```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified"
    targetNamespace="http://www.gleif.org/data/schema/repex/2016"
    xmlns:repex="http://www.gleif.org/data/schema/repex/2016">

    <xs:import namespace="http://www.w3.org/XML/1998/namespace"
        schemaLocation="http://www.w3.org/2001/xml.xsd"/>

    <xs:annotation>
        <xs:documentation>
            <h1>Reporting Exceptions Format Version 1.1</h1> Documentation
last updated: 2017-01-27
                <h2>Introduction</h2> Following the LEI-ROC policy
document, "Collecting data on
            direct and ultimate parents of legal entities in the Global LEI
System â□□ Phase 1" (10
            March 2016), the Global Legal Identifier System (GLEIS)
requires that legal entities
            with an LEI provide information on their ultimate and direct
accounting consolidating parents.<br/>
            <ul>
                <li>Relationship reporting according to the Relationship
Record Common Data File
                    (RR-CDF) format V1.0 is mandatory. The only opt-out
reasons allowed are taken
                    from the LEI-ROC policy document, pages 16-17. A
further general exception
                    case, also based on the LEI-ROC policy document (p. 18)
is also provided to
                    cover situations where the opt-out reasons may not be
precisely applicable: <ul>

                        <li>No LEI - "the parent does not consent to have
an LEI" (LEI-ROC policy,
                            p. 18).</li>
                    </ul>
                </li>
                <li>This format provides a simple record structure linking,
<strong>per
                        record:</strong></li>
                <ul>
                    <li>One LEI from the LOU's current LEI data file;</li>
                    <li>One relationship type (reporting category) that
must be reported;</li>
                    <li>One reason for declining to report that
relationship type for the legal
                        entity referenced by this LEI, plus an optional
reference e.g. to a legal or
                        regulatory provision.</li>
                </ul>
            </ul><ul>All LOUs use this file format to record and submit
Reporting Exceptions to
                GLEIF. </ul>
            <h2>Audience for this document</h2> The target audience for
this standard includes: <ul>
                <li>All Local Operating Units (as well as candidate LOUs)
of the GLEIS</li>
                <li>All users or potential users of LEI data</li>
                <li>All financial regulators who consume LEI data</li>
            </ul>
            <h2>Status of this document</h2> 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<br/>The file format references
            the LEI ROC's published document entitled "LEI ROC Report on
collecting data on direct
            and ultimate parents of legal entities in the Global LEI
System" (10 March 2016;
            available from www.leiroc.org).<br/>
            <h2>Terminology and Typographical Conventions</h2> The
following typographical
            conventions are used throughout the document: <ul>
                <li>ALL CAPS type is used for the special terms enumerated
above.</li>
                <li><code>Monospace type</code> is used to denote
programming language, UML, and XML
                    identifiers, as well as for the text of XML
documents.</li>
            </ul>
            <h2>Cardinalities</h2>
            <ul>
                <li>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:</li>
                <ul>
                    <li>Mandatory, unique: <code>{1,1}</code> - the element
MUST appear, exactly
                        once.</li>
                    <li>Mandatory, repeatable: <code>{1,unbounded}</code> -
the element MUST appear
                        at least once. It may be repeated any number of
times.</li>
                    <li>Optional, unique: <code>{0,1}</code> - the element
NEED NOT appear; it MAY
                        appear once at most.</li>
                    <li>Optional, repeatable: <code>{0,unbounded}</code> -
the element NEED NOT
                        appear. It MAY be repeated any number of
times.</li>
                </ul>
            </ul>Please note: <ul>
                <li>The default cardinality is {1,1} (mandatory, unique).
This document highlights
                    when an element differs from this either by its
<code>minOccurs</code> (minimum
                    occurrences) or <code>maxOccurs</code> (maximum
occurrences) value, or both. </li>
                <li>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
```

```
                        <strong>given</strong> the presence of the
container.</li>
                <li>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.</li>
            </ul>
            <h2>Business Rules</h2> 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.
<br/><br/>
            <h2>Release Notes</h2>

            <h3>Version 1.1</h3>
            <ul>
            <li>Corrections:<ul>
                <li>
            Extension element in Header corrected to
minOccurs="0".</li></ul>
            </li></ul>

            <h3>Version 1.0</h3> The first release. <h2>Change
Management</h2> 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:<br/><br/>
            <hr3>Errata Version</hr3> 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.<br/><br/>
            <h3>Minor Version</h3> 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 (â□□enumâ□□ 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.<br/><br/>
            <h3>Major Version</h3> 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.<br/><br/> 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.<br/><br/>
            <h3>Minor Version Changes to the XML Schema</h3> A minor
version may introduce new XML
            elements and/or adds one or more codes to a code list
(â□□enumâ□□ data type). Minor version
            changes to schema SHALL be made as specified below, in order to
achieve forward and
            backward compatibility.<br/><br/> Forward compatibility means
that an LEI Data File
            which is valid according to the older versionâ□□s schema is
also valid according to the
            newer versionâ□□s schema.<br/><br/> Backward compatibility
means that an LEI Data File
            which is valid according to the newer versionâ□□s schema is
also valid according to the
            older versionâ□□s schema.<br/><br/> 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.<br/><br/> As can be seen
            from the full XML schema presented here, the following rules
SHALL be observed to ensure
            forward and backward compatibility: <ul>
                <li>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.</li>
                <li>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.</li>
                <li>New elements SHALL be introduced in a subsequent minor
version by modifying the
                    declaration for the above type declaration as follows:
<ul>
                        <li>A sequence of the new elements introduced in
the previous version</li>
                        <li>A subsequent NextVersionN element where N is an
index number starting at
                            1 and incremented by 1 with each minor
version</li>
                    </ul>
                    <li>Each new element SHALL be declared
minOccurs=â□□0â□□, 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.</li>
                    <li>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 â□□unique
particle attributionâ□□ constraint
                        of XSDÂ 1.0.</li>
                    <li>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.</li>
                </li>
            </ul>
            <h3>Major Version Changes to the XML Schema</h3> A major
version may make any change to
            the XML schema whatsoever, including incompatible
changes.<br/><br/> 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 â□□aâ□□,
            â□□bâ□□, â□□câ□□, etc., may be appended to the year as
needed.<br/><br/> 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.<br/><br/>
            <h2>XML Syntax</h2> This section specifies the XML schema for
an LEI data file
            conforming to this standard.<br/><br/>
            <h3>XML Design Rules</h3>
            <ul>
                <li>The XSD schema conforms to W3C's XML Schema
specification, version 1.0.</li>
                <li>The XML namespace is
"http://www.gleif.org/data/schema/repex/2016".</li>
                <li>All interior XML elements are namespace-qualified
(element form =
                    qualified).</li>
                <li>All XML attributes are in the null namespace (attribute
form = unqualified),
                    with the exception of <code>xml:lang</code>.</li>
                <li>Element names are upper camel case.</li>
                <li>Attribute name are lower camel case.</li>
                <li>XSD type names are upper camel case.</li>
                <li>Enumeration code list values are all caps with
underscores.</li>
                <li>Elements are used in preference to attributes except
for language and type
                    qualifiers.</li>
                <li>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.</li>
            </ul>
            <h3>XML Schema</h3> An XML file conforming to this standard
SHALL be valid according to
            the following XSD 1.0 schema.<br/><br/>
            <h3>Extension</h3> The optional Extension section of an
Reporting Exception Item 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.<br/> For
            example, an LOU may use Extension to publish additional data
elements it collects as
            part of registration.<br/> The following rules MUST be
observed: <ul>
                <li>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.</li>
                <li>The XML namespace for an extension element SHALL be a
namespace to which the
```

```
                    creator of the extension element is entitled to use;
e.g., a namespace derived
                    from the Internet Domain Name of the creator, a
namespace agreed upon by a group
                    of trading partners, etc.</li>
                <li>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.</li>
                <li>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.</li>
            </ul>
            <h2>Abstract Data Content</h2> 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: <ul>
                <li>A Header.</li>
                <li>Zero or more Reporting Exception Items.</li>
            </ul>
        </xs:documentation>
    </xs:annotation>

    <xs:element name="ReportingExceptionData"
type="repex:ReportingExceptionDataType"/>

    <xs:complexType name="ReportingExceptionDataType">
        <xs:sequence>
            <xs:element name="Header" type="repex:ExceptionHeaderType">
                <xs:annotation>
                    <xs:documentation> Contains the file upload information
for this Reporting
                        Exceptions file </xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="ReportingExceptions"
type="repex:ReportingExceptionsType">
                <xs:annotation>
                    <xs:documentation> A list of reporting exceptions, with
reason(s) for each
                        exception, provided by legal entities, for
declining to provide specified
                        classes of relationship information
</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="ExceptionHeaderType">
```

```
        <xs:sequence>

            <xs:element name="ContentDate" type="repex:LEIDateTimeProfile">
                <xs:annotation>
                    <xs:documentation> The date and time as of which the
data contained in the file
                        is valid.</xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="Originator" type="repex:LEIType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation> The LEI of the entity that created
the content of this file.
                    </xs:documentation>
                </xs:annotation>
            </xs:element>


            <xs:element name="FileContent" type="repex:FileContentEnum">
                <xs:annotation>
                    <xs:documentation>A code describing the content of this
RelationshipRecords
                        file. </xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="DeltaStart" type="repex:LEIDateTimeProfile"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation> The date and time of the baseline
relative to which this file
                        contains new or changed Reporting Exceptions.
</xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="RecordCount" type="xs:nonNegativeInteger">
                <xs:annotation>
                    <xs:documentation> The number of Reporting Exceptions
(RepEx) in the file. Must
                        be a positive whole (integer) number, or zero (0).
</xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="NextVersion"
type="repex:HeaderNextVersionType" minOccurs="0"/>

            <xs:element name="Extension" type="repex:ExtensionType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation> This <code>Extension</code> element
contains any additional
                        elements required to extend the <code>Header</code>
container
                        element.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </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="FileContentEnum">
        <xs:restriction base="xs:string">

            <!-- WARNING: Enumeration values strictly validate-->

            <xs:enumeration value="LOU_FULL_PUBLISHED">
                <xs:annotation>
                    <xs:documentation>The file contains all Reporting
Exceptions published by an LOU
                        (all Reporting Exceptions for which the LOU is the
<code>ManagingLOU</code>)
                        as of the date/time the file is
created.</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="LOU_DELTA_PUBLISHED">
                <xs:annotation>
                    <xs:documentation>The file contains those Reporting
Exceptions published by an
                        LOU (all Reporting Exceptions for which the LOU is
the
                            <code>ManagingLOU</code>) which are new or
changed since the
                            <code>DeltaStart</code> specified in the
header, as of the date/time the
                        file is created.</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="GLEIF_FULL_PUBLISHED">
                <xs:annotation>
                    <xs:documentation>The file contains all Reporting
Exceptions published by GLEIF
                        (including all Reporting Exceptions from all LOUs)
as of the date/time the
                        file is created.</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="GLEIF_DELTA_PUBLISHED">
                <xs:annotation>
                    <xs:documentation>The file contains those Reporting
Exceptions published by
                        GLEIF (including all Reporting Exceptions 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.</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="QUERY_RESPONSE">
                <xs:annotation>
```

```
                    <xs:documentation>The file contains records matching
criteria specified in a
                        query.</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
        </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="ExceptionType">
        <xs:sequence>

            <xs:element name="LEI" type="repex:LEIType">
                <xs:annotation>
                    <xs:documentation> The ISO 17442 compatible identifier
for the legal entity
                        which raised this exception.</xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="ExceptionCategory"
type="repex:ExceptionCategoryEnum">
                <xs:annotation>
                    <xs:documentation>A category of mandatory information
reporting, beyond the
                        minimum required for legal entity identification,
which the legal entity
                        declines, giving reasons and references where
applicable.
                    </xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="ExceptionReason"
type="repex:ExceptionReasonEnum"
                maxOccurs="unbounded">
                <xs:annotation>
                    <xs:documentation>A single reason provided by the legal
entity for declining to
                        provide the mandatory report of a specified type of
information (beyond the
                        minimum reference data needed purely for
identification of the legal
                        entity). </xs:documentation>
                </xs:annotation>
            </xs:element>


            <xs:element name="ExceptionReference"
type="repex:Tokenized500Type" minOccurs="0"
                maxOccurs="unbounded">
                <xs:annotation>
                    <xs:documentation>References of the law, regulation or
other element of the
```

```
                        legal framework to support reason(s) provided by
the legal entity for
                        declining to provide information on its parents.
</xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="NextVersion"
type="repex:ReportingExceptionNextVersionType"
                minOccurs="0"/>

            <xs:element name="Extension" type="repex:ExtensionType"
minOccurs="0">
                <xs:annotation>
                    <xs:documentation> This <code>repex:Extension</code>
element may contain any
                        additional elements required to extend the
Reporting Exception.
                    </xs:documentation>
                </xs:annotation>
            </xs:element>

        </xs:sequence>
    </xs:complexType>


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

            <xs:element name="Exception" type="repex:ExceptionType"
minOccurs="0"
                maxOccurs="unbounded">
                <xs:annotation>
                    <xs:documentation> A single exception to a specified
reporting requirement,
                        giving reasons and references where
applicable.</xs:documentation>
                </xs:annotation>
            </xs:element>

            <xs:element name="NextVersion"
type="repex:ReportingExceptionsContainerNextVersionType"
                minOccurs="0"/>

        </xs:sequence>
    </xs:complexType>

    <xs:simpleType name="ExceptionCategoryEnum">
        <xs:restriction base="xs:string">
            <!-- Enumeration values for LEI-CDF 2.0: -->
            <xs:enumeration value="DIRECT_ACCOUNTING_CONSOLIDATION_PARENT">
                <xs:annotation>
                    <xs:documentation>The legal entity has declined to
report a direct accounting
                        consolidation parent, based on applicable
accounting standards.
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration
value="ULTIMATE_ACCOUNTING_CONSOLIDATION_PARENT">
                <xs:annotation>
                    <xs:documentation>The legal entity has declined to
report an ultimate accounting
                        consolidation parent, based on applicable
accounting standards.
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>

        </xs:restriction>
    </xs:simpleType>


    <xs:simpleType name="ExceptionReasonEnum">
        <xs:restriction base="xs:string">
            <!-- Enumeration values for LEI-CDF 2.0: -->

            <xs:enumeration value="NO_LEI">
                <xs:annotation>
                    <xs:documentation> The parent does not consent to have
an LEI.
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>

            <xs:enumeration value="NATURAL_PERSONS">
                <xs:annotation>
                    <xs:documentation>There is no parent according to the
definition used, because
                        the entity is controlled by natural person(s)
without any intermediate legal
                        entity meeting the definition of accounting
consolidating parent.
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="NON_CONSOLIDATING">
                <xs:annotation>
                    <xs:documentation> There is no parent according to the
definition used, because
                        the entity is controlled by legal entities not
subject to preparing
                        consolidated financial statements.
</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="NO_KNOWN_PERSON">
                <xs:annotation>
                    <xs:documentation> There is no parent according to the
definition used, because
                        there is no known person controlling the entity
(e.g.., diversified
                        shareholding). </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="LEGAL_OBSTACLES">
                <xs:annotation>
                    <xs:documentation> Obstacles in the laws or regulations
of a jurisdiction
```

```
                        prevent providing or publishing this information.
This does not include
                        cases where, under the applicable legal framework
disclosing the parent
                        relationship would require the consent of one of
the entities in the
                        relationship, or both, and such consent could not
be obtained (in these
                        cases "CONSENT_NOT_OBTAINED" is the applicable
code). The LOU is not
                        expected to verify or analyze whether the legal
framework constitutes a
                        legal obstacle. </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="CONSENT_NOT_OBTAINED">
                <xs:annotation>
                    <xs:documentation> Obstacles in the laws or regulations
of a jurisdiction
                        prevent providing or publishing this information:
â□□the consent of the parent
                        was necessary under the applicable legal framework
and the parent did not
                        consent or could not be contactedâ□□. Note that it
is a responsibility of a
                        child entity to seek parent consent when necessary
for disclosing the parent
                        relationship, for instance by inviting in writing
the parent entity to
                        provide consent. The LOU is not expected to verify
or analyze whether the
                        legal framework constitutes a legal obstacle.
</xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            <xs:enumeration value="BINDING_LEGAL_COMMITMENTS">
                <xs:annotation>
                    <xs:documentation> Binding legal commitments (other
than the laws or regulations
                        of a jurisdiction), such as articles governing the
legal entity or a
                        contract, prevent providing or publishing this
information. The LOU is not
                        expected to verify or analyze whether the legal
framework constitutes a
                        legal obstacle. </xs:documentation>
                </xs:annotation>
            </xs:enumeration>

            <!-- this is not in itself a cause of opt out, but only one of
the cumulative reasons for â□□detriment_not_excludedâ□□

            <xs:enumeration value="PARENT_UNREACHABLE">
                <xs:annotation>
                    <xs:documentation>
                        <ul>
                            <li>The parent could not be informed via the
GLEIS and have the
                                possibility to correct the relationship
information before
```

```
                                publication (including raising a cause for
opt out). The disclosure
                                of this information would be detrimental to
the legal entity or the
                                relevant parent. This will include reasons
generally accepted by
                                public authorities in similar
circumstances, based on a declaration
                                by the entity.</li>
                            <li>Business Rules: This exception reason
applies to reporting
                                requirements
<code>ULTIMATE_ACCOUNTING_CONSOLIDATION_PARENT</code>
                                and
<code>DIRECT_ACCOUNTING_CONSOLIDATION_PARENT</code>.</li>
                        </ul>
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
            -->
            <!-- this is not in itself a cause of opt out, but only one of
the cumulative reasons for â□□detriment_not_excludedâ□□
            <xs:enumeration value="RELATIONSHIP_NOT_PUBLIC">
                <xs:annotation>
                    <xs:documentation>
                        <ul>
                            <li>The relationship is not already in the
public domain (information
                                being in the public domain assumes here
that the way the information
                                came into the public domain did not
infringe the applicable legal
                                framework). The disclosure of this
information would be detrimental
                                to the legal entity or the relevant parent.
This will include
                                reasons generally accepted by public
authorities in similar
                                circumstances, based on a declaration by
the entity.</li>
                            <li>Business Rules: This exception reason
applies to reporting
                                requirements
<code>ULTIMATE_ACCOUNTING_CONSOLIDATION_PARENT</code>
                                and
<code>DIRECT_ACCOUNTING_CONSOLIDATION_PARENT</code>.</li>
                        </ul>
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
-->
            <!-- this is not in itself a cause of opt out, but only one of
the cumulative reasons for â□□detriment_not_excludedâ□□
            <xs:enumeration value="DETRIMENT_CONCERN">
                <xs:annotation>
                    <xs:documentation>
                        <ul>
                            <li>The child entity has reasons to believe
that the parent may consider
                                disclosure to be detrimental. The
disclosure of this information
```

```
                                would be detrimental to the legal entity or
the relevant parent.
                                This will include reasons generally
accepted by public authorities
                                in similar circumstances, based on a
declaration by the entity.</li>
                            <li>Business Rules: This exception reason
applies to reporting
                                requirements
<code>ULTIMATE_ACCOUNTING_CONSOLIDATION_PARENT</code>
                                and
<code>DIRECT_ACCOUNTING_CONSOLIDATION_PARENT</code>.</li>
                        </ul>
                    </xs:documentation>
                </xs:annotation>
            </xs:enumeration>
-->
            <xs:enumeration value="DETRIMENT_NOT_EXCLUDED">
                <xs:annotation>
                    <xs:documentation> The child entity has sought to
consult the parent entity
                        about the reporting of the parent information to
the GLEIS but could not
                        confirm the absence of detriment in a way that can
appropriately prevent
                        liability risks for the child entity (or those
acting on its behalf) under
                        the applicable legal framework. The disclosure of
this information would be
                        detrimental to the legal entity or the relevant
parent. This will include
                        reasons generally accepted by public authorities in
similar circumstances,
                        based on a declaration by the entity.
<br/><br/>This reason may be used only when all
                        following cumulative circumstances apply: [i) the
parent could not be
                        informed via the GLEIS and have the possibility to
correct the relationship
                        information before publication (including raising a
cause for opt out,
                        either because the parent does not have an LEI, or
it has an LEI but the
                        GLEIS has not yet implemented such system;] ii) the
relationship is not
                        already in the public domain (information being in
the public domain assumes
                        here that the way the information came into the
public domain did not
                        infringe the applicable legal framework); iii) the
child entity has reasons
                        to believe that the parent may consider disclosure
to be detrimental; (iv)
                        the child entity has sought to consult the parent
entity of the reporting of
                        the parent information to the GLEIS but could not
confirm the absence of
                        detriment in a way that can appropriately prevent
liability risks for the
                        child entity (or those acting on its behalf) under
the applicable legal
```

```
                        framework.</xs:documentation>
                </xs:annotation>
            </xs:enumeration>

            <xs:enumeration value="DISCLOSURE_DETRIMENTAL">
                <xs:annotation>
                    <xs:documentation> The disclosure of this information
would be detrimental to
                        the legal entity or the relevant parent. This will
include reasons generally
                        accepted by public authorities in similar
circumstances, based on a
                        declaration by the entity. </xs:documentation>
                </xs:annotation>
            </xs:enumeration>

        </xs:restriction>
    </xs:simpleType>

    <xs:simpleType name="Tokenized500Type">
        <xs:annotation>
            <xs:documentation>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.</xs:documentation>
        </xs:annotation>
        <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="LEIDateTimeProfile">
        <xs:restriction base="xs:dateTime">

            <xs:annotation>
                <xs:documentation>
                    <ul>
                        <li>Elements of base data type
<code>repex:LEIDateTimeProfile</code> use
                            this datatype to further restrict the ISO 8601
range of date and time
                            format to the single format:
                                <br/><br/><strong>YYYY-MM-
DDThh:mm:ss.sssTZ</strong><br/>
                            <br/> where the components of the above string
are as follows: <ul>
                                <li>YYYY is the year</li>
                                <li>MM is the month (01 = January, â□Ś, 12
= December)</li>
                                <li>DD is the day of the month (01 = first
day of the month)</li>
                                <li>T is the single character â□□Tâ□□</li>
                                <li>hh is the hour (00 â□□ 23)</li>
                                <li>mm is the minute</li>
                                <li>ss.sss is the second and milliseconds.
Two digits must be used
                                    for the seconds. From one to three
digits may be used for
                                    milliseconds, or omitted entirely along
with the decimal
                                    point.</li>
                                <li>TZ is the time zone specifier, which
can be one of:</li>
                                <ul>
                                    <li>Z the single character â□□Zâ□□,
denoting Coordinated Universal
                                        Time (UTC); or</li>
                                    <li>+hh:mm denoting a positive offset
from UTC; or</li>
                                    <li>-hh:mm denoting a negative offset
from UTC</li>
                                </ul>
                            </ul> This pattern therefore adds the
restrictions, beyond the ISO 8601
                            specification: <ul>
                                <li>Only the specified pattern of digits,
indicators and separators
                                    may be used (no spaces or other white
space characters).</li>
                                <li>The time zone (TZ) MUST be present,
although it
                                        <strong>may</strong> be zero (Z)
<strong>if</strong> all
                                    dates and times are expressed as
UTC+00:00.</li>
                                <li>Only 3 decimal places maximum are
allowed in the seconds section
                                    (ss.sss).</li>
                            </ul>
                        </li>
                    </ul>
                </xs:documentation>
            </xs:annotation>

            <xs:pattern
                value="([^\.]*|([^\.]*(\.((\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]))"
            />
        </xs:restriction>
    </xs:simpleType>

    <!-- NextVersion datatypes in analogy to LEI-CDF V1.0 -->

    <xs:complexType name="ReportingExceptionsContainerNextVersionType">
        <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
                namespace="##targetNamespace"/>
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="ReportingExceptionNextVersionType">
        <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
```

```
                namespace="##targetNamespace"/>
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="HeaderNextVersionType">
        <xs:sequence>
            <xs:any minOccurs="0" maxOccurs="unbounded"
processContents="lax"
                namespace="##targetNamespace"/>
        </xs:sequence>
    </xs:complexType>

</xs:schema>
```