



## **IPTC Standards**

---

# **IPTC Core Schema for XMP**

**Version 1.0**

**Supplemental documentation:**

## **Implementation Guidelines**

**Document Revision 3**



Copyright © 2005 by IPTC, the International Press Telecommunications Council. All rights reserved.

IPTC address:

Postal mail: Royal Albert House, Sheet Street, Windsor, Berkshire, SL4 1BE, United Kingdom

Web: [www.iptc.org](http://www.iptc.org)

email: [office@iptc.org](mailto:office@iptc.org)

This project intends to use materials that are either in the public domain or are available by the permission for their respective copyright holders.

Permissions of copyright holder will be obtained prior to use of protected material.

All materials of this IPTC standard covered by copyright shall be licensable at no charge.

---

**Document file name** (+ Word file extension “.doc”): **Ipct4xmpCore\_1.0-doc-ImplGuideLines\_3.doc**

**Document URN: urn:iptc:std:Ipct4xmpCore:1.0:doc:ImplGuideLines:3**

(More information on IPTC URNs in RFC 3937)

### Specification Versioning History

Version	Issue Date	Approved by	Remark
1	2004-10-08	IPTC Standards Committee	

### Document Revision History

Revision	Issue Date	Author (revised by)	Remark
1	2004-12-01	Michael Steidl	Initial draft document
2	2005-02-01	Michael Steidl	Clean up for public release
3	2005-03-15	Michael Steidl	Minor issues resolved



## Table of contents

1	Introduction.....	3
2	Guidelines .....	3
2.1	How the “IPTC Core” and the XMP framework relate .....	3
2.2	Software agent considerations .....	4
2.2.1	Understanding IPTC metadata synchronisation.....	4
2.2.2	Where to store “IPTC Core” properties .....	5
2.3	User interface considerations .....	5
3	Reference documentation.....	6

## 1 Introduction

The “IPTC Core” Schema for XMP specifies a standardised set of metadata properties to be used within the scope of the “Extensible Metadata Platform” (XMP), a metadata framework developed by Adobe Systems Inc.. This guidelines document provides support for implementing this standard either in an automated XMP agent or any piece of software providing a user interface to the XMP properties specified by the “IPTC Core” standard.

The scope of the IPTC Core standard is to define XMP properties with all necessary attributes for XMP and thus the IPTC specifications rely on the XMP specifications by Adobe which itself is built on the W3C standards XML and Resource Description Framework (RDF). These guidelines will focus on the implementation of the “IPTC Core” in an XMP environment and will only point to basic XMP issues but will not discuss them in depth, hence additional reading might be required to understand all technical implications.

The intended audience of this document are

- software developers for XMP enabled software (“software agents”)
- user interface designers for XMP enabled software

## 2 Guidelines

### 2.1 How the “IPTC Core” and the XMP framework relate

The “IPTC Core” defines a set of XMP properties making an “XMP Schema”.

For the XMP Schema as a whole the names and identifiers below are specified by the IPTC:

- An XML namespace URI
- A recommended XML namespace prefix

(The current values are specified by the reference document, see section 3)

The set of information below was specified for each individual XMP property:

- A property name
- A user interface label
- A description for this XMP property
- Notes on this XMP property, primarily addressing implementers!
- An XMP Category
- An XMP Value Type



- An XMP path
- A label used by the user interface of Photoshop CS
- A reference to the IPTC IIM dataset

Find more on this set of information in the “IPTC Core” reference document’s section “XMP Schema Properties”. (See section 3 of this document)

The relationship of an XMP Schema in general and this “IPTC Core” in specific is:

- For a first approach XMP Schema specifies a property at a storage abstract level.
- The “description” of a property specifies its concept and scope, the “name” of the property should relate to this description in a meaningful way. To emphasise this: the “name” of a property does not necessarily relate to the name of any XML element the value of this property is stored to!
- Only the value of the “XMP Path” specifies where to store the value of this property in the XMP framework.

There are two technical options for adopting an XML element as storage location:

- a) store the value into an XML element from the XML namespace specified for this XMP Schema (in this case: the one specified for the “IPTC Core”)
  - b) store the value into an XML element from an XML namespace already defined by another XMP Schema.
- The reason for selecting option b) is to reuse some specific behaviour of such a “pre-defined” XMP property.

This applies to all the XMP properties of the “IPTC Core” XMP Schema which are NOT stored to an element from the IPTC Core XML namespace:

These elements from other XML namespaces, primarily from “photoshop” and “dc”, are “synchronised” with metadata values held in the “legacy” IPTC headers by existing XMP software agents from Adobe. Find more on this issue in section 2.2.1

For more in depth information about XMP read the “XMP Specifications” of Adobe. (See section 3)

## 2.2 Software agent considerations

A “software agent” in terms of XMP is any piece of software able to process the XMP framework.

### 2.2.1 Understanding IPTC metadata synchronisation

The IPTC defined a set of metadata for news objects in its “Information Interchange Model” (IIM) standard which was released for the first time in 1991. A subset of IIM elements was adopted by Adobe for its Photoshop software to hold metadata about images in a technical wrapper structure named “Image Resource Block” (IRB) by Adobe and this IRB can be inserted into popular image file formats like Photoshop (.psd), JPEG and TIFF. This set of metadata was commonly called “IPTC metadata”, or “IPTC header”.

For current image files this “Image Resource Block” can coexist with an XMP packet in the header section of any of these popular file formats and hence holding the values of the legacy “IPTC header” and the XMP packet in synch is an important issue, this kind of synchronisation is currently provided by Photoshop 7.0x and Photoshop CS.



The basic functionality of this synchronising mechanism is:

- To move metadata from the IRB to a new XMP packet in case an image file is opened with such an IRB but without an XMP packet. If the file is saved both the IRB and the XMP packet will be stored to the header section of this file.
- To move metadata from the XMP packet to a new IRB in case an image file is opened with an XMP packet but without an IRB. If the file is saved both the IRB and the XMP packet will be stored to the header section of this file.
- To update the metadata values of the IRB respectively the XMP packet from the “newer” source to the “older” destination. Which set is “newer” or “older” is detected by a hash value over each block of metadata. (More detailed information is available from Adobe’s XMP web pages at [www.adobe.com/xmp](http://www.adobe.com/xmp) and in the software development kits.)

To store the values from the legacy IRB (respectively the IIM) into the XMP framework Adobe specified a set of XMP properties, some of them are from an XML namespace with a prefix of “photoshop”, some are from a Dublin Core XML namespace with “dc” as prefix. Which elements are synchronised mutually is shown in the appendix section of the “IPTC Core” specification. (See section 3)

Therefore it is highly recommended to implementers of any “software agents” for XMP to consider implementing this synchronisation too.

### 2.2.2 Where to store “IPTC Core” properties

The “IPTC Core” specifications (See section 3) define a specific XML element for each XMP property as node of the XMP/XML structure where the value of this property should be stored. This node is specified by as “XMP Path” which aligns with the XPath notation.

## 2.3 User interface considerations

The user interface is the portion of any “IPTC Core” implementation that makes it “tangible” to users.

To avoid confusion about the semantics of a certain metadata field shown to the user for input it is very important to comply with the user interface design recommendations of the IPTC:

- Use only the language of “User interface label” of the “IPTC Core” specification as label to any input field for a property. Using different terms for these labels created a lot of confusion about the intention of this metadata field and corrupted the consistent use of metadata elements.

The IPTC will provide recommended translations to these terms into several languages.

- Add some kind of “easy access” help to the input fields (“Balloon help” or “Hints” while moving the mouse over the field) providing the information given in the “Description” of the “IPTC Core” specifications.

As the labels have to be rather short they can’t tell much about the intended use of this property, having such an additional description at hand supports the user in the right use of the properties.

The IPTC will provide recommended translations to these terms into several languages.

- There is one group of metadata with several members, the “creator’s contact information”. This group comprises XMP properties having a name starting with “Ci...”. These properties should be grouped together by the visual design too.



- As not all fields for “IPTC Core” will fit on a single screen/panel/tab this is the recommended grouping of properties: (XMP property names are given below)
  - \* Group for the photographer’s contact info:  
Creator, CreatorJobTitle, all sub-properties of “ContactInfo”
  - \* Group to describe the visual content of the image:  
Headline, Description, Keywords, SubjectCode, DescriptionWriter
  - \* Group to provide formal descriptive information about the image:  
DateCreated, IntellectualGenre, Scene, Location, City, Province-State, Country, CountryCode
  - \* Group to record workflow and copyright information:  
Title, JobID, Instructions, Provider, Source, CopyrightNotice, RightsUsageTerms
  
- Using the appropriate user interface components:  
XMP specifies several types for the values of its properties with extensions from the Resource Description Framework. The implementer has to apply visual components for the user interface which align with the requirements of the specific value type.  
The XMP Value Types for all XMP properties of the “IPTC Core” are defined in the specification document. (See section 3)

### 3 Reference and support

**Reference documents** for the “IPTC Core XMP Schema” and related technical matters.

“IPTC Core XMP Schema” Version 1.0 specification:

This version and any updates can be obtained from: <http://www.iptc.org/IPTC4XMP>

Extensible Metadata Platform, Adobe Systems Inc:

Overview: <http://www.adobe.com/products/xmp/>

The latest specification can be obtained from:

<http://partners.adobe.com/public/developer/xmp/sdk/index.html>

Extensible Metadata Language XML, W3C

The latest specification can be obtained from: <http://www.w3.org/TR/REC-xml>

Resource Description Framework (RDF) W3C

The latest specification can be obtained from: <http://www.w3.org/RDF/>

**Support sites for the “IPTC Core XMP Schema”:**

Website: <http://www.iptc.org/IPTC4XMP>

Public Yahoo discussion forum: <http://groups.yahoo.com/group/iptc4xmp>

==== END of document ====