

**Security for industrial automation and control systems:
Patch compatibility information**

**A Progress Report for Review and Comment
From ISA99 Work Group 6 (Patch Management)**

The material in this report has been developed by the Patch Management Work Group of ISA99 (Joint with MS-MUG) for the purpose of describing a proposed specification for the exchange of patch compatibility information for industrial automation and control systems (IACS). This document is not in and of itself a work product of the ISA99 committee. Rather the expectation is that the information will appear in a broader technical report (ISA-TR99.02.03) that will be released by the committee at a later date.

This progress report is being made available to anyone who has an interest in the subject for the purpose of collecting comments and feedback to the Patch Management work group. These comments should be submitted to the authors.

ISA-99.xx.xx (Report)
Security for industrial automation and control systems: Patch compatibility information

ISBN: #####

Copyright © 2010 by ISA. All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), without the prior written permission of the publisher.

ISA
67 Alexander Drive
P.O. Box 12277
Research Triangle Park, NC 27709
www.isa.org

PREFACE

This preface, as well as all footnotes and annexes, is included for information purposes and is not part of ISA-99.xx.xx (Report).

This document has been prepared as part of the service of ISA, the Instrumentation, Systems and Automation Society, toward a goal of uniformity in the field of instrumentation. To be of real value, this document should not be static but should be subject to periodic review. Toward this end, the Society welcomes all comments and criticisms and asks that they be addressed to the Secretary, Standards and Practices Board; ISA; 67 Alexander Drive; P. O. Box 12277; Research Triangle Park, NC 27709; Telephone (919) 549-8411; Fax (919) 549-8288; E-mail: standards@isa.org.

The ISA Standards and Practices Department is aware of the growing need for attention to the metric system of units in general and the International System of Units (SI) in particular, in the preparation of instrumentation standards. The Department is further aware of the benefits to USA users of ISA standards of incorporating suitable references to the SI (and the metric system) in their business and professional dealings with other countries. Toward this end, this Department will endeavor to introduce SI-acceptable metric units in all new and revised standards, recommended practices and technical reports to the greatest extent possible. Standard for Use of the International System of Units (SI): The Modern Metric System, published by the American Society for Testing and Materials as IEEE/ASTM SI 10-97, and future revisions, will be the reference guide for definitions, symbols, abbreviations, and conversion factors.

It is the policy of ISA to encourage and welcome the participation of all concerned individuals and interests in the development of ISA standards, recommended practices and technical reports. Participation in the ISA standards-making process by an individual in no way constitutes endorsement by the employer of that individual, of ISA or of any of the standards, recommended practices and technical reports that ISA develops.

CAUTION — ISA does not take any position with respect to the existence or validity of any patent rights asserted in connection with this document, and ISA disclaims liability for the infringement of any patent resulting from the use of this document. Users are advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility.

Pursuant to ISA's Patent Policy, one or more patent holders or patent applicants may have disclosed patents that could be infringed by use of this document and executed a Letter of Assurance committing to the granting of a license on a worldwide, non-discriminatory basis, with a fair and reasonable royalty rate and fair and reasonable terms and conditions. For more information on such disclosures and Letters of Assurance, contact ISA or visit www.isa.org/StandardsPatents.

Other patents or patent claims may exist for which a disclosure or Letter of Assurance has not been received. ISA is not responsible for identifying patents or patent applications for which a license may be required, for conducting inquiries into the legal validity or scope of patents, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory.

ISA requests that anyone reviewing this Document who is aware of any patents that may impact implementation of the Document notify the ISA Standards and Practices Department of the patent and its owner.

Additionally, the use of this standard may involve hazardous materials, operations or equipment. The standard cannot anticipate all possible applications or address all possible safety issues associated with use in hazardous conditions. The user of this standard must exercise sound professional judgment concerning its use and applicability under the user's particular circumstances. The user must also consider the applicability of any governmental regulatory limitations and established safety and health practices before implementing this standard.

The following people served as active members of ISA99 Working Group 6 in the preparation of this document:

Name	Company	Contributor	Reviewer
Bill Cotter, WG Co-Chair	3M	X	
Florian Ott, WG Co-Chair	Siemens AG	X	
Dennis Brandl	BRL Consulting	X	
Jim Gilsinn, ISA99 General Editor	NIST		X
Larry McArthur			X

DRAFT

CONTENTS

PREFACE	3
FORWARD	6
INTRODUCTION	7
1 Scope	8
2 Normative references	8
3 Terms, definitions, abbreviated terms, acronyms, and conventions	8
3.1 Terms and definitions	8
3.2 Abbreviated terms and acronyms	8
4 Patch information concepts	8
4.1 Overview	8
4.2 Patch compatibility information filename convention	9
4.3 VPC file schema	9
4.4 VPC file element definitions	10
5 Patch compatibility information XSD file	12
5.1 Overview	12
6 Core component types	14
6.1 CodeType	14
6.2 DateTimeType	15
6.3 IdentifierType	15
6.4 IndicatorType	15
6.5 TextType	16
BIBLIOGRAPHY	17
Figure 1 – VPC file schema	10
Table 1 – VPC XSD file elements	10
Table 2 – CodeType Optional Attributes	14
Table 3 – DateTimeType Optional Attributes	15
Table 4 – IdentifierType Optional Attributes	15
Table 5 – IndicatorType Optional Attributes	16
Table 6 – TextType Optional Attributes	16

FORWARD

The content of this report is intended for inclusion in a future technical report that will be part of a multipart standard that addresses the issue of security for industrial automation and control systems. It has been developed by Working Group 6 of the ISA99 committee.

DRAFT

INTRODUCTION

This report defines a specification for the exchange of patch compatibility information for industrial automation and control systems (IACS). It defines a small, easy to handle file format, using an extensible markup language (XML) structure, which allows users to identify the patches that are compatible with their automation products prior to installation of the patches in their facilities.

The file format contains only the minimal patch information necessary to identify a patch, an automation product, the compatibility test status, and the compatibility test results. If the user requires additional information, such as a patch description, if a patch supersedes another patch, if this is a formally rereleased patch, etc, they should acquire that information from their patch management system.

Users should also have the possibility to:

- Create tools which could merge XML files from different vendors.
- Create tools which filter XML files associated to vendor's products and product versions.

DRAFT

1 Scope

The scope of this document is the specification of the minimal patch compatibility information necessary to identify a patch, an automation product, the compatibility test status, and the compatibility test results.

2 Normative references

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

ANSI/ISA-99.01.01-2007 – *Security for industrial automation and control systems: Terminology, concepts and models*

3 Terms, definitions, abbreviated terms, acronyms, and conventions

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISA-99.01.01 and the following apply.

3.1.1

patch

piece of software that is used to correct a problem with a software program, operating system, library, or other software element

NOTE 1 Patches are often called fixes.

NOTE 2 Service packs usually contain many different patches.

3.2 Abbreviated terms and acronyms

This subclause defines the abbreviated terms and acronyms used in this document.

CCTS	Core Components Technical Specification
IACS	Industrial automation and control systems
OS	Operating system
TC	Technical committee
UN/CEFACT	United Nations, Centre for Trade Facilitation and Electronic Business
URI	Universal resource identifier
VPC	Vendor patch compatibility
XML	Extensible markup language
XSD	XML schema definition

4 Patch information concepts

4.1 Overview

Patch information is used by industrial automation and control system (IACS) users because these systems are based on operating systems (OSs) and application software programs, and these systems require periodic fixes of newly discovered errors or to correct security deficiencies that have been discovered. Determining the compatibility of OS and library patches with automation software can be a complex task. IACS vendors will perform tests of their products against OS and library patches in order to determine if the patch should be used with the automation product. Because failures in automation products due to incompatibility with a patch may result in the loss of life, property, or product, there is often a requirement that all related automation products have been tested offline with the patch prior to installation of the patch in an online system.

IACS users often have multiple different vendor systems in their overall automation system. Managing the patch compatibility information from multiple vendors is difficult because the

patch information is usually distributed in the vendor's specific format. This report defines a standard format for the minimal patch information necessary to identify a patch and the associated product. It defines what patches may be applied to specific automation software systems. For example, it defines if a specific patch from an OS vendor has been tested against a specific version of the automation software.

The format for the minimal patch compatibility information is based on extensible markup language (XML) technology and is defined through an XML schema definition (XSD) file identified as vendor patch compatibility (VPC).

4.2 Patch compatibility information filename convention

The filename of a VPC file should following the following syntax:

```
<filename> = <vendor_name> "_patch_compatibility" <date> "_" <number> ".xml"
```

where

<vendor_name> the generally recognized short name of the IACS company

<date> the date the compatibility file was released by the vendor (formatted according to ISO 8601 [5])

<number> a number identifying the file if the vendor releases more than one file on a single date.

EXAMPLE 1 Honeywell_patch_compatibility_2010_01_08_01.xml

EXAMPLE 2 Siemens_patch_compatibility_2010_01_08_02.xml

4.3 VPC file schema

Figure 1 illustrates the VPC file schema definition.

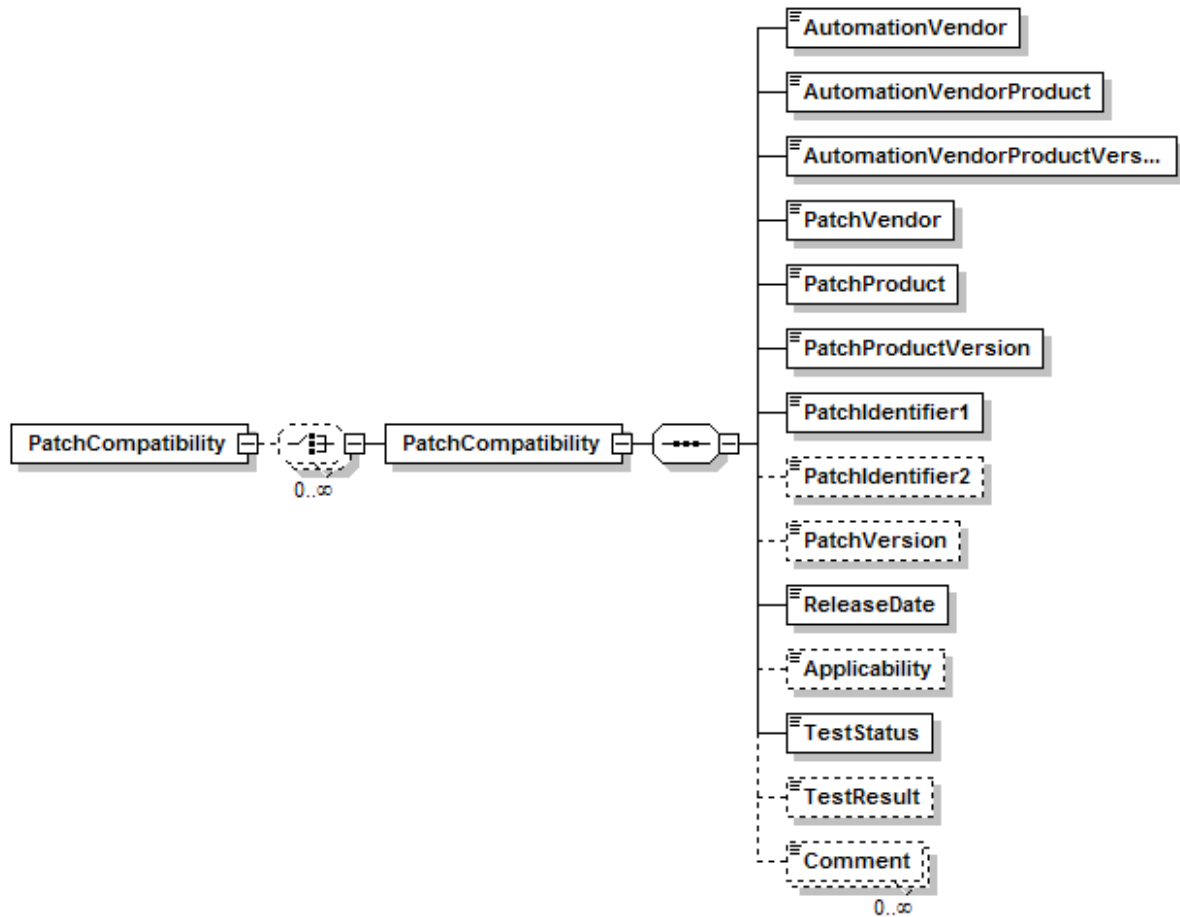


Figure 1 – VPC file schema

4.4 VPC file element definitions

Table 1 defines each element in the VPC XSD file. Only the wording, which is defined in the “Definition” column, should be used. If an element remains blank, “NA” (without quotation marks) should be used.

Table 1 – VPC XSD file elements

Element	Type	Definition
AutomationVendor	IdentifierType	A required string containing the generally recognized name of the automation system vendor. EXAMPLE "Honeywell Process Solutions", "Rockwell Automation", "Siemens"
AutomationVendorProduct	IdentifierType	A required string containing the name of the automation product the patch was tested against.
AutomationVendorProductVersion	IdentifierType	A required string containing the version of the automation product the patch was tested against.
PatchVendor	IdentifierType	A required string containing the generally recognized name of the patch vendor. EXAMPLE "Microsoft", "Adobe", "Oracle", "IBM"
PatchProduct	IdentifierType	A required string containing the vendor's name of the product the patch is for. EXAMPLE "Windows Server 2008 R2", "Acrobat Reader", "Oracle DBMS", "IBM RDBS"

Table 1 (continued)

Element	Type	Definition
PatchProductVersion	IdentifierType	A required string containing the version of the product the patch is for. EXAMPLE "Service Pack 2", "7.15", "A", "R23.9" NOTE For some products that use a service pack numbering scheme, no service pack is considered "Service Pack 0".
PatchIdentifier1	IdentifierType	A required string containing the vendor defined primary identification of the patch. EXAMPLE A Microsoft Patch KB-Number
PatchIdentifier2	IdentifierType	An optional string containing a vendor defined secondary identification of the patch. EXAMPLE A Microsoft Patch MS-Number
PatchVersion	IdentifierType	An optional string containing the version number of the patch. EXAMPLE "1", "1.0", "1.2" NOTE This may be needed if multiple versions of the patch are released due to errors in the previous patch version.
ReleaseDate	DateTimeType	A required string containing the released date of the patch. EXAMPLE "2010-01-17" NOTE Format this string according to ISO 8601.
Applicability	IndicatorType	An optional string containing an indication if the patch is allowed to be installed with the automation product. The value should be one of the following standard enumerations: <ul style="list-style-type: none"> • True → The patch may be installed with the automation product. • False → The patch should not be installed with the automation product.
TestResult	CodeType	A required string containing the results of a completed test. The value should be one of the following standard enumerations: <ul style="list-style-type: none"> • Passed → The automation product works with the patch. • Failed → The automation product does not work with the patch. • Hold → The patch is in testing with the automation product. • Not Tested → The patch was not tested with the automation product. • Conditional → The patch passes under certain conditions. The conditions are defined in the comment element.


```

        <xs:attribute name="listAgencyID" type="xs:normalizedString" use="optional" />
        <xs:attribute name="listAgencyName" type="xs:string" use="optional" />
        <xs:attribute name="listName" type="xs:string" use="optional" />
        <xs:attribute name="listVersionID" type="xs:normalizedString" use="optional" />
        <xs:attribute name="name" type="xs:string" use="optional" />
        <xs:attribute name="languageID" type="xs:language" use="optional" />
        <xs:attribute name="listURI" type="xs:anyURI" use="optional" />
        <xs:attribute name="listSchemeURI" type="xs:anyURI" use="optional" />
    </xs:extension>
</xs:simpleContent>
</xs:complexType>
</xs:complexType>
<!-- ***** -->
<!-- DateTimeType used for any date and/or time representations -->
<!-- ***** -->
<xs:complexType name="DateTimeType">
    <xs:simpleContent>
        <xs:extension base="xs:dateTime">
            <xs:attribute name="format" type="xs:string" use="optional" />
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<!-- ***** -->
<!-- IdentifierType used for any string used to identify an element -->
<!-- ***** -->
<xs:complexType name="IdentifierType">
    <xs:simpleContent>
        <xs:extension base="xs:normalizedString">
            <xs:attribute name="schemeID" type="xs:normalizedString" use="optional" />
            <xs:attribute name="schemeName" type="xs:string" use="optional" />
            <xs:attribute name="schemeAgencyID" type="xs:normalizedString" use="optional" />
            <xs:attribute name="schemeAgencyName" type="xs:string" use="optional" />
            <xs:attribute name="schemeVersionID" type="xs:normalizedString" use="optional" />
            <xs:attribute name="schemeDataURI" type="xs:anyURI" use="optional" />
            <xs:attribute name="schemeURI" type="xs:anyURI" use="optional" />
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<!-- ***** -->
<!-- IndicatorType used for any boolean element -->
<!-- ***** -->
<xs:simpleType name="IndicatorType">
    <xs:restriction base="xs:boolean">
        <xs:pattern value="false" />
        <xs:pattern value="true" />
    </xs:restriction>
</xs:simpleType>
<!-- ***** -->
<!-- TextType used for any element that requires a string value -->
<!-- ***** -->
<xs:complexType name="TextType">
    <xs:simpleContent>
        <xs:extension base="xs:string">
            <xs:attribute name="languageID" type="xs:language" use="optional" />
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<!-- ***** -->
<!-- Patch element used to contain patch compatibility information -->
<!-- ***** -->
<xs:element name="Patch" msdata:IsDataSet="true" msdata:Locale="en-US">
    <xs:complexType>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="Patch">
                <xs:complexType>
                    <xs:sequence>
                        <!-- Identify the Automation vendor, product, and version information -->
                        <xs:element name="AutomationVendor" type="IdentifierType" />
                        <xs:element name="AutomationVendorProduct" type="IdentifierType" />
                        <xs:element name="AutomationVendorProductVersion" type="IdentifierType" />
                        <!-- Identify the Patched product, vendor, version information -->
                        <xs:element name="PatchVendor" type="IdentifierType" />
                        <xs:element name="PatchProduct" type="IdentifierType" />
                        <xs:element name="PatchProductVersion" type="IdentifierType" />
                        <xs:element name="PatchIdentifier1" type="IdentifierType" />
                        <xs:element name="PatchIdentifier2" type="IdentifierType" />
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="PatchVersion" type="IdentifierType" />
        </xs:choice>
    </xs:complexType>

```

```

                                minOccurs="0" />
<xs:element name="ReleaseDate"          type="DateTimeType" />
<!-- Identify the applicability, test status, results, and other comments -->
<xs:element name="Applicability"        type="IndicatorType"
                                minOccurs="0" />
<xs:element name="TestStatus"           type="CodeType" />
<xs:element name="TestResult"           type="CodeType"
                                minOccurs="0" />
<xs:element name="Comment"              type="TextType"
                                minOccurs="0" maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

6 Core component types¹

The base types for most elements are derived from core component types that are compatible with the United Nations, Centre for Trade Facilitation and Electronic Business (UN/CEFACT) [8] core component types. The UN/CEFACT core component types are a common set of types that define specific terms with semantic meaning (for example, the meaning of a quantity, currency, amount and identifier). The UN/CEFACT core components were defined in a Core Components Technical Specification (CCTS) developed by the ebXML [9] project now organized by UN/CEFACT and ISO technical committee (TC) 154.

NOTE The core components contain optional attributes that may be used to specify the context and source of the associated element value. All attributes are optional in the VPC schema.

The core components use several international standards for the representation of semantic and standardized information:

- Country code [2]
- Region code [3]
- Language code [1]
- Currency code [4]
- Date and time representation [5]
- Unit of measure code [6]
- Unit of transport or packaging code [7]

6.1 CodeType

`CodeType` is used to define a character string that is used to represent an entry from a fixed set of enumerations. It is derived from the type `normalizedString`. All of the VPC enumerations are derived from `CodeType`. Table 2 describes the optional attributes for the `CodeType` data type.

Table 2 – CodeType Optional Attributes

Optional Attribute	Base XML Type	Description
listID	normalizedString	An identifier specifying a code list that this is registered with an agency. EXAMPLE UN/EDIFACT data element 3055 code list
listAgencyID	normalizedString	An identifier specifying the agency that maintains one or more lists of codes. EXAMPLE UN/EDIFACT
listAgencyName	string	Text containing the name of the agency that maintains the list of codes.

¹ This section provided courtesy of WBF. [10]

Optional Attribute	Base XML Type	Description
listName	string	Text containing the name of a code list that this is registered with at an agency.
listVersionID	normalizedString	An identifier specifying the version of the code list.
name	string	Text equivalent of the code content component.
languageID	language	An identifier specifying the language used in the code name.
listURI	anyURI	The uniform resource identifier (URI) identifying where the code list is located.
listSchemaURI	anyURI	The URI identifying where the code list schema is located.

6.2 DateTimeType

`DateTimeType` is used to define a particular point in time together with the relevant supplementary information to identify the time zone information. It is derived from the type `dateTime`. In VPC this is a specific instance on time using the ISO 8601 Common Era calendar extended format and abbreviated versions.

EXAMPLE `yyyy-mm-ddThh:mm:ssZ` for UTC as “2002-09-22T13:15:23Z”

Table 3 describes the optional attributes for the `DateTimeType` data type.

Table 3 – DateTimeType Optional Attributes

Optional Attribute	Base XML Type	Description
format	string	A string specifying the format of the date time content. NOTE 1 The format of the attribute is not defined in UN/CEFACT specification. NOTE 2 This attribute is not needed in VPC, but is maintained for compatibility with Open Applications Group Integration Specification (OAGiS) and WBF use.

6.3 IdentifierType

`IdentifierType` is used to define a character string to identify and distinguish uniquely, one instance of an object in an identification schema from all other objects in the same schema. It is derived from the type `normalizedString`. Table 4 describes the optional attributes for the `IdentifierType` data type.

Table 4 – IdentifierType Optional Attributes

Optional Attribute	Base XML Type	Description
schemaID	normalizedString	An identifier specifying the identification schema.
schemaName	string	Text containing the name of the identification schema.
schemaAgencyID	normalizedString	An identifier specifying the agency that maintains the schema.
schemaAgencyName	string	Text containing the name of the agency that maintains the schema.
schemaVersionID	normalizedString	The version (as an identifier) of the schema.
schemaDataURI	anyURI	The URI identifying where schema data is located.
schemaURI	anyURI	The URI identifying where schema is located.

6.4 IndicatorType

`IndicatorType` is used to define a list of two mutually exclusive boolean values that express the only possible states of a property. It is derived from the type `string`. For VPC purposes the defined values for indicator type is “True” and “False”. Table 5 describes the optional attributes for the `IndicatorType` data type.

Table 5 – IndicatorType Optional Attributes

Optional Attribute	Base XML Type	Description
format	string	A string specifying whether the indicator is numeric, textual or binary. NOTE The format of the format attribute is not defined in UN/CEFACT specification.

6.5 TextType

`TextType` is used to define a character string (for example, a finite set of characters) generally in the form of words of a language. It is derived from the type `string`. Table 6 describes the optional attributes for the `TextType` data type.

Table 6 – TextType Optional Attributes

Optional Attribute	Base XML Type	Description
languageID	language	An identifier specifying the language used in the content component.
languageLocaleID	normalizedString	An identifier specifying the locale of the language.

BIBLIOGRAPHY

NOTE The references in this bibliography have been broken into groups depending on the type of source they are.

Standards references:

- [1] ISO 639-1:2002 – *Codes for the representation of names of languages -- Part 1: Alpha-2 code*
- [2] ISO 3166-1:2006 – *Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes*
- [3] ISO 3166-2:2007 – *Codes for the representation of names of countries and their subdivisions -- Part 2: Country subdivision code*
- [4] ISO 4217:2008 – *Codes for the representation of currencies and funds*
- [5] ISO 8601:2004 – *Data elements and interchange formats -- Information interchange -- Representation of dates and times*
- [6] ECE/TRADE/C/CEFACT/2009/24 – *Codes for Units of Measure used in International Trade*
- [7] ECE/TRADE/C/CEFACT/2009/25 – *Codes for Passengers, Types of Cargo, Packages and Packaging Materials (with Complementary Codes for Package Names)*

Websites:

- [8] United Nations, Centre for Trade Facilitation and Electronic Business (UN/CEFACT), available at <http://www.unece.org/cefact>
- [9] Electronic Business using eXtensible Markup Language (ebXML), available at <http://www.ebxml.org>
- [10] Organization for Production Technology, available at <http://www.wbf.org/>