

# XBRL Wprowadzenie do standardu raportowania w biznesie

Barbara Reimschuessel-Wąs  
Asseco Poland SA

*barbara.was@asseco.pl*

**Abstrakt.** Jednym z pytań, jakie często się nasuwają w związku z kryzysem gospodarczym, który ujawnił się jesienią ubiegłego roku jest pytanie o wartość obligatoryjnej sprawozdawczości finansowej. Dlaczego mimo licznych wymogów instytucji nadzorujących nie były one w stanie dostatecznie szybko reagować, a istniejące problemy przez dłuższy czas się nie ujawniały? I zaraz potem rodzą się następne pytania o to, co można zrobić, aby w przyszłości analiza raportów przynosiła lepsze efekty. Odpowiedzi, głoszone zresztą już od paru lat, wskazują na konieczność standardu. Trudno jest mówić o efektywnej kontroli, jeśli przekazywane dane nie są spójne i nie można ich analizować w jednolity sposób.

Z potrzeby standardu wymiany danych zrodził się XBRL, czyli Extensible Business Reporting Language. Jest to, oparta na technologiach XML specyfikacja, definiująca sposób opisu wspólnych pojęć występujących w sprawozdaniach biznesowych oraz zależności między nimi.

XBRL nie jest odkryciem ostatniego roku, już od jakiegoś czasu mieli z nim do czynienia bankowcy, ale, ostatnio, być może w związku z sytuacją na rynkach, której jesteśmy świadkami, nacisk na jego stosowanie znacznie się nasilił. Przyczyniła się do tego także SEC (Security and Exchange Commission – instytucja odpowiadająca za nadzór finansowy w USA), która w grudniu 2008 roku zdecydowała, że do 2011 roku wszystkie amerykańskie firmy swoje obowiązkowe raporty finansowe będą musiały przygotowywać przy użyciu i w formacie XBRL. A to oznacza oczywiście, że już wkrótce każde oprogramowanie do raportowania na rynku amerykańskim będzie tworzyło raporty w formacie XBRL. Może więc niedługo znajomość XBRL będzie częścią elementarza każdego specjalisty od systemów raportowania?

Referat niniejszy stara się pokazać, czym XBRL jest i w jaki sposób można go stosować. Przedstawiona zostanie jego struktura, sposób rozwoju oraz najważniejsze zagadnienia, których rozstrzygnięcie jest istotne dla jego dalszego rozwoju. Pokazane zostaną również przykłady raportów w formacie XBRL i sposoby ich przeglądania.

Już obecnie istnieje wiele rozwiązań programistycznych wspierających XBRL, a jednym z pierwszych był pakiet Oracle Hyperion EPM. Wciąż też anonsowane są nowe systemy. Najbardziej znanym z nich zostanie poświęcona osobna część wystąpienia.

Na koniec omówione będą sposoby wizualizacji raportów XBRL, a wśród nich wykorzystanie Oracle BI Publisher-a, który przecież został stworzony do tego, by w łatwy sposób pokazywać w wielorakich postaciach dane XML.

## 1. Wstęp

Dość często można się dzisiaj spotkać z poglądem, że obecny wokół nas i na rynkach finansowych kryzys dowodzi konieczności bardziej skutecznych działań różnych organów nadzoru finansowego. Niezależnie czy się z tym zgadzamy, czy nie, zauważyć trzeba, że jednym z koniecznych warunków wszelkiego nadzoru jest dostęp do informacji oraz możliwość efektywnego przetwarzania tych informacji. Można bowiem coraz to zwiększać wymagania związane z przekazywaniem danych, ale jeśli jedynym tego efektem będzie konieczność studiowania ton nieporównywalnych ze sobą, papierowych sprawozdań, niewiele to wniesie. Dlatego też tym pilniejsza stała się potrzeba ustalenia formatu wymiany informacji biznesowych, który zapewniałby:

- łatwość przetworzenia takich informacji w systemie informatycznym,
- przejrzystość i jednoznaczność,
- możliwość porównywania między sobą danych pochodzących od różnych podmiotów.

O ile pierwsze wymaganie jest dość proste do spełnienia i możemy zgodnie z prawdą zapewnić, że każdy nasz system je spełnia, bo z łatwością potrafi produkować dane w postaci elektronicznej, to już jeśli dłużej zastanowić się nad przejrzystością, okaże się, nie jest ona łatwa do osiągnięcia. Bo jeśli ktoś czyta raporty z różnych systemów, na których widnieją wartości liczbowe opatrzone nawet podobną etykietą słowną, to jaka jest podstawa do stwierdzenia, że ta wartość jest w obu przypadkach liczona w ten sam sposób? Aby tak było, raporty musiałyby być opatrywane wielkimi komentarzami opisującymi ich kontekst i sposób kalkulacji. A takie komentarze trudno byłoby już przetwarzać elektronicznie.

Najgorzej jednak oczywiście wygląda sprawa spełnienia wymagania ostatniego. Formaty danych, nawet, jeśli uwzględnimy, że wiele z nich wykorzystuje dziś XML są ze sobą zupełnie nieporównywalne. Jeśli wpływają one do jednej instytucji nadzorującej, można je co najwyżej wspólnie przechowywać, angażując do każdej analizy merytorycznej człowieka. Pewnym rozwiązaniem, powszechnie dotąd stosowanym, jest w takiej sytuacji narzucenie jednego konkretnego formatu przez odbiorcę danych. Zapewnia ono pewną porównywalność (pewną, bo z ograniczeniami wynikającymi z braku dostatecznej przejrzystości, o których była mowa wcześniej), jest jednak mało elastyczne. Wymagania informacyjne się zmieniają, stąd stała potrzeba zmiany takiego interfejsu, z czego najbardziej są zadowoleni dostawcy systemów informatycznych.

Wiadomo jednak, że tam gdzie istnieją silne potrzeby, pojawiają się rozwiązania. I właśnie jednym z nich, coraz powszechniej przyjmowanym w wielu krajach, jest XBRL, czyli eXtensible Business Reporting Language. Pod koniec zeszłego roku amerykański nadzór finansowy SEC czyli Security and Exchange Commission (odpowiednik polskiej KPWiG) wydał regulację wymuszającą stopniowe przejście na XBRL dla wszystkich firm raportujących do SEC do połowy 2011 roku. Pierwsza grupa raportów od 500 największych przedsiębiorstw, zgodnie z harmonogramem, wpłynęła już w lipcu. Oznacza to, że już wkrótce każdy system raportowy, operujący na danych finansowych będzie musiał umieć generować XBRL a znajomość jego reguł będzie coraz powszechniejsza wśród informatyków.

## 2. Czym jest XBRL?

### 2.1. Język, metodyka, a może narzędzie?

#### 2.1.1. Język

W pierwszej linii w lewym menu, na głównej stronie ([www.xbrl.org](http://www.xbrl.org)) konsorcjum XBRL International (o którym będzie jeszcze mowa) można znaleźć to samo pytanie. Po kliknięciu dostajemy poniższą odpowiedź:

„XBRL is a language for the electronic communication of business and financial data which is revolutionizing business reporting around the world.”

I dalej jeszcze kilka zdań w podobnym, marketingowym stylu.

Jednak jeśli weźmiemy pod uwagę choćby sam skrót, to już samo jego rozwinięcie sporo wyjaśnia. eXtensible Business Reporting Language oznacza, że mamy do czynienia z językiem do tworzenia raportów biznesowych. Z pojęć użytych w tej nazwie, najwięcej wątpliwości budzi sam obszar zastosowania – „raporty biznesowe”. Nie jest celem tego referatu roztrząsanie co jest a co nie jest przedmiotem raportowania biznesowego i nie ma takiej potrzeby. XBRL jest tak zbudowany, że dysponując kompletnym dokumentem w formacie XBRL, możemy dokładnie i precyzyjnie określić obszar, którego on dotyczy, ponieważ raport XBRL składa się zawsze z 2 części:

- „Słownika”, zwanego taksonomią, zawierającego czasami odwołania do innych, dostępnych „słowników” oraz
- Instancji, czyli zbioru właściwych, raportowanych informacji, zwanych „faktami”

Sformułowanie „eXtensible Language” sugeruje z kolei, że mamy do czynienia z XML i tak rzeczywiście jest. Dodatkowo wiemy, że nie chodzi o żaden konkretny format wymiany danych (w przeciwieństwie do np. EDI), ale o język, używając którego można tworzyć raporty o różnych strukturach. Język ten określa sposób budowy taksonomii i raportów (nazywanych instancjami) na niej opartych. Pozwala także na łatwą rozszerzalność. Jeśli więc mielibyśmy np. taksonomię opisującą wymagania sprawozdawczości w kraju, a firma miałaby jeszcze dodatkowe potrzeby raportowe, wystarczyłoby, żeby stworzyła własną, rozszerzającą, wewnętrzną taksonomię, w której odwoływałaby się do krajowej.

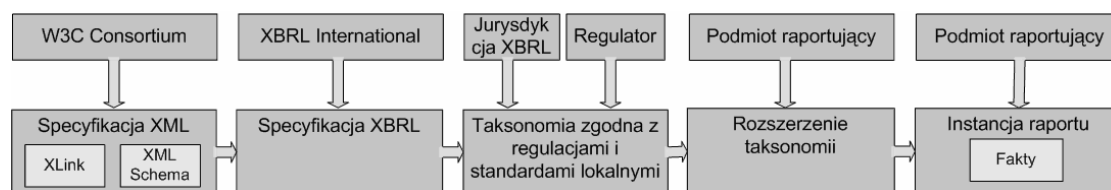
W raporcie XBRL każdemu faktowi towarzyszy odpowiedni znacznik (tag) XML, którego znaczenie opisane jest w dołączonym, utworzonym zgodnie ze specyfikacją XBRL „słowniku”, czyli taksonomii. Ktoś, kto miał do czynienia z XML, nie znajdzie w tym co powyżej, na pierwszy rzut oka, nic nowego. Przecież od dawna do takich właśnie celów wykorzystywana jest XML Schema. Co więc nowego wnosi XBRL?

Rzeczywiście taksonomia XBRL oparta jest na XML Schema, ale nie tylko. Największą różnicą jest to, że taksonomia XBRL opisuje nie tylko formalne cechy poszczególnych znaczników, ale także wyraża ich semantykę, czyli znaczenie (zwane tutaj „concept”). Do tego nie wystarczy specyfikacja samych atrybutów poszczególnych znaczników i ich możliwych wartości, ale konieczne jest także zdefiniowanie relacji między nimi. XBRL to umożliwia, dzięki czemu znając taksonomię raportu XBRL można automatycznie zweryfikować nie tylko jego syntaktykę, jak to jest w przypadku XML, ale także semantykę, na przykład sprawdzając czy podsumowania rzeczywiście są uzyskiwane z sumowania poszczególnych składników.

#### 2.1.2. Proces, choć nie metodyka

Ponieważ taksonomie opisują pojęcia, a te ostatnie w sprawozdawczości finansowej oparte są często na regulacjach prawnych, naturalną rzeczą jest istnienie jednej, wspólnej dla danej regulacji, podstawowej taksonomii. Tak też jest to realizowane. Zbudowana została taksonomia oparta o US-GAAP (Generally Accepted Accounting Principles), jak i taksonomia zgodna z międzynaro-

dowymi standardami rachunkowości. Relacje między specyfikacją XML, XBRL, taksonomią i raportem można przedstawić następująco:



Rys. 1

Schemat ten narzuca pewną kolejność przy budowaniu i wersjonowaniu kolejnych raportów, można więc myśleć o XBRL także jako o pewnej metodzie, choć jeszcze nie metodyce.

### 2.1.3. Nie narzędzie, ale wspierany przez narzędzia

XBRL jako pewna specyfikacja jest narzędziem tylko w takim samym znaczeniu jak jest nim każdy specjalizowany język. Trzeba sobie jednak zdawać sprawę, że posługiwanie się do tworzenia raportów XBRL samym edytorem tekstowym, nawet jeśli jest to edytor przystosowany do pracy z XML jest praktycznie niemożliwe. Taksonomie zgodne z wymogami ustawowymi różnych krajów liczą od kilku do kilkunastu tysięcy powiązanych ze sobą pojęć, a przecież często przedsiębiorstwa rozszerzają je, tworząc własne taksonomie. Istnieje więc duża potrzeba tworzenia i udoskonalania narzędzi XBRL. Więcej o różnych rodzajach narzędzi oraz konkretnych, dostępnych rozwiązaniach powiedziane zostanie w rozdziale 4

## 2.2. Skąd się pojawił XBRL? Historia i terażniejszość

Pomysłodawcą i spiritus movens początków XBRL jest Charles Hoffman, który stworzył pierwszy prototyp XBRL już w 1998 roku, czyli w tym samym roku, w którym pojawiła się pierwsza rekomendacja W3C dla XML. W 2000 roku pojawiła się pierwsza specyfikacja XBRL firmowana przez ówczesny Komitet Sterujący XBRL. Opierała się ona jeszcze na DTD (Document Type Definition), ale już w 2001 roku pojawiła się wersja 2.0 wykorzystująca XML Schema. Obecna specyfikacja – 2.1 powstała w roku 2003, co pewien czas powstają jednak nowe rekomendacje, które ją rozszerzają. Najważniejsze to:

- Wprowadzenie wielu wymiarów – Dimensions 1.0 w 2006 roku
- Dynamiczne odwołania – Generic Links 1.0 22-06-2009 roku (jako część rekomendacji dla formuł)
- Formuły dla bardziej złożonych obliczeń i zależności – Formula Specification 1.0 22-06-2009 roku

Obecnie XBRL jest bezpłatnym standardem rozwijanym przez konsorcjum XBRL International skupiające ok. 550 firm i instytucji z całego świata. Ma ono strukturę składającą się z tzw. lokalnych jurysdykcji, najczęściej o zasięgu krajowym, których jest obecnie 27, w tym 22 stałe i 5 tymczasowych (m.in. XBRL Polska).

Na całym świecie prowadzone są projekty wspierające raportowanie w XBRL i w coraz to więcej obszarach staje się ono standardem wymaganym. Ci, którzy mają do czynienia z bankowością, wiedzą, że od 2007, w ramach wdrażania postanowień Nowej Umowy Kapitałowej, banki polskie są zobowiązane raportować przy pomocy taksonomii COREP (opracowanej przez Committee of European Banking Supervisors), zgodnie z wymaganiami europejskiego nadzoru i KNF. Projekt zastosowania XBRL w obowiązkowej sprawozdawczości prowadzi też Centrum Obsługi Kancelarii Prezesa Rady Ministrów, które obsługuje publikowanie sprawozdań finansowych jednostek, od których wymaga tego ustawa, w Monitorze Polskim B.

Dla rynku rozwoju oprogramowania jednak najważniejszym czynnikiem oznaczającym konieczność implementowania narzędzi XBRL, może być fakt, że w grudniu 2008 roku amerykańska komisja nadzorująca rynek papierów wartościowych (SEC) nakazała wszystkim podlegającym jej podmiotom przejście w ciągu 3 lat na dostarczanie danych w postaci XBRL. Od tego czasu, co i raz kolejny z dużych dostawców (patrz [VRSVR]) ogłasza, że jego aplikacje wspierają w pełni XBRL. W lutym zrobił to SAP (SAP Business Objects XBRL Publishing), marcu Infor, a nieco wcześniej IBM/Cognos. Jak na tym tle wypada Oracle? Dzięki akwizycji Hyperiona całkiem dobrze, o czym będzie więcej w rozdziale 5.

### 3. Co się składa na raport XBRL?

Na pytanie postawione powyżej można odpowiadać w terminach samego XBRL i wtedy odpowiedź jest krótka – raport XBRL to instancja i taksonomia. Można jednak także patrzeć na budowę raportu XBRL także bardziej technicznie, jako na dokument XML i wtedy pytanie o składowe XBRL będzie pytaniem o użyte w nim technologie XML. Są to XML Schema i XLink.

Jak było już powiedziane wcześniej, XBRL jest dialektem XML. Dokumenty XML czasami określane są potocznie jako samo się opisujące. Jednak sama tylko obecność znaczników wcale tego nie gwarantuje o ile znacznikom nie towarzyszy zewnętrzny słownik, który ściśle definiuje ich znaczenie i zależności pomiędzy nimi. Wynikiem jednak takiego podejścia jest sztywna struktura znaczników. Samo wykorzystanie XML Schema niewiele poprawia, bo pozwala tylko na weryfikowanie syntaktycznej postaci dokumentu. W sukurs przychodzą 2 dodatkowe rekomendacje W3C: XML Linking Language (XLink) (patrz [XLREC]) i XPointer ([XPREC]).

XPointer, zgodnie ze swoją nazwą, pozwala wskazać, o który element chodzi, czyli zapewnia składnię potrzebną w XLink do budowania wyrażeń wskazujących konkretny fragment dokumentu XML. Natomiast kluczową rolę odgrywa XLink, który umożliwia definiowanie prostych, ale także złożonych powiązań między pojęciami, mogącymi być zarówno elementami dokumentów XML, jak i zasobami zewnętrznymi. Dzięki temu jedne pojęcia mogą się wywodzić i odwoływać do innych. Rodzaje relacji określone są w specyfikacji XBRL, co pozwala na większą jednoznaczność

O ile schematy XML są powszechnie znane i raczej nie trzeba wyjaśniać ich stosowania, to XLink, z którego pochodzi choćby używany już wcześniej termin „linkbase”, stosowany jest rzadziej, dlatego warto poświęcić mu parę zdań.

#### 3.1. XLink – język definiowania relacji

Oдно́niki (links) są podstawą HTML, stąd od początku zasadność ich użycia w XML była oczywista. Jednak możliwości tworzenia powiązań w HTML są mocno ograniczone. Nie można na przykład:

- Zdefiniować odnośnika nie zmieniając elementu, który chcemy traktować jako źródło, czyli punkt startowy odnośnika. Jest to często konieczne jeśli chce się powiązać ze sobą elementy dokumentów, których nie można zmieniać. Temu służy właśnie linkbase – zewnętrzny w stosunku do źródła i celu dokument definiujący link.
- Określać różne kierunki nawigacji pomiędzy powiązanymi ze sobą elementami. Zdarza się przecież, i to bardzo często, że istnieją relacje w obie strony.
- Pozwalać na tworzenie odnośnika nie tylko od pojedynczego źródła do celu, jak to jest obecnie w HTML, ale powiązania do wielu celów, a nawet relacji wiele do wielu

Wszystkie te możliwości oferuje XML Linking Language - XLink, dostępna jako część technologii XML od 2001 ([XLREC]). Pozwala ona tworzyć 2 rodzaje powiązań: proste (simple), działa-

jące podobnie jak dotychczasowe znaczniki odnośników w HTML, oraz rozszerzone (extended). Właśnie te ostatnie jedną z podstawowych konstrukcji XBRL.

Poniżej prosty przykład, który pokazuje sposób zapisu odnośnika typu rozszerzonego w XML. Załóżmy, że mamy dokument XML opisujący (w bardzo uproszczony sposób) fakturę. Mógłby on wyglądać mniej więcej tak:

```
<?xml version="1.0" encoding="UTF-8"?>
<faktura>
  <dataWystawienia>12-03-2009</dataWystawienia>
  <dataSprzedazy>08-03-2009</dataSprzedazy>
  <klient>
    <ID>23812</ID>
    <nazwa>Usługi Kateringowe sp. z o.o.</nazwa>
    <adres>
      <miasto>Legnica</miasto>
      <ulica>Sienkiewicza</ulica>
      <kod>59-220</kod>
      <nr>12</nr>
    </adres>
  </klient>
  <linia>
    <towar>Talerz głęboki</towar>
    <ilosc>12</ilosc>
    <cena>38</cena>
  </linia>
  <linia>
    <towar>Widelec</towar>
    <ilosc>100</ilosc>
    <cena>4,50</cena>
  </linia>
</faktura>
```

Patrząc jednak na poszczególne części takiego XML jak na dane, wydaje się naturalne, że powinny być one zgrupowane w 3 osobnych dokumentach, co łatwo jest osiągnąć nawet przy pomocy prostych (simple) odnośników:

```
<?xml version="1.0" encoding="UTF-8"?>
<faktura xmlns:xlink="http://www.w3.org/1999/xlink">
  <dataWystawienia>12-03-2009</dataWystawienia>
  <dataSprzedazy>08-03-2009</dataSprzedazy>
  <klient xlink:type="simple" xlink:href="klienci.xml#23812"/>
  <linia>
    <towar xlink:type="simple" xlink:href="towary.xml#452855"/>
    <ilosc>12</ilosc>
    <cena>38</cena>
  </linia>
  <linia>
    <towar xlink:type="simple" xlink:href="towary.xml#733691"/>
    <ilosc>100</ilosc>
    <cena>4,50</cena>
  </linia>
</faktura>

<klienci>
  <klient>
    <id>23812</id>
```

```

    <nazwa>Usługi Kateringowe sp. z o.o.</nazwa>
    <adres>
      <miasto>Legnica</miasto>
      <ulica>Sienkiewicza</ulica>
      <kod>59-220</kod>
      <nr>12</nr>
    </adres>
  </klient>
<klient>
  <id>23816</id>
  <nazwa>Sala Weselna Amor s.c.</nazwa>
  <adres>
    <miasto>Łódź</miasto>
    <ulica>Pojezierska</ulica>
    <kod>91-322</kod>
    <nr>6</nr>
  </adres>
</klient>
</klienci>

<towary>
  <towar>
    <nazwa>Talerz głęboki</nazwa>
    <id>452855</id>
    <cena_ew>36</cena_ew>
  </towar>
  <towar>
    <nazwa>Widelec</nazwa>
    <id>733691</id>
    <cena_ew>2,68</cena_ew>
  </towar>
</towary>

```

Jak można zauważyć na powyższym przykładzie XLink pojawia się w dokumencie XML w postaci atrybutów elementów. W przypadku prostych odnośników są to atrybuty:

- xlink:type – z wartością „simple”
- xlink:href – z wartością wskazującą, gdzie znajduje się punkt docelowy odnośnika

Można też jednak definiować relacje między poszczególnymi elementami na zewnątrz tych elementów, używając odnośników typu „extended”. Poniższy przykład pokazuje, jak wyglądałoby to dla powiązania faktury z klientem. Dokument faktury nie zawiera tym razem żadnego odwołania do klienta:

```

<faktura>
  <dataWystawienia>12-03-2009</dataWystawienia>
  <dataSprzedazy>08-03-2009</dataSprzedazy>
  <linia>
    <towar xlink:type="simple" xlink:href="towary.xml#452855"/>
    <ilosc>12</ilosc>
    <cena>38</cena>
  </linia>
  <linia>
    <towar xlink:type="simple" xlink:href="towary.xml#733691"/>
    <ilosc>100</ilosc>
    <cena>4,50</cena>

```

```
</linia>
</faktura>
```

Pojawia się za to nowy dokument z odnośnikami:

```
<powiazania xmlns:xlink="http://www.w3.org/1999/xlink">
  <fakt_klient_pow xlink:type="extended">
    <klient_loc xlink:type="locator" xlink:href="klienci.xml#23812"
xlink:label="klient" xlink:title="Katering"/>
    <faktura_loc xlink:type="locator"
xlink:href="faktura.xml#xpointer( numer ('45/09'))" xlink:label="faktura"
xlink:title="Faktura 45/09"/>
    <fakt_klient_arc xlink:type="arc" xlink:from="faktura"
xlink:to="klient"/>
  </fakt_klient_pow>
</powiazania>
```

Jak widać odnośniki typu rozszerzonego same są elementami z atrybutem wskazującym typ odnośnika. Mogą one zawierać odsyłacze (locator) do zewnętrznych zasobów jak i wskazania na zasoby lokalne (resources, identyfikowany poprzez atrybut `xlink:type="resource"` – w naszym przypadku, ponieważ odnośnik jest zewnętrzny, nie ma zasobów lokalnych).

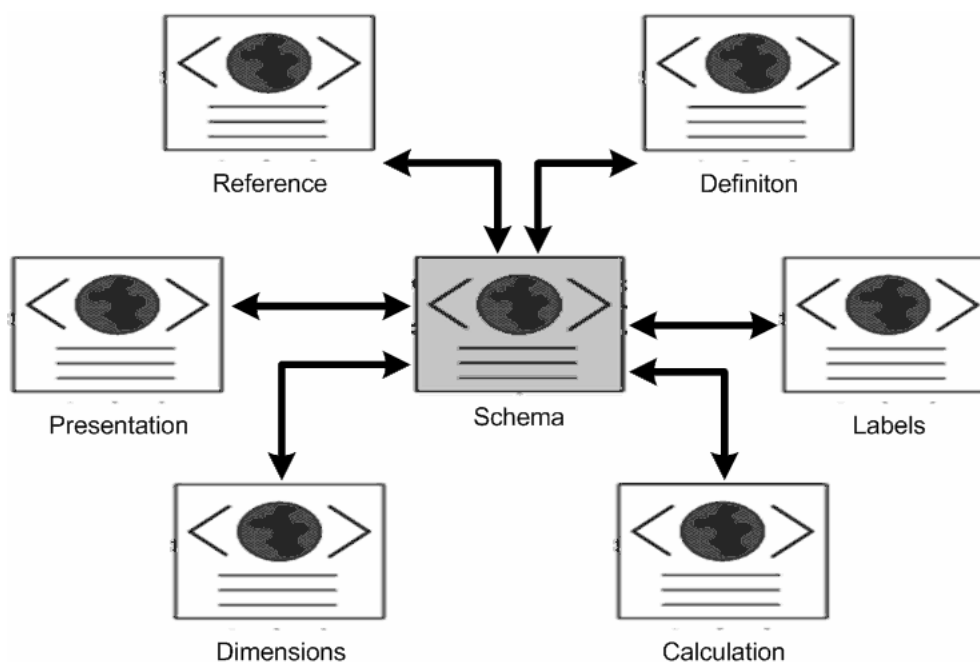
Elementem wewnętrznym odnośnika jest także samo przejście (arc, w powyższym przykładzie `fakt_klient-arc`), posiadający atrybuty:

- `xlink:type="arc"`
- `xlink:from`, przyjmujący na ogół wartość etykiety (`xlink:label`) odsyłacza (locator), będącego punktem startowym
- `xlink:to`, przyjmujący wartość etykiety z punktu docelowego
- `link:arcrole`, za pomocą którego można tworzyć pewne klasy relacji nadając mu wartości ze ściśle określonego zbioru, np. wartość „parent-child”. Należy podkreślić, że sama rekomendacja XLink nic nie mówi o tym, jakie to mają być wartości, zostawiając dowolność konkretnym zastosowaniom. W naszym przypadku takim zastosowaniem jest specyfikacja XBRL, która określa szereg typów relacji, które można stosować w XBRL.

### 3.2. Taksonomia

Jak już było wspomniane wyżej, raport XBRL składa się z instancji, zawierającej wszystkie potrzebne fakty, czyli raportowane informacje oraz taksonomii, stanowiącej słownik definiujący pojęcia (concepts), do których odnoszą się fakty oraz klasyfikację (czyli usystematyzowanie, stąd słowo taksonomia) tych pojęć. Klasyfikacja obejmuje relacje między pojęciami, które mogą obejmować również reguły kalkulacji, potrzebne do weryfikacji danych.

Taksonomia składa się z kilku odwołujących się do siebie części, określanych czasem mianem DTS (Discoverable Taxonomy Set), co pokazuje poniższy diagram:



Rys. 2

Wszystkie pojęcia taksonomii znajdują się w schemacie XSD<sup>1</sup>. Do ich pełnego zdefiniowania potrzebne są jednak dodatkowe informacje, w tym relacje między pojęciami i temu służą tzw. bazy powiązań (linkbases, zdefiniowane w specyfikacji XLink). Standardowo w taksonomii występują następujące bazy powiązań:

- Labels Linkbase – zawiera etykiety pojęć, do wykorzystania przy wizualizacji instancji raportu. Umożliwia tworzenie wersji językowych taksonomii.
- Reference Linkbase – pokazuje odwołania do źródeł zewnętrznych, definiujących dane pojęcie, jak na przykład obowiązujące akty prawne.
- Definition Linkbase – definiuje relacje między pojęciami takie jak: „general-special” (relacja generalizacji, znana każdemu programiście), „essence-alias” (oznaczająca inne ujęcie tego samego pojęcia), „requires-element” (pokazująca, że jeden element wymaga wystąpienia innego)
- Presentation Linkbase – określa jak pojęcia są zorganizowane, czyli ich hierarchię, kolejność oraz ich organizację w postaci np. wierszy tabeli. Umożliwia późniejsze formatowanie raportu. Składa się głównie z zależności rodzic – dziecko (parent-child) definiowanych za pomocą odnośników „extended” z odpowiednim atrybutem xlink:arcrole (odnośniki, czyli linki XLink przedstawione były krótko w rozdziale 3.1).
- Calculation Linkbase – definiuje podsumowania i proste wyliczenia poprzez przypisanie do każdego pojęcia wagi +1 (wielkość jest dodawana) lub -1 (odejmowanie) oraz wskazanie pojęcia nadrzędnego, do którego inne w grupie powinny się sumować.
- Formula Linkbase – opisuje bardziej złożone wyliczenia i reguły biznesowe, oparte na ostatniej rekomendacji XBRL International dotyczącej formuł.

<sup>1</sup> Schematów XML Schema może być czasem więcej, mogą być np. podzielone na Base Taxonomy Schema oraz Extended Taxonomy Schema.

### 3.2.1. Schemat taksonomii

Najważniejszym składnikiem taksonomii jest schemat, bo to on określa zawartość i strukturę instancji, czyli tej części raportu XBRL, która zawiera konkretne informacje.

W schemacie można:

- Definiować pojęcia i to jest jego główna zawartość
- Definiować dodatkowe, nie ujęte w specyfikacji XBRL role dla relacji z baz powiązań (linkbases)
- Importować inne schematy. Każda taksonomia tworzona na niskim poziomie (np. konkretnej firmy) importuje taksonomię opartą o przyjęte uregulowania prawne, np. taksonomię IFRS
- Definiować odwołania do baz powiązań (linkbases)

Poniższy przykład pokazuje początek prostego schematu taksonomii, w którym zaimportowana została taksonomia IFRS:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="file://D:/DokumentyBW/PLOUG/PLOUG2009"
elementFormDefault="qualified"
xmlns:n1="file://D:/DokumentyBW/PLOUG/PLOUG2009" xmlns:n2="http://www.tns.com"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xl="http://www.xbrl.org/2003/XLink"
xmlns:xbrli="http://www.xbrl.org/2003/instance"
xmlns:link="http://www.xbrl.org/2003/linkbase"
xmlns:ref="http://www.xbrl.org/2004/ref"
xmlns:info="http://xbrl.iasb.org/info"
xmlns:ifrs="http://xbrl.iasb.org/taxonomy/2008-03-01/ifrs"
xmlns:xbrldt="http://xbrl.org/2005/xbrldt">
  <xs:import namespace="http://xbrl.iasb.org/taxonomy/2008-03-01/ifrs"
schemaLocation="http://xbrl.iasb.org/taxonomy/2008-03-01/ifrs-cor_2008-03-
01.xsd"/>
  <xs:annotation>
    <xs:appinfo>
      </xs:appinfo>
    </xs:annotation>
  <xs:import namespace="http://www.xbrl.org/2003/instance"
schemaLocation="http://www.xbrl.org/2003/xbrl-instance-2003-12-31.xsd"/>
  <xs:import namespace="http://www.xbrl.org/2004/ref"
schemaLocation="http://www.xbrl.org/2004/ref-2004-08-10.xsd"/>
  <xs:import namespace="http://xbrl.org/2005/xbrldt"
schemaLocation="http://www.xbrl.org/2005/xbrldt-2005.xsd"/>
```

### 3.2.2. Definicja pojęcia (concept)

Specyfikacja XBRL przewiduje 2 rodzaje pojęć: proste (item) oraz złożone (tuple). Pojęcie proste jest elementem XML bez elementów potomnych, natomiast pojęcie złożone je posiada. Formalnie rozróżnienie następuje poprzez zastosowanie w schemacie grup substytucji (substitutionGroup). Z założenia pojęcia złożone miały służyć przedstawianiu informacji, które są zrozumiałe tylko jako grupa. Klasycznym przykładem jest na przykład adres. Warto zauważyć, że wiele z pojęć definiowanych w starszych taksonomiach jako tuple, w nowszych, z powodzeniem i zyskiem dla klarowności można definiować poprzez mechanizm wielu wymiarów, wprowadzony w rekomendacji Dimensions 1.0 w 2006 roku.

Oprócz substitutionGroup każde pojęcie musi posiadać dodatkowe atrybuty:

- Nazwę (name) – identyfikującą pojęcie w ramach schematu. Dodatkowo pojęcia mogą posiadać identyfikatory (id), co jest zalecane, ponieważ upraszcza znacznie składnię wyrażen XPointer
- Typ (type) – określa typ wartości, którą będzie przyjmowało dane pojęcie. Ponieważ zajmujemy się raportami finansowymi, najczęściej spotykanym typem jest „monetaryItemType”. Wszystkie dostępne są typy zawarte są w specyfikacji XBRL.
- periodType – jest wymagany dla pojęć prostych (items) i może przyjmować wartości „instant” i „duration” i dzieli w naturalny sposób pojęcia na te, które mają wartości w danym momencie (na przykład stan zapasów magazynowych) lub w pewnym okresie (jak choćby koszty)
- balance – może być dodatkowym atrybutem stosowanym dla pojęć wywodzących się z księgowości, typu „monetaryItemType” i określa stronę księgowania, przyjmując wartości „debit” lub „credit”.
- Etykiety – label. Etykiety, czyli znaczące dla odbiorcy raportów nazwy pojęć są wykorzystywane do wizualizacji raportów. Znajdują się w osobnej bazie powiązań Labels, co umożliwia tworzenie językowych lub wielojęzycznych wersji taksonomii, bez ingerowania w ich podstawową zawartość.

A oto 2 przykłady definicji pojęć z taksonomii IFRS-GP, pierwsza z nich dotyczy item, druga zaś tuple.

#### Przykład 1. Definicja elementu „item”

```
<element id="ifrs-gp_RevenueTotalByNature"
name="RevenueTotalByNature"
type="xbrli:monetaryItemType"
substitutionGroup="xbrli:item"
xbrli:periodType="duration"
xbrli:balance="credit"
nillable="true" />
```

#### Przykład 2. Definicja elementu „tuple” (część elementów wewnątrz pominięto)

```
<element id="ifrs-gp_SignificantSubsidiary"
name="SignificantSubsidiary"
substitutionGroup="xbrli:tuple"
nillable="true">
<complexType>
<complexContent>
<restriction base="anyType">
<sequence minOccurs="0" maxOccurs="1">
<element ref="ifrs-gp:NameOfSignificantSubsidiary"
minOccurs="1" maxOccurs="1" />
<element
ref="ifrs-gp:CountryOfIncorporationOfSignificantSubsidiary"
minOccurs="1" maxOccurs="1" />
<element
ref=
"ifrs-gp:PercentageOfOwnershipInterestInSignificantSubsidiary"
minOccurs="1" maxOccurs="1" />
```

```

        </sequence>
        <attribute name="id" type="ID" use="optional" />
    </restriction>
</complexContent>
</complexType>
</element>

```

Należy zauważyć, że wprowadzanie pojęcia „tuple” nie może wynikać z postaci, w jakiej raport ma być prezentowany, bo do tego służą powiązania zawarte w bazie powiązań Presentation.

### 3.3. Instancja

Instancja jest tym dokumentem, który zawiera konkretne informacje raportu, czyli fakty. Jej najważniejszym zadaniem jest dostarczać fakty oznaczone znacznikami (tags) z taksonomii. Ale by było to możliwe i rozpoznawalne także dla procesorów XML, oprócz faktów w instancji muszą się też znajdować:

- Odwołania do przestrzeni nazw
- Odwołania do wykorzystywanych taksonomii, w tym baz powiązań (linkbases)
- Przypisy końcowe (footnotes). Przypisy są dodatkowym tekstem, który

Elementem nadrzędnym każdej instancji jest `<xbrl>`. W nim, jak w innych dokumentach XML umieszcza się odwołania do przestrzeni nazw i schematów taksonomii. Pokazuje to poniższy przykład:

Przykład 3. Element root w instancji XBRL

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xbrli:xbrl
xmlns:xbrli="http://www.xbrl.org/2003/instance"
xmlns:dei="http://xbrl.us/dei/2008-03-31"
xmlns:link="http://www.xbrl.org/2003/linkbase"
xmlns:us-gaap="http://xbrl.us/us-gaap/2008-03-31"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xbrl.org/2006/xbrldi
http://www.xbrl.org/2006/xbrldi-2006.xsd">
.....
.....
<context id="c1"> </context>
<unit id="u1"> </unit>
</xbrli:xbrl>

```

Wskazanie na taksonomię danej instancji następuje w elemencie `link:schemaRef`, który definiuje, poprzez atrybut `XLink xlink:href` prosty (czyli typu `simple` – `xlink:type="simple"`) odnośnik (link) do pliku schematu xsd taksonomii, tak jak w poniższym przykładzie.

Przykład 4. Odwołanie do taksonomii w instancji raportu

```

<link:schemaRef xlink:href="my_company-20090112.xsd" xlink:type="simple"/>

```

Każda instancja musi odwoływać się do co najmniej jednej taksonomii.

Główną część instancji stanowią fakty biznesowe. Aby jednak dany fakt był jednoznaczny muszą mu też towarzyszyć dodatkowe informacje:

- Kontekst. Kontekstem jest czas, którego dotyczy raport, informacje o raportującym podmiocie (context entity) lub opcjonalnie scenariusz (scenorio). Dla różnych typów raportów cza-

sem może być konkretny dzień lub przedział czasu. Z kolei scenariusz mówi o tym z jakiego typu faktami mamy do czynienia (actual, budget). Kontekst jest dla faktu identyfikowany poprzez atrybut contextRef, który odwołuje się do elementu „context” podrzędnego do „xbrl”..

- Informacje o używanych jednostkach miar. Każda wartość liczbowa musi być wyrażona w jakiejś jednostce. Informacja o jednostce jest obecna w elemencie tworzącym fakt w postaci atrybutu unitRef odwołującego się do identyfikatora jednostki. Wszystkie użyte jednostki muszą być określone zaraz pod elementem głównym „xbrl”.
- Dokładność. Wszystkie fakty o wartościach liczbowych muszą mieć także określoną dokładność. Dokładność ta może nie być jednak podana wprost (atrybuty „precision” lub „decimals”), bowiem w specyfikacji XBRL istnieją reguły wyznaczania domyślnej dokładności.

#### Przykład 5. Pojedynczy fakt w instancji raportu XBRL

```
<my_company-2008-balance:TotalCurrentAssets
  contextRef="c1"
  unitRef="u1"
  decimals="0">1808578
</my_company-2008-balance:TotalCurrentAssets >
```

Fakty same w sobie, mimo że są zawarte w dokumencie XML, nie mają struktury hierarchicznej. Tak było jeszcze w pierwszych wersjach XBRL, gdzie instancja przypominała klasyczną hierarchię. Obecnie, od wersji 2 XBRL fakty stanowią płaską strukturę, a ich układ na raporcie jest zdeterminowany przez bazę powiązań Presentation. Różnicę pokazuje, w dużym uproszczeniu, poniższy przykład:

#### Przykład 6. Płaska struktura instancji XBRL

```
<balanceSheet>
  <assets>
    <currentAssets>6620</currentAssets>
    .....
    <longTermAssets>14500</longTermAssets>
  </assets>
  <liabilities>
    .....
    <totalLiabilities>21120</totalLiabilities>
  </liabilities>
</balanceSheet>
```

```
<xbrl>
  <schemaRef xlink:href="my_taxonomy.xsd"/>
  <currentAssets>6620</currentAssets>
  <longTermAssets>14500</longTermAssets>
  <totalLiabilities>21120</totalLiabilities>
</xbrl>
```

## 4. Jak to działa?

Z całego dotychczasowego opisu czym jest XBRL wynika, że tworzenie raportów to kilka kolejnych kroków.

### 1. Najpierw trzeba mieć taksonomię

Jeśli szczęśliwie istnieje taksonomia zgodna z wymaganiami, które nas obowiązują, możemy ją pobrać od danego regulatora.

### 2. Mapowanie. Dla każdego pojęcia z raportu, który ma być generowany należy znaleźć jego odpowiednik w taksonomii. Nie jest to proste, o ile weźmie się pod uwagę, że taksonomie krajowe zawierają wiele tysięcy pojęć i relacji pomiędzy nimi (IFRS-GP – 4000 pojęć i 15000 powiązań). Dostępne są bezpłatne przeglądarki taksonomii (adresy niektórych na stronie IASB:

<http://www.iasb.org/XBRL/IFRS+Taxonomy/IFRS+Taxonomy+2009/External+taxonomy+viewers.htm>).

2	A	B	C	D	E	F	J
	URI	SI	PBRNTOKEN		TYPE		SORT1
5425	x	x11	IFRSxGP_PDaDR_UnusedProvisionReversedOnerousContractsProvision		it:xbri:monetaryItem		dd
5426							
5427	x	x11	IFRSxGP_PDaDR_UnusedProvisionReversedOtherProvisions		it:xbri:monetaryItem		dd
5428							
5429	x	x11	IFRSxGP_PDaDR_UnusedProvisionReversedProvisionsTotal		it:xbri:monetaryItem		dd
5430							
5431	x	x11	IFRSxGP_PDaDR_UnusedProvisionReversedRestructuringProvision		it:xbri:monetaryItem		dd
5432							
5433	x	x11	IFRSxGP_PDaDR_UnusedProvisionReversedWarrantyProvision		it:xbri:monetaryItem		dd
5434							
5435	x	x11	IFRSxGP_PDaDR_UnusedProvisionsReversed		it:xbri:monetaryItem		dd
5436							
5437	x	x11	IFRSxGP_PDaDR_WageAndSalaries		it:xbri:monetaryItem		dd
5438							
5439	x	x15	InvestmentInAssociate IFRSxGP_ATaOT_PercentageOfOwnershipInterestInAssociate		it:ifs-gp:typ:Percent		i
5440							
5441	x	x16	InvestmentInAssociate IFRSxGP_ATaOT_PercentageOfVotingPowerInAssociateIfDifferentFromPercentage		it:ifs-gp:typ:Percent		i
5442							
5443	x	x16	InitialDisclosureOfDiscontinuedOperation IFRSxGP_ATaDA_DateOfExpectedCompletionOfDiscontinuance		it:xbri:dateItemType		i
5444							
5445	x	x13	PercentageOrAmountOfMajorCategoryOfIP IFRSxGP_ATaND_AssetsCategoryPercentage		it:xbri:decimalItemT		i
5446							
5447	x	x11	IFRSxGP_ATaND_EffectiveInterestRateForLiabilityComponentOfEmbeddedDerivativ		it:xbri:decimalItemT		i
5448							
5449	x	x13	PrincipalCurrency IFRSxGP_ATaND_ExchangeRateAtPeriodEnd		it:xbri:decimalItemT		i
5450							
5451	x	x21	AmountForCurrentAnnualPeriodAndPrevious IFRSxGP_ATaND_ExperienceAdjustmentArisingOnPlanAssetsExpressedAsPerce		it:xbri:decimalItemT		i
5452							
5453	x	x18	AmountForCurrentAnnualPeriodAndPrevious IFRSxGP_ATaND_ExperienceAdjustmentArisingOnPlanLiabilitiesExpressedAsPer		it:xbri:decimalItemT		i

Rys 3. Przeglądanie taksonomii może być kłopotliwe

### XBRL and Tax Authority Adoption[1].ppt

Tutaj niestety może się okazać, że nasze dane nie w pełni odpowiadają pojęciom w taksonomii i konieczne będą wstępne transformacje danych.

Jeszcze inna sytuacja może zajść, gdy zidentyfikujemy na raporcie wielkość, dla której pojęcia w taksonomii brakuje. Może to oznaczać konieczność rozszerzenia taksonomii, czyli stworzenie własnej. Wtedy konieczny jest krok następny, niestety kosztowny i wymagający większego doświadczenia.

### 3. Modelowanie rozszerzającej, własnej taksonomii

### 4. Przypisywanie znaczników – tagowanie

Jest to najistotniejsza część procesu, czyli przekładanie wewnętrznych źródeł danych na poszczególne, rozpoznane wcześniej znaczniki z taksonomii. O ile w kroku „Mapowanie” tworzone były powiązania wymagań raportowych z pojęciami taksonomii, to tutaj przypisujemy te znaczniki do wewnętrznych źródeł danych, najczęściej pochodzących z systemów informatycznych. Przypię-

sywanie znaczników można przeprowadzić ręcznie (wykorzystując arkusz kalkulacyjny), wykonać specjalistyczne oprogramowanie, które stworzy i zapisze przypisania wiążącą znaczniki taksonomii z naszymi danymi, uwzględniając źródła tych danych lub nawet bezpośrednio w systemach informatycznych, które mają dane generować.

#### 5. Generowanie raportu – tworzenie instancji

Mając zapisane mapowanie, instancję można wygenerować automatycznie.

W powyższym procesie istnieje kilka miejsc, w których dobrze jest zastosować oprogramowanie przetwarzające XBRL. Dostępność takich narzędzi na rynku jest, jak już było wspomniane, coraz większe. Należy także podkreślić, że coraz więcej aplikacji klasy enterprise (tzw. ERP), które przetwarzają dane finansowe, zaczyna także wspierać XBRL (w tym Oracle OeBS w wersji 12). Oznacza to, że jeśli źródłem danych jest taka właśnie aplikacja, to do stworzenia wymaganych raportów potrzebne są tylko, jak zwykle, wysiłek, czas i zaangażowanie.

## 5. Narzędzia XBRL, dedykowane i wspierające

Odkąd SEC (Security and Exchange Commission) ogłosił obligatoryjność wykorzystania XBRL w sprawozdaniach do niego kierowanych, lista dostępnych narzędzi wspierających XBRL szybko rośnie. Wielu dużych dostawców postanowiło pójść najprostszą, ale jedyną dostępną, bo jedyną dostatecznie szybką drogą i po prostu zintegrowała w swoich pakietach oprogramowanie mniejszych firm, które od lat zajmują się XBRL. Tak zrobił w lutym SAP, włączając do swoich SAP Business Objects moduł XBRL Publishing, oparty na aplikacjach firmy UBmatrix. IBM z kolei podpisał umowę z firmą Symansys na dołączenie modułów XBRL do oprogramowania Cognos. Nawet Oracle, choć możliwość tworzenia raportów w formacie XBRL istniała w pakiecie EPM Hyperion-a już od 2003 roku i sam Hyperion był w czasach przed zakupem przez Oracle członkiem XBRL International, postanowił zintegrować narzędzia UBmatrix, tak by jeszcze zwiększyć możliwości swoich produktów.

Lista narzędzi XBRL dostępna jest na <http://xbrl.us/vendors/Pages/default-expand.aspx> i liczy obecnie 55 pozycji. Warto wspomnieć o kilku z nich.

### 5.1. UBmatrix Enterprise Application Suite

Firma UBmatrix była już kilka razy wcześniej wspomniana i nie bez powodu. Jej liderem technicznym jest Charles Hoffman, człowiek który wymyślił XBRL, a UBmatrix powstał rozwinął się m.in. dzięki wykupieniu firmy Hoffmana XBRS Solutions.

Enterprise Application Suite jest całą platformą raportową, stanowiącą rozwiązanie serwerowe dla dużych firm. Jej składowe to Taxonomy Manager oraz Reporting Manager. Dodatkowo platforma integruje się z narzędziami na stacjach roboczych Taxonomy Designer-em i Report Builder-em. Jądrzem i najważniejszym elementem całości jest wydajny procesor XBRL.

### 5.2. Rivet Software Dragon Tag

Firma Rivet także od dawna specjalizuje się w oprogramowaniu XBRL, a jej produkty mają taką zaletę, że integrują się doskonale z Microsoft Office, a zwłaszcza z Excelem. Rivet Software informuje w swoich materiałach promocyjnych, że to ich narzędzia były wybierane przez 6 na 9 firm, które uczestniczyły w programie dobrowolnej sprawozdawczości do SEC, który był ogłoszony w 2005 roku jako testowy „pilot” przed wprowadzeniem XBRL obligatoryjnie.

Ostatnio Rivet Software ogłosił, że udostępnia bezpłatnie przeglądarkę XBRL zintegrowaną z Excelem.

### 5.3. Altova MissionKit 2009

XML Spy jest jednym z bardziej znanych i lubianych przez programistów narzędzi do pracy z XML. Niedawno została wydana nowa wersja tego produktu, tym razem w pełni wspierająca XBRL, na wszelkich etapach jego wykorzystania. MissionKit to szereg aplikacji, w tym:

- XML Spy doposażony w możliwości łatwego tworzenia i modyfikowania oraz przeglądania taksonomii
- Mapforce – aplikacja do mapowania i transformowania danych, w tym danych wszelkich istniejących już w różnych formach w firmie danych na XBRL
- Altova StyleVision – aplikacja do projektowania i generowania raportów w różnych formatach, w tym PDF, RTF, HTML i Word 2007 (tzw. OOXML)

Przy okazji wydania pakietu firma Altova udostępniła też serię tutoriali wprowadzających w zagadnienia XBRL, dostępnych dla każdego przez Internet pod adresem: <http://altova-aot.s3.amazonaws.com/Altova%20XBRL%20Course/player.html>

## 6. Wizualizacja raportów XBRL

Instancje XBRL są dokumentami XML, o płaskiej strukturze, w dodatku o skomplikowanych i nie zawsze czytelnych nazwach znaczników. Nie uwzględniają też żadnych nazw w języku lokalnym. Z drugiej strony, raporty finansowe są przeznaczone nie tylko do przetwarzania przez oprogramowanie, ale powinny być także dostępne bezpośrednio dla zainteresowanych ludzi. Stąd potrzeba ich wizualizacji. Istnieje kilka podejść do wizualizacji raportów XBRL.

### 6.1. Inline XBRL

Jest to inicjatywa XBRL International, polegająca na opracowaniu standardu pozwalającego umieszczać całą zawartość instancji XBRL wewnątrz znaczników HTML, w tak sposób, aby mogły być też ona automatycznie przetwarzana przez aplikacje tak, jak normalny XML. Specyfikacja Inline XBRL ma status Public Working Draft (wersja 0.54) i jest dostępna na witrynie [www.xbrl.org](http://www.xbrl.org). Istnieją już pewne implementacje aplikacji, które przetwarzają strony HTML zgodnie z tą specyfikacją, jak na przykład procesor firmy Semansys.

### 6.2. Wykorzystanie własnych arkuszy XSLT

Jest to podejście najbardziej naturalne, bowiem właśnie XSLT najczęściej służy do transformacji XML w HTML. To podejście ma wiele zalet, jak choćby:

- Dojrzałość technologii
- Dobra znajomość wśród osób zajmujących się XML, w tym rozpoznanie problemów wydajnościowych
- Dostępność licznych bibliotek

Istnieją też niestety wady, w tym 2 główne:

- Mała znajomość XBRL wśród specjalistów XSLT
- Pliki instancji XBRL są duże, a do ich wizualizacji potrzebne są też wszystkie elementy taksonomii. Oznacza to, że przetwarzanie jest dość długie i nie wydaje się, aby mogło być robione on-line

Mimo problemów, próby tworzenia arkuszy XSLT do wizualizacji raportów XBRL są robione i prowadzą do niezłych rezultatów.

### 6.3. Dedykowane narzędzia dla XBRL

Jednym z elementów większości pakietów dedykowanych do obsługi XBRL jest kreator raportów. Wykorzystanie go jest rozwiązaniem najprostszym, ponieważ w chwili przejścia na sprawozdawczość XBRL firma i tak najczęściej jest zmuszona zakupić wspomagające ją w tym oprogramowanie XBRL. Dodatkową zaletą może być fakt, że niektóre z tych kreatorów potrafią łączyć z instancją XBRL także dane z innych źródeł, w tym z baz danych. Należy przy tym zauważyć, że często wykorzystanie takiego narzędzia oznacza tak jak i w poprzednio omawianej metodzie wizualizacji zastosowanie arkuszy XSLT, tyle tylko, że tworzonych w sposób wizualny. Na XSLT oparty jest choćby Altova Stylevision 2009.

### 6.4. Narzędzia do raportowania niededykowane dla XBRL

Istnieje grupa produktów do raportowania, która akceptuje, a nawet czasem wymaga, by dane do raportów były w postaci XML. Takim produktem jest choćby Oracle BI Publisher, którego miałam okazję prezentować przed paru laty w tym właśnie miejscu. Czy można je wykorzystać do wizualizacji? Oczywiście można, ma to jednak w porównaniu choćby ze wspomnianym Altova Stylevision 2009 taką wadę, że tworząc szablony w BI Publisher-e będziemy musieli przeglądać znaczniki instancji (lub taksonomii, szablony w Oracle BI Publisher-e, można tworzyć na podstawie schematu xsd) bez wsparcia, jakie daje przeglądarka taksonomii.

Zaletą dla firm wykorzystujących tego typu narzędzia jako platformę raportową, jest wykorzystanie, również dla potrzeb XBRL tej platformy, bez konieczności zmiany podejścia dla jednej tylko kategorii raportów.

## 7. Konkluzje

XBRL, jak już pewnie wiele osób zdołało się zorientować choćby z tego pobieżnego omówienia nie jest rozwiązaniem idealnym. Taksonomie są tworem skomplikowanym, a raportom muszą towarzyszyć taksonomie, bez nich nie można instancji XBRL przetwarzać i analizować. W procesie tworzenia taksonomii istnieją nierozwiązane problemy, choćby ten związany z wersjonowaniem. Poszczególne taksonomie i na koniec instancja tworzą łańcuch zależności, który, w przypadku modyfikacji jednego z ogniw, wymaga modyfikacji innych. Dodatkowo trudno jest automatycznie znaleźć i interpretować różnice pomiędzy wersjami, co stwarza dodatkowe problemy. Stąd wersjonowaniem jest jednym z ważniejszych zagadnień, którymi zajmuje się obecnie XBRL International.

Jak można było przeczytać wcześniej w części opisującej historię, XBRL istnieje już dość długo. Jego aktualna specyfikacja powstała w 2003 roku i od tego czasu, czyli aż przez 6 lat, pomijając erraty i nowe rekomendacje rozszerzające jest stabilna. Przez ten czas, mimo wsparcia niektórych instytucji udawało się jakoś wielu osobom, nawet tym, które zajmują się zawodowo XML nie słyszeć o tym standardzie. Teraz jednak wygląda na to, że, być może pod wpływem tego, co obserwujemy od ponad roku na rynkach finansowych i w gospodarkach wielu krajów, historia nabrała przyspieszenia. Co i raz można przeczytać doniesienia o nowych wersjach produktów wspierających XBRL dostarczanych przez najważniejszych graczy na rynku Business Intelligence. Można więc przypuszczać, że mimo słabości XBRL, wspomnianych wyżej, nie ma od niego odwrotu, można go tylko doskonalić.

A osoby zajmujące się zawodowo systemami raportującymi powinny powoli zacząć go poznawać.

---

**Bibliografia**

- [XBRLORG] XBRL International: [www.xbrl.org](http://www.xbrl.org)
- [XBRLRE] XBRL 2.1 Recommendation 2003-12-31 with Corrected Errata – 2008-07-02, 2008
- [XLREC] XML Linking Language (XLink) Version 1.0, W3C Recommendation, 2001
- [XPREC] XPointer Framework W3C Recommendation, 2003
- [VRSVR] Ventana Research, Software Vendors Ramp up XBRL, 2009