

# A proposal for an XLink data model

**Report**

**Author(s):**

Wilde, Erik

**Publication date:**

2002-08

**Permanent link:**

<https://doi.org/10.3929/ethz-a-004449705>

**Rights / license:**

[In Copyright - Non-Commercial Use Permitted](#)

**Originally published in:**

TIK Report 148

# A Proposal for an XLink Data Model

Erik Wilde

Computer Engineering and Networks Laboratory  
Swiss Federal Institute of Technology, Zürich

TIK Report 148 (August 2002)

## Abstract

This report describes a proposal for a data model for XLink. It defines the data model as contributions of XLink to the XML Infoset. The data model is meant as a clarification of the link model implicitly defined by XLink. It is also meant as the foundation for future work on XLink, for example a DOM module for XLink support, a CSS module for styling XLinks, or a protocol for accessing XLink linkbases.

## 1 Goals

The goal of this report is to provide a starting point for a discussion for an XLink data model. XLink currently is a W3C recommendation, but it is not widely used or supported. One of the possibilities to promote XLink is to provide potential users with a foundation where they can actually use XLink from within applications (via a DOM interface) or for document formatting (supported by styling mechanisms). XLink needs better support for this, for example a DOM module for XLink, CSS selectors for linking information, CSS properties for link formatting, or linkbase access [8]. An XLink data model is required to built all these things on a shared foundation, and this report proposes to define this data model using the XML Infoset.

## 2 Introduction

The XML Linking Language (described in Section 3) defines how hyperlinks can be embedded into XML documents. It does so by defining a number of XML attributes which can be used in documents to carry hyperlink semantics. By defining these attributes, XLink also implicitly defines a data model, and it is this implicit data model which is described more explicitly by this report.

The XML Information Set (described in Section 4) defines an abstract model for the data model of XML. It is a model that can be extended by other specifications to include additional information in the Infoset, and this is what this report does: it describes the XLink data model in terms of XML Infoset extensions (the term *contribution* is the established way to refer to extensions of the XML Infoset, for example the PSVI contributions of XML Schema [7]).

The goal of this report is to explicitly describe the hyperlink information implicitly defined by XLink. In the same way as the XML Information Set has been created to describe a data model for XML documents, the XLink Infoset Contributions may serve as the foundation for future work with XLink. And in the same way as XML applications may restrict or extend the XML Information Set for their purposes, XLink applications may restrict or extend the XLink Infoset Contributions for their purposes. However, XLink applications requiring interoperability with other XLink applications should fully support the XLink Infoset Contributions, and not depend on any additional information.

## 3 The XML Linking Language

The *XML Linking Language (XLink)* [5] defines how hyperlinks can be expressed in XML. XLink uses the XML Namespace [1] mechanism to define a number of *global attributes*. If these attributes are used according to the constraints defined in XLink, then they are interpreted as hyperlink information.

### 3.1 Simple and Extended Links

In Section 5 ("XLink Elements and Attributes"), XLink defines two syntactic variants of XLink hyperlinks, *simple links* (Section 5.1 of XLink) and *extended links* (Section 5.2 of XLink). Simple links provide a short-hand notation for extended links, so from the model point of view, it is not necessary to distinguish between simple and extended links. However, it may be interesting for applications to know whether a particular link has been specified in simple or extended syntax, so this information should be reflected in the XLink Infoset Contributions.

### 3.2 Value defaulting

XLink defines value defaulting for some attributes, for example an `arc`-type element where the `from` and/or `to` attributes are missing is interpreted in a way as if the missing attribute(s) were standing for all the labels supplied on `locator`-type elements. If the XLink Infoset Contributions are derived from XLink markup, then all the value defaulting constraints defined by XLink must be observed.

On the other hand, if an Infoset containing XLink Infoset Contributions is serialized as XML/XLink, it is at the serializing application's discretion to decide whether the serialization takes advantage of value defaulting (if possible) or not, as long as the serialized result is semantically identical to the XLink Infoset Contributions.

## 4 The XML Information Set

The *XML Information Set (XML Infoset)* [3] defines the data model of XML documents [2]. It omits some of the information that is present in an XML document (such as the order of attributes in an element start tag), and hides some of the specifics of the document (such as the character encoding being used). As such, it is a foundation for other specifications, such as the *Document Object Model (DOM)* [6] or XML Schema [7].

The Infoset defines a number of *information items* with *properties*, and this definition does not prescribe any specific form of representation or interface, it is simply an abstract data model. One possible representation of an Infoset is an XML document, and one possible interface for the Infoset is the Document Object Model.

## 5 XLink Infoset Contributions

If a property specifies *child information items* as possible values, then these child information items may be any XML Infoset items, depending on the content of the respective XML element.

### 5.1 Information Items

The following sections contain definitions of the four types of information items that are defined by the XLink Infoset Contributions.

### 5.1.1 Link Information Item

Link information items represent simple as well as extended links, because simple links conceptually do not differ from extended links, they only provide a more concise syntax for coding link information.

*XLink Definition:* Section 5.1 and Section 5.2

Each Link Information Item has the following properties:

1. *[type]* Either **simple** or **extended**, indicating whether the link has been specified in XLink simple or extended form. If this is unknown, the property has the value **unknown**.
2. *[resources]* An unordered set of Resource Information Items.
3. *[arcs]* An unordered set of Arc Information Items.
4. *[role]* A URI reference, coming from the **role** attribute of the link element (ie, of the **simple**-type element or the **extended**-type element). If the attribute has not been present, this property has the value **novalue**.
5. *[title]* A string, coming from the **title** attribute of the link element (ie, of the **simple**-type element or the **extended**-type element). If the attribute has not been present, this property has the value **novalue**.
6. *[titles]* An ordered list, in document order, of Title Information Items, if the link element has **title**-type children (this is only possible for **extended**-type elements because of the constraints defined by XLink).
7. *[element]* The *Element Information Item* that represents this Link Information Item. If there is no such Element Information Item, this property has the value **novalue**.

It is important to notice that both simple and extended links are represented by Link Information Items. This means that the information item view of simple and extended links is unified, only the *[type]* property indicates what type of XLink has been used. Consequently, a Link Information Item for a simple link contains references to Resource Information Items and Arc Information Items in the same way as a Link Information Item for an extended link, even though both forms of XLinks specify resources and arcs in different ways (simple links as attributes of the **simple**-type element, and extended links as child elements of the **extended**-type element). If the simple link carried behavior attributes, these are represented on the Arc Information Item of the resulting XLink Infoset Contributions.

### 5.1.2 Resource Information Item

Resource information items represent local as well as remote resources. They do both by reference, for local resources by reference to the relevant information items(s), for remote resources by using a URI reference.

*XLink Definition:* Section 5.1.1 and Section 5.1.2

Each Resource Information Item has the following properties:

1. *[reference]* If the resource is a local resource, an ordered list of child information items, in document order. If the resource is a remote a resource, a URI reference coming from the **href** attribute of the **simple**-type or **resource**-type element.
2. *[from arcs]* An unordered set of Arc Information Items that are referencing this Resource Information Item as originating resource.
3. *[to arcs]* An unordered set of Arc Information Items that are referencing this Resource Information Item as target resource.

4. *[role]* A URI reference, coming from the `role` attribute of the `locator`-type element or the `resource`-type element. If the attribute has not been present, this property has the value `novalue`.
5. *[title]* A string, coming from the `title` attribute of the `locator`-type element or the `resource`-type element. If the attribute has not been present, this property has the value `novalue`.
6. *[titles]* An ordered list, in document order, of Title Information Items, if the `locator`-type element has `title`-type children (`title`-type children are not possible for `resource`-type elements because of the constraints defined by XLink).
7. *[label]* An `NCName`, coming from the `label` attribute of the `locator`-type element or the `resource`-type element. If the attribute has not been present, this property has the value `novalue`.
8. *[link]* The Link Information Item that this Resource Information Item is a child of.
9. *[element]* The *Element Information Item* that represents this Resource Information Item. If there is no such Element Information Item, this property has the value `novalue`.

### 5.1.3 Arc Information Item

Arc information items represent arcs connecting resources of a link.

*XLink Definition:* Section 5.1.3

Each Arc Information Item has the following properties:

1. *[arcrole]* A URI reference, coming from the `arcrole` attribute of the `arc`-type element. If the attribute has not been present, this property has the value `novalue`.
2. *[show]* One of the values `new`, `replace`, `embed`, `other`, or `none`, coming from the `show` attribute of the `arc`-type element. If the attribute has not been present, this property has the value `novalue`.
3. *[actuate]* One of the values `onLoad`, `onRequest`, `other`, or `none`, coming from the `actuate` attribute of the `arc`-type element. If the attribute has not been present, this property has the value `novalue`.
4. *[from]* An unordered set of Resource Information Items that are referenced by this Arc Information Item as originating resources.
5. *[from label]* An `NCName`, coming from the `from` attribute of the `arc`-type element. If the attribute has not been present, this property has the value `novalue`.
6. *[to]* An unordered set of Resource Information Items that are referenced by this Arc Information Item as target resources.
7. *[to label]* An `NCName`, coming from the `to` attribute of the `arc`-type element. If the attribute has not been present, this property has the value `novalue`.
8. *[title]* A string, coming from the `title` attribute of the `arc`-type element. If the attribute has not been present, this property has the value `novalue`.
9. *[titles]* An ordered list, in document order, of Title Information Items, if the `arc`-type element has `title`-type children.
10. *[link]* The Link Information Item that this Arc Information Item is a child of.
11. *[element]* The *Element Information Item* that represents this Arc Information Item. If there is no such Element Information Item, this property has the value `novalue`.

### 5.1.4 Title Information Item

Title information items are used for title information for links, resources, or arcs.

*XLink Definition:* Section 5.1.4

Each Title Information Item has the following properties:

1. *[children]* An ordered list of child information items, in document order.
2. *[title]* The Link Information Item, Resource Information Item, or Arc Information Item that this Title Information Item is a child of.
3. *[element]* The *Element Information Item* that represents this Title Information Item. If there is no such Element Information Item, this property has the value `noValue`.

## 5.2 Properties

In addition to the XLink Information Items, the XLink Infoset Contributions also define properties which may be used on the Infoset's information items.

### 5.2.1 XLink Information Item

The XLink Information Item property is used with *Element Information Items* to refer from an element with XLink semantics to the respective XLink-specific information item.

- *[xlink item]* The Link Information Item, Resource Information Item, Arc Information Item, or Title Information Item that has been derived from this element.

It is important to notice that the XLink Information Item property can be used to access the XLink-specific information item that has been derived from an XML element. However, the element may carry additional semantics (such as additional attributes) which are outside the scope of XLink, so depending on the application model it might be necessary to interpret additional properties of an element.

### 5.2.2 XLink Resource

The XLink Resource property is used with any information item to indicate that this particular item is part of a link resource.

- *[xlink resource]* The Resource Information Item that references this information item as a resource.

To use this property, applications must interpret resource identifiers (eg, XPointers).

## 6 Creating Link Information

The link information represented in the XLink Infoset Contributions, can come from any source, XLink markup within XML documents, RDF representations of XLink [4], XLink-structured databases, or any source which can be used for collecting link information. Specifically, it is not necessary that XLink's syntax or link model is the source for constructing XLink Infoset Contributions, as long as there is a possible mapping to the model defined by the XLink Infoset Contributions.

## References

- [1] TIM BRAY, DAVE HOLLANDER, and ANDREW LAYMAN. Namespaces in XML. World Wide Web Consortium, Recommendation REC-xml-names-19990114, January 1999.
- [2] TIM BRAY, JEAN PAOLI, C. M. SPERBERG-MCQUEEN, and EVE MALER. Extensible Markup Language (XML) 1.0 (Second Edition). World Wide Web Consortium, Recommendation REC-xml-20001006, October 2000.
- [3] JOHN COWAN and RICHARD TOBIN. XML Information Set. World Wide Web Consortium, Recommendation REC-xml-infoset-20011024, October 2001.
- [4] RON DANIEL. Harvesting RDF Statements from XLinks. World Wide Web Consortium, Note NOTE-xlink2rdf-20000929, September 2000.
- [5] STEVEN J. DE ROSE, EVE MALER, and DAVID ORCHARD. XML Linking Language (XLink) Version 1.0. World Wide Web Consortium, Recommendation REC-xlink-20010627, June 2001.
- [6] ARNAUD LE HORS, PHILIPPE LE HÉGARET, LAUREN WOOD, GAVIN THOMAS NICOL, JONATHAN ROBIE, MIKE CHAMPION, and STEVEN BYRNE. Document Object Model (DOM) Level 3 Core Specification. World Wide Web Consortium, Working Draft WD-DOM-Level-3-Core-20020409, April 2002.
- [7] HENRY S. THOMPSON, DAVID BEECH, MURRAY MALONEY, and NOAH MENDELSON. XML Schema Part 1: Structures. World Wide Web Consortium, Recommendation REC-xmlschema-1-20010502, May 2001.
- [8] ERIK WILDE and CHRISTIAN STILLHARD. Openly Accessing Linkbases. Technical Report TIK-Report No. 134, Computer Engineering and Networks Laboratory, Swiss Federal Institute of Technology, Zürich, Switzerland, January 2002.