDocuments>{1,1}</ValidationDocuments>
  <ValidationReference>{0,1}</ValidationReference>
  <NextVersion>{0,1}</NextVersion>
  <Extension>{0,1}</Extension>
</Registration>
```

### 2.3.36. Element rr:RegistrationContainerType / InitialRegistrationDate

The date at which the relationship information was first collected by the `ManagingLOU`.

**rr:LEIDateTimeProfile**

content **simple**

pattern

```
(([^\.]*|([^\.]*\.(\\d){1,3})))(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-([01][0-9]|2[0-3]):([0-5][0-9]))
```

### 2.3.37. Element rr:RegistrationContainerType / LastUpdateDate

The date at which the information was most recently updated by the `ManagingLOU`.

**rr:LEIDateTimeProfile**

content **simple**

pattern

```
(([^\.*]|([^\.*]*\.(\\d){1,3})))(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-([01][0-9]|2[0-3]):([0-5][0-9]))
```

### 2.3.38. Element `rr:RegistrationContainerType / RegistrationStatus`

The status of the legal entity's relationship record registration with the `ManagingLOU`.

#### `rr:RegistrationStatusEnum`

content **simple**

enumeration <b>PENDING_VALIDATION</b>	A relationship data report that has been submitted to the LOU and which is being processed and validated, prior to publication.
enumeration <b>PUBLISHED</b>	A relationship data report that has been validated and published, and which is reported by an entity that was an operating legal entity as of the last update.
enumeration <b>DUPLICATE</b>	A relationship data report that has been determined to be a duplicate registration of the same relationship. In many cases this will mean more than one report with e.g. the same 2 entity IDs, the same relationship type, certain status values and the same relationship date(s), but this determination will depend on the relationship type in question.
enumeration <b>LAPSED</b>	A relationship data report that has not been renewed by the <code>NextRenewalDate</code> .
enumeration <b>RETIRED</b>	The relationship is considered to have ended, but the relationship report is kept in publication for historical audit trail purposes.
enumeration <b>ANNULLED</b>	A relationship data report that was marked as erroneous or invalid after it was published. The relationship report is kept in publication for historical audit trail purposes only (so that data recipients can correct their local data).

enumeration <b>TRANSFERRED</b>	A relationship data report that has been transferred to a different LOU as the <i>ManagingLOU</i> . A record in this state is not published, but may be used internally by the prior LOU for audit trail purposes.
enumeration <b>PENDING_TRANSFER</b>	A relationship data report for which a transfer to another LOU has been requested. The request is being processed at the sending LOU. When the receiving LOU is ready, the status will be changed to <i>PENDING_ARCHIVAL</i> by the sending LOU prior to completion of the transfer.
enumeration <b>PENDING_ARCHIVAL</b>	This relationship data report is about to be transferred to a different LOU, after which its registration status will revert to a non-pending status. The <i>PENDING_ARCHIVAL</i> status serves to inform recipients of LOU-provided data files that a relationship record will be removed from that LOU's published file after the transfer is complete.

### 2.3.39. Element rr:RegistrationContainerType / NextRenewalDate

The next date by which the relationship information must be renewed and re-certified by the legal entity.

#### rr:LEIDateTimeProfile

content **simple**

minOccurs **0**

pattern `((^[^\.]*|([^[^\.]*\.(\\d){1,3}))){0,1})(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-[01][0-9]|2[0-3]):([0-5][0-9]))`

### 2.3.40. Element rr:RegistrationContainerType / ManagingLOU

The LEI of the LOU that is responsible for administering this relationship record.

#### rr:LEIType

content **simple**

minLength **20**

maxLength **20**

pattern **([0-9A-Z]{18}[0-9]{2})**

### 2.3.41. Element `rr:RegistrationContainerType / ValidationSources`

Level of relationship validation.

**rr:ValidationSourcesTypeEnum**

content **simple**

enumeration **PENDING**

The validation of the relationship data provided by the registrant has not yet occurred. Records with this `ValidationSources` value **MUST** not be published.

enumeration **ENTITY\_SUPPLIED\_ONLY**

Based on the validation procedures in use by the LOU responsible for the record, the information associated with this record has significant reliance on the information that a submitter provided due to the unavailability of corroborating information.

enumeration **PARTIALLY\_CORROBORATED**

Based on the validation procedures in use by the LOU responsible for the record, the information supplied by the submitter can be partially corroborated by supporting sources (e.g. financial statements with other definitions of the relevant relationship type; quarterly or annual regulatory filings, contracts and other documents used in preparing financial statements).

enumeration **FULLY\_CORROBORATED**

The relationship data provided by the registrant has been validated against an explicit relationship statement found in key sources (e.g. consolidated financial statements).

### 2.3.42. Element rr:RegistrationContainerType / ValidationDocuments

Type of source document(s) used for validating the relationship.

#### rr:ValidationDocumentsTypeEnum

content **simple**

enumeration <b>ACCOUNTS_FILING</b>	A consolidated financial (accounting) statement, prepared and submitted to the relevant authority.
enumeration <b>REGULATORY_FILING</b>	An annual regulatory filing providing public information on parent relationships.
enumeration <b>SUPPORTING_DOCUMENTS</b>	Other documents supporting the preparation of consolidated financial statements.
enumeration <b>CONTRACTS</b>	Contract(s) attesting to the validity of the relationship.
enumeration <b>OTHER_OFFICIAL_DOCUMENTS</b>	Other official document(s) attesting to the validity of the relationship.

### 2.3.43. Element `rr:RegistrationContainerType / ValidationReference`

A reference to a specific document or other source used as the basis of relationship validation for this relationship record.

#### `rr:Tokenized500Type`

content **simple**

minOccurs **0**

minLength **1**

maxLength **500**

pattern **\S+(\S+)\***

### 2.3.44. Element `rr:RegistrationContainerType / NextVersion`

A structure for adding further elements in to the `Registration` section of the Relationship Record in anticipation of a new version, by nesting a series of XML elements with this content model within the `NextVersion` element, one for each new minor version of the schema, postpending a serial number (1,2,3...) to the element name upon each iteration.

#### `rr:RegistrationNextVersionType`

content **complex**

minOccurs **0**

### 2.3.45. Element `rr:RegistrationContainerType` / Extension

This `Extension` element may contain any additional elements required to extend the `Registration` container element.

#### `rr:ExtensionType`

content **complex**

minOccurs **0**

### 2.3.46. Element `rr:RelationshipRecordType / NextVersion`

A structure for adding further elements in to the `Registration` section of the Relationship Record in anticipation of a new version, by nesting a series of XML elements with this content model within the `NextVersion` element, one for each new minor version of the schema, postpending a serial number (1,2,3...) to the element name upon each iteration.

#### `rr:RelationshipRecordNextVersionType`

content **complex**

minOccurs **0**

### 2.3.47. Element rr:RelationshipRecordType / Extension

This `Extension` element may contain any additional elements required to extend the `RelationshipRecord`.

#### **rr:ExtensionType**

content **complex**

minOccurs **0**

### 2.3.48. Simple Type rr:LEIDateTimeProfile

**Type** restriction of `xs:dateTime`

**Facets** pattern

```
((^\.)*|(^\.)*(\.((\d){1,3}))){0,1})(Z|\+([01][0-9]|2[0-3]):([0-5][0-9])|-([01][0-9]|2[0-3]):([0-5][0-9]))
```

### 2.3.49. Simple Type rr:LEIType

**Type** restriction of **xs:string**

minLength 20

**Facets** maxLength 20

pattern **([0-9A-Z]{18}[0-9]{2})**

### 2.3.50. Simple Type rr:FileContentEnum

restriction of **xs:string**

enumeration <b>LOU_FULL_INTERNAL</b>	The file contains all relationship records created for internal use by an LOU (all internal relationship records for which the LOU is the <code>ManagingLOU</code> ) as of the date/time the file is created.
enumeration <b>LOU_DELTA_INTERNAL</b>	The file contains those relationship records created by an LOU for internal use (all internal relationship records for which the LOU is the <code>ManagingLOU</code> ) which are new or changed since the <code>DeltaStart</code> specified in the <code>Header</code> , as of the date/time the file is created.
enumeration <b>LOU_FULL_PUBLISHED</b>	The file contains all relationship records published by an LOU (all relationship records for which the LOU is the <code>ManagingLOU</code> ) as of the date/time the file is created.
enumeration <b>LOU_DELTA_PUBLISHED</b>	The file contains those relationship records published by an LOU (all relationship records for which the LOU is the <code>ManagingLOU</code> ) which are new or changed since the <code>DeltaStart</code> specified in the <code>Header</code> , as of the date/time the file is created.
enumeration <b>GLEIF_FULL_INTERNAL</b>	The file contains all relationship records GLEIF manages internally to the GLEIS (including all internal relationship records from all LOUs) as of the date/time the file is created.
enumeration <b>GLEIF_DELTA_INTERNAL</b>	The file contains those relationship records GLEIF manages internally to the GLEIS (including all internal relationship records from all LOUs) which are new or changed since the <code>DeltaStart</code> date specified in the <code>Header</code> , as of the date/time the file is created.

enumeration <b>GLEIF_FULL_PUBLISHED</b>	The file contains all relationship records published by GLEIF (including all relationship records from all LOUs) as of the date/time the file is created.
enumeration <b>GLEIF_DELTA_PUBLISHED</b>	The file contains those relationship records published by GLEIF (including all relationship records from all LOUs) which are new or changed since the <code>DeltaStart</code> date specified in the <code>Header</code> , as of the date/time the file is created.
enumeration <b>QUERY_RESPONSE</b>	The file contains records matching criteria specified in a query.

### 2.3.51. Simple Type rr:NodeIDTypeEnum

restriction of **xs:string**

enumeration **LEI**

An LEI code taken from the LEI issuing prefix namespace of a GLEIS LOU.

enumeration **ISO\_17442\_COMPATIBLE**

An ISO 17442-compatible code, not taken from the LEI issuing prefix namespace of a GLEIS LOU.

### 2.3.52. Simple Type rr:RelationshipTypeEnum

restriction of **xs:string**

enumeration **IS\_DIRECTLY\_CONSOLIDATED\_BY**

*StartNode* is directly consolidated by *EndNode*. The *StartNode* "child" entity has its accounts fully consolidated by the *EndNode* "parent" entity, in the sense given by the accounting standard(s) specified in *RelationshipQualifiers*; the *EndNode* entity is the closest fully consolidating parent to the *StartNode* entity in any applicable hierarchical ownership structure.

enumeration **IS\_ULTIMATELY\_CONSOLIDATED\_BY**

*StartNode* is ultimately consolidated by *EndNode*. The *StartNode* "child" entity has its accounts fully consolidated by the *EndNode* "parent" entity, in the sense given by the accounting standard(s) specified in *RelationshipQualifiers*; the *EndNode* entity is the most distant fully consolidating parent from the *StartNode* entity in any applicable hierarchical ownership structure.

enumeration **IS\_INTERNATIONAL\_BRANCH\_OF**

*StartNode* is an international branch of the legal entity designated by *EndNode* (in jurisdiction country of *StartNode*). The *EndNode* is the Head Office and MUST be an LEI.

### 2.3.53. Simple Type rr:PeriodTypeEnum

restriction of **xs:string**

enumeration **ACCOUNTING\_PERIOD**

The dates in this instance of `RelationshipPeriod` indicate the accounting period covered by the most recent validation documents for this relationship.

enumeration **RELATIONSHIP\_PERIOD**

The dates in this instance of `RelationshipPeriod` indicate the duration of validity of the relationship itself, as distinct from any administrative or reporting aspects.

enumeration **DOCUMENT\_FILING\_PERIOD**

The dates in this instance of `RelationshipPeriod` indicate the validity period of a regulatory filing, accounting document, or other document demonstrating the relationship's validity.

### 2.3.54. Simple Type rr:RelationshipStatusEnum

restriction of **xs:string**

enumeration **ACTIVE**

As of the last report or update, the reporting legal entity reported that it is legally registered and/or operating, AND that the relationship detailed in this `RelationshipRecord` is still valid.

enumeration **INACTIVE**

It has been determined that the relationship ended, e.g. because entity that reported this relationship is no longer legally registered and/or operating; or the relationship is no longer valid for other reasons.

### 2.3.55. Simple Type `rr:QualifierDimensionEnum`

restriction of `xs:string`

enumeration **ACCOUNTING\_STANDARD**

The accounting standard applied to determine the definition of e.g. ultimate or direct accounting consolidating parent for the relationship detailed in this `RelationshipRecord`. The relevant accounting standard is that applicable to the `EndNode` (the "parent" entity).

### 2.3.56. Simple Type `rr:QualifierCategoryTypeEnum`

restriction of `xs:string`

enumeration `US_GAAP`

United States-Generally Accepted Accounting Principles.

enumeration `IFRS`

International Financial Reporting Standard (developed by the International Accounting Standards Board – IASB see <http://www.ifrs.org>).

enumeration `OTHER_ACCOUNTING_STANDARD`

A financial reporting (accounting) standard not otherwise listed in the latest version of the relationship data file format.

### 2.3.57. Simple Type rr:MeasurementMethodTypeEnum

restriction of **xs:string**

enumeration	<b>ACCOUNTING_CONSOLIDATIO</b>	Accounting consolidation holds when "[in the] financial statements of a group [...] the assets, liabilities, equity, income, expenses and cash flows of the parent and its subsidiaries are presented as those of a single economic entity (please see <a href="http://www.iasplus.com/en/standards/ias/ias-27-2011">http://www.iasplus.com/en/standards/ias/ias-27-2011</a> ).
n	<b>N</b>	

### 2.3.58. Simple Type rr:QuantifierUnitsTypeEnum

restriction of **xs:string**

enumeration

**PERCENTAGE**

### 2.3.59. Simple Type `rr:RegistrationStatusEnum`

restriction of `xs:string`

enumeration <b>PENDING_VALIDATION</b>	A relationship data report that has been submitted to the LOU and which is being processed and validated, prior to publication.
enumeration <b>PUBLISHED</b>	A relationship data report that has been validated and published, and which is reported by an entity that was an operating legal entity as of the last update.
enumeration <b>DUPLICATE</b>	A relationship data report that has been determined to be a duplicate registration of the same relationship. In many cases this will mean more than one report with e.g. the same 2 entity IDs, the same relationship type, certain status values and the same relationship date(s), but this determination will depend on the relationship type in question.
enumeration <b>LAPSED</b>	A relationship data report that has not been renewed by the <code>NextRenewalDate</code> .
enumeration <b>RETIRED</b>	The relationship is considered to have ended, but the relationship report is kept in publication for historical audit trail purposes.
enumeration <b>ANNULLED</b>	A relationship data report that was marked as erroneous or invalid after it was published. The relationship report is kept in publication for historical audit trail purposes only (so that data recipients can correct their local data).
enumeration <b>TRANSFERRED</b>	A relationship data report that has been transferred to a different LOU as the <code>ManagingLOU</code> . A record in this state is not

published, but may be used internally by the prior LOU for audit trail purposes.

enumeration **PENDING\_TRANSFER**

A relationship data report for which a transfer to another LOU has been requested. The request is being processed at the sending LOU. When the receiving LOU is ready, the status will be changed to `PENDING_ARCHIVAL` by the sending LOU prior to completion of the transfer.

enumeration **PENDING\_ARCHIVAL**

This relationship data report is about to be transferred to a different LOU, after which its registration status will revert to a non-pending status. The `PENDING_ARCHIVAL` status serves to inform recipients of LOU-provided data files that a relationship record will be removed from that LOU's published file after the transfer is complete.

### 2.3.60. Simple Type `rr:ValidationSourcesTypeEnum`

restriction of `xs:string`

enumeration **PENDING**

The validation of the relationship data provided by the registrant has not yet occurred. Records with this `ValidationSources` value **MUST** not be published.

enumeration **ENTITY\_SUPPLIED\_ONLY**

Based on the validation procedures in use by the LOU responsible for the record, the information associated with this record has significant reliance on the information that a submitter provided due to the unavailability of corroborating information.

enumeration **PARTIALLY\_CORROBORATED** (e.g. financial statements with other definitions of the relevant relationship type; quarterly or annual regulatory filings, contracts and other documents used in preparing financial statements).

Based on the validation procedures in use by the LOU responsible for the record, the information supplied by the submitter can be partially corroborated by supporting sources

enumeration **FULLY\_CORROBORATED**

The relationship data provided by the registrant has been validated against an explicit relationship statement found in key sources (e.g. consolidated financial statements).

### 2.3.61. Simple Type `rr:ValidationDocumentsTypeEnum`

restriction of `xs:string`

enumeration <b>ACCOUNTS_FILING</b>	A consolidated financial (accounting) statement, prepared and submitted to the relevant authority.
enumeration <b>REGULATORY_FILING</b>	An annual regulatory filing providing public information on parent relationships.
enumeration <b>SUPPORTING_DOCUMENTS</b>	Other documents supporting the preparation of consolidated financial statements.
enumeration <b>CONTRACTS</b>	Contract(s) attesting to the validity of the relationship.
enumeration <b>OTHER_OFFICIAL_DOCUMENTS</b>	Other official document(s) attesting to the validity of the relationship.

### 2.3.62. Simple Type `rr:Tokenized500Type`

An element of this type has minimum length of one character and may not contain any of: the carriage return (`#xD`), line feed (`#xA`) nor tab (`#x9`) characters, shall not begin or end with a space (`#x20`) character, or a sequence of two or more adjacent space characters.

restriction of `xs:string`

<code>minLength</code>	<b>1</b>
<code>maxLength</code>	<b>500</b>
<code>pattern</code>	<b><code>\S+(\S+)*</code></b>