STANDARD ST.36

Version 1.2

RECOMMENDATION FOR THE PROCESSING OF PATENT INFORMATION USING XML (EXTENSIBLE MARKUP LANGUAGE)

Revision adopted by ST.36 Task Force of the Standards and Documentation Working Group (SDWG) on November 23, 2007

TABLE OF CONTENTS

Introduction

1. This Standard recommends the XML (eXtensible Markup Language) resources used for filing, processing, publication, and exchange of all types of patent information. It is based in large part on *Patent Cooperation Treaty, Administrative Instructions, Part 7, Annex F, Appendix I* (hereafter referred to as Annex F). The term “XML resources” is intended to refer to any of the components used to create and operate an XML implementation. Although XML resources normally encompasse style sheets, W3C Schemas, and other objects, this Standard presently includes only document type definitions (DTDs), content models, elements, and a small set of character entities. For further information about the W3C (World Wide Web Consortium), see http://www.w3c.org/.

2. This Standard is an application of the *Extensible Markup Language (XML) 1.1.*
   See: http://www.w3.org/TR/2004/REC-xml11-20040204/:

   “*The Extensible Markup Language (XML) is a subset of SGML that is completely described in this document. Its goal is to enable generic SGML to be served, received, and processed on the Web in the way that is now possible with HTML. XML has been designed for ease of implementation and for interoperability with both SGML and HTML.*”

3. The mark-up included in an XML instance that is in compliance with this Standard is an example of the representation of the contents of a document using XML whereby “*documents are made up of storage units called entities, which contain either parsed or unparsed data. Parsed data is made up of characters, some of which form character data, and some of which form markup. Markup encodes a description of the document’s storage layout and logical structure. XML provides a mechanism to impose constraints on the storage layout and logical structure*” (W3C).

4. XML cannot be used *per se* as the basis for patent document processing – “*This specification does not constrain the semantics,*

use, or (beyond syntax) names of the element types and attributes ..." (W3C).

5. Therefore, this Standard defines elements and their generic identifiers, or "tags", and attributes for marking up patent documents. That is, this Standard provides for some level of the semantics (meaning), the use, and the names of the elements and attributes that make up the various document types it discusses.

   " [*Definition: Each XML document contains one or more elements, the boundaries of which are either delimited by start-tags and end-tags, or, for empty elements, by an empty-element tag. Each element has a type, identified by name, sometimes called its "generic identifier" (GI), and may have a set of attribute specifications.] Each attribute specification has a name and a value.*" (W3C)

   Note: For a complete description and definitions refer to the XML specification at http://www.w3.org/TR/2004/REC-xml11-20040204/.

6. The purpose of the Standard is to provide logical, system-independent structures for patent document processing, whether for text or image data. That means that this Standard may be used in place of WIPO Standards ST.30, ST.32, ST.33, and ST.35 for filing, processing, publishing, and exchanging bibliographic data, abstracts, or full text of all patent document types. This Standard provides XML resources for the following data:

   (a) Full or partial text of patent documents, including bibliographic data, recorded as character coded-data.

   (b) Whole pages of documents represented as one image (page images) irrespective of their content (bibliographic data, text, or images).

   (c) Data, within full-text documents, which cannot be recorded as character-coded data, such as drawings, chemical formulae, especially complex tables (so-called embedded images).

7. XML instances that conform to this Standard must be well-formed XML, conforming to one of the document type definitions (DTDs) contained in Annex F or to an office-specific DTD that itself conforms to this Standard. A DTD that conforms to this Standard must be built from the elements according to the guidelines in this Standard. Annex F DTDs are published at http://www.wipo.int/pct-safe/epct/xml_canon.htm, where the DTDs will be updated as soon as any modification is approved. Once an updated DTD appears at the Web site, it is available for official use.

Definitions

8. For the purposes of this Standard, the following definitions are given:

   (a) The expression **patent document** includes patents for invention, plant patents, design patents, utility certificates, utility models, documents of addition thereto, published applications and specifications, document types related to the prosecution of patents, including post-grant activities, property-rights maintenance, and all office-to-applicant and office-to-office communications.

   (b) **Markup** is defined as text that is added to the content of a document and that describes the structure and other attributes of the document in a non-system-specific manner, independently of any processing that may be performed on it.

   (c) For other definitions see the XML specification at http://www.w3c.org/TR/2004/REC-xml11-20040204/.

Scope of the standard

9. Although the DTDs referenced in Annex F were designed for use under the Patent Cooperation Treaty, it is the ambition of this Standard that they should be used by all patent offices for electronic filing. The model DTD which is reproduced as Annex A to the Standard is intended to guide the use of the international common elements (ICEs) for publishing patent documents. As the Standard evolves, other DTDs may be added to the list below.

Industry-standard DTDs incorporated by reference

- mathml2.dtd
- soextblx.dtd (also referenced as calstblx.dtd)

10. Some Annex F DTDs are also listed below with their corresponding business process as an illustration of their intended use. The table is only a guide to the possible use of these DTDs in the patent business process; different offices may have different needs.

DTD NAME	BUSINESS PROCESS						
	FILING	PUBLISHING	PROSECUTION	GRANT	POST	RE-PUBLISHING	CORRESPONDENCE

					GRANT		
amendment-request	X					X	
application-body	X	X		X		X	
bio-deposit	X	X		X		X	
declaration	X						X
demand	X						X
dispatch	X						
fee-sheet	X						X
iprp			X				X
package-data	X	X	X	X	X	X	X
pkgheader	X	X	X	X	X	X	X
power-of-attorney	X		X	X	X		X
priority-doc	X						X
request	X						
search-report		X				X	X
table-entity	X	X		X		X	
xmit-receipt	X		X		X		X
xx-patent-document		X		X		X	

Requirements of the standard

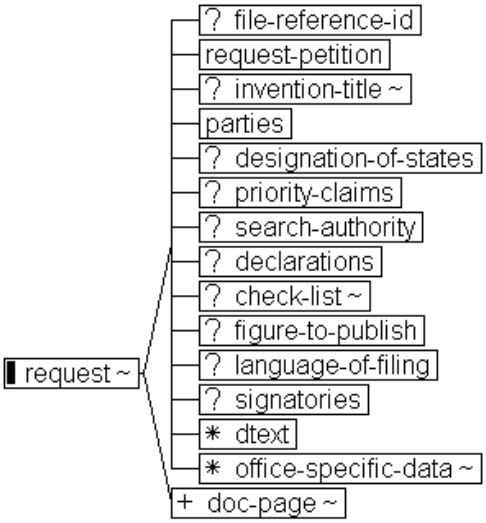
General

11. ICEs are the foundation of this Standard. ICEs are derived from Annex F, WIPO Standard ST.32, and other sources. See http://www.wipo.int/standards/en/xml_material/st36/index.html.

12. ICEs must be used as defined in this Standard, that is, they must have the same name, the same contents, the same attributes and the same meaning as indicated in the list of ICEs. It is understood that this Standard and Annex F cannot possibly include all elements required by all patent offices; in such instances, office-specific elements are allowed as described below.

13. Office-specific information may be treated as follows.

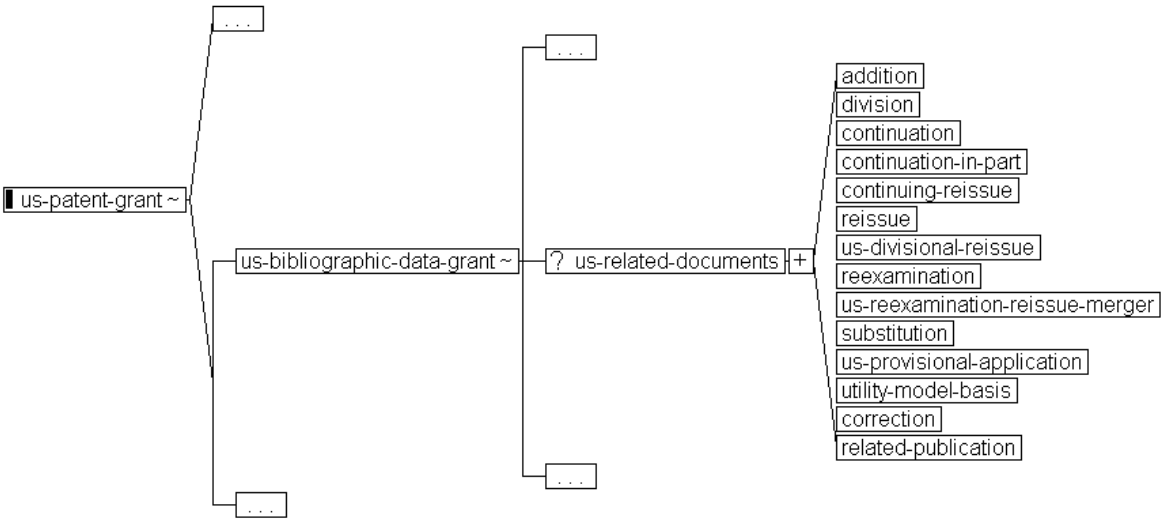
    (a) Segregated in a separate DTD referenced, for example, from the `request` DTD by the `office-specific-data` element (recommended).



    (b) Included directly within the `office-specific-data` element, in which case the element may be changed from empty to include `#PCDATA` or other content models as needed; and add the two-letter country code prefix to `office-specific-data`. For example, `wo-office-specific-data`. The content model of `office-specific-data` must not be modified without adding the office prefix.

(c) Use the XML namespace convention. XML namespaces provide a simple method for qualifying element and attribute names used in XML documents by associating them with namespaces identified by URI (universal resource identifier) references. (see: http://www.w3.org/TR/REC-xml-names/)

(d) Mix office-specific elements with international common elements. For example, in the publishing DTD fragment below, office-specific elements are added to the content model of the `related-documents` element. For further details, see *DTD Conventions* below.



14. At a higher level this data may be included in separate documents referenced from the `package-data` DTD by the `other-documents` element. The DTDs or tags referenced by the `office-specific-data` or `other-documents` are entirely under the control of the responsible office.

15. The name of office-specific DTDs and/or elements shall begin with the ST.3 two-letter country code of the corresponding office, followed by a separator (hyphen or colon) and the name of the entity. Any other names will be understood as being international (generic) DTDs or elements. Therefore, it is advised to restrict the use of names that begin with a two-letter word to only those that represent a valid country code. For example, `request` becomes `ep-request` when modified for use by the EPO, both for the DTD file name and for the root element.

16. For filing interoperability among patent offices, it is necessary to use the following DTDs as defined in Annex F, and not any office-specific alternatives: `application-body, table-external, pdoc-certificate, package-header, package-data,` and `xmit-receipt`.

17. Where instances contain office-specific elements and/or reference office-specific DTDs the issuing authority shall provide constructive notice to other offices and users containing information about the content and meaning of those elements and/or DTDs. Such notice should be given at a readily available web site maintained by the office or at WIPO's web site. The notice should include the DTDs and a complete description of each of the elements.

Characters

18. Although XML permits other character encodings, this Standard recommends Unicode exclusively. It may be useful to add character entities for characters not yet in Unicode, such as those listed in `wipo.ent` (located at http://www.wipo.int/pct-safe/epct/xml_canon.htm). This entity file provides general entity names that can be used in instances in place of the code points from the encodings that they are mapped to in `wipo.ent`. Use of these entities requires the creation of glyphs for presentation, which do not yet exist. See http://www.w3.org/XML/Core/2002/10/charents-20021023 for further information about character entities.

19. Document instances must include an XML declaration as the first line in the file.

```
<?xml version='1.1' encoding='utf-8' ?>
```

20. Note that only UTF-8 is recommended in this Standard. However, in the case of ideographic scripts, Unicode in UTF-8 may produce exceptionally large files since the encoding may use up to four' bytes per character. In such cases, national offices may select an encoding that brings files to manageable sizes. Offices that elect to do so, should be prepared to consult with their exchange partners and to give adequate public notice.

21. The characters that are permitted to appear in an XML document are specified in the *XML 1.1 W3C Recommendation*, and are endorsed by this Standard with the following exception. The characters used in element or attribute names described in this Standard are restricted to the following set:

    {`abcdefghijklmnopqrstuvwxyz1234567890-`}.

22. Offices are strongly encouraged to create document instances for publication and exchange that have been “normalized” in accord with the *Character Model for the World Wide Web* (http://www.w3.org/TR/2003/WD-charmod-20030822/). Parsers that support XML 1.1 can be used to test for normalization. Doing so will significantly improve the consistency of sorting and string comparison operations by ensuring that certain character encoding options available in Unicode will have been applied consistently throughout the international patent community.

Naming international common elements

23. All element names should be words from the English language.

24. Where more than one word is required for an element name, the words shall be separated by a hyphen (-).

25. Element names in this Standard use the Latin alphabet only, limited to the following set of characters: {`abcdefghijklmnopqrstuvwxyz1234567890-`}. Accented characters and uppercase characters are not used. For historical reasons, the element names in the element SDOBI derived from Standard ST.32 retain their uppercase B and other uppercase element names.

26. Names shall be descriptive, not mnemonic or abbreviated, as far as practical. The goal should be that anyone can understand the meaning of the element name with little or no reference to any other documentation. Some notable exceptions include the most common elements used in a patent application, such as `p` for paragraph and others derived from, for example, HTML, and some other widely-used formatting elements (for example, study the `application-body` DTD). It is unlikely that any further exceptions will be required.

27. One or two sentences describing the meaning of the element name and the intended contents of the element shall be provided. The description should cite any applicable rules or regulations and very briefly summarize their substance. In a DTD, the description should be encapsulated in a comment immediately preceding the element or attribute to which it applies.

28. For historical reasons, some elements in ICEs have an entry in the ST.32 Name column in the form `Bnnn` where `n` is a number. These element names are the corresponding elements from *WIPO Standard* ST.32*, Recommendation for the Markup of Patent Documents Using SGML (Standard Generalized Markup Language)*. In due course this column may be removed as offices abandon the use of the `Bnnn` tag names.

Naming office-specific elements

29. The rules for ICEs (previous section) apply.

30. All office-specific element names shall be words from the English language, wherever possible.

31. Each office-specific element name shall be preceded by the ST.3 code for the office that owns the element. The ST.3 code shall be separated from the element name by either a hyphen (-) or a colon (:). For example, `jp:fterm`, or `ep-printer-name`. The colon is used only where the owning office is implementing W3C XML namespaces (see *Namespaces in XML*, http://www.w3.org/TR/REC-xml-names/).

Attributes

32. If an office wishes to add or modify attributes for an ICE, a change request shall be submitted.

33. Office-specific elements may have whatever attributes they require, provided they do not conflict with attributes defined in this Standard (see following paragraphs).

34. Attribute names should not be redefined within a DTD. That is, the name should always have the same meaning, no matter what element it happens to be applied to. A comment should explain the possible values that the attribute can have, what they mean, and, where appropriate, how to construct them. The attribute comment should be included with the element comment to which the attribute belongs.

35. Attribute names should be reused where the permitted attribute values are identical, or the attribute value meaning is the same. If the attribute values are not the same, the names should be different as well. For example, the `file` attribute should be used wherever the attribute value is the name of a computer file. No other attribute name should be used for this purpose.

Adding, deprecating, or changing elements

36. Adding, deprecating, or changing ICEs requires approval according to the procedures established by WIPO.

37. Once it is determined that an ICE is deprecated (it shall no longer be used), the change of status will be noted in the ICEs repository and the date after which the element shall no longer be used.

38. Adding, deprecating, or changing office-specific elements requires only the approval of the industrial property office creating the element. Nevertheless, offices are expected to ensure that office-specific elements are distinct from ICEs in their meaning and description.

39. An office-specific element that is found to be of utility by offices other than the one that created it can be promoted to a common element. Promotion normally consists of deprecating the office-specific element and adding a new ICE that has the same name as the office-specific element minus the ST.3 code prefix.

40. Once an office has determined that an office-specific element is deprecated (it will no longer use the element), the office will request that the status of the element in the repository be changed along with the date after which the office will no longer use the element.

41. Where practical, change requests to modify DTDs affected by the addition, deprecation, or change of a common or office-specific element should be submitted at the same time as the request creating, deprecating, or modifying the element. Implementation of the modified elements and DTDs shall coincide wherever possible.

Element and attribute conventions

42. Certain elements and attributes have special significance within the international common elements. Their intended use is described in this section because of the significant advantages that follow if they are treated uniformly throughout the industrial property community.

43. The `date` element is used extensively throughout the content models of `xx-patent-document` for every occurrence of a date. Consequently, it must always appear as the child of another element that sets the context and explains what event the date corresponds to. Even if the word date appears in the parent element, the `date` element must be used within it to contain the actual date. The content of the element must always be expressed as a string of numbers where the first four positions represent the year, the next two positions represent the month (left padded with zero where needed), and the last two positions represent the day of the month (left padded with zero where needed). For example, `20040717` represents 2004 July 17th.

44. The `document-id` element is used to identify every type of industrial property document to which this Standard applies. Consequently, it must always appear as the child of another element that sets the context and explains what document is being identified. The `date` element within `document-id` represents the date of publication for published documents or the date of filing for unpublished documents. No other date should be used. The `doc-number` should not include the kind code or date or country, all of which are stored separately. Where it is important to display the document identification according to WIPO Standard ST.10/C or some other WIPO standard, use a style sheet to arrange the content as required. The `name` element is used for one or another of the parties associated with the document.

45. Offices are encouraged to tag every occurrence of every kind of document number in a published document. The benefit to search systems will be significant.

46. The `id` attribute is never required, but its use can add significant value to documents. The value assigned to an `id` attribute is required by the XML standard to be unique within a document instance. It has no other purpose than to uniquely identify the object to which it is attached within the instance and should not be used for any other purpose. Although the value of an `id` can be any string, it may be useful for human readability to set the value to represent the type of element that it is assigned to. For example, the paragraphs in a disclosure might be assigned `id` values such as “`p0001`”, “`p0002`” … “`p0101`”, etc.

47. Care must be taken when assigning values in this manner. For example, if more than one set of claims is published (in multiple languages, perhaps), then there will be more than one claim numbered “1”. In this case, the `id` values must still be unique, so the practice described here must be modified to ensure uniqueness, for example, by including an indication of the language,

such as “cl-en-0001”, “cl-de-0001”, “cl-fr-0001”, etc.

48. If amendments are to be processed automatically, it may useful to make the id attribute values unique within a file wrapper. In this case, when an id is assigned to a paragraph, for example, it should never be changed no matter how many times the document itself is modified. When a paragraph is deleted, the id value should be retired for that file wrapper. When a new paragraph is added, an id value new to the file wrapper should be assigned.

49. An id attribute is frequently the target of an idref attribute associated with a claim-ref, crossref, or figref. The pairing of id and idref values supports hypertext linking when the instance is displayed in a browser. The crossref element is intended to point to any arbitrary object in the document instance, other than a claim or a figure. To avoid conflicts with established practice in browsers, it is important to not use these attributes for any other purpose.

50. The core DTDs and content models of elements that are in the core DTDs cannot be modified for national use (referred to herein as core elements). Even where a core element is used in a DTD that is not a core DTD, it cannot be modified. In the case where an office wishes to insert additional information into a core element, id and idref can be used to accomplish this without modifying the core element. Assign an id to the element in question. Create an office-specific element with an idref attribute that points to the core element. In the office-specific element, add whatever additional elements are required for national use.

51. The lang attribute normally contains a two-letter code based on ISO standards for the language of the content of the element to which it is attached. In cases where the two-letter code is not adequate, offices are encouraged to follow the conventions established by the Internet Engineering Task Force and described in *Tags for the Identification of Languages* (http://www.rfc-editor.org/rfc/rfc3066.txt).

DTD conventions

52. The file name of a DTD shall indicate its version. The file name with version number shall be a string resulting from concatenating the document type followed by a hyphen followed by the literal ‘v’ followed by a digit signifying the major version followed by an optional section consisting of a hyphen followed by a minor revision number digit.
Example: request-v1-2.dtd

53. Each DTD shall include a version number and version date in the public identifier. Examples:
PUBLIC "-//WIPO//DTD APPLICATION BODY 1.3//EN" "application-body-v1-3.dtd"

Reference this DTD as PUBLIC "-//USPTO//DTD us-patent-grant v4.0 2004-07-30//EN"
Alias: Grant Red Book (GRB)

54. A revision history at the beginning of the DTD shall document changes with a date and description of each change, in reverse chronological order. Examples:

```
<!--
***** Revision History *****
2004-04-15 H. Li
. Added num attribute to li element.
. Changed content model maths from (img | math) to (img | (math,img?)).
. Added date attribute to element us-patent-application.
. Added math, chemistry, table, program-listing to img-content allowed values
.. and changed dna to DNA.
. Add figref to li, dd, claim-text elements and table cell entry.
. Changed subname? to subname* in serial, article and online elements.
. Renamed table-external element to table-external-ref.
2004-03-04 B. Cox
. Renamed us-continued-prosecution-application to
.. us-issued-on-continued-prosecution-application
.. and renamed attribute from cpa-text to grant-cpa-text
.. and changed attribute value (see below).
***** End Revision History *****
```

```
-->
```

55. Include points of contact in the prologue. Examples:

```
<!--
Contacts:
EPO: Paul Brewin pbrewin-z@not-epo.org
JPO: Shiro Ankyu ankyu-shiro-z@not-jpo.go.jp
USPTO: Bruce B. Cox bruce.cox-z@not-uspto.gov
WIPO: Hideto Tanaka hideto.tanaka-z@not-wipo.int
-->
<!-- Contact: Bruce B. Cox
U.S. Patent and Trademark Office
Crystal Park 8, Suite 1032
Washington, DC 89231
+1-783-906-6062
bruce.cox@uspto.gov-->
```

56. DTDs that incorporate industry-standard DTDs by reference are often unreadable (cannot be parsed) if the external DTDs are not available on the local machine. The following example provides a method for easily hiding such external references temporarily, enabling editing and viewing of the DTD where access to the external references is not convenient. This solution presumes knowledge of marked sections in DTDs. (For further explanation, see http://etext.lib.virginia.edu/TElp4/tei.cgi?div=div2&id=SG17BIS and similar Web sites.)

```
<!--to include mathml2.dtd change MATHML2_DTD value to "INCLUDE",
change MATH_PLACEHOLDER value to "IGNORE", the same for the TABLE_DTD,
TABLE_PLACEHOLDER, and WIPO_ENT
INCLUDE
IGNORE
-->
<!ENTITY % UNICODE_PLANE1D_ESCAPE "IGNORE">
<!ENTITY % WIPO_ENT "IGNORE">
<!ENTITY % MATHML2_DTD "IGNORE">
<!ENTITY % TABLE_DTD "IGNORE">
<!ENTITY % MATH_PLACEHOLDER "INCLUDE">
<!ENTITY % TABLE_PLACEHOLDER "INCLUDE">
<![%UNICODE_PLANE1D_ESCAPE; [
<!ENTITY % plane1D "&#38;#38;#xE">
]]>
<![%WIPO_ENT;[
<!--
import character entity set. Download from:
http://pcteasy.wipo.int/efiling_standards/schemaDocs/wipo.ent
Note that nsgmls-based parsers (SP, Near & Far Designer, etc.)
may not be able to process this file for reasons described below
in MathML comments.
-->
<!ENTITY % wipo PUBLIC "-//WIPO//ENTITIES WIPO 1.0//EN" "wipo.ent">
%wipo;
]]>
```

```
<![%MATHML2_DTD; [
<!-- DTD MathML2: maintained by W3C. Download from:
http://www.w3.org/TR/MathML2/DTD-MathML-20010221.zip
If using nsgmls-based parser (SP, Near & Far Designer, etc.)
Uncomment 'mathml-charent-module' switch below or replace the
Referenced MathML2 DTD with the version downloadable from:
http://www.w3.org/Math/DTD/dtd-sp.zip
This notice copied from: http://www.w3.org/Math/DTD/
"DTD for nsgmls
Some systems (including the popular nsgmls parser) may not be able
to process files using 'plane 1' characters which have Unicode
numbers higher than #xFFFF. The versions of the DTD provided here
incorporate the modifications mentioned above, but the high
characters are replaced by the equivalent mchar construct
<mchar name="..." /> this allows the DTD to be read and for MathML
files to be validated using such systems."
-->
<!--ENTITY % mathml-charent.module "IGNORE" -->
<!ENTITY % MATHML.prefixed "IGNORE">
<!ENTITY % MATHML.xmlns "">
<!--import MathML2 dtd -->
<!ENTITY % mathml2 PUBLIC "-//W3C//DTD MathML 2.0//EN" "mathml2.dtd">
%mathml2;
]]>
<![%TABLE_DTD; [
<!-- DTD OASIS Open XML Exchange Table Model.
Maintained by OASIS; download from:
http://oasis-open.org/specs/soextblx.dtd
Note that the FPI in soextblx.dtd refers to itself as 'calstblx'.
That convention has been followed here.
-->
<!-- create content for title element in table -->
<!ENTITY % title "<!ELEMENT title (#PCDATA | b | i | u | sup | sub | smallcaps)* > ">
%title;
<!--override OASIS Exchange <entry> model -->
<!ENTITY % tbl.entry.mdl "(#PCDATA | b | i | u | sup | sub | smallcaps | br
| patcit | nplcit | bio-deposit | crossref | figref | img
| dl | ul | ol | chemistry | maths)* ">
<!--import OASIS Exchange model -->
<!ENTITY % calstblx PUBLIC "-//OASIS//DTD XML Exchange Table Model 19990315//EN"
"soextblx.dtd">
<!ENTITY % yesorno "NMTOKEN" >
<!ENTITY % tbl.table.att " pgwide %yesorno; #IMPLIED
orient (port | land) #IMPLIED
tabstyle NMTOKEN #IMPLIED”>
```

```
%calstblx;
]]>
<![%MATH_PLACEHOLDER; [
<!--(PLACEHOLDER:w3c math dtd)-->
<!ELEMENT math (#PCDATA)>
]]>
<![%TABLE_PLACEHOLDER; [
<!--(PLACEHOLDER:cals table dtd)-->
<!ELEMENT table (#PCDATA)>
]]>
```

57. The DTD should normally list the elements starting with the root element and sorted branch by branch, in natural document order, with attributes immediately following the element to which they apply. However, any arrangement that aids users to understand the logical structure of the document is acceptable.

58. As stated above, for every element in the DTD, include a comment immediately preceding the element, where necessary to understand its meaning or use. The comment should provide sufficient information to make proper use of the element or attribute and reference more complete documentation, where necessary. Examples:

```
<!--The problem the invention purports to solve (Rule 5.1(a)(iii))-->
<!ELEMENT tech-problem (heading* , p+)+>
<!ATTLIST tech-problem id ID #IMPLIED >
```

59. Rules for ICEs DTDs apply to office-specific DTDs.

60. DTDs should use ICEs wherever possible, reserving office-specific elements only for those cases where national practice differs significantly from international practice.

61. Office-specific elements can be inserted wherever needed in an otherwise WIPO Standard DTD. Such a modified DTD becomes an office-specific DTD and must be named accordingly, that is, the root element, the file name, and the public identifier for an office-specific DTD must begin with the corresponding ST.3 office code.

62. Any DTD that includes even one office-specific element is by definition an office-specific DTD.

Document instance conventions

63. The following specifications govern the construction of document instances in conformance with this Standard.

64. Instances that conform to this Standard shall be well formed and in conformance with XML v1.1.

65. It is recommended that document instances follow the file naming convention described in Annex F (published on WIPO's web site at www.wipo.int/pct/en/texts/index.htm) for filing and processing. For example, the following file list is for a published US patent grant with seven pages of images.

```
US06282717-20010904.xml
US06282717-20010904-D00000.TIF
US06282717-20010904-D00001.TIF
US06282717-20010904-D00002.TIF
US06282717-20010904-D00003.TIF
US06282717-20010904-D00004.TIF
US06282717-20010904-D00005.TIF
US06282717-20010904-D00006.TIF
```

66. For international applications, each PCT receiving office must generate instances that conform to the full complement of Annex F specifications.

67. XML document instances shall begin with a prologue section. The prologue should contain an XML processing instruction that specifies the version of XML and the encoding scheme. The prologue shall also contain a document type declaration, also known as a `DOCTYPE` statement. The document type declaration may optionally include a public identifier (prefaced with the keyword `PUBLIC`) and must contain a system identifier (prefaced by the keyword `SYSTEM`, unless a `PUBLIC` keyword has been used), which specifies a URI (Uniform Resource Identifier) referencing the DTD that the document instance can be validated against. Public identifiers may be found in the comments at the start of each DTD file.

    Example - prolog with Public identifier:

    ```
    <?xml version="1.1" encoding="UTF-8"?>
    <!DOCTYPE request PUBLIC "-//WIPO//DTD REQUEST 1.1//EN" "request-v1-1.dtd">
    ```

    Example - prolog with System identifier:

    ```
    <?xml version="1.1" encoding="UTF-8"?>
    <!DOCTYPE request SYSTEM "request-v1-1.dtd">
    ```

68. Each document instance shall declare unambiguously the DTD to which it conforms, without exception. This declaration imposes no constraints on the processing of the instance at any office.

    Root element with version attribute examples:

    ```
    <request dtd-version="request-v1-1.dtd">
    <request dtd-version="1.1">
    ```

    Prologue with `DOCTYPE` declaration examples:

    ```
    <?xml version="1.1" encoding="UTF-8"?>
    <!DOCTYPE request SYSTEM "request-v1-1.dtd" >
    <?xml version="1.1" encoding="UTF-8"?>
    <!DOCTYPE request PUBLIC "-//WIPO//DTD REQUEST 1.1 2003-06-02//EN"
    SYSTEM "request-v1-1.dtd" >
    ```

69. Optionally, two processing instructions in each instance should indicate the software used to create the instance, one for the software name and one for the version. These processing instructions shall appear in the prologue of the document instance. See http://www.w3.org/TR/2000/REC-xml-20001006#sec-pi for a description of the syntax for processing instructions.

    ```
    <?software_name [name of the software]?>
    <?software_version [version of the software]?>
    ```

70. If style sheets are available, either the DTD or each document instance that conforms to the DTD shall declare the style sheets, in conformance with *W3C Recommendation Associating Style Sheets with XML documents Version 1.0* published at http://www.w3.org/TR/xml-stylesheet/ .

    Stylesheet declaration sample:

    ```
    <?xml-stylesheet href="..\..\stylesheet_factory\pap.xsl" type="text/xsl"?>
    ```

71. The organization of white space in document instances can contribute significantly to the readability of the raw data. For example, document instances could include a line break after the closing tag of every paragraph, so that each paragraph would be on a separate line.

72. Style sheets are based on presumptions about what content is contained between the start and end tags. Although white space is collapsed by XML processors under some circumstances, other software used to process instances might not. Avoid adding extraneous white space that is not part of the content. Content should start immediately after the start tag and end immediately before the end tag.

    For example,

    ```
    <postcode>20231</postcode>
    ```

    rather than

    ```
    <postcode> 20231</postcode>.
    ```

External entities

73. An external entity is any object that accompanies an XML document instance that is referenced from within the document instance. External entities are an integral part of a patent document. Without them, the XML instance cannot be parsed, rendered, or understood successfully.

74. In the case of patent documents, external entities are most frequently drawing pages, but could also include embedded images, computer software listings, tables, sequence listings, undefined characters, or character entities. Embedded images are a commonly used external entity, that is, a reference to an external image file is inserted in a document instance at the point where the image should be displayed when the instance is rendered.

75. Images in patent documents can be complete scanned pages or so-called embedded images. Embedded images are most often document parts that cannot be coded and stored using a character set. These can be drawings, chemical formulae, complex tables, undefined characters, etc. Undefined characters are characters not defined in a character set or not available as a character entity reference (see above). Currently, there are four types of image data allowed in Annex F within the ICEs element `<img>`: TIFF, JPEG, ST.33 and ST.35; these are discussed below:

76. The following subsections provide guidance on the use of the image file types supported by Annex F. If an office chooses to publish document instances that reference external entities that use other encodings, or if an office departs from the recommendations below, information sufficient for proper rendering must be provided to users by the office.

TIFF

77. Embedded images shall be enclosed in a TIFF (tagged image file format) header and follow the profile described in Annex F (published on WIPO's web site at www.wipo.int/pct/en/texts/index.htm). See http://partners.adobe.com/asn/developer/pdfs/tn/TIFF6.pdf for a complete explanation of TIFF.

78. The recommended coding scheme for TIFF image data is based on the Modified READ II data compression technique for ITU-T (CCITT) Group 4 facsimile equipment as described in the ITU-T (CCITT) recommendation T.6, commonly known as Fax Group 4. See Standard ST.35, Annex 3 and 4, for details of Fax Group 4 and the possible content of TIFF header information.

JPEG

79. Embedded images may also be enclosed in a JPEG header and follow the profile described in Annex F (published on WIPO's web site at www.wipo.int/pct/en/texts/index.htm). See http://www.jpeg.org/ for a complete explanation of JPEG.

80. If used, JPEG must conform to office-specific rules and regulations. For example, the PCT admits only black and white images (PCT Rule 11.13: "Drawings shall be executed in durable, black, sufficiently dense and dark, uniformly thick and well-defined, lines and strokes without colorings"). Where another office permits other options in JPEG, those options must be specified and published.

WIPO Standard ST.33

81. When formatting external image files according to Standard ST.33, the office must refer users to WIPO *Standard* ST.33.

WIPO Standard ST.35

82. If an office formats external image files according to Standard ST.35, it must refer users to WIPO *Standard* ST.35.

83. This Standard relates only to the use of ST.35 image data, not to any of the other data types described in Standard ST.35.

PDF

84. External entities that are PDF files shall follow the profile described in Annex F (published on WIPO's web site at www.wipo.int/pct/en/texts/index.htm). For further information about PDF, see http://www.adobe.com/products/acrobat/adobepdf.html.

85. If an office chooses to publish document instances that reference external entities that use proprietary encodings, at minimum, information sufficient for proper rendering shall be provided by the office.

MEGA CONTENT

86. Some patent applications and their resulting publications include content that is so large or voluminous that it prevents or seriously impairs routine machine processing (hereafter referred to as “mega content”). One approach used by some offices to overcome this difficulty is to treat the mega content as an external entity. This section describes how to implement that approach for various types of mega content.

87. As an external entity, the mega content remains an integral part of the document (application-as-filed or publication), but is contained in a distinct file that can be ignored for certain types of processing or processed separately from the instance of which it is a part. Ordinarily, an external entity is referenced from within a document instance at the point where the entity would normally be displayed at the time of rendering, that is, at its logical home within the document. This section provides guidance to offices that choose to treat mega content as external entities.

88. Where some portion of the content, for example, a sequence listing or a table, is larger than a limit set by an industrial property office, that office may choose to require that the mega content be treated as an external entity. For example, an office might require that tables, which would occupy 300 or more printed pages, must be treated as an external entity. By publishing such a large table as an external entity, the table can more easily be excluded from automatic printing, avoiding unexpected and excessive paper usage by examiners or the unsuspecting public. It can also be excluded from downloading, unless explicitly requested by a customer or examiner, to avoid unnecessary burdens on networks and servers.

89. In the case of sequence listings, Standard ST.25 determines the format of the external entity.

90. In the case of tables, the same XML format used for tables within a document instance shall be used for external entities. A slightly modified industry-standard table DTD for this purpose is `table-external.dtd`.

91. In the case of computer program listings, the format of the external entity is usually simple ASCII text, with the layout determined by the syntax of the computer program language. This could be implemented by placing the program listing in a single-cell table referenced from `table-external-doc` in an instance of `table-external.dtd`.

92. In the case of chemical structures or mathematical formulas, where mega content is expected to consist of large numbers of otherwise relatively small objects, the content shall be inserted into one or more tables and the resulting tables shall then be treated as external entities, using `table-external.dtd`.

Industry-standard DTDs

93. Where appropriate to the content of a document, that is, where the content is not unique to the industrial property domain, use industry-standard DTDs. Such DTDs are not included in international DTDs or in office-specific DTDs, but incorporated by reference (see `application-body.dtd` for examples).

94. For mathematical formulas, MathML, version 2, shall be used. See http://www.w3.org/Math/ for a complete description.

95. For tables, the OASIS XML table DTD shall be used. See http://www.oasis-open.org/specs/tm9901.html, which is the *XML Exchange Table Model Document Type Definition, OASIS Technical Memorandum TR 9901:1999*.

96. When an office wishes to use other industry-standard DTDs, a change request shall be submitted.

97. It is recommended that industry-standard DTDs are incorporated by reference only.

98. Such industry-standard DTDs may not be modified in any way when referenced by office-specific DTDs. An exception is the use of an over-ride to add attributes to the root element of the OASIS table DTD in the `xx-patent-publication` DTD. The OASIS table DTD is constructed expressly to permit local definition of cell content. Other exceptions must be submitted as a change request.

Model DTD for patent publications

99. The model DTD `xx-patent-document.dtd` is not intended for use as published. It is intended as a basis from which each office can build its own office-specific DTD with minimal effort and maximum use of common elements and high-level logical structures.

100. When changing the model DTD for patent publications (`xx-patent-document.dtd`) the rules for office-specific names indicated above should be observed. Offices are discouraged from making any changes that are not essential for the intended purpose, since this impairs interoperability.

101. Office-specific constraints not expressed in the DTDs referenced in this specification must be expressed in some other office-specific specification that is public.

References

102. Reference to the following standards and documents are of relevance to this Standard:

(a) WIPO Standard ST.3 – Recommended Standard on Two-Letter Codes for the Representation of States, Other Entities and Intergovernmental Organizations.

(b) WIPO Standard ST.9 – Recommendation Concerning Bibliographic Data on and Relating to Patents and SPCs.

(c) WIPO Standard ST.16 – Recommended Standard Code for the Identification of Different Kinds of Patent Documents.

(d) WIPO Standard ST.25 – Standard for the Presentation of Nucleotide and Amino Acid Sequence Listings in Patent Applications.

(e) WIPO Standard ST.32 – Recommendation for the Markup of Patent Documents Using SGML (Standard Generalized Markup Language).

(f) WIPO Standard ST.33 – Format for Data Exchange of Facsimile Information of Patent Documents.

(g) WIPO Standard ST.35 – Data Exchange of Mixed-Mode Published Patent Information on MMMT.

(h) PCT Administrative Instructions under the Patent Cooperation Treaty: Part 7 Instructions Relating to the Electronic Filing and Processing of International Applications.

(i) Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler. Extensible Markup Language (XML) 1.1. W3C Recommendation 04 February 2004, edited in place 15 April 2004. See: http://www.w3.org/TR/2004/REC-xml11-20040204.

(j) ISO/IEC International Standard 10646-1:1993(E): Information technology -- Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane. International Organization for Standardization, Geneva, 1993.

(k) International Standard ISO/IEC 10646-2, Information technology -- Universal Multiple-Octet Coded Character Set (UCS) – Part 2: Supplementary Planes. First edition, International Organization for Standardization, Geneva, 2001.

[End of Standard]