TECHNICAL SPECIFICATION

ISO/TS 15926-11

First edition
2015-05-01

Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities —

Part 11: Methodology for simplified industrial usage of reference data

Systèmes d'automatisation industrielle et intégration — Intégration de données de cycle de vie pour les industries de "process", y compris les usines de production de pétrole et de gaz —

Partie 11: Méthodologie pour un usage industriel simplifié des données de référence

Reference number
ISO/TS 15926-11:2015(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

ISO 15926 is organized as a series of parts, each published separately. The structure of ISO 15926 is described in ISO 15926-1.

ISO 15926 consists of the following parts, under the general title *Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities*:

— *Part 1: Overview and fundamental principles;*

— *Part 2: Data model;*

— *Part 3: Reference data for geometry and topology* [Technical Specification]*;*

— *Part 4: Initial reference data* [Technical Specification]*;*

— *Part 6: Methodology for the development and validation of reference data* [Technical Specification]*;*

— *Part 7: Implementation methods for the integration of distributed systems: Template methodology* [Technical Specification]*;*

— *Part 8: Implementation methods for the integration of distributed systems: Web Ontology Language (OWL) implementation* [Technical Specification];

— *Part 11: Methodology for simplified industrial usage of reference data* [Technical Specification]

The following parts are under preparation:

— *Part 9: Implementation methods for the integration of distributed systems: Facade implementation* [Technical Specification]*;*

— *Part 10: Conformance testing* [Technical Specification]

Introduction

ISO 15926 is an International Standard for the representation of process industries facility life-cycle information. This representation is specified by a generic, conceptual data model that is suitable as the basis for implementation in a shared database or data warehouse. The model is designed to be used in conjunction with reference data, that is, standard instances that represent information common to a number of users, production facilities, or both. The support for a specific life-cycle activity depends on the use of appropriate reference data in conjunction with the model.

This part of ISO 15926 focuses on a simplified implementation of the afore mentioned data model in the context of engineering data in the area of the process industry, including the oil, gas, process and power industry and is intended for developers of configuration management processes and systems in general.

This part of ISO 15926 provides the capability to express a product model with RDF triples, RDF Named Graphs and a standardized set of natural language relationships resulting in a table that can be exchanged and shared easily in industry.

There is an industry need for this part of ISO 15926.

— The triple relationships are easy to understand by an engineer so that an engineer can understand the product model. This has been proven by the NL Ship Building group who developed a Gellish-RDF based implementation for standardized exchange of product data of HVAC equipment on a daily basis.

— The standard data sheets from API, NORSOK, etc. used in industry for pumps, compressors, instruments, etc. can be supported by a Gellish-RDF product model enabling industry to continue to work with their specific data sheets and yet exchanging the data in standardized way according this new standard. This has been proven by the ICAAMC compressor group in a pilot for the API 617 data sheet.

— It is used in some projects, e.g. in the Pearl project for oil and gas.

— This part of ISO 15926 can be used as a front end engineering layer for the template methodology used by ISO/TS 15926-7 and ISO/TS 15926-8, e.g. in the FIATECH project IIP. This will make the content of those projects easier to access by engineers.

— An EPC contractor has used the draft of this part of ISO 15926 in various tunnel projects for information modelling in the area of systems engineering which was required by the Dutch authority regulations. With this part of ISO 15926 enriched by the knowledge from ISO/IEC 15288, this became possible. They also built a performance measuring system for operational data in tunnel installations where the methodology of this part of ISO 15926 is used to justify the performance to the ministry of transportation in the Netherlands.

Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities —

Part 11: **Methodology for simplified industrial usage of reference data**

1 Scope

This part of ISO 15926 enables a flexible creation of product knowledge models that can be exchanged in the plant engineering supply chain by combining RDF triples within named graphs, reference data dictionaries and a standardized set of relationships.

This part of ISO 15926 is appropriate for use with ISO 15926 reference data libraries, and it is applicable to the oil, gas, process and power industries.

The following are within the scope of this part of ISO 15926:

— process plants in accordance with ISO 15926-1;

— use of RDF triples representing statements as defined in the ISO 15926 series;

— an initial set of relationships required for process plant life cycle representation;

— rules for the use of RDF Named Graphs for product data representation and exchange;

— examples of possible implementations.

The following are outside the scope of this part of ISO 15926:

— definition of reference data libraries;

— the syntax and format of implementations of product data models and/or instance data using this part of ISO 15926;

— any specific methods and guidelines other than RDF Named Graphs for implementing ISO 15926-2.

2 Normative references

The following referenced documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/TS 15926-4, *Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 4: Initial reference data*

3 Terms, definitions and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1
class
category or division of things based on one or more criteria for inclusion and exclusion

Note 1 to entry: A class need not have any members (things that satisfy its criteria for membership).

[SOURCE: ISO 15926-1:2004, 3.1.1, modified]

3.1.2
characteristic data
description of an entity by the class to which it belongs and a set of property values

Note 1 to entry: ISO 13584, ISO 15926, ISO 22745, ISO 13399, and ISO/TS 29002 all include characteristic data in their data models.

[SOURCE: ISO 8000-2:2012, 7.2, modified]

3.1.3
data
representation of information in a formal manner suitable for communication, interpretation, or processing by human beings or computers

[SOURCE: ISO 10303-1:1994, 3.2.14]

3.1.4
formal syntax
specification of the valid sentences of a formal language using a formal grammar

Note 1 to entry: A formal language is computer-interpretable.

EXAMPLE 1 An XML document type definition (DTD) is a formal syntax.

EXAMPLE 2 ISO 10303-21, contains a formal syntax in WSN for ISO 10303 physical files.

[SOURCE: ISO 8000-2:2012, 6.1, modified]

3.1.5
information
facts, concepts, or instructions

[SOURCE: ISO 10303-1:1994, 3.2.20]

3.1.6
Named Graph
key concept of Semantic Web architecture in which a set of Resource Description Framework (RDF triples) statements (a graph) are identified using a unique URI

Note 1 to entry: Named Graphs is the idea that having multiple RDF graphs in a single document/repository and naming them with URIs provides useful additional functionality built on top of the RDF Recommendations

[SOURCE: W3C Recommendation 25 February 2014, modified]

3.1.7
N-Quad statement
N-Quad
sequence of RDF terms representing the subject, predicate, object and graph identifier of an RDF Triple and the graph it is part of in a dataset

Note 1 to entry: These may be separated by white space. This sequence is terminated by a '.' and a new line (optional at the end of a document).

EXAMPLE < http://one.example/subject1> < http://one.example/predicate1> < http://one.example/object1> < http://one.example/graph3> . # comments here.

[SOURCE: W3C Recommendation 25 February 2014, modified]

3.1.8
RDF graph
graph structure formed by a set of RDF triples

[SOURCE: W3C Recommendation 25 February 2014]

3.1.9
reference data
facility life-cycle data that represents information about classes or individual things which are common to many facilities or of interest to many users

[SOURCE: ISO 15926-1:2004, 3.1.18, modified]

3.1.10
reference data library
RDL
managed collection of reference data

[SOURCE: ISO 15926-1:2004, 3.1.19]

3.1.11
relationship
abstract object that indicates something that one thing has to do with another

[SOURCE: ISO 15926-2:2003, 4.6.4, modified]

3.1.12
semantic encoding
technique of replacing natural language terms in a message with identifiers that reference data dictionary entries

[SOURCE: ISO 8000-2:2012, 6.2]

3.1.13
statement
fact
information that is regarded as indivisible

Note 1 to entry: A statement can be recorded as an instance of the entity relationship in ISO 15926-2. A set of one or more statements can be recorded in shorthand form as a single item as an instance of a template, as defined in ISO/TS 15926-7.

[SOURCE: ISO/TS 15926-6:2013, 3.1.25]

3.1.14
thing
actual part of the real world, perceived part of the real world, or subject of thought

Note 1 to entry: A thing can be a material or non-material object, idea or action.

Note 2 to entry: This definition is adapted from ISO 15926-2, within which “thing” is an entity, but not a defined term.

[SOURCE: ISO/TS 15926-6:2013, 3.1.26]

3.1.15
triple
RDF-triple, representation of a relation between the objects or data that it links

Note 1 to entry: A triple comprises at least:

— an object called “subject”;

— a predicate (also called property) that denotes a relationship between a subject and an object;

— an object or data called "object".

[SOURCE: W3C Recommendation 25 February 2014]

3.1.16
product
thing or substance produced by a natural or artificial process

[SOURCE: ISO 10303-1:1994, 3.2.26]

3.1.17
product data
representation of information about a product in a formal manner suitable for communication, interpretation, or processing by human beings or by computers

[SOURCE: ISO 10303-1:1994, 3.2.27]

3.2 Abbreviated terms

IDM	Information Delivery Manual
RDL	Reference Data Library
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SPARQL	Protocol and RDF Query Language
TriX	Triples in XML
URI	Uniform Resource Identifier
W3C	World Wide Web Consortium

4 Fundamental concepts and assumptions

4.1 Purpose and objectives

This part of ISO 15926 provides a semantic modelling methodology of engineering data that can be relatively "easily" understood by engineers and that is flexible in terms of tailoring the methodology for a specific domain or project. The provided modelling methodology is based on the existing parts ISO 15926-2 and ISO/TS 15926-4. To achieve this, the triple concept of the RDF standard of W3C supplemented with the Named Graph concept of W3C is adopted and augmented with a set of relationships further called the "initial set of relationships".

Reason for developing this methodology can be found in the fact that product specialists and engineers, especially within Small and Medium Enterprises (SMEs), who in general have limited skills in the area of information modelling and related techniques, and they should be supported in their product and or engineering knowledge modelling activities by a simple to use methodology, close to natural languages. Using this methodology will lead to models that are upgradable to full ISO 15926-2 compliant models. In other words this methodology provides a bridge to the much more complex ISO 15926-2 world and provide a low entry threshold to ISO 15926. Also the fact in development work is that humans like to work with simple table based structures rather than relatively complicated schemes should be respected if possible.

In line with this, this part of ISO 15926 offers a way to industry groups to set up and exchange their product model using a low level modelling methodology based on statements in a table manner. This part of ISO 15926 provides for that purpose a normative set of rules that allows engineers to build a product and plant lifecycle models using statements based on a normative set of relationships and normative reference data. A statement is: that which is the case, independent of natural language. A statement can be used to classify things as ‘being the case’. Statements can be expressed in languages as relationships between two roles of things (respectively “thing playing role 1” and “thing playing role 2”).

With the help of this part of ISO 15926, one is able to express any kind of product or engineering information and exchanging this information based on a managed set of reference data, including information concerning all relevant (system life cycle) processes to realize and maintain the product. This part of ISO 15926 provides a semantic modelling methodology for creating and exchanging engineering data, originating from Systems Engineering processes.

4.2 Positioning of this part of ISO 15926

Figure 1 shows the layers that can be distinguished looking at data exchange in general. Within this figure the upper layer represents the role that a specific organization or enterprise plays in the exchange of information. The origin and scope of the information is represented by the second layer. For this part of ISO 15926, the origin can be seen as applicable quality aspects of information as defined in ISO 8000-1, while the scope of the information exchange is ISO/IEC 15288, specific in the context of process industry.

The content layer represents the meaning of the objects that are exchanged as defined by the RDL ISO/TS 15926-4.

The semantic layer is the scope of this part of ISO 15926 and is represented by the RDF Named Graph methodology as described within this part of ISO 15926.

The syntax layer represents the technology that is used to physically exchange the data that represents the information to be exchanged. This part of ISO 15926 prescribes not a standard for any exchange syntax, only gives examples of possible syntax formats as given in Annex B.

The storage layer describes the technology that is used to store the information and can be done by means of a triple store, a graph database or a traditional relational database management system.

Within a specific project each layer can be specified by means of a project-specific “Information Delivery Manual”.

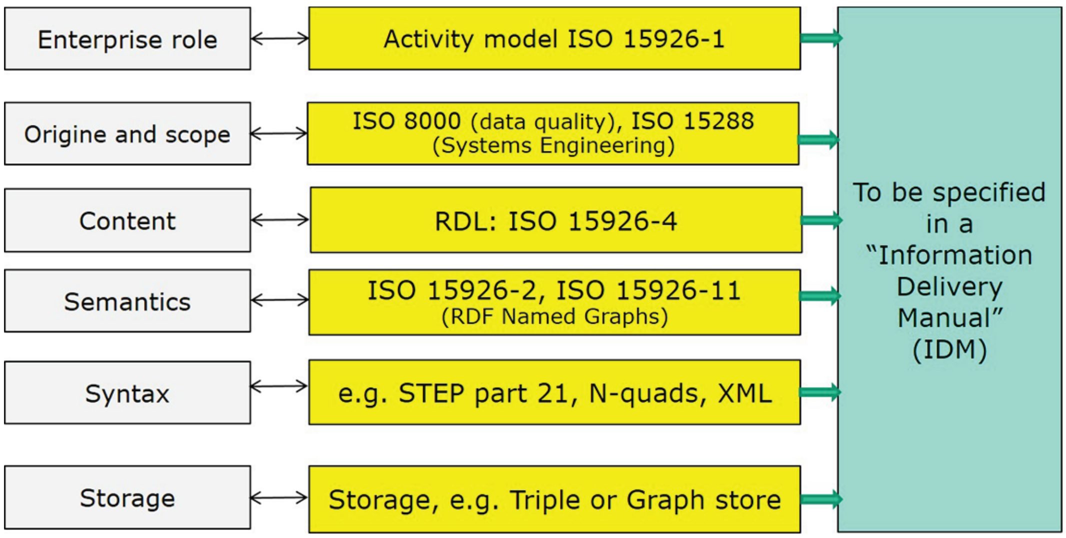

Figure 1 — Positioning of this part of ISO 15926 in a view on data exchange

4.3 Use of statements within ISO 15926

The aim of this part is to realize a human readable expression of engineering data that can be supported and managed by computers where the aim of ISO 15926-2 is a full understanding of engineering data by computers. Starting point of this part is that engineering data expressed according to this part can be (manually) transformed to the level of ISO 15926-2. For this reason some principles of ISO 15926-2 are respected and other are simplified by means of "short cut" relationships but anyway consistent with part ISO 15926-2

The expression of a statement consists of "object (playing a role 1) –> relationship –> object (playing a role 2)" following the simplest possible grammatical pattern: subject – > verb – > object. This is also called a triple in the context of Resource Description Framework (RDF) developed within the W3C community (see also Figure 4).

RDF is a language for expressing data models using statements in triple formats and sharing them with other people and machines. Since it is a W3C "Recommendation", large collections of tools and services are available. The principle of using statements describing a "specific world" (an ontology) can replace fixed data models and provides an extensible ontology with reference data with the aim to defining, customizing and harmonizing systems. For this aim, the methodology described in this part of ISO 15926 uses for this goal taxonomy of relationships and a taxonomy of "things" to be able to describe the world in a consistent and explicit way. This is provided by the rule that relevant things in the real world must be classified by a class in a reference data library.

In Figure 2, the principle of the object - relation - object mechanism is shown. The left side object (playing a role 1) "airco xyz" is classified as the right side object (playing a role 2) being an "air-conditioning unit". In the same way the "capacity airco xyz" is classified as a "nominal cooling capacity". The given information is for human explicit and readable and correspondents with the information that usually can be found in a datasheet. As example the "capacity airco xyz" is quantified in kW and has a magnitude of 100. Engineers are integrating the facts expressed in sentences with all that they already know such that engineers are able to utilize the meaning that they derived from the statements given to create new knowledge which can be easily communicated with colleagues resulting in actions.

While statements (data model expressed in triple formats) that RDF uses is very simple and easy to understand, but the serialized representation tends to get complicated for engineers to visualize the whole structure of knowledge and communicate it to colleagues. An RDF graphical representation of statements is here to help engineers since RDF also provides a standard way of expressing graphs of statements and sharing them with colleagues and computers on line and/or off line. Further ontologies provide engineers to express the rules for inference from statements. An engineer shares his ontology with his colleague it should have all knowledge he should have to draw same conclusions from statements on hands. And same thing will be true with machines if they commit to share the same ontologies.

object (playing role 1)	*relationship*	*object (playing role 2)*
airco xyz	Is classified as	air-conditioning unit
airco xyz	has property	capacity airco xyz
capacity airco xyz	is classified as	nominal cooling capacity
capacity airco xyz	is quantified in	Kw
capacity airco xyz	has magnitude	100

Figure 2 — Representation of a piece of product information in statements

Both objects in a statement have each a specific role, related to the meaning of the relationship. As a result of this a statement always must be readable in two directions, from left to right and from right to left. So each relationship as defined in the initial set has two names: the "prescribed" name, reading from an object playing a role 1 and an object playing a role 2 and the reverse name, reading from an object playing a role 2 to an object playing a role 1 (airco xyz has role 1 and capacity airco xyz has role 2 with respect to the relation). The reverse statements of Figure 2 are shown in Figure 3.

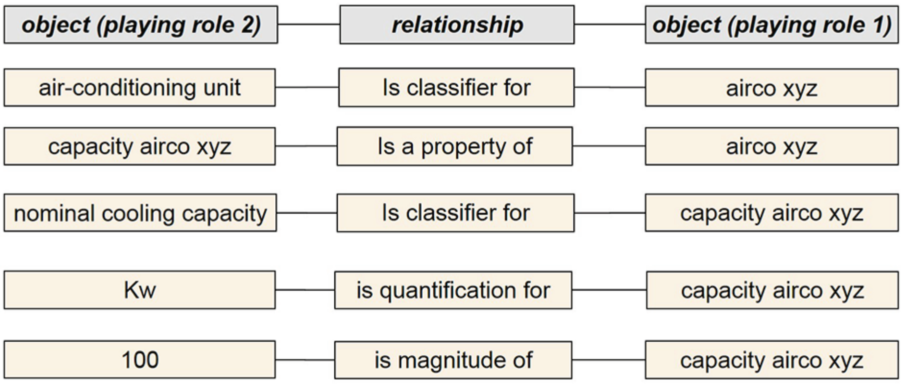

Figure 3 — Statements of Figure 2 in reverse direction

One of the advantages of using statements as shown in this clause is that one is able with one and the same methodology to describe product model data and create instance data in a repository as well but also exchange both kinds of data between parties and or configuration management systems.

4.4 Requirements of statements

In general there is, in the context of engineering data, a need to be able to make statements about statements. This ability is within in standard realized by identifying each statement by a unique identifier. Therefore each statement is provided with a Uniform Resource Identifier (URI). By referring to a specific URI one can make a statement about the statement that is represented by that URI.

EXAMPLE Who has stated a specific statement and when has he made that statement in time are examples of statement about statements.

When in industry parties communicate (exchanging statements) with each other in general there is a need for explicit statements about exchanged statements. These statements about statements support the workflow of processing industrial data along the supply chain. Together with providing provenance, these are fundamental requirements for exchanging as well as referencing and archiving industrial data electronically.

In this part the following statements about statements are recognized to facilitate engineering transactional usage:

— creator of a statement (party or role);

— date and time of creation of the statement;

— modifier of a statement (party or role);

— date and time of modification of a statement;

— certainty of a statement; possible values (instances) for certainty can be “estimated”, “calculated” or “as-build”), supporting of ranking the probability of the correct value of property values;

— modality of a statement; possible modal verbs to express the modus of a statement: possible values (instances) of modality are “Can be”, “Shall be” and “Shall have” supporting requirement management and modelling product knowledge. Default the modus of a statement will be “is the case”;

— intention of a statement, possible values (instances) of intention are: “requested”, “proposed”, “approved”. Supporting the workflow in the exchange process and change management, especially for property values;

— cardinality of an element (specific the “object playing role 2”) within a statement, supporting requirement management and product knowledge modelling;

— versioning of a statement, supporting library versioning and base lining in the context of configuration management;

— relating a statement to a specific system life cycle, making the difference between information that is relevant for the conceptual design, detailed design or the maintenance stage of a system;

— defining the “begin of life” and “end of life” of a statement in order to be able to pinpoint the period in time that the specific statement is valid.

In 5.4 and 5.5 the implementation of afore mentioned requirements is been made explicit by means of examples.

This method can define schemas as well as instances of the scheme defined in neutral format and can be implemented using any syntax layer, e.g. a spread sheet, XML, as shown in Annex B.

4.5 Representing statements in RDF triples

The Resource Description Framework (RDF) is a formal language from the W3C community that makes it possible to describe the semantics of information in similar way as the statement mechanism described in 4.3. RDF is a general-purpose language; originally meant for representing information on the Web. It defines a language for describing relationships among resources in terms of named properties and values. Properties in the context of RDF are instances of the class rdf:Property and describe a relationship between subject resources (left side of the relationship within a statement, the role 1 element) and object resources (right side of the relationship, the role 2 element within a statement). When used as such a property is a predicate in the context of RDF.

RDF provides no mechanisms for describing properties, nor does it provide any mechanisms for describing the relationships between properties and other resources. That is the role of the RDF vocabulary description language, RDF Schema (RDFS). RDFS defines classes and properties that may be used to describe classes, properties and other resources. These resources are used to determine characteristics of other resources, such as the domains and ranges of properties. RDFS vocabulary descriptions are written in RDF.

Most of the abstract model of RDF comes down to four simple rules:

— a statement is expressed as a Subject-Predicate-Object triple: this is similar to a (short) English sentence;

— subjects, predicates, and objects are given as names for entities, also called resources (dating back to the application of RDF to metadata for web resources) or nodes (from graph terminology): entities represent something, a person, website, or something more abstract like states and relations;

— names are URIs, which are global in scope, always referring to the same entity in any RDF document in which they appear;

— objects can also be given as text values, called literal values, which may or may not be typed using XML Schema data types.

The simplicity and flexibility of the triple in combination with the use of URIs for globally unique names for subjects, predicates and objects makes RDF unique, and very powerful. It is a specification that fills a very particular niche for decentralized, distributed knowledge and provides a framework to enable computer applications to answer complex questions.

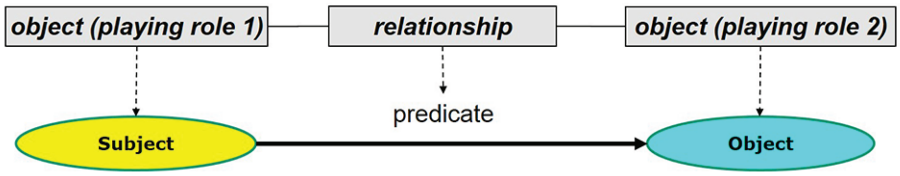

Figure 4 — Adopt RDF triple principle to represent a statement

The design of RDF can be characterized as:

— is based on a simple data model;

— enables a certain level of formal semantics and provable, traceable inference;

— uses an extensible URI-based vocabulary;

— uses an XML-based syntax, supporting XML schema data-types;

— allows anyone to make statements about any resource.

The underlying structure of any expression in RDF is a collection of triples, each consisting of a subject, a predicate and an object. A set of such triples is called an RDF graph. This can be seen in the diagram shown in Figure 4, a triple being described by as a node-arc-node link. An RDF triple is conventionally written in the order subject, predicate, object. The direction of the arc is significant: it always points toward the object by means of an arrow. This way of visualization of statements allows users to gain better grasps of models and instances of models they defined in table manner.

The assertion of an RDF triple says that some relationship, indicated by the predicate, holds between the things denoted by subject and object of the triple, in other words, visualization as an arc is useful to clearly identify the predicate (equal to a relationship) between two things. The assertion of an RDF graph amounts to asserting all the triples in it, so the meaning of an RDF graph is the conjunction (logical AND) of the statements corresponding to all the triples it contains:

— RDF provides a data model for objects and relations between them and provides a simple semantics for the data model;

— RDF Schema provides a vocabulary for describing properties and classes of RDF objects, with semantics for generalization-hierarchies of such properties and classes.

RDF is using the principle of Linked Data. Linked Data describes a method of publishing structured data so that it can be interlinked and become more useful. It enables integrating knowledge from other vocabularies into a product model making a formal reference to elements of those vocabularies in a way that can be read automatically by computer by means of SPARQL, the query language for RDF. Linked Data uses URIs to denote things. Specific HTTP URIs are used so that these things can be referred to and looked up (“dereferenced”) by people and user agents.

In this part of ISO 15926, URIs are as follows:

— a URI (Uniform Resource Identifier) is a compact string of characters that is used to identify a subject, predicate, object or Named Graph;

— a hash (#) URI is used whenever one wants to refer to something that doesn’t live on the web, with the base URI providing information about that thing;

— a slash (/) URI is used whenever one wants to refer to something that is live and addressable on the web;

— in general, a namespace is a container for a set of identifiers (names). Namespaces usually group names based on their functionality. A namespace can be part of a URI;

— an absolute URI reference consists of three parts: a scheme, a scheme-specific part and a fragment identifier.

