

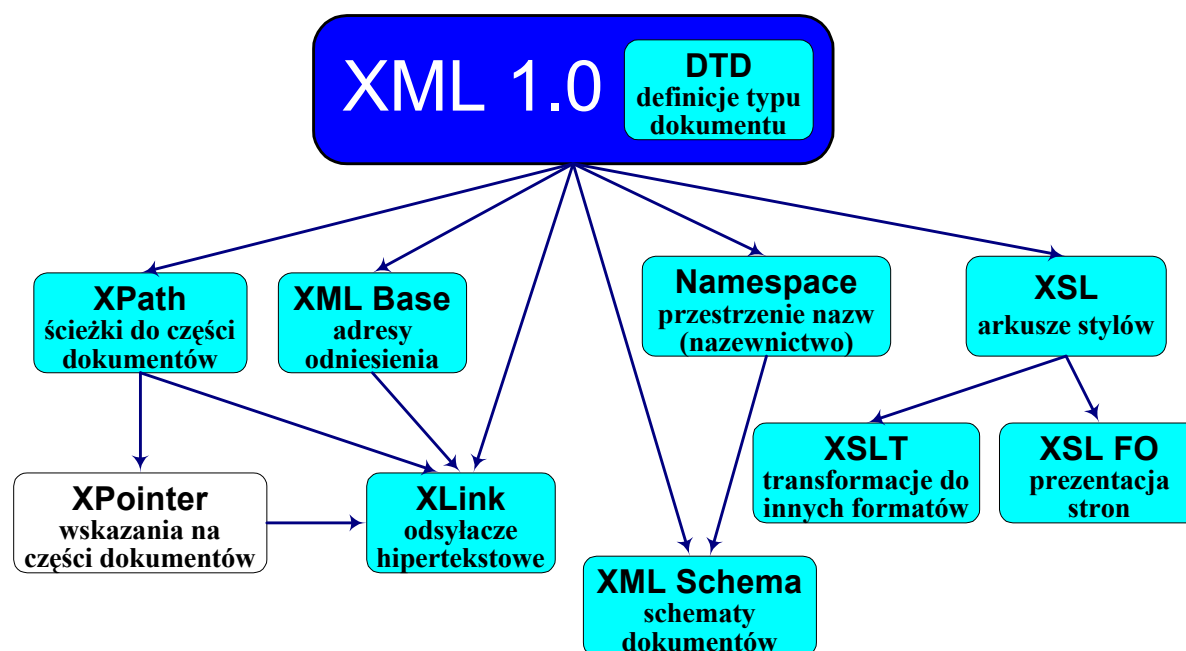
Rodzina języków XML

Przemysław Kazienko

Rok 2001 przyniósł kilka ważnych wydarzeń związanych z językiem XML. Można powiedzieć, że ukończony został drugi etap jego rozwoju (pierwszym było opublikowanie XML 1.0 w 1998r.).

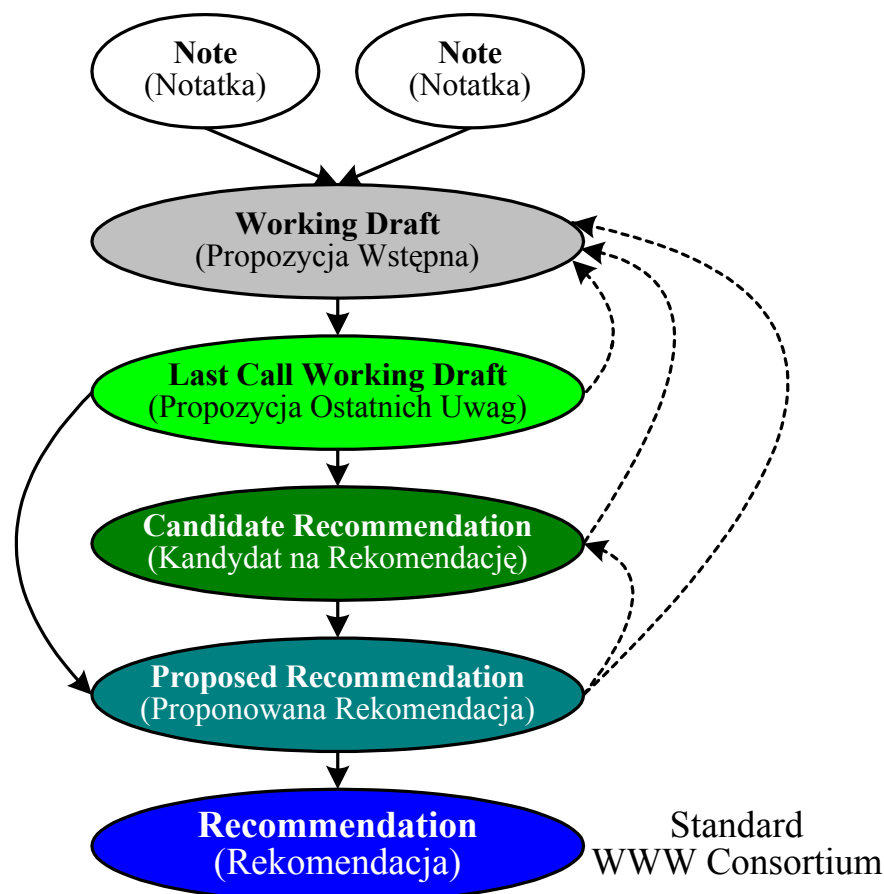
Obeenie istnieje kilkanaście standardowych języków, wywodzących się z XML-a. Można już mówić o rodzinie standardowych języków XML (rysunek 1). Każdy z tych języków jest zdefiniowany poprzez pewien zbiór — charakterystycznych dla niego — elementów i atrybutów. Jednocześnie poszczególne osoby lub organizacje (poprzez definicje typu dokumentu DTD lub schematy XML Schema) mogą tworzyć własne języki przydatne jedynie dla pewnego kręgu zainteresowanych, np. język do opisu dyskografii zespołu muzycznego czy umożliwiający tworzenie w aptece zamówień na leki.

XML to nie pojedynczy język, to bardziej metajęzyk, z pomocą którego tworzone są kolejne standardowe języki a także języki wykorzystywane jedynie dla własnych potrzeb.



Rysunek 1. Rodzina najważniejszych języków XML

Język XML jest owocem wysiłków organizacji WWW Consortium, która jest odpowiedzialna za rozwój systemu WWW. Jej rekomendacje (ang. *Recommendations*) są *de facto* standardami dla środowiska WWW. Jednak, aby standard został zatwierdzony, musi przejść przez kolejne fazy zaprezentowane na rysunku 2.



Rysunek 2. Proces powstawania standardu

W drugiej połowie lat dziewięćdziesiątych język HTML, fundament WWW, z jednej strony przeżywał burzliwy rozwój, a z drugiej — jako nośnik informacji — zaczął wykazywać istotne braki. Powstanie XML miało umożliwić ekspansję WWW na nowe obszary zastosowań — tam gdzie ważna jest informacja a nie tylko jej atrakcyjne zaprezentowanie. Zamysł się powiódł i obecnie można obserwować dynamiczny wzrost zainteresowania językiem XML. Jest on przy tym wykorzystywany nie tylko w serwisach internetowych, ale także np. w elektronicznej wymianie danych czy bazach danych.

Od ukazania się pierwszego numeru Software 2.0, w pełni poświęconego językowi XML — 6/2001, zatwierdzono kilka istotnych rekomendacji WWW Consortium. Aktualną listę najważniejszych standardów języka XML umieszczono w tabeli 1.

Tabela 1. Podstawowe standardy związane z językiem XML

Standard	Nazwa	Przeznaczenie	Data ogłoszenia rekomendacji (standardu) WWW Consortium
XML	Rozszerzalny język znaczników	Pierwszy standard języka XML (XML 1.0)	Luty 1998
Struktura dokumentów XML			
DTD	Definicje typu dokumentu	Opis struktury (typu) dokumentu. Część specyfikacji XML 1.0	Luty 1998
Namespaces	Przestrzenie nazw	Rozwiązanie problemu kolizji nazw elementów i atrybutów. Ważne przy korzystaniu w jednym dokumencie z wielu języków (standardowych lub własnych)	Styczeń 1999

XML Schema	Schematy	Schematy zawartości dokumentu XML. Rozwinięcie DTD	Maj 2001
Prezentacja dokumentów XML			
XSL	Arkusze stylów	Składa się dwóch części: XSLT oraz XSL FO	Listopad 1999 (XSLT), październik 2001 (XSL FO)
XSL FO	XSL obiekty formatujące	Arkusze stylów zawierające obiekty formatujące służące do prezentacji informacji (z podziałem na poszczególne strony)	Październik 2001
CSS	Kaskadowe arkusze stylów	Arkusze stylów przeznaczone dla języka HTML, często wykorzystywane w transformacjach (XSLT) z dokumentów XML do HTML	Level 1 - grudzień 1996, Level 2 - maj 1998
Wyszukiwanie, wskazywanie i połączenia części dokumentów XML			
XPath	XPath	Wyszukiwanie i wskazania na fragmenty (węzły) dokumentu	Listopad 1999
XLink	Odsyłacze XML	Hipertekstowe, zewnętrzne odsyłacze łączące wiele zasobów	Czerwiec 2001
XML Base	Adresy odniesienia	Adresy odniesienia dla względnych adresów URI	Czerwiec 2001
XPointer	Wskaźniki XML	Rozwinięcie XPath umożliwiające m.in. wskazywanie na konkretne punkty i zakresy (od - do) w dokumencie XML	Na razie posiada status kandydata na rekomendację. Prawdopodobne zatwierdzenie – druga połowa 2002
Przetwarzanie dokumentu XML			
XSLT	Transformacje XSL	Transformacja dokumentów XML do innych języków i formatów, czyli np. do stron HTML, dokumentów XSL FO, dokumentów XML o innej strukturze czy nawet zwykłych dokumentów tekstowych	Listopad 1999
DOM	Obiektowy model dokumentu	Przetwarzanie dokumentu XML traktowanego jak drzewo obiektów (węzłów)	Level 1 - październik 1998, Level 2 - listopad 2000
Terminologia			
XML Information Set	Słownik terminologii XML	Słownik i definicje terminów związanych z językiem XML	Październik 2001
HTML w XML			
XHTML	HTML jako XML	XHTML — język HTML 4 jako język wywodzący się z XML, czyli HTML z dodatkowymi ograniczeniami na poprawność składniową (HTML dobrze uformowany). XHTML Basic — okrojona (podstawowa) wersja XHTML, przeznaczona zwłaszcza dla urządzeń przenośnych Modularization of XHTML — umożliwia wydzielanie z XHTML podzbioru elementów (modułów) potrzebnych w danym zastosowaniu; otwiera równocześnie drogę do rozszerzeń i	XHTML - styczeń 2000, XHTML Basic- grudzień 2000, Modularization of XHTML - kwiecień 2001

łączenia języków			
Specjalistyczne zastosowania języka XML			
XML Signature	Podpisy cyfrowe	Podpisy cyfrowe dla dokumentów XML (i nie tylko)	Luty 2002
SVG	Grafika wektorowa	XML-owy język grafiki wektorowej	Wrzesień 2001
SMIL	Synchronizacja multimedii	Język prezentacji multimedialnych. Synchronizacja mediów: tekstu (także strumieni tekstowych), obrazów, video, animacji i dźwięków	Wersja 1.0 - czerwiec 1998, wersja 2.0 - sierpień 2001
SMIL Animation	Animacje SMIL	Język opisu animacji dla prezentacji SMIL	Wrzesień 2001
MathML	Matematyczny XML	Język opisu wzorów matematycznych	Wersja 1.0 - kwiecień 1998, wersja 2.0 - luty 2001
RDF	Metadane	Język opisu metadanych służących do łączenia różnych rodzajów danych, np. po to, by ułatwić wyszukiwanie po wszystkich tych rodzajach na raz	Luty 1999

Standard XML 1.0

Zatwierdzony cztery lata temu standard języka XML 1.0 zawiera podstawowe zasady tworzenia dokumentów XML. Jeżeli jakiś dokument tekstowy ma być uznany za dokument XML, to musi rozpoczynać się od wiersza deklaracji XML, np.:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```

Po niej mogą wystąpić deklaracje typu dokumentu DTD a następnie zasadnicza treść składająca się z pojedynczego elementu głównego – korzenia. Element ten może posiadać elementy – dzieci (podelementy) i (lub) treść tekstową. Podelementy korzenia mogą zawierać kolejne podelementy. W ten sposób uzyskujemy hierarchię elementów. Niepusty element jest ograniczony dwoma znacznikami: początkowym i końcowym, zaś pusty to zwykle pojedynczy znacznik elementu pustego. W znacznikach początkowych lub znacznikach elementu pustego można umieszczać atrybuty.

Dodatkowo, w dokumencie XML mogą się znajdować komentarze, instrukcje przetwarzania, bloki tekstu (sekcje CDATA) a także jednostki (encje).

Standard określa także pojęcie dokumentu dobrze uformowanego (ang. *well-formed*) czyli poprawnego składniowo. To w skrócie wszystko, co zawiera specyfikacja XML 1.0. Dużo, a jednocześnie za mało.

Kontrola struktury

W standardzie XML 1.0 przewidziano możliwość tworzenia **definicji typu dokumentu DTD** (ang. *Document Type Definition*). Umożliwiają one kontrolę poprawności strukturalnej dokumentu, czyli sprawdzenie czy elementy posiadają odpowiednie nazwy, atrybuty i podelementy. Istnienie DTD ułatwiło korygowanie zawartości dokumentu XML już w trakcie jego tworzenia.

Od początku istnienia standardu zaczęły powstawać różne definicje typu dokumentu, specyficzne dla określonych zastosowań. Można powiedzieć, że za pomocą pojedynczego DTD definiuje się jeden nowy język wyprowadzony z metajęzyka XML. Każdy taki nowy język może być stosowany w wielu dokumentach XML. Przykładowo, DTD opisujące fakturę VAT tworzy XML-owy język dla faktur VAT, DTD dla sprawozdania finansowego firmy określa język sprawozdań finansowych firm, DTD dla płytoteki wyznacza język dla płytotek, itd.

Nowe języki zaczęły tworzyć firmy dla własnych wewnętrznych potrzeb a także grupy organizacji dla usprawnienia przepływu dokumentów pomiędzy tymi organizacjami¹. Ta dowolność tworzenia języków (przepisów na dokument) wcale nie rozwiązała problemu wymiany danych, gdyż różne organizacje mogły tworzyć własne, odmienne DTD. Siłą języka XML tkwi jednak w tym, że jeżeli posiadamy opisy dwóch struktur w postaci DTD to możliwa jest konwersja dokumentów XML zgodnych z jednym DTD (z jednego formatu) do dokumentów drugiego formatu (o innej strukturze, zgodnych z drugim DTD). Do automatyzacji tego procesu istnieją nawet na rynku odpowiednie produkty, np. *SQL BizTalk Server* firmy *Microsoft Corp.* czy *EntireX Orchestrator* firmy *Software AG*.

Aby umożliwić koegzystencję wielu języków w jednym dokumencie XML wprowadzono koncepcję **przestrzeni nazw** (ang. *namespace*). Dzięki przestrzeniom można stosować elementy o takiej samej nazwie, ale o innym znaczeniu i zawartości, pod warunkiem, że pochodzą z różnych przestrzeni nazw. Poszczególne nazwy są rozróżnialne dzięki temu, że poprzedza je prefiks odpowiadający przestrzeni, z której pochodzi nazwa — listing 1.

Listing 1. Dokument XML, w którym wykorzystano nazwy z czterech przestrzeni

```
<?xml version="1.0" ?>

  <!-- element faktura jest określony w przestrzeni z prefiksem wspólne -->
<wspólne:faktura xmlns:wspólne="file://DTD/faktura.dtd"
  xmlns:moje="file://DTD/moje.dtd"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:html="http://www.w3.org/TR/REC-html40">

  <!-- nazwa z przestrzeni nazw o prefiksie wspólne -->
<wspólne:nazwa>Faktura VAT</wspólne:nazwa>

  <!-- nazwa z przestrzeni nazw o prefiksie moje -->
<moje:nazwa>Zwykły dokument od kontrahenta</moje:nazwa>

  <!-- element p pochodzi z języka HTML -->
<html:p>Paragraf HTML</html:p>

  <!-- odsyłacz korzysta z atrybutów z przestrzeni języka XLink -->
<odsyłacz xlink:type="simple" xlink:href="http://www.przyklad.pl/fakturaOpis.xml" />

</wspólne:faktura>
```

Jednak przestrzenie powstały rok później niż XML 1.0 zawierający specyfikację DTD. Z tego powodu definicje DTD nie obsługują przestrzeni nazw co niejednokrotnie było istotnym ograniczeniem.

XML Schema

Definicje typu dokumentu posiadają sporo ograniczeń, z których wiele zostało przedstawionych w Software 2.0 nr 6/2001. W związku z tym rozpoczęto, długotrwałe prace nad nowym sposobem opisywania struktury dokumentów. I tak powstał zatwierdzony rok temu język **XML Schema**. Dzięki niemu można tworzyć schematy dokumentów². W porównaniu do DTD wprowadzono do niego bardzo wiele dodatkowych mechanizmów umożliwiających:

- określanie wzorców zawartości elementów i atrybutów (przede wszystkim poprzez aspekty — ang. *facets*, w których wykorzystano m.in. wyrażenia regularne);
- definiowanie własnych typów prostych lub złożonych; typy złożone zawierają atrybuty i (lub) podelementy; dodatkowo, jedne typy mogą być wyprowadzane z innych typów (zarówno wbudowanych³ jak i własnych);
- pełną obsługę przestrzeni nazw: deklarowane typy, elementy i atrybuty można przydzielać do własnych przestrzeni nazw (nawet do kilku w jednym schemacie); równocześnie w schematach można używać wielu przestrzeni, wykorzystując np. typy zdefiniowane w innych schematach;

¹ Taka elektroniczna wymiana danych pomiędzy organizacjami jest określana mianem EDI (ang. *Electronic Data Interchange*) a jeżeli dodatkowo wykorzystuje język XML, to jest to EDI XML.

² Przed pojawieniem się standardu XML Schema wielu autorów mianem *schematu* określało definicje typu dokumentu DTD.

³ XML Schema posiada także 44 typy wbudowane, określone przez standard (liczbowe, znakowe, czasowe, logiczne, binarne oraz związane bezpośrednio z językiem XML).

- wymuszenia (konieczność) występowania — zarówno dla elementów jak i atrybutów — a także precyzyjne określenia liczby wystąpień elementów;
- deklarowanie dowolnej kolejności podelementów — składacz (ang. *compositor*) *all*;
- domyślne oraz stałe wartości elementów (dla wartości atrybutów można je było deklarować już w DTD);
- wartości niepowtarzalne (ang. *unique*) oraz kluczowe (ang. *key*), jak również odesłania do tych wartości;
- wyliczenia dla elementów (dla atrybutów istniały już w DTD) oraz listy (zbiór wartości tego samego typu) i kombinacje (zbiór wartości różnych typów);
- precyzyjne określenie treści elementów o zawartości mieszanej (tekst i podelementy, łącznie);
- zastępcze nazwy elementów (istotne zwłaszcza w kontaktach międzynarodowych);
- pozostawienie miejsca na nowe elementy lub atrybuty, dodawane bezpośrednio w dokumencie XML, poza schematem;
- importowanie typów z innych schematów;
- złożone komentarze (ang. *annotation*).

Jednocześnie schematy XML Schema nie są pozbawione wad: zwykle są one bardziej rozbudowane niż deklaracje DTD. Obecnie istnieje więc dylemat: czy do opisu struktury dokumentu wybrać prostsze i popularniejsze definicje DTD czy zastosować bardziej skomplikowane, ale oferujące więcej możliwości schematy XML Schema?

Prezentacja informacji zawartych w dokumencie XML

XML powstał przede wszystkim jako język wizualizacji informacji w systemie WWW. Ponieważ jednak dokumenty XML — w przeciwieństwie do stron HTML — nie zawierają informacji na temat sposobu prezentacji, więc należało szybko opracować narzędzie umożliwiające wyświetlanie dokumentów w zwykłych przeglądarkach. Do tego celu miał służyć **język arkuszy stylów XSL** (ang. *Extensible Stylesheet Language*). Jednakże to co zatwierdzono pod koniec 1999 r., czyli XSLT (ang. *XSL Transformation*), trudno nazwać arkuszami stylów. Jest to język transformacji dokumentów XML na inne dokumenty, w szczególności transformacji do stron HTML. Tak naprawdę to nie ma w nim żadnych stylów dla prezentacji, przynajmniej w takim znaczeniu, jak się style rozumie potocznie czyli zbiór atrybutów wyglądu, któremu nadano pewną nazwę i którą to nazwę przyporządkowuje się jakimś fragmentom, po to, by wyglądały jednakowo.

XSLT ma jednak bardzo istotne znaczenie dla zastosowań języka XML. To właśnie transformacje do języka HTML spowodowały, że XML jest obecnie wykorzystywany w bardzo wielu portalach internetowych, zwłaszcza dużych i średnich.

W transformacjach XSLT z dokumentów XML do HTML można stosować **kaskadowe arkusze stylów CSS** (ang. *Cascade Style Sheet*), które są już typowymi arkuszami stylów. Standard CSS powstał po to, by ułatwić zarządzanie wyglądem stron HTML. Dzięki wykorzystaniu stylów łatwiej jest dokonywać modyfikacji wyglądu a także powielać ten sam wygląd dla wielu różnych fragmentów stron. CSS jest szczególnie przydatny dla stron o złożonej strukturze, zawierającej wiele elementów. Z tego powodu style CSS są często stosowane w dużych serwisach WWW o stronach zarówno tworzonych wprost w języku HTML jak i transformowanych z dokumentu (lub wielu dokumentów) XML.

Od kilku miesięcy istnieje jeszcze jeden język prezentacji: XSL FO czyli **XSL obiekty formatujące** (ang. *formatting objects*). Z pewnością nie zastąpi on języka HTML jako docelowego przy prezentacjach dokumentów XML w przeglądarkach. Powodem jest to, że wygląd określa się w nim, traktując dokument jako stronę (w sensie kartki papieru a nie dokumentu WWW). Być może zyska on jednak dużą popularność tam, gdzie ważna jest możliwość uzyskania postaci gotowej do wydruku.

Dokumenty XSL FO powstają zwykle jako efekt transformacji XSLT dokonywanych na zwykłych dokumentach XML. W tym sensie XSLT oraz XSL FO można — za WWW Consortium — traktować łącznie, jako całość, czyli XSL.

Obecnie wciąż wiele osób mówiąc *XSL* ma na myśli *XSLT*, chociaż nie są to synonimy. Wiąże się to m.in. z tym że XSL przez długi czas składał się jedynie z XSLT (nie było XSL FO). Dodatkowo jako zwyczajowy prefiks przestrzeni nazw dla XSLT przyjmuje się **xsl**, np. element główny transformacji to: **xsl:stylesheet**.

Wyszukiwanie i wskazywanie

Dla transformacji XSLT niezbędny był mechanizm wskazywania na fragmenty dokumentu XML, po to by określić które węzły dokumentu XML mają podlegać jakim przekształceniom. Z tego powodu wraz z XSLT zatwierdzono **język XPath**. Do podobnych celów XPath jest stosowany w schematach — po to by wskazać kluczowe i unikatowe elementy (lub atrybuty). W języku odsyłaczy hipertekstowych XLink XPath umożliwia precyzyjne łączenie konkretnych miejsc (elementów) a co za tym idzie łatwą nawigację zarówno po dokumencie jak i między dokumentami XML.

Podstawą XPath są wyrażenia, w których określa się warunki, jakie mają spełniać węzły na które wskazujemy, np. piąty element `faktura` lub elementy z atrybutem `wartość` pomiędzy 40 a 50. XPath posiada także mechanizmy przydatne przy wyszukiwaniu informacji, zwłaszcza operatory i funkcje. Z tego powodu jest on wykorzystywany jako język do wyszukiwania, np. w bazach XDBMS (ang. *XML Database Management Systems*), w których dane przechowywane są wprost jako struktury XML.

XPath posiada jednak pewne ograniczenia związane głównie z tym, że operuje on na węzłach (elementach). Nie jest możliwe np. wskazanie konkretnego punktu za znacznikiem początkowym lub fragmentu treści tekstowej elementu. Do jego możliwości wyszukiwawczych również można mieć zastrzeżenia, np. nie można wyszukiwać po tekstach rozciągających się na dwa elementy. Trudności te pokonano w innym języku, **języku wskazań — XPointer**, będącym rozszerzeniem XPath. Nie jest on jednak jeszcze standardem (rekomendacją) WWW Consortium, aczkolwiek wszystko wskazuje na to, że wkrótce się nią stanie. Można w nim wskazywać na punkty (zbiory punktów) lub zakresy (zbiory zakresów). Zakres to fragment dokumentu pomiędzy elementami lub punktami umieszczonymi nawet w różnych elementach. Zakres może zaczynać się w jednym elemencie i kończyć w innym. Przykładowo wyrażenie:

```
xpointer(//faktura/pozycja[3]/range-to(id("Podsumowanie2")))
```

określa zakres pomiędzy trzecią pozycją faktury a elementem o identyfikatorze `Podsumowanie2`.

Za pomocą wyrażen XPointer możliwe jest wyszukiwanie po połączonej treści tekstowej elementów, z pominięciem znaczników. Dokonuje się tego w ramach danego zakresu, którym w szczególności może być cały dokument, np. dla dokumentu z listingu 1 wyrażenie:

```
xpointer(string-range(//*, "VATZw"))
```

wskazuje miejsce rozciągające się od ostatnich 3 znaków elementu `wspólne:nazwa` do pierwszych 2 znaków elementu `moje:nazwa`.

Zarówno XPath jak i XPointer są stosowane w **języku odsyłaczy hipertekstowych XLink**.

Na tym jednak nie koniec. Obecnie WWW Consortium prowadzi intensywne prace, które podążają w dwóch kierunkach: XPath 2.0 oraz **XQuery**. Ich celem jest przede wszystkim rozszerzenie możliwości wyszukiwania w dokumentach XML. Dotychczasowe języki nie uwzględniają m.in. typów wbudowanych oraz wzorców zawartości zawartych w standardzie XML Schema. Nie jest więc możliwe np. wyszukanie wszystkich dat występujących w dokumencie XML. Nie można także dokonywać wyszukiwań w wielu dokumentach na raz. Na zatwierdzenie nowych standardów w dziedzinie wyszukiwania (oprócz XPointer) w ciągu najbliższych kilku miesięcy nie ma jednak co liczyć.

XML Base

XML Base jest chyba najprostszym i najkrótszym standardem związanym z językiem XML. Umożliwia on wstawianie do elementów atrybutu predefiniowanego⁴ `xml:base`, określającego adres odniesienia. Dzięki temu można w innych miejscach tego elementu (w jego elementach potomnych) stosować adresy względne — listing 2.

Listing 2. Atrybut `xml:base`

```
<?xml version="1.0" ?>
<filmoteka xml:base="http://www.przyklad.pl/filmy/całe"
  xmlns:xlink="http://www.w3.org/1999/xlink">

  <video xlink:type="simple" xlink:href=" ../fragmenty/film2.mpg">Fragment 2</video>

</filmoteka>
```

⁴ Dotychczas istniały jedynie dwa atrybuty predefiniowane: `xml:lang` (określający język, np. `pl` to polski) oraz `xml:space` (wyznaczający sposób przetwarzania *białych znaków*)

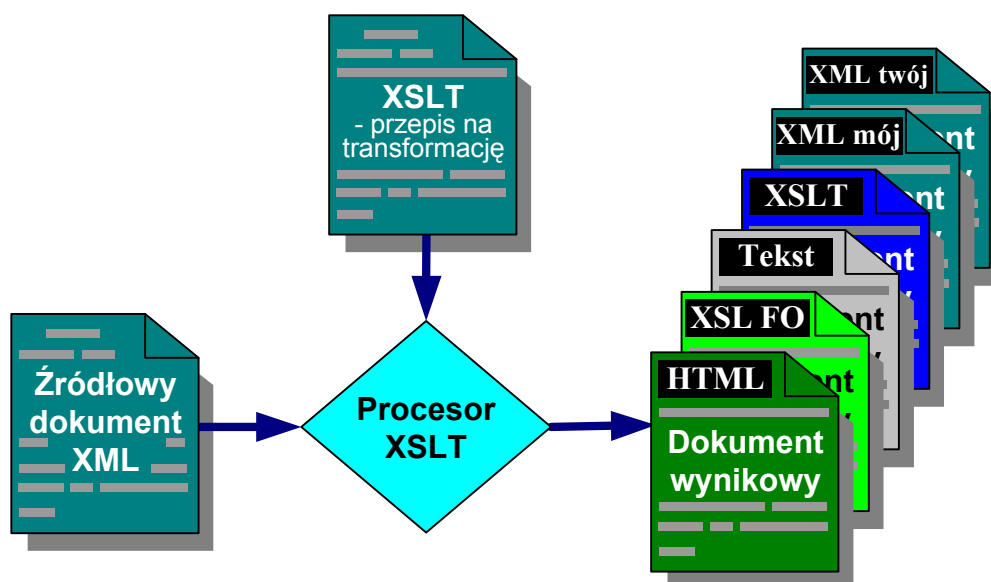
Adres względny zawarty w atrybucie `href` odsyłacza `video` (`../fragmenty/film2.mpg`) zostanie połączony z adresem odniesienia (`http://www.przyklad.pl/filmy/całe`). Innymi słowy, odsyłacz `video` wskazuje na plik:

`http://www.przyklad.pl/filmy/fragmenty/film2.mpg`

Ponieważ adresy wykorzystuje się przede wszystkim w odsyłaczach, więc standard XML Base jest ważny zwłaszcza dla języka XLink.

Przetwarzanie dokumentów

Zasadniczym językiem związanym z przetwarzaniem dokumentów XML jest **XSLT**. Nie jest to jednak język programowania, chociaż posiada pewne cechy takiego języka, np. pętle czy zmienne. Umożliwia on utworzenie *przepisu na transformację*. Przepis taki (dokument XSLT) składa się z listy wzorców. Podczas procesu transformacji — wykonywanego przez procesor XSLT — w dokumencie XML wyszukiwane są fragmenty wskazane we wzorcach. We wzorcach zawarta jest również informacja o docelowej postaci pasującego doń fragmentu dokumentu źródłowego. Dzięki temu możliwe jest dowolne kształtowanie efektu końcowego procesu transformacji. Język XSLT opisuje więc sposób na uzyskanie ze źródłowego dokumentu XML innego dokumentu: strony HTML, zwykłego tekstu lub innego dokumentu XML, np. XSL FO — rysunek 3. O tym, że XSLT w zupełności wystarcza do opisu przekształceń przeprowadzanych na dokumentach XML niech świadczy fakt, że obecnie nie prowadzone są żadne poważne prace związane z jego rozwojem.



Rysunek 3. Przetwarzanie dokumentu XML

Znaczenie języka XSLT jest bardzo duże. Najważniejsze jest to (było to przyczyną tak szybkiego zatwierdzenia standardu XSLT), że dzięki niemu można z dokumentów XML otrzymać strony HTML, które są prezentowane w dowolnych przeglądarkach⁵. Dodatkowo, tę samą transformację XSLT można zastosować do wielu dokumentów XML, co zapewnia wszystkim stronom wynikowym jednakową postać — ten sam wygląd. Zmiana wyglądu wielu stron (nawet całego, dużego serwisu internetowego) wymaga wtedy jedynie zmiany jednego pliku XSLT.

Oprócz tego, stosując różne transformacje XSLT do tego samego dokumentu XML można uzyskać różną postać tej samej informacji⁶. Dotyczy to zarówno wyglądu, układu (np. tabela, wielopoziomowa lista), jak i zawartości prezentowanej informacji, np. ukrycie pewnych fragmentów.

⁵ Transformacja dokumentów XML do HTML odbywa się zwykle po stronie serwera WWW. W przypadku konieczności wielorakiego wykorzystania przez klienta informacji zawartej w dokumencie XML, np. przesłanie jej do innej aplikacji lub bazy danych, dokument XML wraz ze związanym z nim plikiem XSLT może być przekazany klientowi i przetwarzany przez procesor XSLT zawarty w przeglądarce.

⁶ Nie możliwe jest dokonanie tego wprost, gdyż dokument XML może być powiązany tylko z jednym dokumentem XSLT. Jednym z rozwiązań jest zastosowanie wielu krótkich dokumentów nadrzędnych, z których każdy jest związany z inną transformacją. W tych krótkich dokumentach za pomocą jednostki (encji) zewnętrznej włączany jest ten sam dokument

Z przetwarzaniem dokumentów XML wiążą się także inne standardy: **DOM** i **SAX**⁷. Tworzą one programowy interfejs API pomiędzy aplikacją a dokumentem XML. W implementacjach DOM (ang. *Document Object Model*) wczytywany jest cały dokument XML a jego zawartość jest odwzorowywana na drzewo węzłów. Drzewo jest udostępniane aplikacji, umożliwiając manipulowanie poszczególnymi fragmentami tego drzewa, w tym także zmianę jego struktury lub dodawanie nowych elementów.

SAX (ang. *Simple API for XML*) natomiast daje dostęp do aktualnie, strumieniowo (sekwencyjnie) przetwarzanej części dokumentu. Ponieważ nie każde przetwarzanie hierarchicznego dokumentu XML można dokonać poprzez strumień (mamy dostęp tylko do aktualnie przetwarzanego fragmentu), więc SAX nie nadaje się np. do zmiany struktury dokumentu XML. Obie koncepcje dostępu do dokumentu z poziomu aplikacji zostały opisane w numerze 6/2001.

XHTML

Ostatnią zatwierdzoną wersją języka HTML jest wersja 4.01. Wydaje się, że na tym rozwój tego języka właściwie się zakończył i obecnie zmierza on w kierunku języka XML, czyli przeobraża się w **XHTML** (ang. *Extensible HyperText Markup Language*). Pierwszym wymiernym krokiem na tej drodze było zatwierdzenie standardu XHTML 1.0, w którym HTML 4 zdefiniowano jako język wyprowadzony z XML (a nie SGML). Można powiedzieć, że dokumenty XHTML to po prostu poprawnie sformułowane dokumenty HTML, czyli takie, w których: małe i duże litery w nazwach elementów i atrybutów są rozróżniane, niepuste elementy są zakończone znacznikiem końcowym a wartości atrybutów umieszczone są cudzysłowach lub apostrofach.

Kolejnym krokiem jest umożliwienie dodawania do XHTML nowych elementów (zbiór elementów HTML jest stały) poprzez wydzielenie rozszerzalnych modułów oraz udostępnienie mechanizmów łączenia z innymi językami np. z MathML. W ten sposób, rok temu, powstał XHTML 1.1.

Jednocześnie wydzielono z języka XHTML 1.0 minimalny podzbiór elementów, który ułatwiłby prezentację dokumentów XHTML w urządzeniach o ograniczonej możliwości przetwarzania, w szczególności w urządzeniach przenośnych PDA (ang. *Personal Digital Assistant*). Można jednak mieć wątpliwości czy ograniczanie możliwości języka jest tym, czego oczekuje rynek urządzeń przenośnych, których moce obliczeniowe są już bardzo duże i dla których istnieją przecież graficzne systemy operacyjne, np. Windows CE.

Języki specjalistyczne

WWW Consortium prowadzi i prowadzi nadal prace jedynie nad niektórymi, wybranymi zastosowaniami języka XML — patrz tabela 1. Obecnie rozwijane są np. standardy związane z zabezpieczaniem dokumentów XML, co ma ułatwić ich stosowanie m.in. w handlu elektronicznym. W efekcie tego powstał standard podpisu elektronicznego dla dokumentów XML, zaś bliskie zatwierdzenia są standardy związane z szyfrowaniem i deszyfrowaniem.

Dużo więcej specjalistycznych standardów jest opracowywanych przez zespoły a nawet pojedyncze osoby blisko związane z dziedziną ich zastosowania, z pominięciem organizacji międzynarodowych, np. język dla astronomii AML zaproponował D. Guillaume, zaś XML-owy odpowiednik standardu opisu bibliograficznego — BiblioML opracowała grupa sponsorowana przez Francuskie Ministerstwo Kultury. Wiele przykładów specjalistycznych zastosowań języka XML zostało umieszczonych przez Robina Covera na stronach OASIS:

<http://xml.coverpages.org/xml.html#applications>.

Podsumowanie

Język XML to obecnie nie pojedynczy standard, ale rodzina języków. Dzięki nim jest on coraz szerzej wykorzystywany w bardzo wielu dziedzinach. Jedną z najważniejszych z nich są serwisy internetowe. Obecnie już bardzo wiele portali, także polskich, stosuje XML jako warstwę pośrednią pomiędzy źródłami informacji, np. bazami danych a stronami HTML przekazywanymi przez serwer WWW do przeglądarek klientów.

źródłowy. Dokumenty nadrzędne mają więc taką samą zawartość, ale inny dowiązany plik XSLT. W przeglądarce są więc inaczej prezentowane.

⁷ SAX w przeciwieństwie do DOM nie jest standardem WWW Consortium

Narzędzia do opisywania struktury dokumentów (DTD a zwłaszcza język XML Schema) odpowiadają za to, że XML na dobre zadomowił się w systemach elektronicznej wymiany danych EDI, o czym świadczy dynamiczny rozwój różnych tzw. inicjatyw EDI XML.

Precyzyjne definicje struktury, język wyszukiwania i wskazywania — XPath oraz mechanizmy manipulowania danymi poprzez DOM API umożliwiły powstanie baz danych XML, czyli XDBMS.

Oprócz tego język XML można stosować jako format do opisu danych pseudostrukturalnych (notatek, sprawozdań, wiadomości) a także specjalistycznych, np. struktur chemicznych (CML — ang. *Chemical Markup Language*) czy wzorów matematycznych (MathML).

Dynamiczny rozwój obszarów zastosowań jest równocześnie przyczyną powstawania coraz to nowych standardów związanych z tym językiem. Już czekają na zatwierdzenie kolejne propozycje rekomendacji (ang. *Proposed Recommendations*) i kandydaci na rekomendację (ang. *Candidate Recommendations*).

ONLINE:

Aktualna lista standardów WWW Consortium: <http://www.w3.org/TR>

XML Schema: <http://www.w3.org/XML/Schema>

Języki służące do wyszukiwania: <http://www.w3.org/XML/Query>, <http://xml.coverpages.org/xmlQuery.html>

Języki związane z odsyłaczami: <http://www.w3.org/XML/Linking>

Zastosowania XML: <http://xml.coverpages.org/xml.html#applications>