



13 listopada 2003

## Standardy pokrewne



## XLink – dowiązania w XML-u

**Linki jakie znamy (HTML):**


- łączą dwa dokumenty: źródło i cel linku,
- źródłem linku jest zawsze element opisujący link (A, IMG).

**XLink – rozszerzona koncepcja dowiązań:**

- dowolne elementy przechowują informacje o linkach,
- informacja o linkach poza połączonymi dokumentami,
- więcej niż dwa końce linku.

**Status:**

- rekomendacja W3C z 27 czerwca 2001,
- korzenie historyczne: HyTime.



## Terminologia

**Zasób (resource)** – dowolna adresowalna jednostka informacji lub usługa.


**Dowiązanie (link)** – jawnie wyrażona (przy pomocy **elementu wiążącego (linking element)**) relacja pomiędzy zasobami.

- te zasoby **uczestniczą (participate)** w dowiązaniu.

**Przejście (traversal)** – użycie pary zasobów połączonej dowiązaniem.

**Łuk (arc)** – informacja o przejściu między dwoma zasobami (kierunek, zachowanie aplikacji, itp.):

- wychodzący (outbound),
- wchodzący (inbound),
- niezależny (third party).




## Dowiązania XLink

**Extended link:**

- wiąże dowolną liczbę zasobów:
  - zasoby zewnętrzne (np. odwołania do innych dokumentów),
  - zasoby lokalne (zawarte w elemencie wiążącym).
- łuki opisujące sposoby przechodzenia pomiędzy zasobami,
- role zasobów uczestniczących w linku,
- role łuków.

**Simple link:**

- link wychodzący,
- wiąże dokładnie 2 zasoby: 1 lokalny i 1 zewnętrzny,
- jeden łuk z zasobu lokalnego do zewnętrznego.




## Simple link – przykład

```
<osoba xmlns:xlink="http://www.w3.org/1999/xlink">
  <nazwisko>Kopernik, Mikołaj</nazwisko>
  <biogram>Wybitny polski astronom, matematyk, lekarz, prawnik,
  tłumacz poezji włoskiej i ekonomista, pochodził z rodziny
  wywodzącej się z mieszczan krakowskich. Urodzony w <geogr
  xlink:type="simple" xlink:href="Torun.xml">
  Toruniu</geogr>.</biogram>
</osoba>
```



## Extended link - przykład

```
<fikcja xmlns:xlink="http://www.w3.org/1999/xlink"
  xlink:type="extended">
  <wypowiedz xlink:type="resource">Kopernik była kobieta!
  </wypowiedz>
  <film xlink:type="locator" xlink:href="seksmisja.xml"
  xlink:title="Seksmisja"/>
  <osoba xlink:type="locator" xlink:href="kopernik.xml"
  xlink:title="Kopernik, Mikołaj"/>
  <pojecie xlink:type="locator" xlink:href="kobieta.xml"
  xlink:title="kobieta"/>
</fikcja>
```



## Składnia

### Przestrzeń nazw XLink:

- `http://www.w3.org/1999/xlink`,
- zawiera globalne atrybuty:
  - type definition attribute: `type`,
  - locator attribute: `href`,
  - semantic attributes: `role`, `arcrole`, `title`,
  - behavior attributes: `show`, `actuate`,
  - traversal attributes: `label`, `from`, `to`.

### Dostępne typy:

- `simple`, `extended`, `locator`, `resource`, `arc`, `title`.



## Łuki i role

### Łuk - atrybuty:

- `from`, `to` – etykiety zasobów połączonych łukiem,
- `show` – sposób wyświetlania (`new`, `replace`, `embed`, `none`, `other`),
- `actuate` – czas ładowania celu łuku (`onLoad`, `onRequest`, `other`, `none`).

```
<plec xlink:type="arc" xlink:from="Kopernik"
      xlink:to="kobieta"/>
```

### Rola (zasobu – atrybut `role`, łuku – atrybut `arcrole`):

- znaczenie zasobu/łuku w kontekście linku,
- wartość: bezwzględny URI wskazujący na zasób identyfikujący rolę.

```
<osoba xlink:type="locator" xlink:href="kopernik.xml"
       xlink:title="Kopernik, Mikołaj"
       xlink:role="http://encyklopedia.pl/role/wspomnienie"/>
```



## XLink a DTD

### Brak ograniczeń na nazwy i zawartość elementów wiążących.

#### Deklarowanie elementów wiążących w DTD:

- atrybut `xmlns:xlink` jako `#FIXED`,
- atrybut `xlink:type` jako `#FIXED`,
- pozostałe atrybuty: `#REQUIRED` lub `#IMPLIED`.

```
<!ELEMENT film EMPTY>
<!ATTLIST film
  xmlns:xlink CDATA #FIXED "http://www.w3.org/1999/xlink"
  xlink:type (locator) #FIXED "locator"
  xlink:title CDATA #IMPLIED
  xlink:role CDATA #IMPLIED
  xlink:label NMTOKEN #IMPLIED
  xlink:href CDATA #REQUIRED>
```



## Przyszłość XLink

### Zastosowania:

- organizowanie, kojarzenie zasobów, nawet gdy nie mamy prawa zapisu,
- dostarczanie wartości dodanej – zbiorów linków.

### Zasięg:

- lokalny: serwery linków operujące na bazie linków,
- Internet?

### Problemy:

- wizualizacja extended links,
- synchronizacja zasobów i linków (Internet).



## XPointer – adresowanie fragmentów dokumentu XML

### XPath "opakowany" w składnię URI, np.:

`http://www.sejm.gov.pl/ustawa.xml#xpointer(/art[5]/par[2])`

#### Dodatkowe możliwości:

- proste adresowanie elementów opatrzonych identyfikatorami, np.:  
`urlopy`
- adresowanie elementów bez znajomości struktury, z wykorzystaniem identyfikatorów jako kotwic, np.:  
`element (/4/2/3)`  
`element (urlopy/2/3)`
- wskazanie punktu w dokumencie,
- wskazanie zakresu pomiędzy dwoma punktami,
- wskazanie punktów w tekście i fragmentów tekstu.



## XPointer – status

### Rekomendacje W3C z 25 marca 2003:

- XPointer Framework,
- XPointer element() scheme,
- XPointer xmlns() scheme.

### W3C Working Draft:

- XPointer xpointer() scheme.



## XQuery – zapytania o dokumenty XML

### Język zapytań o dokumenty XML:

- podobny do SQL-a,
- pozwala pytać o kolekcje dokumentów,
- rozszerza funkcjonalność XPath m. in. o:
  - zmienne,
  - wyrażenia FLWOR (for, let, where, order by, return),
  - złączenia,
  - wyrażenia warunkowe,
  - definiowanie funkcji,
  - typ sekwencji,
  - konstruktory, umożliwiające tworzenie struktur XML w zapytaniu,
  - wyrażenia walidujące względem schematu.

Dostępne składnie: SQL-opodobna i XML-owa.

Status: W3C Last Call.



## XQuery – przykład

### Zwróć nazwę każdego wydawcy i średnią cenę jego książek:

```
FOR $p IN distinct(document("bib.xml")//publisher)
LET $a := avg(document("bib.xml")//book[publisher = $p]/price)
RETURN
<publisher>
  <name>{ $p/text() }</name>
  <avgprice>{ $a }</avgprice>
</publisher>
```

Zródło: XML Syntax for XQuery 1.0 (XQueryX), <http://www.w3.org/TR/xqueryx>



## XInclude – łączenie zawartości dokumentów

### Załączanie zawartości jednego dokumentu XML do innego:

- pozwala załączyć fragment dokumentu (wskazany XPointer-em),
- pozwala określić zawartość używaną w razie błędu.

### Przykład:

```
<file name="salatka.xml">
  <xi:include href="salatka.xml#xpointer(/przepis/tytul)"/>
</file>
```

- po przetworzeniu procesorem XInclude:

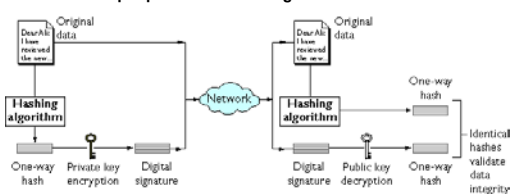
```
<file name="salatka.xml">
  <tytul>Sałatka z ogórków</tytul>
</file>
```

Status: W3C Working Draft



## XML Signature – podpis elektroniczny (1)

### Zasada działania podpisu elektronicznego:



### Kluczowa rola:

- jakości algorytmu haszującego (funkcji skrótu),
- jakości asymetrycznego algorytmu szyfrowania/desyfrowania,
- zaufania do wystawcy certyfikatu.



## XML Signature – podpis elektroniczny (2)

### Podpis dokumentu XML-owego:

- zapisany w postaci struktury XML-owej,
- umieszczany w elemencie Signature:
  - w osobnym dokumencie,
  - dołączonym do podpisywanego dokumentu,
  - zawierającym podpisywane dane.

### Możliwości XML Signature:

- podpisywanie fragmentów dokumentu XML,
- podpisy wielokrotne.



## XML Signature – przykład

```
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
  <SignedInfo>
    <CanonicalizationMethod Algorithm=
      "http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
    <SignatureMethod Algorithm=
      "http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
    <!-- w URI znajduje się wskazanie na podpisywane dane -->
    <Reference URI="http://przyklad.pl/pliki/do-podpisu.xml">
      <Transforms>
        <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#base64"/>
      </Transforms>
      <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
      <DigestValue>60NvZvtdTB+7UnlP/H24p7h4bs=</DigestValue>
    </Reference>
  </SignedInfo>
  <!-- zaszyfrowany skrót z SignedInfo - podpis -->
  <SignatureValue>0sH9AljTNL...</SignatureValue>
  <KeyInfo><KeyValue><DSAKeyValue>
    <P>imup61m...</P><Q>xDve3j7...</Q><G>NlugAf...</G>
    <Y>W7dOmH/v...</Y>
  </DSAKeyValue></KeyValue></KeyInfo>
</Signature>
```

Zródło: P. Kazienko, Co tam panie w XML-u?, Software 2.0, 6/2003



## XML Encryption – szyfrowanie XML-a

**Cel: zagwarantowanie poufności danych XML-owych.**

**Najczęstszy scenariusz:**

- wygenerowanie losowego klucza symetrycznego (sesyjnego),
- zaszyfrowanie nim danych źródłowych,
- zaszyfrowanie klucza sesyjnego kluczem publicznym odbiorcy.

```

<InfoPlatnicza
  xmlns='http://przyklad.pl/platnosci1' >
  <Nazwa>Jozef Nowak</Nazwa>
  <KartaKredytowa Limit='2,000'
    Waluta='PLN' System='Visa' >
    <NrKarty>4019244502775567</NrKarty>
    <Wystawca>Nasz Bank S.A.</Wystawca>
    <DataWaznosci>10/03</DataWaznosci>
  </KartaKredytowa>
</InfoPlatnicza>

<InfoPlatnicza
  xmlns='http://przyklad.pl/platnosci1' >
  <Nazwa>Jozef Nowak</Nazwa>
  <EncryptedData
    Type='http://www.w3.org/2001/04/
      xmlns#Element'
    xmlns='http://www.w3.org/2001/04/
      xmlns#' >
    <CipherData>
      <CipherValue>A2e3B4f5gChDyBReHwWc
        ScX6wQ3q5tVw</CipherValue>
    </CipherData>
  </EncryptedData>
</InfoPlatnicza>
  
```

empolis

## XForms – zaawansowane formularze

**Odpowiedź na ograniczenia formularzy w HTML-u:**

- kontrola dziedziny wprowadzanych danych po stronie klienta,
- specyfikowanie pól obowiązkowych i opcjonalnych,
- zależności między polami, np.:
  - wartość pola jest funkcją wartości innych pól,
  - pole jest widoczne tylko przy określonej zawartości innych pól;
- wyzwalacze aktywowane zdarzeniami interfejsu użytkownika,
- bogaty zasób kontroltek, np.:
  - powtórzenia, grupy, wielopoziomowe wybory,
  - suwaki do wyboru wartości z zakresu,
  - kontrolki wyboru plików.

**Status:**

- rekomendacja W3C z 14 października 2003 r,
- możliwość zanurzania w XHTML-u.

empolis

## XForms – przykład (program X-Smiles)

Źródło: P. Kazienko, Co tam panie w XML-u?, Software 2.0, 6/2003

empolis

## Gdzie szukać dalej

**Przemysław Kazienko, Co tam panie w XML-u?**

- Software 2.0, nr 6/2003, Wydawnictwo Software

**Fabio Arciniegas A., What is XLink?**

- 📍 [www.xml.com/pub/a/2000/09/xlink](http://www.xml.com/pub/a/2000/09/xlink)

**Leslie Carr, Initial Experiences of an XLink Implementation**

- 📍 [journals.ecs.soton.ac.uk/xml4j/xlinkexperience.html](http://journals.ecs.soton.ac.uk/xml4j/xlinkexperience.html)

**XPointer tutorial**

- 📍 [www.zvon.org/xxl/xpointer/tutorial/OutputExamples/xpointer\\_tut.html](http://www.zvon.org/xxl/xpointer/tutorial/OutputExamples/xpointer_tut.html)

**Tamino XQuery Demo**

- 📍 [tamino.demozone.softwareag.com/demoXQuery](http://tamino.demozone.softwareag.com/demoXQuery)

**Micah Dubinko, Ten Favorite XForms Engines**

- 📍 [www.xml.com/pub/a/2003/09/10/xforms.html](http://www.xml.com/pub/a/2003/09/10/xforms.html)



empolis