

Zdzisław SROCYŃSKI
Politechnika Śląska, Instytut Matematyki

SERWER WWW JAKO SERWER DOKUMENTÓW ZAWIERAJĄCYCH ZŁOŻONĄ NOTACJĘ MATEMATYCZNĄ

Streszczenie. W pracy przedstawiono propozycję rozszerzenia możliwości serwera WWW o automatyczne przetwarzanie dokumentów zawierających notację w języku MathML do postaci czytelnej w obecnie rozpowszechnionych przeglądarkach internetowych. Proponowana metoda uniezależnia poprawny odbiór dokumentu od obecności w systemie klienta dodatkowych niestandardowych rozszerzeń wymagających uprzedniego pobrania z Internetu i instalacji. Podejście takie daje możliwość wygodnej dystrybucji poprzez Internet dokumentów matematycznych dostępnych dla szerokiego grona odbiorców. Istotnym elementem pracy jest porównanie różnych metod rozszerzania możliwości serwerów WWW oraz opis praktycznej implementacji proponowanego rozwiązania.

SERVING DOCUMENTS CONTAINING COMPLEX MATHEMATICAL NOTATION OVER THE WEB

Summary. The paper describes a proposal of extending Web server capabilities with automatic translation of MathML documents into common HTML understandable for popular Web browsers. This way it should be easy to spread over Internet many math documents readable not only for experienced users, but also for beginners using popular software not extended by non-standard plug-ins. Important parts of the article are a comparison of different ways of extending Web servers capabilities and description of experimental implementation of proposed extension.

1. Wprowadzenie – problemy związane z właściwą prezentacją dokumentów matematycznych na stronach WWW

Prezentacja dokumentów zawierających złożoną notację matematyczną w sieci Internet jest niewątpliwie problemem niezwykle istotnym zarówno z punktu widzenia autorów profesjonalnych publikacji naukowych, jak i odbiorców wszelkiego rodzaju materiałów dydaktycznych. Jak dotąd, przedstawiono propozycje standardu opartego na XML języka MathML, pozwalającego opisywać zarówno wygląd (warstwa prezentacji), jak i znaczenie (warstwa struktury) wzorów matematycznych [1,2]. Pojawiły się również stosunkowo nieliczne rozwiązania mające na celu wprowadzenie tego standardu w życie. Jedno z nich opisano w pracy [3]. Przedstawiono w niej metodę rozszerzenia funkcjonalności przeglądarki internetowej o obsługę języka MathML. Rozwiązanie to jednak uzależnia poprawną prezentację całości dokumentu od wstępnego przetworzenia kodu przez autora oraz od zainstalowania w systemie klienta odpowiedniego modułu rozszerzającego przeglądarkę. Jak wykazały próby, użycie rozwiązań niezależnych od systemu operacyjnego (takich jak Java) nie sprawdza się w tym zastosowaniu ze względu na zbyt małą wydajność, więc kolejną przeszkodą okazuje się konieczność przygotowania modułu rozszerzającego dla każdej platformy systemowej osobno. Ponadto, z chwilą pojawienia się przeglądarek obsługujących w pełni standard MathML, tak specjalnie przygotowane dokumenty wymagać będą kolejnych przeróbek.

Najlepszym rozwiązaniem opisanych powyżej problemów wydaje się odpowiednie przetworzenie dokumentu jeszcze w serwerze. Pozwoliłoby to na wstępne rozpoznanie rodzaju przeglądarki, którą dysponuje odbiorca dokumentu i translację dokumentu na czytelny dla niej format. Szczególnie istotne są trzy aspekty tego podejścia:

- można całkowicie zrezygnować z translacji w przypadku rozpoznania przeglądarki zgodnej ze standardem MathML (np. W3C Amaya),
- można rozpoznać połączenia ze wszelkiego rodzaju serwisami wyszukiwawczymi, co pozwoli przekazać im również nie zmodyfikowany dokument do skatalogowania,
- można rozpoznać przeglądarki wspomagające pracę osób niewidomych i słabowidzących i przetworzyć zapis w języku MathML na odpowiedni kod w notacji Braile'a.

Pozostałe przeglądarki mogą otrzymać w odpowiedzi od serwera wzory MathML przekształcone na pliki graficzne lub uproszczony zapis tekstowy (w przypadku przeglądarek wyłącznie tekstowych).

Zaproponowane powyżej rozwiązanie wymaga odpowiedniego rozszerzenia funkcjonalności serwera WWW. Można to osiągnąć na kilka sposobów, dlatego też warto uprzednio rozważyć ich wady i zalety.

2. Przegląd metod rozszerzania funkcjonalności serwerów WWW

Aplikacje działające jako serwery WWW pozwalają z reguły na rozszerzanie możliwości na kilka sposobów. Z punktu widzenia zadania polegającego na zbudowaniu serwera przetwarzającego duże i skomplikowane dokumenty niebagatelne znaczenie ma przede wszystkim wydajność stosowanego rozwiązania.

Najpowszechniejsze jest umożliwienie wykonania programów w środowisku operacyjnym serwera. Programy takie najczęściej komunikują się z serwerem za pomocą standardu CGI (*Common Gateway Interface*). Większość serwerów udostępnia też pewne API pozwalające na bezpośrednią komunikację z modulem rozszerzającym i przekazywanie do niego informacji o zdarzeniach występujących w trakcie przetwarzania zapytań klienta. Do najbardziej znanych rozwiązań tego rodzaju należy standard ISAPI zaprojektowany przez Microsoft dla serwera Internet Information Services – IIS (zaimplementowany w większości innych serwerów działających na platformie Windows) oraz moduły Apache. Rozszerzenia wykorzystujące API są realizowane jako dynamicznie dołączane biblioteki (so w Linuxie, dll w Windows). Aby zwiększyć niezależność projektowanych rozwiązań od platformy systemowej, stworzono również wiele języków skryptowych, takich jak Perl czy bardzo ostatnio popularne PHP. Ich interpretery mogą być zarówno klasycznymi aplikacjami CGI, jak i rozszerzeniami wykorzystującymi API.

Program CGI jest każdorazowo uruchamiany przez serwer w momencie otrzymania żądania. W przypadku wielu równoległych żądań czas odpowiedzi serwera jako całości zależy więc od wydajności sprzętu, na którym pracuje serwer. W szczególności istotna jest dostępna pamięć operacyjna. Programy CGI mają jednak niepodważalne zalety, do których można zaliczyć łatwość implementacji i przenośność (na poziomie kodu źródłowego).

Dynamicznie dołączana biblioteka jest ładowana do pamięci przy pierwszym użyciu, tak więc zapotrzebowanie na zasoby powinno być w tym przypadku znacznie mniejsze. Mniejszy też powinien być wpływ zwiększania ilości żądań na ewentualne wydłużenie czasu odpowiedzi serwera. Należy w tym miejscu podkreślić największą wadę tego rozwiązania – awaria biblioteki może spowodować całkowite załamanie serwera, co w przypadku CGI jest znacznie mniej prawdopodobne, gdyż każde wywołanie programu CGI korzysta z własnej przestrzeni adresowej. Również implementacja rozszerzenia API może rodzić problemy, ponieważ wymaga ona zapewnienia izolacji wątków oraz dostosowania kodu do konkretnego serwera WWW.

Języki skryptowe pozornie nie mogą równać się z opisanymi powyżej rozwiązaniami pod względem wydajności, ale warto rozważyć wielkość narzutu czasu związanego z załadowaniem i uruchomieniem interpretera. Gdyby się bowiem okazało, że straty te są stosunkowo

niewielkie, korzyści wynikające z łatwej implementacji i przenośności kodu mogłyby je w wielu projektach przewyższyć.

Do porównania wydajności wybrano program CGI, bibliotekę ISAPI oraz skrypt w języku PHP. Program CGI testowano na platformie Windows i serwerach IIS 5 oraz Xitami, a także pod kontrolą systemu Linux na serwerze Apache 1.3.19. Bibliotekę ISAPI testowano na platformie Windows 2000 Server i serwerze IIS 5. Skrypt PHP testowano na platformach Windows (IIS 5, Xitami) i Linux (Apache).

Programom CGI oraz językowi PHP poświęcono wiele miejsca w licznych pozycjach literaturowych, np. [6,7], stąd też poniżej przedstawiono jedynie dokładniejszy opis specyfikacji rozszerzeń ISAPI.

2.1. ISAPI

Jest to opracowane przez firmę Microsoft rozszerzenie serwera WWW (IIS, Personal Web Server) oparte na API serwera. Jego najważniejszą zaletą jest wysoka wydajność. Podczas gdy program CGI jest ładowany, wykonywany i usuwa wykorzystane zasoby osobno dla każdego zapytania a do przekazywania informacji wykorzystywane są pliki, biblioteka API serwera jest jednorazowo ładowana do pamięci, a komunikacja następuje poprzez bufor pamięci.

Podstawowa wada bibliotek ISAPI to ścisła integracja z serwerem: awaria lub błąd biblioteki może wpływać w istotny sposób na pracę serwera (włącznie z załamaniem). Rozszerzenia serwera powinny zapewniać izolację wątków. W celu wymiany lub usunięcia biblioteki dll trzeba wyłączyć serwer lub stosować dodatkowe zabiegi.

ISAPI DLL może występować jako filtr lub rozszerzenie. Filtr przetwarza różne etapy zapytania klienta, może modyfikować i zapamiętywać przekazywane dane i parametry. Rozszerzenie ISAPI wykonuje pewne zadanie na podstawie danych przekazanych od klienta (w większym stopniu przypomina więc tradycyjny program CGI). W przypadku rozszerzenia biblioteka eksportuje co najmniej dwie pierwsze z poniższych funkcji:

- GetExtensionversion
- HttpExtensionProc
- TerminateExtension

Główna funkcja HttpExtensionProc jest wywoływana przez serwer dla każdego żądania wysłanego do DLL. Parametrem funkcji jest struktura danych zawierająca pola:

- lpszpathInfo – informacje o ścieżce
- lpszMethod – metodę żądania
- lpszQueryString – tekst zapytania

oraz adresy czterech funkcji: GetServerVariable do dostępu do zmiennych serwera (tak jak zmienne CGI), WriteClient – do zapisu danych w buforze wyjściowym, ReadClient – pozwalają-

cej na odczyt danych z klienta i `ServerSupportFunction` – umożliwiającej dostęp do funkcji ogólnego przeznaczenia, zależnych od wersji serwera.

Rozszerzenia wykorzystujące API serwerów innych producentów budowane są na podobnych zasadach jak ISAPI.

2.2. Skrypty PHP

Wywołanie skryptu PHP wymaga uprzedniego załadowania interpretera. Może on pracować jako klasyczna aplikacja CGI lub jako rozszerzenie korzystające z API serwera. W każdym wypadku należy się spodziewać pewnego narzutu czasu związanego z tą operacją. Z założenia PHP zaimplementowane jako rozszerzenie API powinno wykazywać większą wydajność dzięki wykorzystaniu omówionych powyżej zalet takich rozwiązań.

2.3. Wyniki eksperymentów

Podana powyżej charakterystyka różnych metod rozszerzania funkcjonalności serwera WWW pozwala uszeregować je pod względem wydajności od najmniej do najbardziej wydajnych w następujący sposób:

- języki skryptowe (PHP),
- aplikacje CGI,
- moduły korzystające z API serwera.

Przeprowadzone testy zasadniczo potwierdziły te przypuszczenia. Jednak różnice w czasie wykonania zadania testowego dla różnych sposobów realizacji nie były wyjątkowo wyraźne. Nie udało się wykonać pełnego zestawu testów dla aplikacji ISAPI. Załamanie serwera WWW IIS następowało przy obciążeniu, które nie sprawiało kłopotów innym rozwiązaniom. Tabela 1 wskazuje na największą wydajność programów CGI w środowisku Linux-Apache. Ciekawe są dobre wyniki (szczególnie w porównaniu z IIS) stosunkowo mało popularnego serwera Xitami. Generalnie jednak większość przetestowanych stabilnych rozwiązań osiągnęła bardzo podobne wyniki, a różnice mieszczą się czasami w granicach błędu.

Wybór rozszerzenia powinien w związku z powyższym następować po rozważeniu korzyści wynikających z łatwości implementacji, przenośności kodu i łatwości późniejszej pielęgnacji. Biorąc pod uwagę tylko wydajność nie sposób podać technologii zdecydowanie przeważającej.

Interesujące wnioski pozwala przy tym wyciągnąć porównanie wydajności analogicznych rozwiązań zaimplementowanych na różnych platformach systemowych i dla różnych serwerów WWW. Okazuje się bowiem, że i ten aspekt nie ma wielkiego znaczenia, różnice w czasie wy-

konania testu były bowiem pomijalne dla analogicznych rozwiązań uruchamianych na platformie Linux i Windows.

Wszystkie testy przeprowadzono za pomocą programu MS Web Application Stress 1.1.

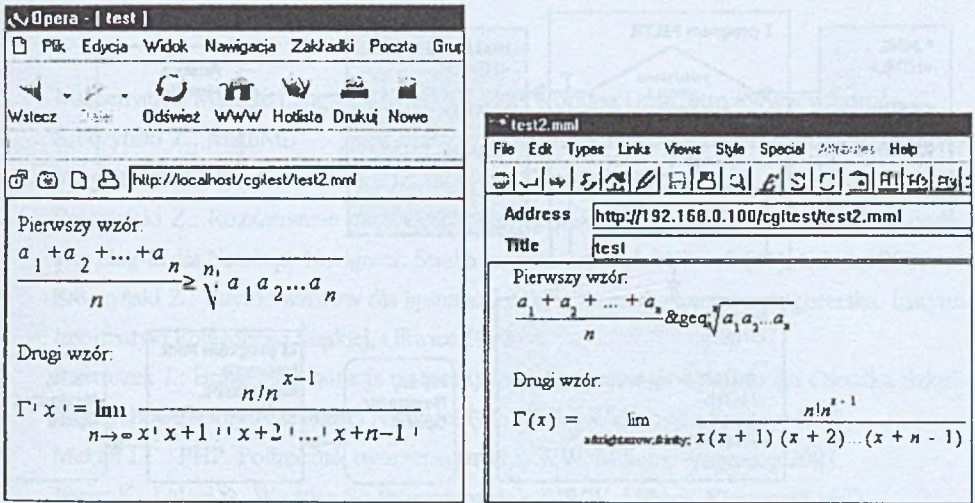
Tabela 1

Prędkość wykonania przykładowego zadania za pomocą różnych metod rozszerzania możliwości serwerów WWW. Wyniki podano znormalizowane do 100 względem najlepszego przy danym obciążeniu

Rozszerzenie	Serwer WWW	System operacyjny	Wynik przy obciążeniu			Łącznie
			małym	średnim	dużym	
CGI	Apache 1.3.19	Linux	104	104	100	100
PHP moduł	Apache 1.3.19	Linux	103	118	102	105
PHP exe	Xitami	Windows 2000	174	176	165	167
CGI	Xitami	Windows 2000	103	100	107	100
CGI	IIS 5.0	Windows 2000	115	100	107	104
PHP exe	IIS 5.0	Windows 2000	100	108	104	101

3. Algorytm działania rozszerzenia serwera WWW o obsługę języka MathML

Ze względu na łatwość przeniesienia istniejących rozwiązań w zakresie wizualizacji wzorów matematycznych [4] rozszerzenie serwera WWW o obsługę języka MathML zrealizowano w postaci programów CGI skompilowanych za pomocą pakietu Borland Delphi 3 Standard. Pozwala to na uruchomienie rozszerzania praktycznie na dowolnym serwerze WWW pracującym pod kontrolą systemu Windows. Nie powinno jednak sprawić wielkich trudności ewentualne przeniesienie działającego rozwiązania na platformę IIS w postaci bibliotek ISAPI. Możliwa jest również kompilacja w systemach Linux za pomocą pakietu Borland Kylix.

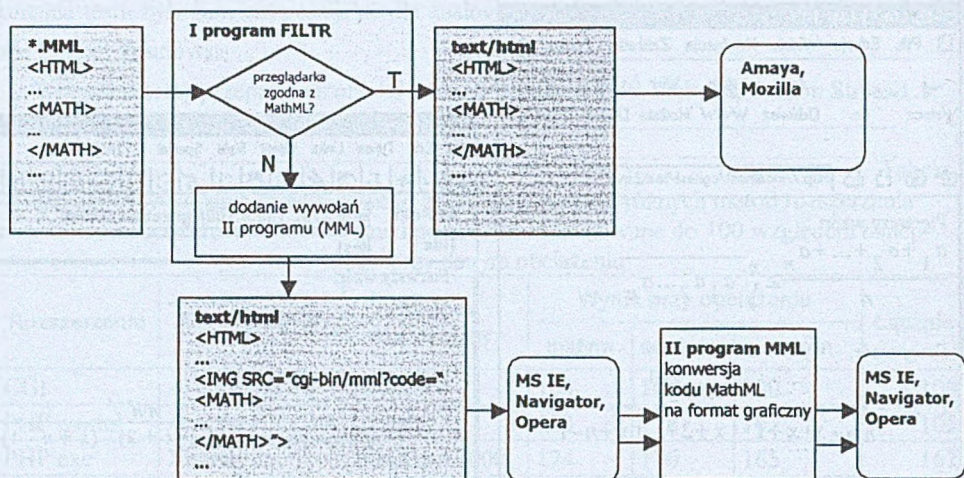


Rys. 1. Wizualizacja dokumentu MathML z wykorzystaniem zaimplementowanego rozszerzenia serwera WWW (z lewej w przeglądarce Opera) oraz bezpośrednio w przeglądarce Amaya (z prawej)

Fig. 1. The visualization of a web page with MathML equations as an output from web server extension (on the left in Opera browser) and rendered directly in the browser (on the right in Amaya)

Zaimplementowane rozszerzenie składa się z dwóch programów. Pierwszy z nich działa jako filtr, dokonując analizy kodu HTML przed odesłaniem go do przeglądarki klienta. Jeżeli przeglądarka klienta zostanie zidentyfikowana jako zgodna z MathML, nie są dokonywane żadne modyfikacje. W przeciwnym wypadku przed znacznikiem <math> dodawany jest kod wywołujący drugą część rozszerzenia. Takie podejście nie wymaga żadnych modyfikacji dokumentów w stosunku do specyfikacji MathML i zapewnia łatwość zmiany konfiguracji w miarę rozwoju możliwości interpretacyjnych przeglądarek. Można również rozpoznać wyszukiwarki internetowe i przesłać im niezmodyfikowany kod strony, co zapewni poprawne przeszukiwanie dokumentów zawierających znaczniki MathML.

Druga część rozszerzenia to program konwertujący kod MathML na grafikę w formacie JPG. Pobiera on kod MathML do interpretacji jako parametr CGI, dekoduje i zwraca dane graficzne. Schemat działania rozszerzenia zawiera rys. 2.



Rys. 2. Schemat działania rozszerzenia serwera WWW o obsługę języka MathML
Fig. 2. The structure of MathML web server extension

Detekcja przeglądarki używanej przez klienta pozwala nie tylko na odróżnienie przeglądarek obsługujących standard MathML od pozostałych. Można w tym miejscu podjąć decyzję o translacji dokumentu do innego formatu, w szczególności zaś do kodów notacji matematycznej języka Braile'a [5] lub nawet bezpośrednio kodów drukarek drukujących w tym standardzie. Budowa takiego rozszerzenia dałaby szansę publikowania dokumentów dostępnych powszechnie w Internecie, a przy tym czytelnych dla osób niewidomych bez konieczności ich specjalnego uprzedniego przetwarzania.

4. Podsumowanie

Przedstawione w pracy porównanie wydajności różnych metod rozszerzania funkcjonalności serwerów WWW nie wskazało zdecydowanie najlepszej technologii. Eksperymentalne rozszerzenie interpretujące język MathML zrealizowano jako programy CGI i przetestowano w praktyce. Jego wdrożenie daje szansę na łatwe upowszechnienie standardu MathML, który ze względu na trudności w przygotowaniu i publikacji dokumentów nie zdobył dotąd zaufania szerokiego grona użytkowników. Istotną zaletą proponowanego rozwiązania jest możliwość jego adaptacji na potrzeby osób niewidomych.

LITERATURA

1. Mathematical Markup Language (MathML) 2.0 Working Draft, <http://www.w3.org/>.
2. Sroczyński Z.: MathML — język opisu wyrażeń matematycznych w dokumentach internetowych, *Studia Informatica* Vol. 21 No. 3 (41), Gliwice 2000.
3. Sroczyński Z.: Rozszerzanie możliwości przeglądarek internetowych na przykładzie modułów plug-in dla Netscape Navigator, *Studia Informatica* Vol. 21 No. 3 (41), Gliwice 2000.
4. Sroczyński Z.: Edytor wzorów dla systemu LaTeX, Praca dyplomowa magisterska. Instytut Informatyki Politechniki Śląskiej, Gliwice 1997.
5. Świerczek J.: Brajlowska notacja matematyczna. Opracowanie Specjalnego Ośrodka Szkolno-Wychowawczego dla Dzieci Niewidomych i Słabo Widzących, Kraków 1998.
6. Meloni J.C.: PHP. Podręcznik tworzenia stron WWW, Mikom, Warszawa 2001.
7. Jamsa K., Lalani S., Weakley S.: Programowanie WWW. Mikom, Warszawa 1997.
8. Specyfikacja ISAPI firmy Microsoft.

Recenzent: Prof. dr hab. inż. Tadeusz Czachórski

Wpłynęło do Redakcji 2 kwietnia 2002 r.

Abstract

Web documents are in general intended to contain formatted text and some graphic. But very often a necessity to show a special kind of data appears, especially containing math expressions. Interpretation of such documents can be done on a client side, but requires special activities from a user (installing plug-ins). Extending web server seems to be better solution, because it can be completely transparent to a user.

Comparison of ways of extending web server possibilities is the next part of this article. But results did not point the best solution in performance-sensitive tasks (Table 1).

Next part of the paper is a description of the experimental web server extension translating MathML documents into common HTML with graphic (Fig. 2). Client web browser is detected prior to that to ensure no translation for MathML-aware browsers. Graphic is generated every time on-the-fly so document can be modified at any time and there is no need to prepare it before publication in a special way (Fig. 1).