

Open Group Standard

Open Data Element Framework (O-DEF), Version 1.0



Copyright © 2016, The Open Group

The Open Group hereby authorizes you to use this document for any purpose, PROVIDED THAT any copy of this document, or any part thereof, which you make shall retain all copyright and other proprietary notices contained herein.

This document may contain other proprietary notices and copyright information.

Nothing contained herein shall be construed as conferring by implication, estoppel, or otherwise any license or right under any patent or trademark of The Open Group or any third party. Except as expressly provided above, nothing contained herein shall be construed as conferring any license or right under any copyright of The Open Group.

Note that any product, process, or technology in this document may be the subject of other intellectual property rights reserved by The Open Group, and may not be licensed hereunder.

This document is provided “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Any publication of The Open Group may include technical inaccuracies or typographical errors. Changes may be periodically made to these publications; these changes will be incorporated in new editions of these publications. The Open Group may make improvements and/or changes in the products and/or the programs described in these publications at any time without notice.

Should any viewer of this document respond with information including feedback data, such as questions, comments, suggestions, or the like regarding the content of this document, such information shall be deemed to be non-confidential and The Open Group shall have no obligation of any kind with respect to such information and shall be free to reproduce, use, disclose, and distribute the information to others without limitation. Further, The Open Group shall be free to use any ideas, concepts, know-how, or techniques contained in such information for any purpose whatsoever including but not limited to developing, manufacturing, and marketing products incorporating such information.

If you did not obtain this copy through The Open Group, it may not be the latest version. For your convenience, the latest version of this publication may be downloaded at www.opengroup.org/bookstore.

Open Group Standard

Open Data Element Framework (O-DEF), Version 1.0

ISBN: 1-937218-75-1

Document Number: C163

Published by The Open Group, May 2016.

Comments relating to the material contained in this document may be submitted to:

The Open Group, Apex Plaza, Forbury Road, Reading, Berkshire, RG1 1AX, United Kingdom
or by electronic mail to:

ogspeccs@opengroup.org

Contents

1	Introduction.....	1
1.1	Objective.....	1
1.2	Overview.....	1
1.3	Conformance.....	1
1.4	Normative References.....	2
1.5	Terminology	2
1.6	Future Directions	2
2	Definitions.....	4
2.1	Classification	4
2.2	Core Index	4
2.3	Data Element.....	4
2.4	Data Element Concept	4
2.5	Node.....	5
2.6	Object Class	5
2.7	Object Class Tree.....	5
2.8	Plugin.....	5
2.9	Property.....	5
2.10	Property Tree	5
2.11	Relative Description	5
2.12	Relative Identifier	5
2.13	Relative Name.....	6
2.14	Role.....	6
2.15	Role Tree	6
2.16	Root.....	6
2.17	Status.....	6
2.18	Tree	6
3	The Index	7
3.1	Overall Structure.....	7
3.2	Languages	7
3.3	Changes.....	8
3.4	Trees	8
3.4.1	Node Identifiers.....	8
3.4.2	Node Names	8
3.4.3	Node Descriptions	9
3.5	Unstructured Plugins.....	9
4	Classification of Data Elements	10
4.1	Introduction (Informative)	10
4.1.1	Overall Classification	10

4.1.2	Process Overview	10
4.1.3	Unclassifiable Data Elements.....	11
4.1.4	Observations (Informative)	11
4.2	Determining the Object.....	11
4.2.1	Indirect Properties	11
4.2.2	Identifier Properties.....	12
4.3	Determining the Object Class	12
4.4	Determining the Role.....	13
4.5	Determining the Property.....	13
5	Classification Syntax.....	15
5.1	Data Element Concepts.....	15
5.1.1	Data Element Concept Identifiers	15
5.1.2	Data Element Concept Names.....	15
5.2	Object Classes.....	15
5.2.1	Object Class Identifiers	15
5.2.2	Object Class Names.....	16
5.3	Roles	16
5.3.1	Role Identifiers	16
5.3.2	Role Names	17
5.4	Properties	17
5.4.1	Property Identifiers.....	17
5.4.2	Property Names	17
5.5	Nodes	18
5.5.1	Node Identifiers.....	18
5.5.2	Node Names	18
5.6	Plugins	18
5.6.1	Plugin Identifiers	18
5.6.2	Plugin Names	18
5.6.3	Plugin Member Identifiers.....	19
5.6.4	Plugin Member Names	19
5.7	Basic Components	19
5.7.1	Positive Integer.....	19
5.7.2	Name String.....	19
5.7.3	Bracketed String	19
6	Index Files.....	20
6.1	Main Description Files.....	20
6.2	Description RDF Resources.....	20
6.3	Plugin Specifications	22
6.4	Object Class, Role, and Property Specifications	22
6.5	Tree Structures	22
6.5.1	Object Classes	23
6.5.2	Roles.....	23
6.5.3	Properties.....	23
7	Initial Index	24
7.1	Object Classes.....	24

7.2	Roles	25
7.3	Properties	25
7.4	Plugins	27
7.4.1	The O-DEF Plugin.....	28
7.4.2	The Enterprise Operation Plugin.....	29
7.4.3	The Purchase Plugin.....	30
A	Initial Core Main Description File	32
B	O-DEF Plugin Main Description File	45
C	Enterprise Operation Plugin Main Description File.....	49
D	Purchase Plugin Main Description File.....	51
E	Some Fundamental Enterprise Data Elements	54
E.1	Enterprise Name	54
E.2	Person Family Name.....	54
E.3	Person Birth Date.....	55
E.4	Employee Person Birth Date.....	56
E.5	Customer Person Name	57
F	Example Web Shopping Trolley Classification	59
G	Example Dublin Core Plugin	60
G.1	Plugin Identification	60
G.2	Plugin Definition.....	60
G.3	Examples of Use	67
G.3.1	Online Bookstore.....	67
G.3.2	ArchiMate Models.....	68
H	Example UNSPSC Product Item Plugin	70
H.1	Plugin Identification	70
H.2	Plugin Definition.....	70
H.3	Example of Use.....	71
I	Example Charitable Foundation Case Study.....	72
I.1	Plugin Identification	72
I.2	Plugin Definition.....	72
I.3	Examples of Use	76
J	Example Electric Vehicle Battery Status Reporting	78
J.1	Message Format.....	78
J.2	Data Model	79
J.3	Data Item Classifications	79
J.3.1	Electric Vehicle VIN	79
J.3.2	Battery Serial Number.....	80

	J.3.3	Battery GTIN.....	81
	J.3.4	Battery Effective Capacity	82
J.4		EV Plugin.....	83
	J.4.1	Plugin Identification	83
	J.4.2	Plugin Definition	84
J.5		Moving Definitions into the Index.....	84

Preface

The Open Group

The Open Group is a global consortium that enables the achievement of business objectives through IT standards. With more than 500 member organizations, The Open Group has a diverse membership that spans all sectors of the IT community – customers, systems and solutions suppliers, tool vendors, integrators, and consultants, as well as academics and researchers – to:

- Capture, understand, and address current and emerging requirements, and establish policies and share best practices
- Facilitate interoperability, develop consensus, and evolve and integrate specifications and open source technologies
- Offer a comprehensive set of services to enhance the operational efficiency of consortia
- Operate the industry's premier certification service

Further information on The Open Group is available at www.opengroup.org.

The Open Group publishes a wide range of technical documentation, most of which is focused on development of Open Group Standards and Guides, but which also includes white papers, technical studies, certification and testing documentation, and business titles. Full details and a catalog are available at www.opengroup.org/bookstore.

Readers should note that updates – in the form of Corrigenda – may apply to any publication. This information is published at www.opengroup.org/corrigenda.

This Document

This document is The Open Group Standard for the Open Data Element Framework, Version 1.0. It has been developed and approved by The Open Group.

The Open Data Element Framework (O-DEF) enables basic units of data to be classified, so that equivalences and similarities between them can be determined easily. This simplifies the development of interface software and contributes to improved management and organization of data. While it may have other applications, the O-DEF is primarily intended for deployment within extended enterprises, and by groups of collaborating enterprises.

The term “classification” is used in a general sense, as in the well-known Dewey Decimal Classification scheme,¹ and not in the specialized sense of a security classification. The use of the word “classification” does not imply a taxonomy; in particular, various subclasses at any point in the index may or may not be mutually-exclusive.

¹ See Wikipedia: https://en.wikipedia.org/wiki/Dewey_Decimal_Classification.

The O-DEF consists of an index and a method for using it to classify data elements. Following this method, a practitioner can tag a data element by assigning a name and numeric identifier to it according to its meaning.

The index is federated. It consists of a core index and a collection of plugins. A plugin can be developed by a standards organization, by a body of specialists in a particular area, by a user enterprise, or by a group of user enterprises.

The core index, and each plugin, is potentially infinitely extensible. The initial version of the index is defined in this standard. It will also be contained in separate files that have the structure defined in this standard and that can be added to over time.

This standard is structured as follows:

- [Chapter 1](#) of this standard contains a general introduction.
- [Chapter 2](#) contains the definitions of key terms.
- [Chapter 3](#) defines the structure of the index.
- [Chapter 4](#) describes the method for its use.
- [Chapter 5](#) specifies the syntax of the names and identifiers that can be assigned to data elements.
- [Chapter 6](#) specifies the organization and the syntax of the files containing the index.
- [Chapter 7](#) defines the initial index, comprising the initial core index and set of plugins.
- [Appendix A](#) contains the initial core index.
- [Appendix B](#) contains the O-DEF Plugin of the initial index.
- [Appendix C](#) contains the Enterprise Operation Plugin of the initial index.
- [Appendix D](#) contains the Purchase Plugin of the initial index.
- [Appendix E](#) shows the classification of some fundamental data elements commonly used by enterprises.
- [Appendix F](#) contains an example classification of the data elements of a web shopping trolley.
- [Appendix G](#) describes an example plugin containing the elements of the Dublin Core Metadata Initiative [[DCMI](#)].
- [Appendix H](#) describes an example plugin containing object classes defined by the United Nations Standard Products and Services Code [[UNSPSC](#)].
- [Appendix I](#) contains an example case study of the use of the O-DEF by a charitable foundation.
- [Appendix J](#) contains an example case study of the use of the O-DEF to classify data in status reports on electric vehicles.

Trademarks

ArchiMate®, DirecNet®, Making Standards Work®, OpenPegasus®, The Open Group®, TOGAF®, UNIX®, UNIXWARE®, X/Open®, and the Open Brand X® logo are registered trademarks and Boundaryless Information Flow™, Build with Integrity Buy with Confidence™, Dependability Through Assuredness™, FACE™, the FACE™ logo, IT4IT™, the IT4IT™ logo, O-DEF™, Open FAIR™, Open Platform 3.0™, Open Trusted Technology Provider™, Platform 3.0™, the Open O™ logo, and The Open Group Certification logo (Open O and check™) are trademarks of The Open Group.

Dublin Core® is a registered trademark of the Dublin Core Metadata Initiative.

United Nations Standard Products and Services Code® and UNSPSC® are registered trademarks of the United Nations Development Programme (UNDP).

All other brands, company, and product names are used for identification purposes only and may be trademarks that are the sole property of their respective owners.

Acknowledgements

The Open Group gratefully acknowledges the contribution of the following people in the development of this standard:

- Kary Främling, Aalto University
- Chris Harding, The Open Group
- Hrad Hekimian, Time Critical
- Nikolaos Maris, EPFL
- Arnold van Overeem, Semantix4all
- David Potter, Promise Innovation
- Ron Schuldt, Data-Harmonizing, LLC
- Roberta Shauger, Bobbi Shauger, LLC
- Christopher Voigt, IBM
- Min-Jung Yoo, EPFL

The Open Group further gratefully acknowledges the following people who assisted with the review of the standard prior to publication:

- Phil Archer, W3C
- Stuart Boardman, KPN
- Don Brancato, HPE
- TJ Virdi, The Boeing Company

Finally, the O-DEF standard was developed from the previous Universal Data Element Framework (UDEF) standards. Many people participated in their development, over a period of years, including in the development of different language versions. They are not here acknowledged individually, but their collective contribution to the present O-DEF standard is fundamental, and much appreciated.

Referenced Documents

The following documents are referenced in this standard.

(Please note that the links below are good at the time of writing but cannot be guaranteed for the future.)

Normative References

Normative references for this standard are defined in Section 1.4.

Informative References

- [C154] ArchiMate[®] Model Exchange File Format, an Open Group Standard, C154, August 2015, published by The Open Group; refer to: www.opengroup.org/bookstore/catalog/c154.htm.
- [CHEN] The Entity-Relationship Model – Toward a Unified View of Data, P. P-S Chen (1976); refer to: www.inf.unibz.it/~nutt/IDBs1011/IDBPapers/chen-ER-TODS-76.pdf.
- [DC 2012] The terms of the Dublin Core Metadata Initiative as of June 14, 2012; refer to: <http://dublincore.org/2012/06/14/dcterms.rdf>.
- [DCMI] The Dublin Core Metadata Initiative; refer to: www.dublincore.org.
- [GTIN] Global Trade Item Number; refer to: www.gs1.org/gtin.
- [HOBERMAN] Data Modeling Made Simple, S. Hoberman, Technics Publications; refer to: <https://technicspub.com/core/#ORMFundamentals>.
- [ISO 2382] ISO/IEC 2382:2015, Information Technology – Vocabulary; refer to: www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=63598.
- [ISO 11179] ISO/IEC 11179, Information Technology – Metadata Registries (MDR), Parts 1-6; refer to: www.iso.org/iso/catalogue_detail.htm?csnumber=35343.
- [O-DF] Open Data Format (O-DF), an Open Group Internet of Things (IoT) Standard, C14A, published by The Open Group, October 2014; refer to: www.opengroup.org/bookstore/catalog/c14a.htm.
- [OGPUBS] The Open Group Online Bookstore; refer to: www.opengroup.org/bookstore.
- [RDF] Resource Description Framework (RDF), World-Wide Web Consortium; refer to: www.w3.org/RDF.

- [RFC 3986] IETF RFC 3986: Uniform Resource Identifier (URI): Generic Syntax, T. Berners-Lee, R. Fielding, L. Masinter, Internet Engineering Task Force, January 2005; refer to: www.ietf.org/rfc/rfc3986.txt.
- [RFC 5013] IETF RFC 5013: The Dublin Core Metadata Element Set, J. Kunze, T. Baker, Internet Engineering Task Force, August 2007; refer to: www.ietf.org/rfc/rfc5013.txt.
- [UNSPSC] United Nations Standard Products and Services Code; refer to www.unspsc.org.

1 Introduction

1.1 Objective

The objective of this Open Group standard is to define the Open Data Element Framework (O-DEF) so that organizations can use, define, and extend its index, and practitioners can use that index to classify data elements.

1.2 Overview

The operation of an extended enterprise, or other group of collaborating enterprises, is typically supported by a number of data processing systems that exchange data. These systems may be operated by, and have been developed by, different organizations. The organizations operating and developing them generally use different terms to identify the data elements. Substantial effort is needed when programming and configuring a system to determine how data elements supplied by other systems should be interpreted. This effort is reduced by use of the O-DEF.

By using the core index and a set of plugins, and possibly by defining some of those plugins, an extended enterprise or set of collaborating enterprises simplifies and greatly reduces the cost of interface development.

Even where collaborating enterprises have not agreed to use a set of plugins specific to their purpose, their use of the core index and industry-standard plugins will make interfacing their systems much easier.

The standard is based on the concepts of data modeling (see, for example, [[CHEN](#)] and [[HOBERMAN](#)]), and of ISO/IEC 11179 [[ISO 11179](#)]. In terms of that standard, data elements are classified according to their data element concepts. The index files used in the classification are defined using the Resource Description Framework [[RDF](#)] of the World-Wide Web Consortium.

1.3 Conformance

This standard defines conformance requirements for the index of the O-DEF, and for the classification of data elements using that index.

The index and its component parts SHALL have the structure defined under The Index ([Chapter 3](#)), have the syntax defined under Classification Syntax ([Chapter 5](#)), and be contained in files with the structure and syntax defined under Index Files ([Chapter 6](#)).

Classification of data elements SHALL be carried out as described under Classification of Data Elements ([Chapter 4](#)).

1.4 Normative References

The following standards contain provisions which, through references in this standard, constitute provisions of The Open Group O-DEF standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

- [ISO 3166-1] ISO 3166-1:2013, Codes for the Representation of Names of Countries and their Subdivisions – Part 1: Country Codes; refer to:
www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=63545.
- [RDF-SYNTAX] RDF 1.1 XML Syntax, W3C Recommendation, F. Gandon, G. Schreiber, The World-Wide Web Consortium, 25 February 2014; refer to:
www.w3.org/TR/rdf-syntax-grammar.
- [RFC 4646] IETF RFC 4646: Tags for Identifying Languages; A. Phillips, M. Davis, Internet Engineering Task Force, September 2006; refer to:
www.ietf.org/rfc/rfc4646.txt.

1.5 Terminology

For the purposes of the O-DEF standard, the following terminology definitions apply:

CAN	Describes a possible feature or behavior available to the user or application.
MAY	Describes a feature or behavior that is optional. To avoid ambiguity, the opposite of “may” is expressed as “need not”, instead of “may not”.
SHALL	Describes a feature or behavior that is a requirement.
SHALL NOT	Describes a feature or behavior that is an absolute prohibition.
SHOULD	Describes a feature or behavior that is recommended but not required.

1.6 Future Directions

This standard includes the definition of an initial index. This index will be expanded over time in index files published by The Open Group and other bodies.

The classification defined in this standard can be used to describe entities and their attributes, but not relationships between entities. A future standard could define a classification for relationships between entities in a data set.

The classification defined in this standard does not include representations of data elements. A future edition of this standard, or a different future standard, could include a classification of representations.

While definition sets that are structured as arbitrary graphs can be incorporated in the index as plugins, the tagging of data elements does not then use their structure. Use of the structure of such sets may be addressed in a future version of the standard.

2 Definitions

For the purposes of this standard, the following terms and definitions apply. Merriam-Webster's Collegiate Dictionary should be referenced for terms not defined in this section.

2.1 Classification

A collection of classes in some domain, each of which has:

- A description that characterizes its members
- An identifier
- A name
- A status

2.2 Core Index

The part of the index described in this standard that is identified as the core.

2.3 Data Element

A unit of data that is considered in context to be indivisible. [[ISO 2382](#)]

Notes:

1. Data elements can be stored in databases, exchanged in messages, and manipulated by software programs. A data element could, for example, be a database field, an RDF triple, an XML attribute or atomic element, a program variable or array element, or a JSON value.
2. A data element can be of any size; for example, a bit containing a binary value, or a large file of unstructured data.

2.4 Data Element Concept

A concept that can be represented in the form of a data element described independently of any particular representation. [[ISO 11179](#)]

2.5 Node

A member of a tree.

2.6 Object Class

A set of ideas, abstractions, or things in the real world that are identified with explicit boundaries and meaning and whose properties and behavior follow the same rules. [ISO 11179]

Note: An object class is similar to an entity used by the data modeling community. An example object class is *person*. Any attribute applicable to an object class is a valid attribute for all instances of this object class.

2.7 Object Class Tree

A tree whose nodes are object classes that form a classification of objects, and whose parent-child relation is the subclass relation.

2.8 Plugin

A part of the index that can be governed as a unit separate from the core index and other plugins and that has a description, an identifier, a name, and a status.

2.9 Property

A characteristic applicable to all members of an object class. [ISO 11179]

Note: A property is similar to an *attribute* used by the data modeling community. An example property is *date of birth*.

2.10 Property Tree

A tree whose nodes are properties, whose parent-child relation is the sub-property relation, and that defines a classification of properties in which the sub-properties of each node are a class.

2.11 Relative Description

A description that is added to the description of a parent to produce the description of a child.

2.12 Relative Identifier

An identifier that is added to the identifier of a parent to produce the identifier of a child.

2.13 Relative Name

A name that is added to the name of a parent to produce the name of a child.

2.14 Role

A function that can be performed by an object in a particular context.

2.15 Role Tree

A tree whose nodes are roles, whose parent-child relation is the sub-role relation, and that defines a classification of roles in which the sub-roles of each node are a class.

2.16 Root

A node that has no parent.

Note: Each tree has exactly one root.

2.17 Status

An indication of the stage in the lifecycle of a plugin, node, context, or role that can be used to determine its validity for use in tagging in a particular context.

2.18 Tree

A designated set, with a parent-child relation defined on its members such that one member has no parent and every other member has exactly one parent.

3 The Index

3.1 Overall Structure

The index SHALL consist of a core index and a number of plugins. A plugin CAN be tree-structured or unstructured.

The core index SHALL contain one or more object class trees and one or more property trees.

A tree-structured plugin CAN contain one or more object class trees, CAN contain one or more role trees, and CAN contain one or more property trees.

An unstructured plugin CAN contain one or more object classifications, CAN contain one or more role classifications, and CAN contain one or more property classifications.

NOTE: Unstructured plugins allow the adoption as plugins of externally-defined standards that are not structured as trees.

The identifiers of the plugins in the index SHALL be distinct positive integers and they SHALL be assigned without gaps.

The names of the plugins in the index SHALL be distinct.

The descriptions of the plugins in the index SHALL be distinct and SHALL describe distinct plugins.

NOTE: Plugins outside the index can be used in the classification of data elements by arrangement between the parties concerned. Their identifiers, names, and descriptions are not constrained by this standard.

3.2 Languages

A member of a classification CAN have multiple names and multiple descriptions in different languages.

The descriptions of any classification SHOULD be semantically identical across all supported languages.

In the core index, each classification member SHALL have an English name and description, and the English description SHALL be regarded as authoritative in case of differences of interpretation between descriptions in different languages.

Each plugin SHALL have a specified language that is regarded as authoritative in case of differences of interpretation between descriptions in different languages.

3.3 Changes

A change to the index CAN add new plugins, add new classifications in existing plugins or the core index, and add new members to existing classifications.

A change to the index SHALL NOT remove any plugin, classification, or classification member, or change its identifier, or materially change its description or name. A change to the index CAN make minor editorial changes to descriptions or names.

Status values SHALL include *retired*. A change to the index CAN give a plugin, classification, or classification member a status of *retired*. This value SHALL be used to indicate that a plugin, classification, or classification member is not to be used in future classifications. It SHALL remain in the index as a retired entry, and its identifier SHALL NOT be re-used.

If a node with children is given *retired* status then its children will have *retired* status by implication.

A change to the index SHALL NOT change the status of a plugin, classification, or classification member that has a status of *retired*.

3.4 Trees

3.4.1 Node Identifiers

The identifiers of the roots of object class trees of the core index or a plugin SHALL be distinct positive integers, and they SHALL be assigned without gaps.

The identifiers of the roots of role trees of a plugin SHALL be distinct positive integers, and they SHALL be assigned without gaps.

The identifiers of the roots of property trees of the core index or a plugin SHALL be distinct positive integers, and they SHALL be assigned without gaps.

The relative identifiers of the children of any node SHALL be distinct positive integers, and they SHALL be assigned without gaps.

3.4.2 Node Names

The names of the roots that do not have *retired* status of object class trees of the core index or a plugin SHALL be distinct.

The names of the roots that do not have *retired* status of property trees of the core index or a plugin SHALL be distinct.

The names of the roots that do not have *retired* status of role trees of a plugin SHALL be distinct.

The relative names of the children that do not have *retired* status of any node SHALL be distinct.

3.4.3 Node Descriptions

The descriptions of the roots of object class trees of the core index or a plugin SHALL be distinct and SHALL characterize distinct object classes.

The descriptions of the roots of role trees of a plugin SHALL be distinct and SHALL characterize distinct roles.

The descriptions of the roots of property trees of the core index or a plugin SHALL be distinct and SHALL characterize distinct properties.

The relative descriptions of the children of any node SHALL be distinct and SHALL characterize distinct object classes, roles, or properties.

3.5 Unstructured Plugins

The identifiers of the members of an unstructured plugin SHALL all be distinct.

The names of the members that do not have *retired* status of an unstructured plugin SHALL all be distinct.

The descriptions of the members of an unstructured plugin SHALL all be distinct.

Note: The identifiers, names, and descriptions of members of unstructured plugins are not further constrained by this standard.

4 Classification of Data Elements

4.1 Introduction (Informative)

The O-DEF applies to data elements, not to instances of data elements. A data element is classified according to its data element concept. This takes account of the meaning of the data element but not its representation. For example, a piece of text can be represented using different character encodings. This difference is ignored when the data element is classified.

The terms *data element concept*, *object class*, and *property* are defined in [ISO 11179] and, in that set of standards, data element concepts are characterized by object class and property. In the O-DEF standard, roles are used as well. An O-DEF role might be considered as a particular form of ISO/IEC 11179 object class, characterized by a function that an object can perform.

4.1.1 Overall Classification

A data element represents information about some thing – its *object*. A data element SHALL be classified according to:

- Its *object class*: the kind of thing that the object is.
- Its *role*: the function that the object has.
- Its *property*: the kind of information about the object that it represents.

Every data element has an object class and a property, and may also have a role. Some data elements may just have an object class and a property. For some others, the role dominates the meaning and the object class can be neglected.

4.1.2 Process Overview

A data element SHALL be classified by first determining what the object is, and then determining the object class, role, and property.

Note: The determination of what the object is effectively assumes a data model. The data model may be explicitly documented, or may exist in the practitioner's head. Explicitly defining and documenting the data model before attempting to classify data elements generally saves substantial time and effort in the long run.

The object class, role, and property SHALL be determined by identifying sequences, each of whose members narrows the scope. The next member of a sequence is a child of the previous member, or the root of a tree-structured plugin, or a member of an unstructured plugin. If it is not a child, its description NEED NOT be a restriction of the description of the previous member. In this case, the scope SHALL be narrowed by combining its description with that of the previous member.

Note: For example, the *Type* property of the example [Dublin Core plugin](#) (see [Appendix C](#)) can have code or text values. Its description is not a restriction of the description of the *Text* property of the core index, but it can follow the core index *Text* property to narrow the scope to text properties that describe the nature or genre of a resource.

When selecting a classification member in a sequence, members with *retired* status SHALL be ignored.

4.1.3 Unclassifiable Data Elements

For some data elements, the process does not result in a classification, because the index does not contain an appropriate object class, role, or property. When a data element cannot be classified, the practitioner SHALL consider requesting a change to the core index or to a plugin, or developing a new plugin.

Note: Changes to the core index and to many plugins are likely to require considerable time. In such a case, creation of a new application-specific plugin outside the index should be considered. If appropriate, and once it has been proven in use, the plugin could be included in the index or incorporated in another plugin that is already in the index.

4.1.4 Observations (Informative)

The process appears complex as described, but can be carried out very quickly by an experienced practitioner.

The index may be large, but for most classification operations there will be just a few plugins that could be applicable, and only these need be considered when searching. Applicable plugins are determined by considering the nature of the application, the contexts of the data elements, and whether specialized knowledge is needed to understand the information. They could be in the index or outside it.

Tooling can further speed up the process by presenting the appropriate classes of object classes, roles, or properties to choose from at each stage.

4.2 Determining the Object

A data element represents information about some thing. The first step is to characterize that thing. This will often be straightforward, but the following situations need particular consideration.

4.2.1 Indirect Properties

The data element may be directly about something other than the object, and only indirectly about the object. If the thing that the data element is directly about is not important for the application in question, then the object is something else that is important to the application.

Note: A person's address is an example of this. This is directly about a location (usually the person's home), not directly about the person. It is, however, the person, not the location, that is the object.

4.2.2 Identifier Properties

Identifier properties are often used to indicate relationships between objects. This is particularly the case in relational databases. An identifier field in a database table typically identifies an object in another table. In O-DEF, an identifier is considered to be a property of the object that is identified, not of the object that the identified object relates to. The O-DEF does not provide a means of indicating relationships between objects.

Note: For example, a purchase order may contain a *customer identifier* that identifies the customer making the purchase. The object of this data element is the customer, not the purchase order.

4.3 Determining the Object Class

The object class is determined by identifying a sequence of object classes from the index, each of which successively makes the characterization of the object more precise.

The classification **SHOULD NOT** proceed beyond the point where the class of objects to which the kind of information given by the data element is relevant has been identified.

Note: For example, age is information that is relevant to all people. The classification of a data element giving the age of a person should identify the *Person* object class, but should not consider subclasses such as *Person.Male* and *Person.Female*, because the meaning of a person's age is not different for male or female persons.

The first object class is selected by reviewing the roots of the object class trees of the core index and, if none of their descriptions fits the object, then reviewing the roots of the object class trees of applicable tree-structured plugins, and then the object classes of applicable unstructured plugins.

At each stage, if the latest object class in the sequence has children, its children are reviewed and, if the description of one of them fits the object and makes the classification of the object more precise, it is identified as the next one in the sequence.

If the latest object class in the sequence does not have children, or if none of its children has a description that meets the criteria for selection, then the root object classes of applicable tree-structured plugins and then the object classes of applicable unstructured plugins are considered. Plugins that have been used already in the sequence are not applicable at this stage.

The question to ask when considering whether an object class description fits the object is: "Does the object always conform to the description?"

Note: For example, when classifying a *customer identifier* field in an enterprise database, the first step would characterize the object as a customer of the enterprise. Review of the root object classes of the core index would then consider (amongst others) the *Person* object class. Most customers of the enterprise might be people, but some might be companies. The *Person* object class is therefore not selected. (In this case, there is no root object class of the core index that characterizes the object. It is the role – customer – that characterizes what the data element is about, and the object class is neglected in this data element classification.)

4.4 Determining the Role

The role is classified by selecting a sequence of roles from the index, each of which successively makes the characterization of the role more precise.

The characterization SHALL characterize a specific role, but SHALL NOT proceed beyond the point where the meaning of the data element that is being described is not affected by further characterization of subclasses of the selected role, even though more specialized roles are available for consideration.

Notes:

1. Some roles are very imprecise. For example, a role tree might be defined with a node *Customer-Facing* that has children *Sales-Person*, *Technical-Support*, and so on. *Customer-Facing* is so imprecise as hardly to be a defined role at all. The characterization of a role should not stop at *Customer-Facing*, but should select one of the children.
2. In many enterprises, *Sales-Person* would contain a single role. Other enterprises might further sub-divide it; for example, into *On-Commission* and *Salaried*. When classifying a data element such as *commission percentage*, which clearly only applies to the specialized *On-Commission* role, that specialized role should be selected. When classifying a data element such as *sales territory*, which applies to both specialized roles, the classification should not proceed beyond the point where the generic *Sales-Person* role is identified.

The first role class is selected by reviewing the roots of the role trees of applicable tree-structured plugins and the roles of applicable unstructured plugins. If the description of one of them describes the role performed by the object, then it is selected.

At each stage, if the latest role in the sequence has children, its children are reviewed and, if the description of one of them describes the role performed by the object, and makes the characterization of the data element more precise, it is selected as the next one in the sequence.

If the latest role in the sequence does not have children, or if none of its children has a description that meets the criteria for selection, then the roots of the role trees of applicable tree-structured plugins and the roles of applicable unstructured plugins are considered. Plugins that have been used already in the sequence are not applicable at this stage.

4.5 Determining the Property

The property is classified by selecting a sequence of properties from the index, each of which successively makes the characterization of the property more precise.

The characterization SHALL characterize the property as precisely as possible.

Note: For example, the *Name* property of the core index is not very precise, when considered as the name of a person. It would be selected as the first property of the sequence, but shall if possible be further specialized by a property such as *given name*, *family name*, or *full name*, if an appropriate property is available in the core index or an applicable plugin.

Some properties cannot be indirect. If the thing that the data element is directly about is not important for the application, then the selected property SHALL be one that that can be indirect.

Notes:

1. An instance is the *Name* property, which names the object. *Mother's maiden name* provides a good illustration of a situation where *Name* should not be selected, since *Name* in this case applies to the child (direct) and not the mother (indirect). It is the child, not the mother, that is important to the application. The practitioner should select a *Text* property with the child as an indirect object rather than the *Name* property.
2. *Text.Content* and *Text.Description* are also properties that cannot be indirect.

The first property is selected by reviewing the roots of the property trees of the core index and, if none of them is appropriate, the roots of the property trees of applicable tree-structured plugins and the properties of applicable unstructured plugins. If the description of one of them covers the kind of information provided about the object, then it is selected.

At each stage, if the latest property in the sequence has children, its children are reviewed and, if the description of one of them covers the kind of information provided about the object, and makes the characterization of that information more precise, it is selected as the next one in the sequence.

If the latest property in the sequence does not have children, or if none of its children has a description that meets the criteria for selection, then the roots of the properties of applicable tree-structured plugins and the properties of applicable unstructured plugins are considered. Plugins that have been used already in the sequence are not applicable at this stage.

The question to ask when considering whether a description covers the kind of information provided about the object is: "Does the description always apply to all of the information that the data element can give about the object?". If it does not – even if it usually applies to all the information, or always applies to most of the information – the property is not selected.

Note: For example, the description "the person's family name" would not cover the information provided in a *Last-Name* field of a personnel record because in some cultures the family name does not come last.

5 Classification Syntax

A data element that has been classified as belonging to a data element concept CAN be tagged with the data element concept identifier, the data element concept name, or both. This chapter describes the syntax of data element concept identifiers and names.

5.1 Data Element Concepts

5.1.1 Data Element Concept Identifiers

The identifier of a data element concept whose classification includes both an object class and a role consists of an object class identifier, followed by a tilde and a role identifier, and then by an underscore and a property identifier. If the object class is neglected in the classification, then the object class identifier is replaced by an asterisk. If the classification does not include a role, then the tilde and the role identifier are omitted.

```
Data-Element-Concept-Id := Object-Class-Id "~" Role-Id "_" Property-Id
                          | "*"~" Role-Id "_" Property-Id | Object-Class-Id "_" Property-Id
                          | "*"_" Property-Id
```

5.1.2 Data Element Concept Names

The name of a data element concept whose classification includes both an object class and a role consists of an object class name, followed by a tilde and a role name, and then by an underscore and a property name. If the object class is neglected in the classification, then the object class name is replaced by an asterisk. If the classification does not include a role, then the tilde and the role name are omitted.

```
Data-Element-Concept-Name := Object-Class-Name "~" Role-Name "_"
                             Property-Name | "*"~" Role-Name "_" Property-Name |
                             Object-Class-Name "_" Property-Name | "*"_" Property-Name
```

5.2 Object Classes

5.2.1 Object Class Identifiers

An object class identifier consists of a core object class identifier followed by zero or more plugin object class identifiers, each of which is preceded by a semicolon, or of one or more plugin object class identifiers, each of which is preceded by a semicolon.

A core object class identifier consists of a node identifier and is the identifier of an object class node in the core index.

A plugin object class identifier consists of a plugin identifier followed by a colon and the identifier of the object class member of the plugin.

```

<Object-Class-Id> ::= <Core-Object-Class-Id>
    <Opt-Plugin-Object-Class-Ids> | <Plugin-Object-Class-Ids>
<Core-Object-Class-Id> ::= <Node-Id>
<Opt-Plugin-Object-Class-Ids> ::= "" | ";" <Plugin-Object-Class-Id>
    <Opt-Plugin-Object-Class-Ids>
<Plugin-Object-Class-Ids> ::= ";" <Plugin-Object-Class-Id>
    <Opt-Plugin-Object-Class-Ids>
<Plugin-Object-Class-Id> ::= <Plugin-Id> ":" <Plugin-Member-Id>

```

5.2.2 Object Class Names

An object class name consists of a core object class name followed by zero or more plugin object class names, each of which is preceded by a semicolon, or of one or more plugin object class names, each of which is preceded by a semicolon.

A core object class name consists of a node name and is the name of an object class node in the core index.

A plugin object class name consists of a plugin name followed by a colon and the name of the object class member of the plugin.

```

<Object-Class-Name> ::= <Core-Object-Class-Name>
    <Opt-Plugin-Object-Class-Names> | <Plugin-Object-Class-Names>
<Core-Object-Class-Name> ::= <Node-Name>
<Opt-Plugin-Object-Class-Names> ::= "" | ";" <Plugin-Object-Class-Name>
    <Opt-Plugin-Object-Class-Names>
<Plugin-Object-Class-Names> ::= ";" <Plugin-Object-Class-Name>
    <Opt-Plugin-Object-Class-Names>
<Plugin-Object-Class-Name> ::= <Plugin-Name> ":" <Plugin-Member-Name>

```

5.3 Roles

5.3.1 Role Identifiers

A role identifier consists of a core role identifier followed by zero or more plugin role identifiers, each of which is preceded by a semicolon, or of one or more plugin role identifiers, each of which is preceded by a semicolon.

A core role identifier consists of a node identifier and is the identifier of a role node in the core index.

A plugin role identifier consists of a plugin identifier followed by a colon and the identifier of the role element within the plugin.

```

<Role-Id> ::= <Core-Role-Id> <Opt-Plugin-Role-Ids>
    | <Plugin-Role-Ids>
<Core-Role-Id> ::= <Node-Id>
<Opt-Plugin-Role-Ids> ::= "" | ";" <Plugin-Role-Id>
    <Opt-Plugin-Role-Ids>
<Plugin-Role-Ids> ::= ";" <Plugin-Role-Id> <Opt-Plugin-Role-Ids>
<Plugin-Role-Id> ::= <Plugin-Id> ":" <Plugin-Member-Id>

```

5.3.2 Role Names

A role name consists of a core role name followed by zero or more plugin role names, each of which is preceded by a semicolon, or of one or more plugin role names, each of which is preceded by a semicolon.

A core role name consists of a node name and is the name of a role node in the core index.

A plugin role name consists of a plugin name followed by a colon and the name of the role element within the plugin.

```
<Role-Name> ::= <Core-Role-Name> <Opt-Plugin-Role-Names>
               | <Plugin-Role-Names>
<Core-Role-Name> ::= <Node-Name>
<Opt-Plugin-Role-Names> ::= "" | ";" <Plugin-Role-Name>
                           <Opt-Plugin-Role-Names>
<Plugin-Role-Names> ::= ";" <Plugin-Role-Name> <Opt-Plugin-Role-Names>
<Plugin-Role-Name> ::= <Plugin-Name> ":" <Plugin-Member-Name>
```

5.4 Properties

5.4.1 Property Identifiers

A property identifier consists of a core property identifier followed by zero or more plugin property identifiers, each of which is preceded by a semicolon, or of one or more plugin property identifiers, each of which is preceded by a semicolon.

A core property identifier consists of a node identifier and is the identifier of a property node in the core index.

A plugin property identifier consists of a plugin identifier followed by a colon and the identifier of the property member of the plugin.

```
<Property-Id> ::= <Core-Property-Id> <Opt-Plugin-Property-Ids> |
                  <Plugin-Property-Ids>
<Core-Property-Id> ::= <Node-Id>
<Opt-Plugin-Property-Ids> ::= "" | ";" <Plugin-Property-Id>
                           <Opt-Plugin-Property-Ids>
<Plugin-Property-Ids> ::= ";" <Plugin-Property-Id>
                           <Opt-Plugin-Property-Ids>
<Plugin-Property-Id> ::= <Plugin-Id> ":" <Plugin-Member-Id>
```

5.4.2 Property Names

A property name consists of a core property name followed by zero or more plugin property names, each of which is preceded by a semicolon, or of one or more plugin property names, each of which is preceded by a semicolon.

A core property name consists of a node name and is the name of a property node in the core index.

A plugin property name consists of a plugin name followed by a colon and the name of the property member of the plugin.

```
<Property-Name> ::= <Core-Property-Name> <Opt-Plugin-Property-Names> |  
    <Plugin-Property-Names>  
<Core-Property-Name> ::= <Node-Name>  
<Opt-Plugin-Property-Names> ::= "" | ";" <Plugin-Property-Name>  
    <Opt-Plugin-Property-Names>  
<Plugin-Property-Names> ::= ";" <Plugin-Property-Name>  
    <Opt-Plugin-Property-Names>  
<Plugin-Property-Name> ::= <Plugin-Name> ":" <Plugin-Member-Name>
```

5.5 Nodes

5.5.1 Node Identifiers

A node identifier is a root identifier followed by zero or more relative identifiers, each preceded by a period. Root identifiers and relative identifiers are positive integers.

```
<Node-Id> ::= <Root-Id> <Opt-Relative-Ids>  
<root-Id> ::= <Positive-Integer>  
<Opt-Relative-Ids> ::= "" | "." <Relative-Id> <Opt-Relative-Ids>  
<Relative-Id> ::= <Positive-Integer>
```

5.5.2 Node Names

A node name is a root name followed by zero or more relative names, each preceded by a period. Root names and relative names are name strings.

```
<Node-Name> ::= <Root-Name> <Opt-Relative-Names>  
<Root-Name> ::= <Name-String>  
<Opt-Relative-Names> ::= "" | "." <Relative-Name> <Opt-Relative-Names>  
<Relative-Name> ::= <Name-String>
```

5.6 Plugins

5.6.1 Plugin Identifiers

The identifier of a plugin in the index is a positive integer. The identifier of a plugin outside the index is a sequence of characters enclosed in brackets.

```
<Plugin-Id> ::= <Positive-Integer> | <Bracketed-String>
```

5.6.2 Plugin Names

The name of a plugin in the index is a name string. The name of a plugin outside the index is a sequence of characters enclosed in brackets.

```
<Plugin-Name> ::= <Name-String> | <Bracketed-String>
```

5.6.3 Plugin Member Identifiers

The identifier of a member of a tree-structured plugin is a node identifier. The identifier of a member of an unstructured plugin is a sequence of characters enclosed in brackets.

`<Plugin-Member-Id> ::= <Node-Id> | <Bracketed-String>`

5.6.4 Plugin Member Names

The name of a member of a tree-structured plugin is a node name. The name of a member of an unstructured plugin is a sequence of characters enclosed in brackets.

`<Plugin-Member-Name> ::= <Node-Name> | <Bracketed-String>`

5.7 Basic Components

Positive integers, name strings, and bracketed strings are basic components of identifiers and names.

5.7.1 Positive Integer

A positive integer is a sequence of Unicode decimal digit characters without a leading zero.

5.7.2 Name String

A name string is a sequence of Unicode characters that does not include any of the following:

space ! " # % ' () * + , . / : ; < = > ? \ ^ _ ` { | } ~ []

A name string that is in a language that distinguishes upper and lowercase SHALL start with an uppercase character and SHALL consist of one or more words or acronyms separated by hyphen characters (-). Each word SHALL start with an uppercase character and its other characters SHALL be lowercase or numeric. Each acronym SHALL consist of uppercase and/or numeric characters.

5.7.3 Bracketed String

A bracketed string is a sequence of Unicode characters that starts with an open-bracket character ([) and ends with a close-bracket character (]) and does not include another close-bracket character.

6 Index Files

6.1 Main Description Files

The core index SHALL be described in a main description file, and each plugin in the index SHALL be described in a main description file.

Each main description file SHALL have a primary URL at which it can be accessed.

Other descriptions using different formats CAN be provided. The main description files SHALL take precedence in case of any doubt or conflict.

The main description file of the core index SHALL use RDF/XML format as defined in [RDF-SYNTAX].

The main description file of a plugin NEED NOT use RDF/XML format.

The main description file of the core index SHALL define [RDF resources](#) (as described in [Section 6.2](#)) for use in description files of the core index and plugins.

The main description file of the core index SHALL contain a [specification of each plugin in the index](#) (see [Section 6.3](#)).

The main description file of the core index or a plugin SHALL contain a [specification of each of its object classes, roles, and properties](#) (see [Section 6.4](#)), and SHALL describe the [structure of its trees](#) (see [Section 6.5](#)).

Note: This chapter and Appendices A, F, G, and H assume that the primary URL of the main description file of the core index is `http://o-def.info/core`.

6.2 Description RDF Resources

The [table](#) ([Table 1](#)) shows the RDF resources used in core index and plugin description files.

Table 1: Description RDF Resources

Resource	URI	Description
Object Class	<code>http://o-def.info/core#ObjectClass</code>	The RDF class of O-DEF object classes.
Role	<code>http://o-def.info/core#Role</code>	The RDF class of O-DEF roles.
Property	<code>http://o-def.info/core#Property</code>	The RDF class of O-DEF properties.

Resource	URI	Description
Plugin	http://o-def.info/core#Plugin	The RDF class of O-DEF plugins.
Objects	http://o-def.info/core#Objects	The O-DEF class of all O-DEF objects.
Roles	http://o-def.info/core#Roles	The ultimately generic O-DEF role of which all O-DEF roles are specializations.
Properties	http://o-def.info/core#Properties	The ultimately generic O-DEF property of which all O-DEF properties are specializations.
identifier	http://o-def.info/core#id	The RDF property that is the relationship between something in the index and its O-DEF identifier.
name	http://o-def.info/core#name	The RDF property that is the relationship between something in the index and an O-DEF name that it has. In the case of an object class, role, or property, this is a classification-local name.
description	http://o-def.info/core#description	The RDF property that is the relationship between something in the index and an O-DEF description that it has. In the case of an object class, role, or property, this is a classification-local description.
uri	http://o-def.info/core#uri	The RDF property that is the relationship between something in the index and a URI that it has.
language	http://o-def.info/core#language	The RDF property that is the relationship between a plugin and its authoritative language.
isChild ObjectClassOf	http://o-def.info/core#isChildObjectClassOf	The RDF property that is the relationship between an object class and its parent.
isChild RoleOf	http://o-def.info/core#isChildRoleOf	The RDF property that is the relationship between a role and its parent.
isChild PropertyOf	http://o-def.info/core#isChildPropertyOf	The RDF property that is the relationship between a property and its parent.

6.3 Plugin Specifications

The specification of a plugin shall describe the plugin as an RDF resource that is of type `http://o-def.info/core#plugin` and whose URI is the primary URL of its main description file.

The specification of a plugin shall give its identifier, names, descriptions, URI, and language, using the [appropriate RDF properties](#) (see [Section 6.2](#)).

For names and descriptions not in English, the language of the name or description shall be given using the `xml:lang` XML attribute.

6.4 Object Class, Role, and Property Specifications

The specification of an object class, role, or property SHALL give its classification-local name and description. For root nodes of the core index or tree-structured plugins, and for members of unstructured plugins, these are the name and description. For child nodes of the core index or tree-structured plugins, these are the relative name and relative description.

In a file that has RDF/XML format:

- Each object class SHALL be described as an RDF resource of type `http://o-def.info/core#ObjectClass`. Its URI SHALL be the principal URL of the main description file of the core index or plugin containing it, with an added fragment identifier consisting of the string “OC” and its O-DEF identifier.
- Each role SHALL be described as an RDF resource of type `http://o-def.info/core#Role`. Its URI SHALL be the principal URL of the main description file of the plugin containing it, with an added fragment identifier consisting of the string “R” and its O-DEF identifier.
- Each property SHALL be described as an RDF resource of type `http://o-def.info/core#Property`. Its URI SHALL be the principal URL of the main description file of the core index or plugin containing it, with an added fragment identifier consisting of the string “P” and its O-DEF identifier.
- Names and descriptions SHALL be specified using the [appropriate RDF properties](#) (see [Section 6.2](#)).
- For names and descriptions in the core index that are not in English, and names and descriptions in a plugin that are not in its specified language, the language shall be given using the `xml:lang` XML attribute.

6.5 Tree Structures

The main description file of the core index or a tree-structured plugin SHALL describe the structures of its object class trees, role trees, and property trees.

6.5.1 Object Classes

A root object class SHALL be described as a child of the *Objects* resource using the *isChildObjectClassOf* RDF property.

The relationship between a child object class and its parent SHALL be described using the *isChildObjectClassOf* RDF property.

6.5.2 Roles

A root role SHALL be described as a child of the *Roles* resource using the *isChildRoleOf* RDF property.

The relationship between a child role and its parent SHALL be described using the *isChildRoleOf* RDF property.

6.5.3 Properties

A root property SHALL be described as a child of the *Properties* resource using the *isChildPropertyOf* RDF property.

The relationship between a child property and its parent SHALL be described using the *isChildPropertyOf* RDF property.

7 Initial Index

This chapter defines the initial O-DEF index, which will be extended in the course of time.

7.1 Object Classes

The object classes of the initial index SHALL be as shown in the [table](#) (Table 2).

Table 2: Object Classes

Identifier	Name	Description
1	Person	A human being, whether man, woman, or child.
1.1	Male	A male person.
1.2	Female	A female person.
2	Enterprise	A collection of people organized to achieve a common set of goals.
3	Resource	<p>A source or supply from which benefit is produced.</p> <p>Typically resources are materials, energy, services, knowledge, or other assets that are transformed, used, or consumed to produce benefit and in the process may be consumed or made unavailable.</p> <p>Note: A resource as defined here is not the same as a resource as defined in the W3C RDF standard.</p>
4	Location	<p>A place or position.</p> <p>Typically, locations provide geographic context that characterizes the data element.</p>
5	Event	An occurrence happening at a determinable time or times, with or without the participation of human agents.
6	Condition	A particular mode of being of a person or thing; existing state; situation with respect to circumstances.
7	Environment	The natural or man-made surroundings of someone or something.
8	Law-Rule	A law (natural or man-made) or policy that governs a process.
9	Process	A series of actions, changes, or functions bringing about a result.
10	Product	The result of a process.

Identifier	Name	Description
11	Substance	A material of a particular kind.
12	Information-Set	A collection of information, whether in physical form (such as a book), machine-readable form (such as a file on disc), or human-based form (such as a person's recollection or an oral tradition).

7.2 Roles

The initial index has no roles.

7.3 Properties

The properties of the initial index SHALL be as shown in the [table](#) (Table 3).

Table 3: Properties

Identifier	Name	Description
1	Identifier	A value that is intended to identify uniquely the object of the data element. Example identifiers are social security numbers, automobile Vehicle Identification Numbers (VINs), drivers' license numbers, engineering drawing numbers, and part serial numbers. A name can be an identifier in some contexts, but names are usually not used as identifiers because they are often not unique.
1.1	URI	A Uniform Resource Identifier as defined in [RFC 3986] .
2	Name	A word or phrase that refers to the object of the data element but is not intended to identify it uniquely. Name is a specialized form of <i>Text</i> . Note: Names are complicated and the full complexity is not addressed in this version of the standard.
2.1	Family	A name shared by the people in a family.
2.2	Given	A name given to a person, as distinguished from an inherited family name.
3	Code	A value from an enumerated list of possible values that is not an <i>Indicator</i> and that is not subject to continual change. Example codes are telephone area codes, postal zip codes, and country codes.
3.1	Language	A code that indicates a particular human language.
3.1.1	RFC-4646	A language tag structured and registered as described in [RFC 4646] .

Identifier	Name	Description
3.2	Country	A code that indicates a country.
3.2.1	ISO-3166	A code as defined by [ISO 3166] .
3.2.1.1	Alpha-2	A two-letter country code.
3.2.1.2	Alpha-3	A three-letter country code.
3.2.1.3	Numeric	A three-digit country code.
4	Indicator	A value that is one of only two (true or false) possible choices. Example indicators are <i>temperature high</i> , <i>pressure low</i> , and <i>power on</i> .
5	Measure	A numeric quantity that has an explicit or implied unit of measure other than a monetary unit. Example units of measure are miles, degrees centigrade, liters, and meters.
5.1	Temperature	A temperature.
5.1.1	Centigrade	A temperature measured in degrees centigrade.
5.1.2	Fahrenheit	A temperature measured in degrees Fahrenheit.
6	Quantity	An integral numeric quantity that has no unit of measurement. Examples are the quantity ordered of a product or the number of items in stock.
7	Amount	An explicit or implied monetary unit of measure. Example monetary units are Euros, British pounds, US dollars, and Canadian dollars.
8	Rate	A real (in the mathematical sense) numeric quantity, possibly expressed as a fraction, but not as a percentage. For example, a divorce rate of 10 marriages per 1000 is a <i>Rate</i> .
9	Percent	A numeric quantity expressed as a percentage. Percent is a specialized <i>Rate</i> . If the above divorce rate was expressed as one marriage per hundred, it would be a Percent. Another example Percent is an interest rate on a loan.
10	Date	A specific date in the progression of time. An example is date-of-birth expressed as year + month + day.

Identifier	Name	Description
11	Date-Time	A specific date plus time in the progression of time. An example is the date and time at which a patient's temperature is taken, expressed as year + month + day + hour + minute. Other examples might specify time more or less precisely; for example, with seconds and microseconds, or without minutes.
12	Time	A specific time within a specific timeframe. For example, day of month, hour of day, and minute of hour might be used to specify when a regular meeting repeats.
13	Text	Text that is not an identifier, name, code, indicator, measure, quantity, amount, rate, percent, date, date-time, or time. Examples are descriptions of products offered for sale, the text of contracts, and passwords.
13.1	Text.Content	The text of the object of the data element. For example, the text of a book.
13.2	Text.Description	Text that describes the object of the data element.
14	Binary	A value that is not an identifier, name, code, indicator, measure, quantity, amount, rate, percent, date, date-time, time, or text.
14.1	Binary.Graphic	A binary object that is a diagram, graph, mathematical curve, or similar representation.
14.2	Binary.Picture	A binary object that is a visual representation of a person, object, or scene.
14.3	Binary.Sound	A binary object that is capable of creating audio.
14.4	Binary.Video	A binary object that is capable of creating motion pictures.

7.4 Plugins

The plugins of the initial index SHALL be as shown in the [table](#) (Table 4).

Table 4: Plugins

Identifier	Name	Description
1	O-DEF	Metadata used to describe the O-DEF itself.
2	Enterprise-Operation	Metadata used to describe the basic operation of an enterprise.
3	Purchase	Metadata used to describe a purchase of goods or services.

7.4.1 The O-DEF Plugin

Authoritative Language

The authoritative language of the O-DEF plugin SHALL be English.

Object Classes

The object classes of the O-DEF plugin of the initial index SHALL be as shown in the [table \(Table 5\)](#).

Table 5: O-DEF Plugin Object Classes

Identifier	Name	Description
1	Identified-Resource	An O-DEF plugin, object class, role, or property, inside or outside the index, that has an identifier.
1.1	Plugin	A plugin.
1.2	Object-Class	An object class.
1.3	Role	A role.
1.4	Property	A property.

Roles

The O-DEF plugin has no roles.

Properties

The properties of the O-DEF plugin of the initial index SHALL be as shown in the [table \(Table 6\)](#).

Table 6: O-DEF Plugin Properties

Identifier	Name	Description
1	Plugin-Id	The O-DEF identifier of a plugin, inside or outside the index.
2	Member-Id	The identifier of a classification member that is unique within its classification, inside or outside the index.
3	Member-Local-Id	The identifier of a classification member that is unique within a level of the classification, inside or outside the index. This can be: the identifier of a root object class, role, or property; the relative identifier of a child object class, role, or property; or the identifier of a member of an unstructured plugin.

Identifier	Name	Description
4	Plugin-Name	The name of an O-DEF plugin, inside or outside the index.
5	Member-Name	The name of a classification member within its classification, inside or outside the index.
6	Member-Local-Name	The name of a classification member within a level of the classification, inside or outside the index. This can be: the name of a root object class, role, or property; the relative name of a child object class, role, or property; or the name of a member of an unstructured plugin.
7	Plugin-Description	The description of an O-DEF plugin, inside or outside the index.
8	Member-Description	The description of a classification member within its classification, inside or outside the index.
9	Member-Local-Description	The description of a classification member within a level of the classification, inside or outside the index. This can be: the description of a root object class, role, or property; the relative description of a child object class, role, or property; or the description of a member of an unstructured plugin.

7.4.2 The Enterprise Operation Plugin

Authoritative Language

The authoritative language of the Enterprise Operation plugin SHALL be English.

Object Classes

The Enterprise Operation plugin has no object classes.

Roles

The roles of the Enterprise-Operation plugin of the initial index SHALL be as shown in the [table \(Table 7\)](#).

Table 7: Enterprise-Operation Roles

Identifier	Name	Description
1	Customer	A person or enterprise to which the enterprise has supplied, supplies, or may in future supply goods or services.
2	Supplier	An enterprise that has supplied, supplies, or may in future supply goods or services to the enterprise.

Identifier	Name	Description
3	Employee	A person employed by the enterprise.
4	Contractor	A person, other than an employee, who is engaged by the enterprise to carry out work.
5	Manager	A person with responsibility for managing employees, contractors, or assets of the enterprise to achieve enterprise objectives.
6	Asset	A resource, other than human, which is used, consumed, or available for use or consumption by the enterprise.
7	Product	A product that is provided by the enterprise and is intended to be used or consumed by entities outside of the enterprise.

Properties

The Enterprise-Operation plugin has no properties.

7.4.3 The Purchase Plugin

Authoritative Language

The authoritative language of the Purchase plugin SHALL be English.

Object Classes

The object classes of the Purchase plugin of the initial index SHALL be as shown in the [table \(Table 9\)](#).

Table 8: Purchase Object Classes

Identifier	Name	Description
1	Purchase-Order	A purchaser's request to purchase.
2	Line-Item	A good or service, or set of goods or services of the same kind, to be purchased.
3	Discount	A price reduction that applies to a purchase.
4	Conditions-Of-Sale	A Law-Rule defined by a vendor to govern purchases.

Roles

The roles of the Purchase plugin of the initial index SHALL be as shown in the [table](#) (Table 9).

Table 9: Purchase Roles

Identifier	Name	Description
1	Purchaser	The person or enterprise making the purchase.
2	Vendor	The person or enterprise supplying the purchased goods or services.
3	Approver	A person assigned by an enterprise that is a purchaser to approve a purchase on its behalf.

Properties

The properties of the Purchase plugin of the initial index SHALL be as shown in the [table](#) (Table 10).

Table 10: Purchase Properties

Identifier	Name	Description
1	Purchase-Order-Number	An identifier of the purchase order assigned by the purchaser.
2	Order-Number	An identifier of the purchase order assigned by the vendor.
3	Number-Of-Units	The quantity of units ordered of a line item.
4	Unit-Price	The amount charged by the vendor per unit for a line item.
5	Total-Price	A total amount charged by the vendor for a number of items.
6	Discount	An amount or percentage by which the vendor reduces a price.

A Initial Core Main Description File

This appendix shows the main description file of the initial core index. In case of any conflict with the provisions in the body of this standard, those provisions take precedence over this appendix.

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:odef="http://o-def.info/core#">

  <!-- DESCRIPTION RDF RESOURCES -->

  <!-- ObjectClass -->
    <rdf:Description rdf:about="http://o-def.info/core#ObjectClass">
      <rdf:type rdf:resource=
        "http://www.w3.org/2000/01/rdf-schema#Class"/>
    </rdf:Description>

  <!-- Role -->
    <rdf:Description rdf:about="http://o-def.info/core#Role">
      <rdf:type rdf:resource=
        "http://www.w3.org/2000/01/rdf-schema#Class"/>
    </rdf:Description>

  <!-- Property -->
    <rdf:Description rdf:about="http://o-def.info/core#Property">
      <rdf:type rdf:resource=
        "http://www.w3.org/2000/01/rdf-schema#Class"/>
    </rdf:Description>

  <!-- Plugin -->
    <rdf:Description rdf:about="http://o-def.info/core#Plugin">
      <rdf:type rdf:resource=
        "http://www.w3.org/2000/01/rdf-schema#Class"/>
    </rdf:Description>

  <!-- Objects -->
    <rdf:Description rdf:about="http://o-def.info/core#Objects">
      <rdf:type rdf:resource="http://o-def.info/core/ObjectClass"/>
    </rdf:Description>

  <!-- Roles -->
    <rdf:Description rdf:about="http://o-def.info/core#Roles">
      <rdf:type rdf:resource="http://o-def.info/core/Role"/>
    </rdf:Description>

  <!-- Properties -->
    <rdf:Description rdf:about="http://o-def.info/core#Properties">
```

```

        <rdf:type rdf:resource="http://o-def.info/core/Property"/>
    </rdf:Description>

<!-- id -->
    <rdf:Description rdf:about="http://o-def.info/core#id">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- name -->
    <rdf:Description rdf:about="http://o-def.info/core#name">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- description -->
    <rdf:Description rdf:about="http://o-def.info/core#description">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- uri -->
    <rdf:Description rdf:about="http://o-def.info/core#uri">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- language -->
    <rdf:Description rdf:about="http://o-def.info/core#language">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- isChildObjectClassOf -->
    <rdf:Description rdf:about=
        "http://o-def.info/core#isChildObjectClassOf">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- isChildRoleOf -->
    <rdf:Description rdf:about="http://o-def.info/core#isChildRoleOf">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- isChildPropertyOf -->
    <rdf:Description rdf:about=
        "http://o-def.info/core#isChildPropertyOf">
        <rdf:type rdf:resource=
            "http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    </rdf:Description>

<!-- PLUGINS -->

```

```

<!-- O-DEF -->
  <rdf:Description rdf:about="http://o-def.info/plugins/1">
    <rdf:type rdf:resource="http://o-def.info/core#Plugin"/>
    <odef:id>1</odef:id>
    <odef:name>O-DEF</odef:name>
    <odef:description>Metadata used to describe the
      O-DEF itself.</odef:description>
    <odef:uri>http://o-def.info/plugins/1</odef:uri>
    <odef:language>EN</odef:language>
  </rdf:Description>

<!-- Enterprise-Operation -->
  <rdf:Description rdf:about="http://o-def.info/plugins/2">
    <rdf:type rdf:resource="http://o-def.info/core#Plugin"/>
    <odef:id>2</odef:id>
    <odef:name>Enterprise-Operation</odef:name>
    <odef:description>Metadata used to describe the basic
      operation of an enterprise.</odef:description>
    <odef:uri>http://o-def.info/plugins/2</odef:uri>
    <odef:language>EN</odef:language>
  </rdf:Description>

<!-- Purchase -->
  <rdf:Description rdf:about="http://o-def.info/plugins/3">
    <rdf:type rdf:resource="http://o-def.info/core#Plugin"/>
    <odef:id>3</odef:id>
    <odef:name>Purchase</odef:name>
    <odef:description>Metadata used to describe a purchase
      of goods or services.</odef:description>
    <odef:uri>http://o-def.info/plugins/3</odef:uri>
    <odef:language>EN</odef:language>
  </rdf:Description>

<!-- OBJECT CLASSES -->
<!-- Person -->
  <rdf:Description rdf:about="http://o-def.info/core#OC1">
    <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
    <odef:isChildObjectClassOf rdf:resource=
      "http://o-def.info/core#Objects"/>
    <odef:name>Person</odef:name>
    <odef:description>A human being, whether man, woman,
      or child.</odef:description>
  </rdf:Description>

<!-- Male Person -->
  <rdf:Description rdf:about="http://o-def.info/core#OC1.1">
    <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
    <odef:isChildObjectClassOf rdf:resource=
      "http://o-def.info/core#OC1"/>
    <odef:name>Male</odef:name>
    <odef:description>A male person.</odef:description>
  </rdf:Description>

<!-- Female Person -->

```

```

    <rdf:Description rdf:about="http://o-def.info/core#OC1.2">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/core#OC1"/>
      <odef:name>Female</odef:name>
      <odef:description>A female person.</odef:description>
    </rdf:Description>

<!-- Enterprise -->
    <rdf:Description rdf:about="http://o-def.info/core#OC2">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/core#Objects"/>
      <odef:name>Enterprise</odef:name>
      <odef:description>A collection of people organized to achieve
        a common set of goals.</odef:description>
    </rdf:Description>

<!-- Resource -->
    <rdf:Description rdf:about="http://o-def.info/core#OC3">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/core#Objects"/>
      <odef:name>Resource</odef:name>
      <odef:description>A source or supply from which benefit is
        produced. Typically resources are materials, energy,
        services, knowledge, or other assets that are transformed,
        used or consumed to produce benefit and in the process may
        be consumed or made unavailable.</odef:description>
    </rdf:Description>

<!-- Location -->
    <rdf:Description rdf:about="http://o-def.info/core#OC4">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/core#Objects"/>
      <odef:name>Location</odef:name>
      <odef:description>A place or position. Typically locations
        provide geographic context that characterizes the data
        element.</odef:description>
    </rdf:Description>

<!-- Event -->
    <rdf:Description rdf:about="http://o-def.info/core#OC5">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/core#Objects"/>
      <odef:name>Event</odef:name>
      <odef:description>An occurrence happening at a determinable
        time or times, with or without the participation of
        human agents.</odef:description>
    </rdf:Description>

<!-- Condition -->

```

```

<rdf:Description rdf:about="http://o-def.info/core#OC6">
  <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
  <odef:isChildObjectClassOf rdf:resource=
    "http://o-def.info/core#Objects"/>
  <odef:name>Condition</odef:name>
  <odef:description>A particular mode of being of a person or
    thing; existing state; situation with respect to
    circumstances.</odef:description>
</rdf:Description>

<!-- Environment -->
<rdf:Description rdf:about="http://o-def.info/core#OC7">
  <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
  <odef:isChildObjectClassOf rdf:resource=
    "http://o-def.info/core#Objects"/>
  <odef:name>Environment</odef:name>
  <odef:description>The natural or man-made surroundings of
    someone or something.</odef:description>
</rdf:Description>

<!-- Law-Rule -->
<rdf:Description rdf:about="http://o-def.info/core#OC8">
  <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
  <odef:isChildObjectClassOf rdf:resource=
    "http://o-def.info/core#Objects"/>
  <odef:name>Law-Rule</odef:name>
  <odef:description>A law (natural or man-made) or policy that
    governs a process.</odef:description>
</rdf:Description>

<!-- Process -->
<rdf:Description rdf:about="http://o-def.info/core#OC9">
  <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
  <odef:isChildObjectClassOf rdf:resource=
    "http://o-def.info/core#Objects"/>
  <odef:name>Process</odef:name>
  <odef:description>A series of actions, changes, or functions
    bringing about a result.</odef:description>
</rdf:Description>

<!-- Product -->
<rdf:Description rdf:about="http://o-def.info/core#OC10">
  <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
  <odef:isChildObjectClassOf rdf:resource=
    "http://o-def.info/core#Objects"/>
  <odef:name>Product</odef:name>
  <odef:description>The result of a process.</odef:description>
</rdf:Description>

<!-- Substance -->
<rdf:Description rdf:about="http://o-def.info/core#OC11">
  <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
  <odef:isChildObjectClassOf rdf:resource=
    "http://o-def.info/core#Objects"/>

```



```

        <odef:name>Substance</odef:name>
        <odef:description>A material of a particular
            kind.</odef:description>
    </rdf:Description>

<!-- Information-Set -->
    <rdf:Description rdf:about="http://o-def.info/core#OC12">
        <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
        <odef:isChildObjectClassOf rdf:resource=
            "http://o-def.info/core#Objects"/>
        <odef:name>Information-Set</odef:name>
        <odef:description>A collection of information, whether in
            physical form (such as a book), machine-readable form
            (such as a file on disc), or human-based form
            (such as a person's recollection or an oral
            tradition).</odef:description>
    </rdf:Description>

<!-- PROPERTIES -->
<!-- Identifier -->
    <rdf:Description rdf:about="http://o-def.info/core#P1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Identifier</odef:name>
        <odef:description>A value that is intended to identify
            uniquely the object of the data element.
            Example identifiers are social security numbers,
            automobile Vehicle Identification Numbers (VINs), drivers'
            license numbers, engineering drawing numbers, and part
            serial numbers.
            A name can be an identifier in some contexts, but names
            are usually not used as identifiers because they are often
            not unique.</odef:description>
    </rdf:Description>

<!-- URI -->
    <rdf:Description rdf:about="http://o-def.info/core#P1.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P1"/>
        <odef:name>URI</odef:name>
        <odef:description>A Uniform Resource Identifier as defined
            in RFC 3986.</odef:description>
    </rdf:Description>

<!-- Name -->
    <rdf:Description rdf:about="http://o-def.info/core#P2">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Name</odef:name>
        <odef:description>A word or phrase that refers to the object
            of the data element but is not intended to identify

```

```

        it uniquely.
        Name is a specialized form of Text.</odef:description>
    </rdf:Description>

<!-- Family Name -->
    <rdf:Description rdf:about="http://o-def.info/core#P2.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P2"/>
        <odef:name>Family</odef:name>
        <odef:description>A name shared by the people in a
            family.</odef:description>
    </rdf:Description>

<!-- Given Name -->
    <rdf:Description rdf:about="http://o-def.info/core#P2.2">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P2"/>
        <odef:name>Given</odef:name>
        <odef:description>A name given to a person, as distinguished
            from an inherited family name.</odef:description>
    </rdf:Description>

<!-- Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Code</odef:name>
        <odef:description>A value from an enumerated list of possible
            values that is not an Indicator and that is not subject to
            continual change.
            Example codes are telephone area codes, postal zip codes,
            and country codes.</odef:description>
    </rdf:Description>

<!-- Language Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3"/>
        <odef:name>Language</odef:name>
        <odef:description>A code that indicates a particular human
            language.</odef:description>
    </rdf:Description>

<!-- RFC-4646 Language Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.1.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3.1"/>
        <odef:name>RFC-4646</odef:name>
        <odef:description>A language tag structured and registered

```

```

        as described in RFC 4646.</odef:description>
    </rdf:Description>

<!-- Country Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.2">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3"/>
        <odef:name>Country</odef:name>
        <odef:description>A code that indicates a
            country.</odef:description>
    </rdf:Description>

<!-- ISO-3166 Country Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.2.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3.2"/>
        <odef:name>ISO-3166</odef:name>
        <odef:description>A code as defined by
            ISO 3166.</odef:description>
    </rdf:Description>

<!-- Alpha-2 ISO-3166 Country Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.2.1.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3.2.1"/>
        <odef:name>Alpha-2</odef:name>
        <odef:description>A two-letter country code.</odef:description>
    </rdf:Description>

<!-- Alpha-3 ISO-3166 Country Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.2.1.2">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3.2.1"/>
        <odef:name>Alpha-3</odef:name>
        <odef:description>A three-letter country
            code.</odef:description>
    </rdf:Description>

<!-- Numeric ISO-3166 Country Code -->
    <rdf:Description rdf:about="http://o-def.info/core#P3.2.1.3">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P3.2.1"/>
        <odef:name>Numeric</odef:name>
        <odef:description>A three-digit country
            code.</odef:description>
    </rdf:Description>

<!-- Indicator -->
    <rdf:Description rdf:about="http://o-def.info/core#P4">

```

```

        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Indicator</odef:name>
        <odef:description>A value that is one of only two
            (true or false) possible choices.
            Example indicators are temperature high, pressure low,
            and power on.</odef:description>
    </rdf:Description>

<!-- Measure -->
    <rdf:Description rdf:about="http://o-def.info/core#P5">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Measure</odef:name>
        <odef:description>A numeric quantity that has an explicit
            or implied unit of measure other than a monetary unit.
            Example units of measure are miles, degrees centigrade,
            liters, and meters.</odef:description>
    </rdf:Description>

<!-- Temperature Measure -->
    <rdf:Description rdf:about="http://o-def.info/core#5.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#5"/>
        <odef:name>Temperature</odef:name>
        <odef:description>A temperature.</odef:description>
    </rdf:Description>

<!-- Centigrade Temperature Measure -->
    <rdf:Description rdf:about="http://o-def.info/core#5.1.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#5.1"/>
        <odef:name>Centigrade</odef:name>
        <odef:description>A temperature measured in degrees
            centigrade.</odef:description>
    </rdf:Description>

<!-- Fahrenheit Temperature Measure -->
    <rdf:Description rdf:about="http://o-def.info/core#5.1.2">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#5.1"/>
        <odef:name>Fahrenheit</odef:name>
        <odef:description>A temperature measured in degrees
            Fahrenheit.</odef:description>
    </rdf:Description>

<!-- Quantity -->
    <rdf:Description rdf:about="http://o-def.info/core#P6">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>

```

```

        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Quantity</odef:name>
        <odef:description>An integral numeric quantity that has no
            unit of measurement.
            Examples are the quantity ordered of a product or the
            number of items in stock.</odef:description>
    </rdf:Description>

<!-- Amount -->
    <rdf:Description rdf:about="http://o-def.info/core#P7">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Amount</odef:name>
        <odef:description>An explicit or implied monetary unit of
            measure.
            Example monetary units are Euros, British pounds, US
            dollars, and Canadian dollars.</odef:description>
    </rdf:Description>

<!-- Rate -->
    <rdf:Description rdf:about="http://o-def.info/core#P8">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Rate</odef:name>
        <odef:description>A real (in the mathematical sense) numeric
            quantity, possibly expressed as a fraction, but not as a
            percentage.
            For example, a divorce rate of 10 marriages per 1000 is a
            Rate.</odef:description>
    </rdf:Description>

<!-- Percent -->
    <rdf:Description rdf:about="http://o-def.info/core#P9">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Percent</odef:name>
        <odef:description>A numeric quantity expressed as a percentage.
            Percent is a specialized Rate. If the above divorce rate
            was expressed as one marriage per hundred, it would be a
            Percent. Another example Percent is an interest rate on a
            loan.</odef:description>
    </rdf:Description>

<!-- Date -->
    <rdf:Description rdf:about="http://o-def.info/core#P10">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Date</odef:name>
        <odef:description>A specific date in the progression of time.

```

```

        An example is date-of-birth expressed as year + month +
        day.</odef:description>
    </rdf:Description>

<!-- Date-Time -->
    <rdf:Description rdf:about="http://o-def.info/core#P11">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Date-Time</odef:name>
        <odef:description>A specific date plus time in the progression
            of time.
            An example is the date and time at which a patient's
            temperature is taken, expressed as year + month + day +
            hour + minute. Other examples might specify time more or
            less precisely; for example, with seconds and microseconds,
            or without minutes.</odef:description>
    </rdf:Description>

<!-- Time -->
    <rdf:Description rdf:about="http://o-def.info/core#P12">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Time</odef:name>
        <odef:description>A specific time within a specific timeframe.
            For example, day of month, hour of day, and minute of hour
            might be used to specify when a regular meeting
            repeats.</odef:description>
    </rdf:Description>

<!-- Text -->
    <rdf:Description rdf:about="http://o-def.info/core#P13">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Text</odef:name>
        <odef:description>Text that is not an identifier, name, code,
            indicator, measure, quantity, amount, rate, percent, date,
            date-time, or time.
            Examples are descriptions of products offered for sale, the
            text of contracts, and passwords.</odef:description>
    </rdf:Description>

<!-- Text.Content -->
    <rdf:Description rdf:about="http://o-def.info/core#P13.1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#P13"/>
        <odef:name>Content</odef:name>
        <odef:description>The text of the object of the data element.
            For example, the text of a book.</odef:description>
    </rdf:Description>

```

```

<!-- Description Text -->
  <rdf:Description rdf:about="http://o-def.info/core#P13.2">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#P13"/>
    <odef:name>Description</odef:name>
    <odef:description>Text that describes the object of the data
      element.</odef:description>
  </rdf:Description>

<!-- Binary -->
  <rdf:Description rdf:about="http://o-def.info/core#P14">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#Properties"/>
    <odef:name>Binary</odef:name>
    <odef:description>A value that is not an identifier, name,
      code, indicator, measure, quantity, amount, rate, percent,
      date, date-time, time, or text.</odef:description>
  </rdf:Description>

<!-- Graphic Binary -->
  <rdf:Description rdf:about="http://o-def.info/core#P14.1">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#P14"/>
    <odef:name>Graphic</odef:name>
    <odef:description>A binary object that is a diagram, graph,
      mathematical curve, or similar
      representation.</odef:description>
  </rdf:Description>

<!-- Picture Binary -->
  <rdf:Description rdf:about="http://o-def.info/core#P14.2">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#P14"/>
    <odef:name>Picture</odef:name>
    <odef:description>A binary object that is a visual
      representation of a person, object, or
      scene.</odef:description>
  </rdf:Description>

<!-- Sound Binary -->
  <rdf:Description rdf:about="http://o-def.info/core#P14.3">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#P14"/>
    <odef:name>Sound</odef:name>
    <odef:description>A binary object that is capable of creating
      audio.</odef:description>
  </rdf:Description>

<!-- Video Binary -->

```

```
<rdf:Description rdf:about="http://o-def.info/core#P14.4">
  <rdf:type rdf:resource="http://o-def.info/core#Property"/>
  <odef:isChildPropertyOf rdf:resource=
    "http://o-def.info/core#P14"/>
  <odef:name>Video</odef:name>
  <odef:description>A binary object that is capable of creating
    motion pictures.</odef:description>
</rdf:Description>

</rdf:RDF>
```


B O-DEF Plugin Main Description File

This appendix shows the main description file of the O-DEF Plugin in the initial index. In case of any conflict with the provisions in the body of this standard, those provisions take precedence over this appendix.

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:odef="http://o-def.info/core#"

  <!-- OBJECT CLASSES -->
  <!-- Identified-Resource -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#OC1">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/core#Objects"/>
      <odef:name>Identified-Resource</odef:name>
      <odef:description>An O-DEF plugin, object class, role, or
        property, inside or outside the index, that has an
        identifier.</odef:description>
    </rdf:Description>

    <!-- Plugin Identified-Resource -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#OC1.1">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/plugins/1#OC1"/>
      <odef:name>Plugin</odef:name>
      <odef:description>A plugin.</odef:description>
    </rdf:Description>

    <!-- ObjectClass Identified-Resource -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#OC1.2">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/plugins/1#OC1"/>
      <odef:name>Object-Class</odef:name>
      <odef:description>An object class.</odef:description>
    </rdf:Description>

    <!-- Role Identified-Resource -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#OC1.3">
      <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
      <odef:isChildObjectClassOf rdf:resource=
        "http://o-def.info/plugins/1#OC1"/>
      <odef:name>Role</odef:name>
      <odef:description>A role.</odef:description>
    </rdf:Description>
```

```

<!-- Property Identified-Resource -->
  <rdf:Description rdf:about="http://o-def.info/plugins/1#OC1.4">
    <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
    <odef:isChildObjectClassOf rdf:resource=
      "http://o-def.info/plugins/1#OC1"/>
    <odef:name>Property</odef:name>
    <odef:description>A property.</odef:description>
  </rdf:Description>

<!-- PROPERTIES -->
<!-- Plugin-Id -->
  <rdf:Description rdf:about="http://o-def.info/plugins/1#P1">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#Properties"/>
    <odef:name>Plugin-Id</odef:name>
    <odef:description>The O-DEF identifier of a plugin, inside
      or outside the index.</odef:description>
  </rdf:Description>

<!-- Member-Id -->
  <rdf:Description rdf:about="http://o-def.info/plugins/1#P2">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#Properties"/>
    <odef:name>Member-Id</odef:name>
    <odef:description>The identifier of a classification member
      that is unique within its classification, inside or
      outside the index.</odef:description>
  </rdf:Description>

<!-- Member-Local-Id -->
  <rdf:Description rdf:about="http://o-def.info/plugins/1#P3">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#Properties"/>
    <odef:name>Member-Local-Id</odef:name>
    <odef:description>The identifier of a classification member
      that is unique within a level of the classification,
      inside or outside the index. This can be: the identifier
      of a root object class, role, or property; the relative
      identifier of a child object class, role, or property;
      or the identifier of a member of an unstructured
      plugin.</odef:description>
  </rdf:Description>

<!-- Plugin-Name -->
  <rdf:Description rdf:about="http://o-def.info/plugins/1#P4">
    <rdf:type rdf:resource="http://o-def.info/core#Property"/>
    <odef:isChildPropertyOf rdf:resource=
      "http://o-def.info/core#Properties"/>
    <odef:name>Plugin-Name</odef:name>
    <odef:description>The name of an O-DEF plugin, inside or

```

```

        outside the index.</odef:description>
    </rdf:Description>

<!-- Member-Name -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#P5">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Member-Name</odef:name>
        <odef:description>The name of a classification member within
            its classification, inside or outside the
            index.</odef:description>
    </rdf:Description>

<!-- Member-Local-Name -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#P6">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Member-Local-Name</odef:name>
        <odef:description>The name of a classification member within
            a level of the classification, inside or outside the index.
            This can be: the name of a root object class, role, or
            property; the relative name of a child object class, role,
            or property; or the name of a member of an unstructured
            plugin.</odef:description>
    </rdf:Description>

<!-- Plugin-Description -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#P7">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Plugin-Description</odef:name>
        <odef:description>The description of an O-DEF plugin, inside
            or outside the index.</odef:description>
    </rdf:Description>

<!-- Member-Description -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#P8">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Member-Description</odef:name>
        <odef:description>The description of a classification member
            within its classification, inside or outside the
            index.</odef:description>
    </rdf:Description>

<!-- Member-Local-Description -->
    <rdf:Description rdf:about="http://o-def.info/plugins/1#P9">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>

```

```
<odef:name>Member-Local-Description</odef:name>
<odef:description>The description of a classification member
  within a level of the classification, inside or outside
  the index. This can be: the description of a root object
  class, role, or property; the relative description of a
  child object class, role, or property; or the description
  of a member of an unstructured plugin.</odef:description>
</rdf:Description>

</rdf:RDF>
```

C Enterprise Operation Plugin Main Description File

This appendix shows the main description file of the Enterprise Operation Plugin in the initial index. In case of any conflict with the provisions in the body of this standard, those provisions take precedence over this appendix.

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:odef="http://o-def.info/core#">

  <!-- ROLES -->
  <!-- Customer -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R1">
      <rdf:type rdf:resource="http://o-def.info/core#Role"/>
      <odef:isChildRoleOf rdf:resource=
        "http://o-def.info/core#Roles"/>
      <odef:name>Customer</odef:name>
      <odef:description>A person or enterprise to which the
        enterprise has supplied, supplies, or may in future
        supply goods or services.</odef:description>
    </rdf:Description>

  <!-- Supplier -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R2">
      <rdf:type rdf:resource="http://o-def.info/core#Role"/>
      <odef:isChildRoleOf rdf:resource=
        "http://o-def.info/core#Roles"/>
      <odef:name>Supplier</odef:name>
      <odef:description>An enterprise that has supplied, supplies,
        or may in future supply goods or services to the
        enterprise.</odef:description>
    </rdf:Description>

  <!-- Employee -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R3">
      <rdf:type rdf:resource="http://o-def.info/core#Role"/>
      <odef:isChildRoleOf rdf:resource=
        "http://o-def.info/core#Roles"/>
      <odef:name>Employee</odef:name>
      <odef:description>A person employed by the
        enterprise.</odef:description>
    </rdf:Description>

  <!-- Contractor -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R4">
      <rdf:type rdf:resource="http://o-def.info/core#Role"/>
      <odef:isChildRoleOf rdf:resource=
        "http://o-def.info/core#Roles"/>
```

```

        <odef:name>Contractor</odef:name>
        <odef:description>A person, other than an employee, who is
            engaged by the enterprise to carry out
            work.</odef:description>
    </rdf:Description>

<!-- Manager -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R5">
        <rdf:type rdf:resource="http://o-def.info/core#Role"/>
        <odef:isChildRoleOf rdf:resource=
            "http://o-def.info/core#Roles"/>
        <odef:name>Manager</odef:name>
        <odef:description>A person with responsibility for managing
            employees, contractors, or assets of the enterprise to
            achieve enterprise objectives.</odef:description>
    </rdf:Description>

<!-- Asset -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R6">
        <rdf:type rdf:resource="http://o-def.info/core#Role"/>
        <odef:isChildRoleOf rdf:resource=
            "http://o-def.info/core#Roles"/>
        <odef:name>Asset</odef:name>
        <odef:description>A resource, other than human, which is used,
            consumed, or available for use or consumption by the
            enterprise.</odef:description>
    </rdf:Description>

<!-- Product -->
    <rdf:Description rdf:about="http://o-def.info/plugins/2#R7">
        <rdf:type rdf:resource="http://o-def.info/core#Role"/>
        <odef:isChildRoleOf rdf:resource=
            "http://o-def.info/core#Roles"/>
        <odef:name>Product</odef:name>
        <odef:description>A product that is provided by the enterprise
            and is intended to be used or consumed by entities outside
            of the enterprise.</odef:description>
    </rdf:Description>

</rdf:RDF>

```

D Purchase Plugin Main Description File

This appendix shows the main description file of the Purchase Plugin in the initial index. In case of any conflict with the provisions in the body of this standard, those provisions take precedence over this appendix.

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:odef="http://o-def.info/core#"

    <!-- OBJECT CLASSES -->
    <!-- Purchase-Order -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#OC1">
        <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
        <odef:isChildObjectClassOf rdf:resource=
            "http://o-def.info/core#Objects"/>
        <odef:name>Purchase-Order</odef:name>
        <odef:description>A purchaser's request to
            purchase.</odef:description>
    </rdf:Description>

    <!-- Line-Item -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#OC2">
        <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
        <odef:isChildObjectClassOf rdf:resource=
            "http://o-def.info/core#Objects"/>
        <odef:name>Line-Item</odef:name>
        <odef:description>A good or service, or set of goods or
            services of the same kind, to be
            purchased.</odef:description>
    </rdf:Description>

    <!-- Discount -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#OC3">
        <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
        <odef:isChildObjectClassOf rdf:resource=
            "http://o-def.info/core#Objects"/>
        <odef:name>Discount</odef:name>
        <odef:description>A price reduction that applies to a
            purchase.</odef:description>
    </rdf:Description>

    <!-- Conditions-Of-Sale -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#OC4">
        <rdf:type rdf:resource="http://o-def.info/core#ObjectClass"/>
        <odef:isChildObjectClassOf rdf:resource=
            "http://o-def.info/core#Objects"/>
        <odef:name>Conditions-Of-Sale</odef:name>
```

```

        <odef:description>A Law-Rule defined by a vendor to govern
        purchases.</odef:description>
    </rdf:Description>

<!-- ROLES -->
<!-- Purchaser -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#R1">
        <rdf:type rdf:resource="http://o-def.info/core#Role"/>
        <odef:isChildRoleOf rdf:resource=
            "http://o-def.info/core#Roles"/>
        <odef:name>Purchaser</odef:name>
        <odef:description>The person or enterprise making the
            purchase.</odef:description>
    </rdf:Description>

<!-- Vendor -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#R2">
        <rdf:type rdf:resource="http://o-def.info/core#Role"/>
        <odef:isChildRoleOf rdf:resource=
            "http://o-def.info/core#Roles"/>
        <odef:name>Vendor</odef:name>
        <odef:description>The person or enterprise supplying the
            purchased goods or services.</odef:description>
    </rdf:Description>

<!-- Approver -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#R3">
        <rdf:type rdf:resource="http://o-def.info/core#Role"/>
        <odef:isChildRoleOf rdf:resource=
            "http://o-def.info/core#Roles"/>
        <odef:name>Approver</odef:name>
        <odef:description> A person assigned by an enterprise that
            is a purchaser to approve a purchase on its
            behalf.</odef:description>
    </rdf:Description>

<!-- PROPERTIES -->
<!-- Purchase-Order-Number -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#P1">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Purchase-Order-Number</odef:name>
        <odef:description>An identifier of the purchase order assigned
            by the purchaser.</odef:description>
    </rdf:Description>

<!-- Order-Number -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#P2">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Order-Number</odef:name>
        <odef:description>An identifier of the purchase order

```



```

        assigned by the vendor.</odef:description>
    </rdf:Description>

<!-- Number-Of-Units -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#P3">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Number-Of-Units</odef:name>
        <odef:description>The quantity of units ordered of a
            line item.</odef:description>
    </rdf:Description>

<!-- Unit-Price -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#P4">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Unit-Price</odef:name>
        <odef:description>The amount charged by the vendor per unit
            for a line item.</odef:description>
    </rdf:Description>

<!-- Total-Price -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#P5">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Total-Price</odef:name>
        <odef:description>A total amount charged by the vendor
            for a number of items.</odef:description>
    </rdf:Description>

<!-- Discount -->
    <rdf:Description rdf:about="http://o-def.info/plugins/3#P6">
        <rdf:type rdf:resource="http://o-def.info/core#Property"/>
        <odef:isChildPropertyOf rdf:resource=
            "http://o-def.info/core#Properties"/>
        <odef:name>Discount</odef:name>
        <odef:description>An amount or percentage by which the
            vendor reduces a price.</odef:description>
    </rdf:Description>

</rdf:RDF>

```

E Some Fundamental Enterprise Data Elements

This appendix shows the classification of some fundamental data elements commonly used by enterprises, as an illustration of the application of the classification method. The elements are *Enterprise Name*, *Person Family Name*, *Person Birth Date*, *Employee Person Birth Date*, and *Customer Person Name*.

E.1 Enterprise Name

This is the name of an enterprise. An example value is “The Open Group”.

Object

The object of this data element is an enterprise.

Object Class

The object fits in the *Enterprise* object class of the core index, and this class characterizes it completely. Its object class is thus:

```
2 Enterprise
```

Role

The object does not perform a particular function, so its classification does not include a role.

Property

The *Name* property of the core index characterizes the information provided about the object. The property is thus:

```
2 Name
```

Complete Classification

The complete classification is:

```
2_2 Enterprise_Name
```

E.2 Person Family Name

This is the family name of a person. Example values are “Smith”, “Sharma”, and “Wang”.

Object

The object of this data element is a person.

Object Class

The object fits in the *Person* object class of the core index, and this class characterizes it completely. Its object class is thus:

```
1 Person
```

Role

The object does not perform a particular function, so its classification does not include a role.

Property

The *Name* property of the core index covers the information provided about the object, but does not characterize it completely. The *Family* child of this property covers the information provided about the object, and does characterize it completely. The property is thus:

```
2.1 Name.Family
```

Complete Classification

The complete classification is:

```
1_2.1 Person_Name.Family
```

E.3 Person Birth Date

This is the date of birth of a person. An example value is “August 4, 1991”.

Object

The object of this data element is a person.

Object Class

The object fits in the *Person* object class of the core index, and this class characterizes it completely. Its object class is thus:

```
1 Person
```

Role

The object does not perform a particular function, so its classification does not include a role.

Property

The *Date* property of the core index covers the information provided about the object, but does not characterize it completely. It is selected as the first property of the sequence.

None of its children covers the information provided about the object, nor does any root property of a tree-structured plugin, or any member of an unstructured plugin.

Birth Date is an obvious candidate for addition to the core index, but is not in the initial index defined in this standard. If a classification is needed before the addition is made to the core index, then a plugin can be defined outside the index for use by the parties concerned.

If a group of parties defines a tree-structured plugin with name *Temporary* and Id 99 (for example), containing a root node with name *Birth* and Id 1, then the property classification used by that group is:

```
10;[99]:1 Date;[Temporary]:Birth
```

If *Birth* were at some point in the future to be added to the core index as a child of *Date* with relative Id 1, then the standard property classification would be:

```
10.1 Date.Birth
```

Complete Classification

A complete classification that could be used by a group of parties before the core index is changed is:

```
1_10;[99]:1 Person_Date;[Temporary]:Birth
```

If *Birth* were at some point in the future to be added to the core index as a child of *Date* with relative Id 1, then, after the change, the standard classification would be:

```
1_10.1 Person_ Date.Birth
```

E.4 Employee Person Birth Date

In the context of an enterprise, this is the date of birth of a person who is an employee of the enterprise.

Object

The object of this data element is a person who is an employee of an enterprise.

Object Class

The object fits in the *Person* object class of the core index, and this class characterizes the kind of thing that it is completely. Its object class is thus:

```
1 Person
```

Role

The person performs the function of the employee. This is characterized by the *Employee* role of the *Enterprise Operation* plugin of the core index. The role is thus:

```
;2:3 ;Enterprise-Operation:Employee
```

Property

The property is as analyzed above:

- 10;[99]:1 Date:[Temporary]:Birth could be used by agreement between concerned parties before any change is made to the core index.
- 10.1 Date.Birth would be a possible classification if the core index is changed.

Complete Classification

A complete classification that could be used by agreement between concerned parties before any change is made to the core index is:

```
1~;2:3_10;[99]:1  
Person~;Enterprise-Operation:Employee_Date:[Temporary]:Birth
```

A possible classification if the core index is changed is:

```
1~;2:3_10.1 Person~;Enterprise-Operation:Employee_Date.Birth
```

E.5 Customer Person Name

In the context of an enterprise, this is the name of a person that is a customer of the enterprise.

Object

The object of this data element is a person who is a customer of an enterprise.

Object Class

The object fits in the *Person* object class of the core index, and this class characterizes the kind of thing that it is completely. Its object class is:

```
1 Person
```

Role

The person performs the function of customer. This is characterized by the *Customer* role of the *Enterprise Operation* plugin of the core index. The role is:

```
;2:1 ;Enterprise-Operation:Customer
```

Property

The *Name* property of the core index characterizes the information provided about the object.
The property is:

2 Name

Complete Classification

The complete classification is:

1~;2:1_2 Person~;Enterprise-Operation:Customer_Name

F Example Web Shopping Trolley Classification

This appendix shows the O-DEF classifications for the data elements of a web shopping trolley. The data elements were identified by inspecting the website of a well-known web retailer.

In most cases, the data elements can be classified using the core index only. The exceptions are:

- **Quantity in stock:** This would not be found in a traditional purchase order, and usage varies between web retailers. Some put text such as “Available again soon” or “Only two left” in this field when there are no or only a few items in stock. Some show an arbitrary number such as 50 when there are many items in stock.
- **Gift:** This would not be found in a traditional purchase order, but is used consistently by web retailers, and is a candidate for addition to the `Purchase` plugin of the core index.

The properties that are not in the core index are shown as being in a web retailer-defined plugin, named `XX` and identified as `xx`.

Data Element	O-DEF Identifier	O-DEF Name
Product	;2:2_2	;Purchase:Line-Item_Name
Product description	;2:2_13.2	;Purchase:Line-Item_Text.Description
Quantity in stock	;2:2_6;[xx]:[SL]	;Purchase:Line-Item_Quantity;[XX]:[Stock Level]
Quantity ordered	;2:2_6;2:3	;Purchase:Line-Item_Quantity;Purchase:Number-Of-Units
Whether intended as a gift	;2:2_4;[xx]:[G]	;Purchase:Line-Item_Indicator;[XX]:[Gift]
Product price	;2:2_7;2:4	;Purchase:Line-Item_Amount;Purchase:Unit-Price
Saving on list price	;2:2_7;2:7	;Purchase:Line-Item_Amount;Purchase:Discount
Saving percent	;2:2_9;2:7	;Purchase:Line-Item_Percent;Purchase:Discount
Total price of trolley	;2:1_7;2:6	;Purchase:Purchase-Order_Amount;Purchase:Total-Price
Total saving	;2:1_7;2:7	;Purchase:Purchase-Order_Amount;Purchase:Discount

G Example Dublin Core Plugin

The Dublin Core plugin described here is an example of a plugin that is outside the index and could be used by agreement between collaborating parties. (It could be a candidate for inclusion in the index once experience has been gained in its use.) It is an unstructured plugin.

G.1 Plugin Identification

The plugin identifier, name, and description are shown in the [table](#) (Table 11).

Table 11: The Dublin Core Plugin

Identifier	Name	Description
dc	Dublin-Core	The elements defined by the Dublin Core Metadata Initiative [DCMI].

G.2 Plugin Definition

Authoritative Language

English is the authoritative language of the Dublin Core plugin.

Object Classes

The Dublin Core plugin does not contain any object classes.

Roles

The Dublin Core plugin does not contain any roles.

Properties

The properties of the Dublin Core plugin are the elements of the Dublin Core Metadata Element Set. An initial definition is contained in [[RFC 5013](#)]. This has subsequently been extended. At the time of publication of this standard, these elements are as defined in [[DC 2012](#)]. The list of elements can be extended by future publications of the Dublin Core Metadata Initiative [[DCMI](#)].

The identifiers of these properties in the plugin are their *Element Names*. The names of these properties in the plugin are their *Labels*. The names are the same as the identifiers except that their first characters are uppercase. The descriptions of these properties in the plugin are their

Definitions. (The terms *Element Name*, *Label*, and *Definition* are as used in [RFC 5013]. In [DC 2012], the *Labels* and *Definitions* are given by label and comment RDF constructs.)

The properties of this plugin will usually narrow the scope of properties of the core index, as shown in the [table](#) (Table 12).

Table 12: Plugin Properties in Relation to Core Index Properties

Plugin Property	Narrowed Core Index Properties
Title	Name
Creator	Text
Subject	Text Code
Description	Text.Description
Publisher	Text Identifier
Contributor	Text Identifier
Date	Date Date-Time
Type	Text Code
Format	Text Code
Identifier	Identifier
Source	Text Code Identifier
Language	Text Code
Relation	Text Identifier
Coverage	Text Code

Plugin Property	Narrowed Core Index Properties
Rights	Text Code
Audience	Text Identifier
Alternative Title	Name
Table Of Contents	Text
Abstract	Description.Text
Date Created	Date Date-Time
Date Valid	Date Date-Time
Date Available	Date Date-Time
Date Issued	Date Date-Time
Date Modified	Date Date-Time
Extent	Text
Medium	Text
Is Version Of	Identifier Text
Has Version	Identifier Text
Is Replaced By	Identifier Text
Replaces	Identifier Text
Is Required By	Identifier Text
Requires	Identifier Text

Plugin Property	Narrowed Core Index Properties
Is Part Of	Identifier Text
Has Part	Identifier Text
Is Referenced By	Identifier Text
References	Identifier Text
Is Format Of	Identifier Text
Has Format	Identifier Text
Conforms To	Identifier Text
Spatial Coverage	Text
Temporal Coverage	Text Time
Mediator	Identifier Text
Date Accepted	Date Date-Time
Date Copyrighted	Date Date-Time
Date Submitted	Date Date-Time
Audience Education Level	Text Code
Access Rights	Text Code
Bibliographic Citation	Text Identifier

Plugin Property	Narrowed Core Index Properties
License	Text Identifier
Rights Holder	Text Identifier
Provenance	Text
Instructional Method	Text Code
Accrual Method	Text Code
Accrual Periodicity	Text
Accrual Policy	Text
Agent	Identifier Text
Agent Class	Identifier Code Text
Bibliographic Resource	Identifier Text
File Format	Identifier Text
Frequency	Text
Jurisdiction	Text Identifier Code
License Document	Identifier Text
Linguistic System	Text Code
Location	None Identifier
Location, Period, or Jurisdiction	Text Identifier

Plugin Property	Narrowed Core Index Properties
Media Type	Text Code
Media Type Or Extent	Text Code
Method of Instruction	Text Code
Method of Accrual	Text Code
Period of Time	Text
Physical Medium	Text Code
Physical Resource	Identifier Text
Policy	Identifier Text
Provenance Statement	Text
Rights Statement	Text
Size or Duration	Quantity Measure
Standard	Text
ISO 639-2	Code.Language
RFC 1766	Code.Language
URI	Identifier.URI
DCMI Point	Text
ISO 3166	Language.Code.ISO-3166
DCMI Box	Text
DCMI Period	Text
W3C DTF	Date-Time Text

Plugin Property	Narrowed Core Index Properties
ISO 3066	Code.Language
ISO 5646	Code.Language
RFC 4646	Code.Language.RFC-4646
ISO 639-3	Code.Language
LCSH	Code
MeSH	Code
DDC	Code
LCC	Code
UDC	Code
DCMI Type Vocabulary	Code
IMT	Code
TGN	Code
NLM	Code

Use of the Dublin Core *Description* property to narrow the scope of the *Text.Description* property of the core index is redundant, but may be appropriate in some circumstances.

Use of the Dublin Core *Date* property to narrow the scope of the *Date* or *Date-Time* property of the core index is redundant, but may be appropriate in some circumstances.

Use of the Dublin Core *Identifier* property to narrow the scope of the *Identifier* property of the core index is redundant, but may be appropriate in some circumstances.

Use of the Dublin Core *URI* property to narrow the scope of the *Identifier.URI* property of the core index is redundant, but may be appropriate in some circumstances.

Use of the Dublin Core *ISO 3166* property to narrow the scope of the *Language.Code.ISO-3166* property of the core index, or of the Dublin Core *ISO 4646* property to narrow the scope of the *Language.Code.ISO-4646* property of the core index, is redundant, but may be appropriate in some circumstances.

G.3 Examples of Use

G.3.1 Online Bookstore

The Open Group Online Bookstore [OGPUBS] gives information about each Open Group publication. This information could be represented by data elements.

The object of each such data element would be a publication. The core index does not contain such an object class. For the purpose of this example, it is assumed that The Open Group has defined a publications structured plugin outside the index, with name *ogpubs* and containing the object classes, roles, and properties that are needed in connection with The Open Group publications but are not found amongst the Dublin Core elements. The object class is then `[ogpubs]:Publication`. The names of the properties are shown in the table, with example values taken from the catalog entry for the ArchiMate® Model Exchange File Format [C154].

Table 13: Publication Reference Data Element Property Name Components

Data Element	Example Value	Property Name
Document title	ArchiMate® Model Exchange File Format	Name;[dc]:[Title]
Description	The ArchiMate® Model Exchange File Format, an Open Group standard, defines a file format that can be used to exchange data between systems that wish to import and export ArchiMate models. ...	Text.Description;[dc]:[Description]
Reference	C154	Identifier;[dc][Identifier];[ogpubs]:Publication-Id
US ISBN	1-937218-67-6	Identifier;[dc][Identifier];[ogpubs]:ISBN.US
Date of publication	August 2015	Date-Time;[dc][Date];[ogpubs]:Publication-Date
Number of pages	77	Quantity;[ogpubs]:Pages
See also	C13L: ArchiMate® 2.1 Specification, December 2013	Text;[dc][Relation];[ogpubs]:See-Also
Supersedes	S151: ArchiMate® Model Exchange File Format, Version 2, March 2015	Text;[dc][Relation];[ogpubs]:Supersedes
Subject	ArchiMate	Text;[dc]:[Subject]
Type	Standards	Code;[dc]:[Type];[ogpubs]:Publication-Type

An example of a complete data element concept name is:

; [ogpubs]:Publication_Name; [dc]:[Title]

Notes:

1. The *Reference* and *US ISBN* properties are classified as Identifier; [dc]:[Identifier], but this does not narrow the scope of *Identifier*. In particular, it does not indicate that the identifier is one that is assigned by The Open Group, or that it is the standard globally-unique US ISBN identifier. The classifications are made precise by the addition of the ; [ogpubs]:Publication-Id and ; [ogpubs]:ISBN.US properties.
2. Additions of ogpubs plugin properties for precision are similarly made to *Date of publication*, *See also*, *Supersedes*, and *Type*.
3. The *Type* is classified as a Code because its value is selected from a defined set of publication types. The definition of the [ogpubs]:Publication-Type property in the ogpubs plugin specifies this set.

If the Dublin Core plugin is incorporated in the core index, the names will remain the same except for the removal of '[' and ']' characters. The identifiers will remain the same except for the removal of '[' and ']' characters and the change of the plugin identifier "dc" to a positive integer.

G.3.2 ArchiMate Models

The ArchiMate® Model Exchange File Format, an Open Group standard [C154], recommends the use of the Dublin Core Metadata Element Set to describe ArchiMate models (see its Section 4.1.6). Assuming an *ArchiMate* structured plugin containing a *Model* object class, this would result in data elements with classifications such as the following, possibly extended for precision:

- ; [ArchiMate]:Model_Name; [dc]:[Title]
- ; [ArchiMate]:Model_Text; [dc]:[Creator]
- ; [ArchiMate]:Model_Text; [dc]:[Subject]
- ; [ArchiMate]:Model_Text.Description; [dc]:[Description]
- ; [ArchiMate]:Model_Text; [dc]:[Publisher]
- ; [ArchiMate]:Model_Text; [dc]:[Contributor]
- ; [ArchiMate]:Model_Date; [dc]:[Date]
- ; [ArchiMate]:Model_Code; [dc]:[Type]
- ; [ArchiMate]:Model_Code; [dc]:[Format]
- ; [ArchiMate]:Model_Identifier; [dc]:[Identifier]
- ; [ArchiMate]:Model_Text; [dc]:[Source]
- ; [ArchiMate]:Model_Code; [dc]:[Language]

- ;[ArchiMate]:Model_Text;[dc]:[Relation]
- ;[ArchiMate]:Model_Code;[dc]:[Coverage]
- ;[ArchiMate]:Model_Text;[dc]:[Rights]

H Example UNSPSC Product Item Plugin

The UNSPSC Product Item plugin described here is another example of a plugin that is outside the index and could be used by agreement between collaborating parties. (It also could be a candidate for inclusion in the index once experience has been gained in its use.) It is an unstructured plugin. It is used in the [Example Electric Vehicle Battery Status Reporting use case](#).

H.1 Plugin Identification

The plugin identifier, name, and description are shown in the [table](#) (Table 14).

Table 14: The UNSPSC Product Item Plugin

Identifier	Name	Description
X1	UNSPSC	The classes of item defined by the United Nations Standard Products and Services Code [UNSPSC].

H.2 Plugin Definition

Authoritative Language

English is the authoritative language of the UNSPSC Product Item plugin.

Object Classes

The object classes of the UNSPSC Product Item plugin are the classes of product and service items that are allocated UNSPSC codes. Their names are the code names, and their identifiers are the code numbers. At the time of publication of this standard, these codes are as defined in Version 17.1001 of the UNSPSC. The list of codes can be extended by future publications of the United Nations Development Programme (UNDP).

The term “product” can be used to mean an individual item or a type of item resulting from a production process. The members of the object classes of the UNSPSC Product Item plugin are individual items, not types of item. So, for example, the object class *25100000: Motor Vehicles* contains individual automobiles but does not contain “Ford Model-T”. An additional UNSPSC Product Type plugin could be defined to contain object classes whose members are products in the “type of item” sense.

Roles

The UNSPSC Product Item plugin does not contain any roles.

Properties

The UNSPSC Product Item plugin does not contain any properties.

H.3 Example of Use

An example of the use of this plugin is given in [Appendix F](#).

I Example Charitable Foundation Case Study

This appendix describes part of a plugin that could be used by a charitable foundation dedicated to the support of children, and gives some examples of its use to classify data elements. It is a tree-structured plugin.

The foundation deals with the children, their parents or guardians, the communities they live in, their churches, and their schools, among a host of people and organizations. It maintains a database of information about these entities and, in some cases, exchanges electronic messages about them. The plugin could be used to classify the information in the database and in the messages.

This study is based on a real-life example, but only a small amount of the total metadata required is shown here.

I.1 Plugin Identification

The plugin identifier, name, and description are shown in the [table](#) (Table 15).

Table 15: Example Charitable Foundation Plugin

Identifier	Name	Description
cf	Charitable-Foundation	Metadata for data used by the charitable foundation.

I.2 Plugin Definition

Authoritative Language

English is the authoritative language of the Charitable Foundation plugin.

Object Classes

Some of the object classes of the plugin are shown in the [table](#) (Table 16). The foundation maintains data about conditions in the countries in which it operates, the communities it works in, the churches it works with, and many other things. For example, it keeps records of the infant mortality rates in the countries to help it assess the need for and effectiveness of its operations. The Charitable Foundation plugin contains the object classes that are needed for the classification of this data that are not in the core index or in other available plugins.

Table 16: Charitable Foundation Selected Object Classes

Identifier	Name	Description
1	Country	A country in which the foundation operates or could operate.
2	Community	A collection of people that interact with each other socially.
3	Church	A particular religious group in a local area.
...

Roles

Some of the roles of the plugin are shown in the [table](#) (Table 17).

Table 17: Charitable Foundation Selected Roles

Identifier	Name	Description
1	Child	A person that is a child supported by the foundation.
2	Development-Sponsor	A person that sponsors the development of a child in the foundation's program.
3	Party	A person or enterprise with which the foundation has a business or contractual relationship.
4	Partner	An enterprise designated by the foundation as a partner.
4.1	Child-Development	A partner for activities concerned with development of children that are not necessarily in their early childhood.
4.2	Early-Childhood-Development	A partner for activities concerned with early childhood development.
4.3	Leadership-Development	A partner for activities concerned with leadership development.
4.4	Local	An enterprise that contributes to the operation of the foundation's program in a local area.
...
11	Building-Material	A substance used as a building material by a community.
...

Properties

Some of the properties of the plugin are shown in the [table](#) (Table 18).

Table 18: Charitable Foundation Selected Properties

Identifier	Name	Description
1	Party	An attribute of a party (a person or enterprise).
1.1	Id	The identifier of a party that is assigned by the foundation.
...
4	Partner	An attribute of a partner enterprise.
4.1	Cognitive-And-Vocational-Activities	A description of the cognitive and vocational activities undertaken by the partner.
4.2	Contact-Title	The name of a contact in the partner enterprise.
...
4.5	Latitude-Low-Precision	The GPS coordinate latitude of the partner, given with low precision.
...
4.10	Socio-Emotional-Activities	A description of the socio-emotional activities undertaken by the partner.
4.11	Non-School Child-Activities	A description of the activities for non-school children undertaken by the partner.
...
4.17	Disburse-Gifts	Whether the partner can disburse gifts.
4.18	Disburse-Unsponsored-Funds	Whether the partner can disburse unsponsored funds.
...
4.27	Church	An attribute of a local partner that is a church.
4.27.1	Church-Has-Electricity	Whether the church has electricity.
4.27.2	Church-Has-Phone	Whether the church has a telephone.
...

Identifier	Name	Description
4.27.8	Paid-Staff	The number of paid staff that the church has.
4.27.9	Not-In-School-Reason	Rationale explaining the percentage of children in the local area that are not in full or part-time education.
4.27.10	Not-In-School	The percentage of children in the local area that are not in full or part-time education.
...
8	Country	An attribute of a country.
8.1	Country-Code	The two-character code for the country defined by ISO.
8.2	Primary-School-Years	The average number of years taken by a pupil to complete primary school in the country.
...
8.9	Infant-Mortality-Rate	The country's infant mortality rate.
...
8.15	Per-Capita-Income	The country's per-capita income in US dollars.
8.16	Attain-5th-Grade	The country's percent of children that reach US 5 th grade equivalent.
...
9	Community	An attribute of a community.
9.1	Id	The identifier assigned to the community by the foundation.
...
9.7	Hunger-Months	A list of the months when people do not have enough to eat.
...
9.15	Warmest-Month-Temperature	The average temperature over the warmest month.
9.16	Warmest-Month	The warmest month.
...
11	Occupation	An attribute of a person connected with the person's occupation.

Identifier	Name	Description
11.1	Occupation-ID	A code that identifies the type of occupation.
...
11.13	Works-In-Market	Whether the person works as a market trader.
...

I.3 Examples of Use

Data Element Description	O-DEF Identifier	O-DEF Name
An identifier assigned by the foundation to identify a person or enterprise with which it has a business or contractual relationship.	*~;[cf]3_1;[cf]:1.1	*~;[Charitable-Foundation]:Party_Identifier;[Charitable-Foundation]:Party.Id
The name of a contact in the partner organization.	*~;[cf]4_13;[cf]:4.2	*~;[Charitable-Foundation]:Partner_Text; [Charitable-Foundation]:Partner.Contact-Title
The GPS co-ordinate latitude of a partner at low precision.	*~;[cf]4_5;[cf]:4.5	*~;[Charitable-Foundation]:Partner_Measure; [Charitable-Foundation]:Partner.Latitude-Low-Precision
Activities for non-school children run by a child-development partner.	*~;[cf]:4.1 _13;[cf]:4.11	*~;[Charitable-Foundation]:Partner.Child-Development_Text~; [Charitable-Foundation]:Partner.Non-School-Child-Activities
Whether a child-development partner can disburse unsponsored funds.	*~;[cf]:4.1_4;[cf]:4.18	*~;[Charitable-Foundation]:Partner.Child-Development_Indicator~; [Charitable-Foundation]:Partner.Disburse-Unsponsored-Funds
Whether a local church partner church has electricity.	;[cf]:3~;[cf]:4.4 _4;[cf]:4.27.1	;[Charitable-Foundation]:Church~;[Charitable-Foundation]:Partner.Local_Indicator; [Charitable-Foundation]:Partner.Church.Church-Has-Electricity

Data Element Description	O-DEF Identifier	O-DEF Name
The number of paid staff that a local church partner has.	;[cf]:3~;[cf]:4.4_6;[cf]:4.27.8	;[Charitable-Foundation]:Church~;[Charitable-Foundation]:Partner.Local_Quantity; [Charitable-Foundation]:Partner.Church.Paid-Staff
The percent of children supported by a local church partner that are not receiving school education.	;[cf]:3~;[cf]:4.4_9;[cf]:4.27.10	;[Charitable-Foundation]:Church~;[Charitable-Foundation]:Partner.Local_Percent; [Charitable-Foundation]:Partner.Church.Not-In-School
The 2-character ISO code of a country.	;[cf]:1_3.2.1.1	;[Charitable-Foundation]:Country_Code.Country.ISO-3166.Alpha-2
The infant mortality rate of a country.	;[cf]:1_8[cf]:8.9	;[Charitable-Foundation]:Country_Rate;[Charitable-Foundation]:Country.Infant-Mortality
The percent of children of a country that reach US 5 th grade equivalent educational level.	;[cf]:1_9[cf]:8.16	;[Charitable-Foundation]:Country_Percent; [Charitable-Foundation]:Country.Reaching-5th-Grade
The name of a community.	;[cf]:2_2	;[Charitable-Foundation]:Community_Name
The identifier assigned by the foundation to identify a community.	;[cf]:2_1;[cf]:9.1	;[Charitable-Foundation]:Community_Identifier; [Charitable-Foundation]:Community.Id
A list of months when the members of a community do not have enough to eat.	;[cf]:2_13;[cf]:9.7	;[Charitable-Foundation]:Community_Text; [Charitable-Foundation]:Community.Hunger-Months
The average temperature of a community's warmest month.	;[cf]:2_5.1.1;[cf]:9.15	;[Charitable-Foundation]:Community_Measure.Temperature; [Charitable-Foundation]:Warmest-Month-Average
Building Material Description.	7~;[cf]:11_13.2	Substance~;[Charitable-Foundation]:Building-Material_Text.Description

Note: The name of a contact in the partner organization is classified as a *Text* property, not a *Name* property, because it is not the name of the partner, which is the object of the data element. See under Indirect Properties ([Section 4.2.1](#)).

J Example Electric Vehicle Battery Status Reporting

This appendix contains an example classification of data elements in messages reporting the status of a battery in an electric vehicle. It shows the use of plugins, and illustrates how material in plugins can be incorporated in the core index.

An electric vehicle sends regular messages to a service center giving the effective total charge capacity of its battery. This is used by the service center, in combination with other information, for battery lifecycle management, so that the service center can arrange battery maintenance and replacement operations.

The messages are defined by a group of vehicle manufacturers, battery manufacturers, and service centers, and this group agrees on and uses their O-DEF classifications.

The classifications are made using the initial O-DEF index. This is not sufficient to classify the data elements involved. The group agrees to use the United Nations Standard Products and Services Code [UNSPSC] as an unstructured plugin for the classification of products and services, and to develop an additional application-specific tree-structured Electric Vehicle plugin for other object classes, roles, and properties. They agree that the UNSPSC plugin will have name “UNSPSC” and identifier “X1”, and that the Electric Vehicle plugin will have name “EV” and identifier “X2”. The UNSPSC Product Item plugin is described in [Appendix D](#). The elements of the EV plugin are described as they are needed for the example classifications, and are summarized [later in this Appendix](#).

J.1 Message Format

The messages are formatted in the Open Data Format [O-DF]. An example message is shown in the [figure](#) (Figure 1).

Figure 1: Example Battery Capacity Report Message

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Example electric vehicle battery charge capacity report. -->
<Objects xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="odf.xsd">
  <Object type="Electric Vehicle">
    <id idType="VIN">S59N12345LAZ67890</id>
    <Object type="Battery">
      <id idType="ACME Serial Number">0123456</id>
      <InfoItem name="GTIN">
        <description>The Global Trade Item Number (GTIN) of
          the battery. Note that the manufacturer and model
          can be determined from the GTIN.</description>
        <value>01122334455667</value>
        <InfoItem name="Effective Capacity">
          <description>Effective total charge capacity
```

```

        value in Kilowatt-Hours.</description>
        <value>45.5</value>
    </InfoItem>
</Object>
</Object>
</Objects>

```

J.2 Data Model

This message is based on the data model shown in the [figure \(Figure 2\)](#). (In reality, this would be part of a larger data model including other data about the vehicle.)

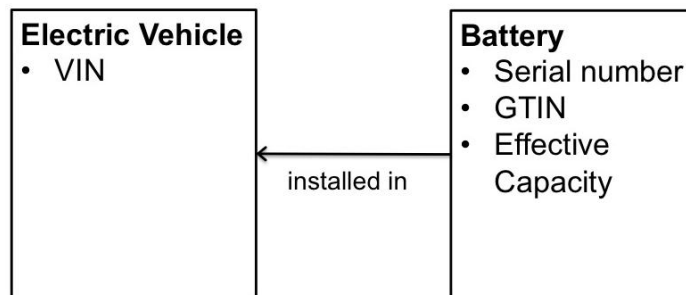


Figure 2: Electric Vehicle Example Data Model

Note that the data model does not include the *Infoitem* descriptions, which are regarded as metadata rather than data. This is a choice made by the modeler in creating a clear and useful model of the data.

J.3 Data Item Classifications

J.3.1 Electric Vehicle VIN

Object

The object is the electric vehicle.

Object Class

The *Product* [object class of the initial index \(Section 7.1\)](#) includes electric vehicles. This is selected as the first object class of the sequence.

The first object class does not categorize the object precisely enough. It does not have any children. The UNSPSC Product Item plugin has object class *25101509: Electrically Powered Vehicle* which is applicable to this example. This is selected as the second and final object class of the sequence.

The O-DEF object class name is thus:

```
Product;[UNSPSC]:[Electrically Powered Vehicle]
```

Its identifier is:

```
10;[X1]:[25101509]
```

Role

The electric vehicle does not perform a specific function. The classification does not include a role.

Property

The Vehicle Identification Number (VIN) is an identifier of the electric vehicle, which is the object of the data element. The *Identifier* [property of the initial index \(Section 7.2\)](#) is selected as the first property of the sequence.

The first property does not categorize the information given about the object precisely enough. A sub-property of it, *VIN*, is added to the EV plugin. This property then forms the second and final member of the sequence.

The O-DEF property name is thus:

```
Identifier;[EV]:VIN
```

Its identifier is:

```
1;[X2]:1
```

Complete Classification

The name of the classification of the data element is thus:

```
Product;[UNSPSC]:[Electrically Powered Vehicle]_Identifier;[EV]:VIN
```

Its identifier is:

```
10;[X1]:[25101509]_1;[X2]:1
```

J.3.2 Battery Serial Number

Object

The object is the battery in the electric vehicle.

Object Class

The *Product* [object class of the initial index \(Section 7.1\)](#) includes batteries.

The first object class does not categorize the object precisely enough. It does not have any children. The UNSPSC Product Item plugin has a member *26111703: Vehicle batteries*. This is

selected as the second object class of the sequence. It categorizes the object precisely enough, and no further object classes are needed to complete the sequence.

The O-DEF object class name is thus:

```
Product;[UNSPSC]:[Vehicle batteries]
```

Its identifier is:

```
10;[X1]:[26111703]
```

Role

The battery does not perform a specific function. The classification does not include a role.

Property

The serial number is an identifier of the battery, which is the object of the data element. The *Identifier property of the initial index* (Section 7.2) is selected as the first property of the sequence.

The first property does not categorize the information given about the object precisely enough. A sub-property of it, *Manufacturer-Assigned*, is added to the EV plugin. This property then forms the second and final member of the sequence.

The O-DEF property name is thus:

```
Identifier;[EV]:Manufacturer-Assigned
```

Its identifier is:

```
1;[X2]:2
```

Complete Classification

The name of the classification of the data element is thus:

```
Product;[UNSPSC]:[Vehicle batteries]  
_Identifier;[EV]:Manufacturer-Assigned
```

Its identifier is:

```
10;[X1]:[26111703]_1;[X2]:2
```

J.3.3 Battery GTIN

Object

The object is the battery in the electric vehicle.

Object Class

The O-DEF object class name is as for the Battery Serial Number:

Product;[UNSPSC]:[Vehicle batteries]

Its identifier is:

10;[X1]:[26111703]

Role

The battery does not perform a specific function. The classification does not include a role.

Property

The information given by the data element is the Global Trade Item Number [GTIN]. A GTIN can be used to identify types of products at any packaging level, but does not identify individuals. Individuals can be uniquely identified using a GTIN plus a serial number. It is a text field that is not an identifier, name, code, indicator, measure, quantity, amount, rate, percent, date, date-time, or time, so the *Text* property of the initial index (Section 7.2) is selected as the first property of the sequence.

The first property does not categorize the information given about the object precisely enough. A sub-property of it, *GTIN*, is added to the EV plugin. This property then forms the second and final member of the sequence.

The O-DEF property name is thus:

Text;[EV]:GTIN

Its identifier is:

13;[X2]:3

Complete Classification

The name of the classification of the data element is thus:

Product;[UNSPSC]:[Vehicle batteries]_Text;[EV]:GTIN

Its identifier is:

10;[X1]:[26111703]_13;[X2]:3

J.3.4 Battery Effective Capacity

Object

The object is the battery in the electric vehicle.

Object Class

The O-DEF object class name is as for the Battery Serial Number:

Product;[UNSPSC]:[Vehicle batteries]

Its identifier is:

```
10; [X1] : [26111703]
```

Role

The battery does not perform a specific function. The classification does not include a role.

Property

The information given by the data element is the effective capacity of the battery in kWh. It is a numeric quantity that has an explicit or implied unit of measure other than a monetary unit, so the *Measure* [property of the initial index \(Section 7.2\)](#) is selected as the first property of the sequence.

The first property does not categorize the information given about the object precisely enough. A sub-property of it, *Effective-Capacity*, is added to the EV plugin. This is still not sufficiently precise, and a further sub-property Kilowatt-Hours is added. This property then forms the third and final member of the sequence.

The O-DEF property name is thus:

```
Measure; [EV] :Effective-Capacity.Kilowatt-Hours
```

Its identifier is:

```
5; [X2] : 4.1
```

Complete Classification

The name of the classification of the data element is thus:

```
Product; [UNSPSC] : [Vehicle batteries]  
_Measure; [EV] :Effective-Capacity.Kilowatt-Hours
```

Its identifier is:

```
10; [X1] : [26111703]_5; [X2] : 4.1
```

J.4 EV Plugin

J.4.1 Plugin Identification

The plugin identifier, name, and description are shown in the [table \(Table 19\)](#).

Table 19: Example Electric Vehicle Plugin

Identifier	Name	Description
X2	EV	Metadata for data contained in electric vehicle status reports.

J.4.2 Plugin Definition

Authoritative Language

English is the authoritative language of the EV plugin.

Object Classes

There are no object classes in the plugin.

Roles

There are no roles in the plugin.

Properties

The properties of the plugin that are used in the example are shown in the [table](#) (Table 20).

Table 20: Electric Vehicle Status Selected Properties

Identifier	Name	Description
1	VIN	A Vehicle Identification Number.
2	Manufacturer-Assigned	An identifier of the object of the data element that is assigned by its manufacturer.
3	GTIN	The Global Trade Item Number (GTIN) of the object of the data element.
4	Effective-Capacity	The effective capacity of a battery, measured at a particular point in time.
4.1	Kilowatt-Hours	An effective capacity measured in kilowatt-hours.
...

J.5 Moving Definitions into the Index

The UNSPSC Product Item plugin, and many of the definitions in the EV plugin, would be suitable for inclusion in the O-DEF index.

All that would be needed to include the UNSPSC Product Item plugin in the index would be to assign it an O-DEF identifier.

The *VIN*, *Manufacturer-Assigned*, *Manufacturer*, *Model*, *Effective-Capacity*, and *Kilowatt-Hours* properties are sufficiently general that they could be added to the core index, and retired from the EV plugin.

Assuming that the UNSPSC Product Item plugin is incorporated in the core index, the *VIN*, *Manufacturer-Assigned*, *GTIN*, *Effective-Capacity*, and *Kilowatt-Hours* properties are added to the core index, the names of the data element classifications discussed above would be as follows:

```
Product;UNSPSC:[Electrically Powered Vehicle]_Identifier.VIN
Product;UNSPSC:[Vehicle batteries]_Identifier.Manufacturer-Assigned
Product;UNSPSC:[Vehicle batteries]_Text.GTIN
Product;UNSPSC:[Vehicle batteries]_
    Measure.Effective-Capacity.Kilowatt-Hours
```

The previous classifications would remain valid and meaningful, but would not be used for future applications.

Acronyms

DCMI	Dublin Core Metadata Initiative
GTIN	Global Trade Item Number
JSON	JavaScript Object Notation
MDR	Metadata Registries
O-DEF	Open Data Element Framework
O-DF	Open Data Format
RDF	Resource Description Framework
UNDP	United Nations Development Programme
UNSPSC	United Nations Standard Products and Services Code
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
VIN	Vehicle Identification Number
XML	eXtensible Markup Language

Index

bracketed string	19	Person Birth Date	55
change (to index)	8	Person Family Name	54
characterization	13	plugin	1, 5, 7, 27
classification	1, 4, 7, 10	plugin identifier	18
classification syntax	15	plugin member identifier	19
core index	4, 7	plugin member name	19
Customer Person Name	57	plugin name	18
data element	1, 4, 10, 15, 54	plugin specification	22
data element concept	4, 15	positive integer	19
data model	10	property	5, 13, 25
Dublin Core	60	property identifier	17
Employee Person Birth Date	56	property name	17
Enterprise Name	54	property tree	5
Enterprise Operation plugin	29	Purchase plugin	30
identifier	7, 8	RDF/XML	20, 22
identifier properties	12	relative description	5
index	1, 7	relative identifier	5
indirect properties	11	relative name	6
language	7	role	6, 13
main description file	20	role identifier	16
name string	19	role name	17
node	5	role tree	6
node description	9	root	6
node identifier	8, 18	root object class	23
node name	8, 18	root property	23
object class	5, 12, 24	root role	23
object class identifier	15	status	6
object class name	16	tree	6, 8, 22
object class tree	5	tree-structured plugin	7, 72
O-DEF	1	Unicode	19
O-DEF index	24	UNSPSC Product Item	70
O-DEF plugin	28	unstructured plugin	7, 9, 60, 70