

MATEMATYKA W SIECI

Minęło kilka lat wykorzystywania sieci WWW. Przyjęta do tworzenia witryn normalizacja języka HTML wydaje się być przyjętym językiem Internetu. Jedni są zadowoleni; powiadają, że dobrze, iż Internet się rozwija. Inni są nieco bardziej sceptyczni. HTML nie czyni z Internetu narzędzia w pełni interaktywnego, sieć globalna pozostaje właściwie jednym wielkim urządzeniem do faksowania. A gdzie miejsce na elektroniczny biznes, gdzie bezpieczeństwo transakcji, gdzie autoryzacja użytkowników. Mimo coraz szerszego kręgu użytkowników jest oczywiste, że przyjęta do tworzenia witryn normalizacja jest niewystarczająca. Tekst, obrazki i tabele to stanowczo za mało.

Komunikowanie się jest podstawą wszelkiego działania. Przekazywanie komunikatów wymaga obecności nadawcy i odbiorcy oraz kanału, którym komunikaty będą przekazywane. Aby jednak odebrany przez odbiorcę sygnał był komunikatem, musi istnieć zrozumiały dla nadawcy i odbiorcy język, w którym to języku komunikaty są wyrażane.

Podstawą komunikowania się jest język. Język zaś składa się z słów. I jest tak dla każdego rodzaju języka. Posługujemy się codziennie różnymi językami. Mamy język gestów rąk, język mimiki twarzy, język postawy. Mamy język gestów celowych; wiemy, jak zatrzymać samochód, jak okazać swoją niechęć, jak gestem obrazić.

Mamy w końcu język mówiony, którym staramy posługiwać się poprawnie. Język, którym się posługujemy, jest żywym i zmieniającym się tworem. Zmieniają się słowa i ich znaczenie. Zmieniają się słowa i ich rozumienie. Dodajmy do tego, że zmienia się także nauka o języku i będziemy mieli obraz może nie pełny, ale zamknięty w ramy. Wszystkim zainteresowanym językami naturalnymi można polecić niezwykle ciekawą książkę Devlina [2]

Szczególnym językiem jest język matematyki. Prezentacja wzorów matematycznych na stronach WWW nie jest jeszcze znormalizowana. I nic dziwnego. Z dotychczasowych doświadczeń typograficznych wynika bowiem, że logika wzorów matematycznych jest bardzo skomplikowana i tylko nauczycielom matematyki (i też chyba tylko niektórym) udaje się wmawiać uczniom, że zapis matematyczny jest prosty. Spróbujmy opisać niektóre cechy wzorów matematycznych.

Po pierwsze, symbole matematyczne są wieloznaczne. Oznacza to, że te same symbole mogą oznaczać różne logicznie rzeczy. I tu często jedynym rozróżnieniem rzeczy różnych logicznie jest typografia. Na przykład, symbol ‘litera d z następującą za nią pomniejszoną literą i’; owa pomniejszona litera ‘i’ może oznaczać albo kolejny indeks dolny i wtedy piszemy obie litery kursywą d_i , albo też ‘i’ może być częścią nazwy zmiennej i wtedy piszemy mniejszą literę czcionką zwykłą d_i . Typografia łąta więc dziury zapisu ręcznego.

Z drugiej strony, często jest tak, że różne symbole znaczą dokładnie to samo, np. pochodna funkcji może być zapisana jako $f'(x)$ lub też df/dx .

Po trzecie, sama notacja matematyczna ma podwójną naturę graficzno-logiczną. Z jednej bowiem strony jest zupełnie obojętne czy napiszemy e^3 , czy też e^3 , ponieważ obie notacje mogą być dobrze zrozumiałe. Jeśli jednak nie piszemy na papierze, a drukujemy, to chcielibyśmy zobaczyć wyłącznie drugą formę. Typografia nic, rzecz jasna, w logice matematyki nie zmienia, ale pozwala ową logikę uwypuklić i prościej zaprezentować.

Można powiedzieć, że wszelkie prace nad przekazywaniem wzorów matematycznych poprzez sieć są nieporozumieniem. Istnieje przecież TeX, który prosto i łatwo koduje nawet

najbardziej skomplikowane wzory. Prezentacja TeX-a nie jest celem tej pracy. Wszyscy wykorzystujący matematykę znają doskonale zalety tego metajęzyka. TeX jest jednak, o czym czasem zapominamy, wyłącznie językiem przeznaczonym do składu dokumentów, podczas gdy HTML jest językiem opisu stron.

Od kilkunastu miesięcy konsorcjum W3C prowadzi intensywne prace nad projektem języka MathML. Jest to język znacznikowy przeznaczony do tekstowego kodowania wzorów matematycznych. Dodajmy, że jest to projekt bardzo ambitny, ponieważ zdecydowano się na stworzenie funkcjonalnego odpowiednika tradycyjnej notacji matematycznej.

Programowa oprawa tego przedsięwzięcia polega na stworzeniu dwóch programów: edytora-przeglądarki *AMAYA* i działającego po stronie serwera programu *JADE*. Ten ostatni ma wiele cennych właściwości: może generować dokumenty w wielu formatach na podstawie zadanego dokumentu SGML-owego.

JĘZYK SGML

Wielu autorów nazywa słusznie SGML (*Standard Generalized Markup Language*) metajęzykiem. I jest w tym sporo racji. SGML można określić jako język opisu wszelakich dokumentów, język bardzo skomplikowany i obszerny słownikowo.

Historia tego języka wydaje się bardzo prosta. Od wielu pokoleń drukarze załączali notatki do manuskryptów przeznaczonych dla zecerów. W notatkach tych opisywali sposób adjustacji tekstu. Adjustacja rozwijała się niezależnie w wielu krajach aż do roku 1986, kiedy to Międzynarodowa Organizacja Normatywna ISO zatwierdziła system tworzenia nowych języków adjustacyjnych.

Ten metajęzyk był i jest przeznaczony do opisywania języków. W rzeczy samej HTML, język stron WWW, został zdefiniowany za pomocą SGML. Także opisywany dalej XML. Używanie bezpośrednio języka SGML jest skomplikowane, ale daje bardzo dobre rezultaty w dużych publikacjach. Słownik języka SGML jest niezwykle obszerny i bogaty w wiele wyrafinowanych funkcji.

Praca z SGML przypomina pracę z znanym językiem składu dokumentów, jakim jest TeX. W przypadku TeX-a mamy język, w którym są zdefiniowane polecenia, mamy plik opisujący szczegółowo cechy składanego dokumentu oraz kompilator, który sprawdza poprawność generowanego dokumentu. I niby wszystko jest jasne i proste. Ale... nie wspomnieliśmy jeszcze o obecnym w systemie TeX prostym mechanizmie tworzenia nowych poleceń i zmiany sposobu działania starych komend.

Podobnie jest w języku SGML. Mamy język poleceń, dany jako słownik znaczników. Mamy zdefiniowane wcześniej pliki opisujące elementy struktury dokumentu (DTD) oraz możliwość tworzenia własnych DTD [1].

Do utworzenia dokumentu w kodzie SGML potrzebny nam jest: edytor strukturalny i parser. Edytor strukturalny wspomaga oznaczanie struktury tekstu lub ogólniej mówiąc – danych. Oznaczanie struktury odbywa się znacznikami zdefiniowanymi w DTD. Parser SGML-owy sprawdza, czy utworzony dokument jest poprawnym dokumentem. Parser może być wbudowany w edytor, lecz najczęściej jest to oddzielny program.

Jednym z największych problemów dokumentów sieci WWW jest prezentacja wzorów matematycznych. Do tej pory radzono sobie tak, że w miejsce wzorów matematycznych wstawiano w dokumencie HTML-owym obrazki wzorów. Nie trzeba dodawać, że obrazek wzoru nie nadaje się do dalszego przetwarzania przez programy matematyczne. Ba, nie da się z pomocą obrazka wzoru wykonać nawet prostych rachunków.

Konsorcjum W3C [1] opracowało specjalny język MathML, który powstał w oparciu o standard XML. Założenia twórców języka MathML były ambitne, bo chciano opracować funkcjonalny odpowiednik notacji matematycznej. I tu pojawiła się pierwsza przeszkoda, jako że zapis matematyczny ma podwójną strukturę: graficzną i logiczną. Rozwiązano to tak, że ten sam wzór można w języku MathML zapisać na dwa sposoby: albo wizualny (odpowiadający graficznej strukturze wzoru), albo znaczeniowy (odpowiadający logicznej strukturze wyrażenia matematycznego).

Choć MathML nie jest najlepiej udanym tworem, to jednak zapisane w nim wzory dają się łatwo konwertować do innych znanych formatów zapisu wzorów matematycznych. Jako przykład działania MathML przytoczmy wzór [3], którego graficzna postać jest

$$\sqrt{\Delta} = \sqrt{b^2 - 4ac}$$

Gdyby wzór ten zapisać w ogólnie przyjętym w matematyce kodzie TeX-a, to wzór ten przyjąłby następującą zwartą postać

Opis tego wyrażenia jest dość oczywisty. W miejsce znaczników ograniczających działanie do wyodrębnionego fragmentu danych, w systemie TeX mamy nawiasy klamrowe, które równie dobrze spełniają funkcję ograniczającą.

MathML-owa notacja wizualna ma postać:

MathML-owa notacja znaczeniowa ma postać:

JĘZYK XML

Istnieje jeden powszechnie akceptowany język świata WWW. Językiem tym jest HTML (*Hypertext Markup Language*), język znaczników hipertekstowych. Każda strona WWW jest dokumentem napisanym w języku HTML. Wszystkie współczesne przeglądarki rozumieją ten język.

HTML nie opisuje treści strony WWW. Jest to język wyłącznie opisu strony. W języku tym wyrażamy sposób, w jaki przeglądarka ma wyświetlić treść strony na ekranie. Należy tu podkreślić różnice, jakie dzielą HTML od języków służących do składu publikacji (np. TeX). Otóż języki służące do składu tekstu dają pełną kontrolę nad wyglądem strony po wydrukowaniu. HTML takiej kontroli nad wyglądem strony WWW na ekranie nie daje.

Zanim przejdziemy do krytyki języka HTML, przypomnijmy o wielkiej roli, jaką język ten odegrał przy tworzeniu współczesnych witryn WWW i wielkiej roli, jaką język ten będzie jeszcze odgrywał. Przecież świat WWW bez HTML po prostu nigdy by nie zaistniał.

W początkowej fazie rozwoju świata WWW istotne było udostępnienie samego dokumentu tekstowego w pełnej krasie czcionek rozmaitej wielkości i różnego kroju, wzbogaconego rysunkami i obrazkami. Szybko też opracowano mechanizm zestawień tabelarycznych. Taki mechanizm służy nie tylko do tworzenia tabel lecz, a może przede wszystkim, do urozmaicania strony łamami wielokolumnowymi.

Język HTML był sprawnie rozwijany, a jego ostatnia zatwierdzona wersja normatywna ma numer 4. (Skoro mowa o zatwierdzaniu przypomnijmy, że nad stroną standardów w świecie WWW panuje nieformalna organizacja zrzeszająca reprezentantów zainteresowanych firm oraz wielu wolontariuszy. W3C – bo o tym mowa – jest dostępny pod adresem [1]). Niedostatki HTML były łatanie na wiele sposobów, że wspomnimy o korzystaniu z języka Java, o kontrolkach ActiveX, o dynamicznym HTML itd.

Niemniej świat poszedł do przodu. Wymagania użytkowników są kształtowane już nie przez nastoletnich zapaleńców, a przez nowe zdalne usługi, w szczególności przez e-biznes. Skrót e-biznes powstał jako paralela do wyrażenia e-mail, które oznacza pocztę elektroniczną. E-biznes – to biznes elektroniczny, realizowany przez sieć informacyjną.

Od współczesnych witryn WWW ludzie żądają znacznie więcej niż do tej pory. Dotychczasowa witryna WWW to przecież wyrafinowany fax, przekazujący dokumenty wszystkim, które o takie dokumenty proszą. Firmy chcą jednak, aby ich witryny przyjmowały zamówienia od klientów, przesyłały dane bezpiecznie i pewnie, aby sterowały fabrykami i urządzeniami z drugiego końca świata. HTML nie może spełnić tych oczekiwań, ponieważ nie był projektowany do tego celu.

występujących w pliku znaczników. Jeśli zapomnimy, co jakiś znacznik symbolizuje, to możemy znaleźć interpretację w jednym z wielu podręczników.

Z tekstowym plikiem zawierającym kod XML jest znacznie gorzej, ponieważ język znaczników może być różny dla różnych aplikacji i zwykły człowiek może mieć tutaj kłopoty. Jeśli zapomnimy, co dany znacznik symbolizuje, to plik XML pozostanie nieczytelny. Rzecz jasna, trudności tych nie mają programiści, ale trudno ich uważać za zwykłych ludzi.

Po kolejne czwarte, XML jest rodziną technik. Jeśli mówimy o czystym XML, to mamy na myśli XML 1.0 – specyfikację pewnych znaczników i atrybutów. Jednak wokół tej podstawowej wersji XML stworzono wiele dodatkowych modułów. Oto przykłady:

- Xlink (świeżo utworzony w czerwcu 1999 r.) opisuje normatywny sposób włączania odsyłaczy i innych odwołań do dokumentów kodowanych w XML.
- Xpointer & Xfragments (wersja niestabilna, w rozwoju) opisują sposoby wskazywania na różne części i fragmenty danego dokumentu XML. Xpointer przypomina w działaniu URL, ale o ile ten ostatni wskazuje na dokument w sieci WWW, to Xpointer wskazuje na fragment wewnątrz danego dokumentu kodowanego w XML.
- XSL (koniec prac przewidziano na jesień 1999 r.) jest językiem specjalizowanym do transformacji dokumentu XML w inny dokument XML, poprzez rearanżację, dodanie lub skasowanie niektórych znaczników.
- DOM jest zbiorem funkcji do sterowania dokumentem XML, które to funkcje można wywołać z poziomu języka kompilowanego, np. języka C.

Po piąte, XML jest rozwlekły, co jednak nie stanowi problemu. Alternatywą jest zapisywanie w postaci binarnej, o czym mówiliśmy już wcześniej.

Po szóste i ostatnie, XML nie wymaga licencji, jest niezależny od sprzętu i doskonale programowo wspomagany. Ponieważ XML powstał jako niekomercyjny projekt internetowej społeczności, toteż rozwija się szybko. Jego znaczenie dla rozwoju świata WWW przyrównuje się nieraz do roli, jaką w korzystaniu z baz danych odgrywa SQL.

BIBLIOGRAFIA:

- 1) Witryna: <http://www.w3.org>
- 2) Devlin K.: *Żegnaj, Kartezjusz. Rozstanie z logiką w poszukiwaniu nowej kosmologii umysłu*. Prószyński i S-ka, Warszawa 1999.
- 3) Bolek P.: „Kodowanie, przetwarzanie i formatowanie dokumentów SGML-owych”. *Biuletyn Polskiej Grupy Użytkowników Systemu TeX*. Nr 12, 1999 r.