

» Idź do

- Spis treści
- Przykładowy rozdział

» Katalog książek

- Katalog online
- Zamów drukowany katalog

» Twój koszyk

- Dodaj do koszyka

» Cennik i informacje

- Zamów informacje o nowościach
- Zamów cennik

» Czytelnia

- Fragmenty książek online

» Kontakt

Helion SA
ul. Kościuszki 1c
44-100 Gliwice
tel. 032 230 98 63
e-mail: helion@helion.pl
© Helion 1991-2008

PHP Web 2.0. Tworzenie aplikacji typu mashup

Autor: Shu-Wai Chow

Tłumaczenie: Maciej Jezierski, Radosław Meryk

ISBN: 978-83-246-1609-1

Tytuł oryginału: [PHP Web 2.0 Mashup Projects](#)

Format: 170x230, stron: 240



- Jak stosować mechanizmy dostarczane przez gigantów Web 2.0?
- Jak stworzyć swoją własną, prostą wyszukiwarkę?
- Jak zintegrować treści z Flickr i Google Maps?

Serwisy takie, jak Google Maps, Flickr, YouTube, MySpace, to symbole nurtu Web 2.0. Nie da się ukryć, że dorównać takim gigantom będzie Ci trudno. Ważne jednak jest to, że treści i mechanizmów dostarczanych przez nich możesz używać również i Ty. Wykorzystując ich API oraz dostępne usługi sieciowe, możesz stworzyć całkowicie nowy serwis, oferujący takie możliwości, o których nikomu wcześniej się nawet nie śniło! Właśnie dzięki tej książce dowiesz się, w jaki sposób połączyć treści i możliwości wielu serwisów w ramach nowej – Twojej – aplikacji WWW, korzystając z języka PHP. Poznasz zasady działania stron agregujących treści, a używanie technologii XML-RPC nie będzie stanowiło dla Ciebie problemu. Lektura książki „PHP Web 2.0 Mashup” pozwoli Ci poznać zasady pracy z żądaniami typu REST. Poznasz metody GET i POST oraz nauczysz się przetwarzać odpowiedzi w formacie XML, korzystając z biblioteki SAX. Już wkrótce będziesz mógł sam stworzyć prostą wyszukiwarkę czy też własną "szafę grającą", korzystającą z danych YouTube i Last.fm. Dodatkowo będziesz miał możliwość poznania API dostępnego między innymi w Google Maps, internetowej bazie danych UPC oraz księgarni Amazon. Jeżeli chcesz stworzyć nowy serwis agregujący treści, jest to obowiązkowa pozycja dla Ciebie!

- Funkcjonowanie stron agregujących treści
- Wykorzystanie techniki XML-RPC
- Przetwarzanie odpowiedzi przy użyciu SAX
- Żądania REST
- Działanie metod POST i GET
- Język WSDL
- Zastosowanie kanałów RSS
- Interfejs API Amazon
- Interfejs API Google Maps
- Interfejs API internetowej bazy danych UPC
- Interfejs API Flickr
- Zastosowanie obiektu XMLHttpRequest i jego rola w technologii AJAX

Stwórz nowy projekt, korzystający z możliwości gigantów Web 2.0!

Spis treści

O autorze	9
O recenzencie	11
Wstęp	13
Rozdział 1. Wprowadzenie do aplikacji typu mashup	17
Web 2.0 i mashup	19
Znaczenie danych	19
Społeczności użytkowników	20
Jak będziemy tworzyć aplikacje typu mashup	21
Więcej aplikacji typu mashup	22
Rozdział 2. Zrób zakupy w Amazon	23
Podsumowanie projektu	23
XML-RPC	24
Struktura XML-RPC	24
Odpowiedź XML-RPC	29
Obsługa XML-RPC w PHP	31
Wykonywanie żądania XML-RPC	31
Przetwarzanie odpowiedzi XML-RPC	40
Tworzenie klasy parsującej XML-RPC	41
Wykorzystanie PEAR do obsługi XML-RPC	44
REST	46
Praca z REST w PHP	48
Wykonywanie żądania REST	48
Przetwarzanie odpowiedzi REST	55
Interfejs API internetowej bazy danych UPC	65
Interfejs API Amazon	67
Przegląd ECS	68
Budowa żądania REST ECS	69
Tworzenie aplikacji typu mashup	71
Wyszukiwanie produktów	71
Podsumowanie	84

Rozdział 3. Stwórz własną wyszukiwarkę	85
Podsumowanie projektu	85
SOAP	86
Web Services Descriptor Language (WSDL) i XML Schema Data (XSD)	87
Podstawowa struktura WSDL	87
Komunikat SOAP	99
Klasa SoapClient w PHP	103
Tworzenie parametrów	104
Tworzenie instancji SoapClient	105
Wykonywanie wywołania za pomocą metod SoapClient	107
Obsługa odpowiedzi SOAP	110
Usługa sieciowa Microsoft Live Search	113
Korzystanie z wyszukiwania	113
Usługa Yahoo! Search	117
Korzystanie z wyszukiwania stron internetowych	117
Tworzenie strony agregującej	119
Podsumowanie	124
Rozdział 4. Własna szafa grająca z teledyskami	125
Podsumowanie projektu	125
XSPF	126
RSS	129
Przegląd YouTube	136
Interfejs programistyczny YouTube	137
Przegląd Last.fm	139
Usługi internetowe Audioscrobbler	140
Parsowanie za pomocą PEAR	141
Instalacja i korzystanie z pakietów	142
File_XSPF	143
Services_YouTube	144
XML_RSS	147
Tworzenie aplikacji typu mashup	150
Architektura aplikacji typu mashup	150
Strona główna	151
Strona nawigacyjna	152
Strona z zawartością	153
Korzystanie z aplikacji typu mashup	155
Podsumowanie	158
Rozdział 5. Zdjęcia londyńskiego metra	159
Podsumowanie projektu	159
Wstępne planowanie	160
Znalezienie informacji o stacjach metra	161
Integracja serwisów Google Maps i Flickr	162
Kolejność operacji w aplikacji	163
RDF (Resource Description Framework)	164

SPARQL	166
Analiza przedmiotu zapytania	167
Anatomia zapytania SPARQL	168
Klauzule WHERE w języku SPARQL	169
Dodatkowe własności języka SPARQL	177
RDF API for PHP (RAP)	177
Obiekt XMLHttpRequest	180
Przegląd informacji na temat obiektu XMLHttpRequest	181
Korzystanie z obiektu	182
Notacja JSON (JavaScript Object Notation)	186
Przegląd obiektów JavaScript	187
Struktura JSON	187
Korzystanie z właściwości JSON	188
Serializacja odpowiedzi JSON	189
Interfejs API Google Maps	190
Tworzenie mapy	191
Geokodowanie	192
Znaczniki	194
Zdarzenia	195
Obiekty InfoWindow	195
Interfejs API Flickr Services	198
Wykonywanie operacji wyszukiwania	199
Interpretacja wyników zwróconych przez usługę	200
Pobieranie fotografii lub strony z fotografiami	202
Tworzenie aplikacji typu mashup	203
Tworzenie bazy danych i wypełnianie jej danymi	204
Klasa interfejsu z bazą danych — TubeSource	214
Główny interfejs użytkownika	216
Wykorzystanie usług Flickr i technologii AJAX	220
Podsumowanie	230
Skorowidz	231

Wprowadzenie do aplikacji typu mashup

Strony agregujące (mashup), określane również przez Wikipedię jako hybrydowe aplikacje internetowe, stają się w ostatnich latach coraz popularniejszym kierunkiem rozwoju sieci. Są one dokładnie tym, na co wskazuje ich nazwa — aplikacjami internetowymi, które pobierają dane z jednego lub kilku źródeł i prezentują je w nowy sposób. Właściciele danych chętnie udostępniają swoje zasoby zewnętrznym użytkownikom. W wielu przypadkach oferują interfejsy programistyczne (API) oparte na standardowych protokołach usług internetowych, które z łatwością mogą być zaimplementowane w różnych językach programowania, włączając w to PHP. Każdego dnia pojawiają się strony agregujące tworzone przez ludzi, którzy w innowacyjny sposób łączą dane.

Jednym z przykładów jest witryna Wii Seeker. Kiedy w listopadzie 2006 roku pojawiła się na rynku konsola Nintendo Wii, wiadomo było, że zrodzą się problemy z jej zakupem. Zadaniem witryny Wii Seeker była pomoc w odnalezieniu sklepu, w którym dostępne są konsole, poprzez połączenie informacji o dostawach z danymi Google Maps. Znacznik na mapie Google Maps wskazywał sklep. Kiedy użytkownik kliknął znacznik, wyświetlane były informacje o sklepie, m.in. adres. Oprócz tego pojawiała się również informacja o liczbie konsol oferowanych do sprzedaży w sklepie w dniu premiery. Dzięki temu, że na mapie wyświetlane były dane na temat wielkości dostawy, użytkownik mógł odszukać sklep położony najbliżej swojego domu i tak zaplanować zakupy, aby mieć jak największe szanse nabycia Wii.

Po wypuszczeniu konsoli na rynek strona została uzupełniona o informacje o aukcjach w serwisie eBay i produktach dostępnych w Amazon. Dodano także sklepy sieciowe takie jak Circuit

City i Wal-Mart. Zamiast pokazywać informacje o zapasach konsol Nintendo Wii w każdym sklepie, witryna umożliwia użytkownikom wymianę informacji na temat dostępności konsol w poszczególnych sklepach.

Innym przykładem witryny mashup jest Astrolicio.us. Pobiera ona dane z takich witryn jak Digg.com, Google News oraz Google Videos i prezentuje je na jednej stronie. W ten sposób powstał portal z bieżącymi informacjami o astronomii.

Na stronie głównej użytkownik może szybko przejrzeć tematy i wybrać te, które go interesują. Wiadomości prezentowane są w postaci wypunktowanej listy zawierającej nagłówki i skróconą

treść. Zamiast filmów widnieją ich miniaturki. Po kliknięciu łącza użytkownik zostanie przeniesiony na stronę źródłową artykułu lub filmu. Strona jest czytelna, prosta i zawiera dużo informacji. Bardzo łatwo ją stworzyć za pomocą interfejsów API źródeł danych. Prawdopodobnie jej zrobienie nie zajęło autorowi więcej niż jeden wieczór.

Web 2.0 i mashup

Dlaczego w ciągu zaledwie kilku lat aplikacje typu mashup znalazły tak powszechne zastosowanie? Ich historia zaczęła się kilka lat temu. Po pęknięciu bańki internetowej w 2001 roku firmy internetowe przekształciły się. Musiały zmienić podejście biznesowe, poddać technologie ponownej ocenie i wziąć pod uwagę sposób, w jaki użytkownicy korzystają z internetu. W połowie dekady ujawniły się kluczowe trendy. Pojawiło się określenie **Web 2.0**, pozwalające oddzielić nowe witryny od witryn, które zdobyły popularność pod koniec lat 90. Określenie to było mało precyzyjne i na pierwszy rzut oka wydawało się podejrzanie efekciarskie. Jednak różnice między nowymi i starymi witrynami są rzeczywiste. Nie chodzi tylko o zaszczoły historyczne czy zmiany wynikające z upływu czasu. Witryny takie jak Google, YouTube czy Flickr zademonstrowały nowe podejście do biznesu internetowego. Miały one często proste interfejsy, w pełni rozwinięte usługi internetowe i dawały użytkownikowi dużo swobody. Wiele z nich bazowało na zawartości dostarczanej przez użytkowników. We wrześniu 2005 roku wydawca książek technicznych Tim O'Reilly napisał artykuł zatytułowany *Czym jest Web 2.0*, w którym zwięźle opisał cechy odróżniające witryny Web 2.0 od witryn poprzedniej generacji. Uwzględnił w swoim wykazie dwie cechy, które bezpośrednio wpłynęły na rozwój stron mashup:

- znaczenie danych,
- społeczność użytkowników.

Znaczenie danych

Pierwszą cechą jest znaczenie danych. Kwestia tego, kto jest właścicielem danych i co zamierza z nimi zrobić, stała się poważnym problemem. Dlaczego firmy miałyby inwestować miliony dolarów, żeby gromadzić dane w systemach bazodanowych, a następnie za darmo udostępniać je innym? Dlatego, że dzięki temu twórcy stron agregujących ułatwiają dotarcie do właścicieli danych.

Żeby to pokazać, O'Reilly posłużył się przykładem firmy MapQuest, która była liderem na rynku map w drugiej połowie lat 90. Jednak jej system był zamknięty i nie umożliwiał zewnętrznym użytkownikom wykorzystywania jej danych. W pierwszych latach XXI wieku strony z mapami zaczęły walczyć z tymi słabościami. Na rynek weszły Yahoo! Maps, Microsoft Virtual Earth i Google Maps, udostępniając interfejsy API. Pomimo ogromnego udziału w rynku, MapQuest szybko zaczął tracić przewagę na rzecz firm udostępniających dane. Jest wiele podobnych przykładów. Amazon udostępnił dane poprzez Amazon E-Commerce Service (ECS). Wiele witryn agregujących wykorzystało tę usługę do stworzenia własnych sklepów. Amazon organizuje

sprzedaż, płacąc prowizję twórcom i właścicielom witryn mashup. Dzięki temu firma uruchomiła nowe kanały sprzedaży produktów, choć oczywiście *www.amazon.com* nadal działa. Z kolei firma BarnesAndNoble.com nie udostępniła swoich danych. Jedynek kanałem ich sprzedaży jest główna witryna. Spowodowało to nie tylko utratę możliwości sprzedaży, ale również mniejszą lojalność użytkowników niż w przypadku Amazon.

We wcześniejszych przykładach przedstawiliśmy witrynę Wii Seeker, która pomagała sprzedawcom poprzez kierowanie nabywców do sklepów. W zamian za to pobierana jest opłata za reklamę i członkostwo. Google Videos, Google News i Digg.com zdobywają użytkowników, kiedy ktoś kliknie łącze na witrynie Astrolicious.us. Z kolei Astrolicious.us otrzymuje opłatę za reklamy, wkładając niewielki wysiłek w ich stworzenie.

Spółeczności użytkowników

Drugą cechą jest to, że dane wprowadzone przez użytkowników są o wiele cenniejsze, niż dotąd sądziliśmy. Recenzje produktów pisane przez użytkowników na stronach komercyjnych nie są jakąś nowością. Nie są nią również fora internetowe. Jednak ważne stało się to, w jaki sposób witryny korzystają z tych informacji i kto jest ich właścicielem. Witryna wypożyczająca filmy, Netflix, od zawsze umożliwiała użytkownikom ocenianie filmów, które oglądali. Na podstawie tych ocen Netflix sugeruje inne filmy, które mogłyby się im podobać. Niedawno dodano nową funkcjonalność społecznościową o nazwie „Przyjaciele”, która umożliwia obejrzenie ocen przyznanych filmom przez Twoich przyjaciół oraz zdobycie informacji o tym, jakie filmy oglądają. Jedną z opcji jest ocena podobieństwa. Dzięki porównaniu Twoich rekomendacji z rekomendacjami przyjaciół Netflix wylicza procentowy wskaźnik zgodności gustów.

Niektóre witryny są całkowicie zależne od danych użytkowników. YouTube i Flickr udostępniają za darmo miejsce na filmy i zdjęcia. Ich popularność nie wzięła się jednak po prostu z tego, że usługi są darmowe. Nie ma w tym niczego nowego. Różnica ponownie polegała na tym, co obie witryny robią z danymi dodawanymi przez użytkowników. Obie udostępniają funkcjonalności społecznościowe. Możesz ocenić i skomentować zdjęcia i filmy innych użytkowników, a także śledzić zmiany w ich profilach. Za każdym razem, kiedy ktoś doda jakieś dane, zostaniesz o tym poinformowany. Obydwie witryny umożliwiają także swobodne oznaczanie treści, dzięki czemu użytkownik może opisać dodany element za pomocą własnych słów kluczowych. Odwiedzający mogą użyć tych słów podczas wyszukiwania zawartości. Oznaczanie treści okazało się doskonałą pomocą przy tworzeniu algorytmów wyszukiwania.

Te dwie cechy nowych witryn umożliwiają małym firmom internetowym sprawianie wrażenia, że są o wiele większe. Korzystając z danych udostępnianych przez większe firmy, twórcy witryn agregujących znajdują zastosowania, o których nie pomyśleliby właściciele danych lub których nie mogliby wprowadzić z powodu ograniczeń biznesowych.

Jak będziemy tworzyć aplikacje typu mashup

Z technologicznego punktu widzenia zjawisko agregacji danych nie narodziłoby się, gdyby właściciele witryn internetowych nie oddzielili wyraźnie danych, które są wykorzystywane na witrynach, od właściwej ich prezentacji. To zawsze było celem podczas tworzenia aplikacji internetowych, dlatego też nie powinien dziwić kierunek rozwoju ich architektury. Dzięki temu podziałowi sieć przekształca się w coś, co jest znane jako **semantyczna sieć stron internetowych** — zawartość internetowa jest prezentowana nie tylko w celu czytania przez ludzi, ale również może być łatwo przetwarzana przez komputery. Przeszliśmy od stron statycznych do stron opartych na bazach danych, od prezentacyjnych znaczników FONT do kaskadowych arkuszy stylów. Prawdopodobnie nieuniknione było to, że internet stanie się środowiskiem, w którym rozwiną się witryny mashup.

Źródła danych dla witryn agregujących są zróżnicowane. Często właściciele udostępniają dane poprzez oficjalne interfejsy API. Interfejsy te używają usług sieciowych, które korzystają z wielu protokołów. Innym powszechnie stosowanym źródłem danych, które pomogło w rozwoju witryn agregujących, jest rodzina formatów RSS. Kiedy niedostępne są oficjalne metody, programiści bardzo kreatywnie poszukują rozwiązań umożliwiających pobieranie danych. Przetwarzanie całych stron jest często stosowaną metodą. Bez względu na metodę, witryny agregujące korzystają z różnych formatów danych i zasadniczo są dość proste w tworzeniu. Mimo to ich twórcy muszą być elastyczni i dobrze znać używane narzędzia.

Oprogramowanie o otwartym kodzie szczególnie dobrze zaspokaja potrzeby twórców witryn agregujących. Połączenie serwera Apache i języka PHP umożliwia szybkie tworzenie oprogramowania. Dzięki otwarciu kodu, programiści bez przerwy i szybko dodają nowe funkcjonalności zapewniające spójność z usługami internetowymi.

W tej książce przedstawię, w jaki sposób korzystać z popularnych źródeł danych w PHP. Większość oficjalnych interfejsów API oparta jest na jednym z trzech protokołów internetowych — XML-RPC, REST lub SOAP. Oczywiście przyjrzymy się tym protokołom. Interfejsy API i bezpośrednie ręczne odwoływanie się do usług internetowych to nie jedyne sposoby pobierania danych. Pokażę również, jak korzystać z zewnętrznych bibliotek w celu komunikacji z popularnymi witrynami. Innym ważnym źródłem danych są kanały informacyjne. Poznanie szerokiego zestawu narzędzi wykorzystywanych do tworzenia stron agregujących pozwoli Ci na szybkie rozpoczęcie pracy nad własną witryną.

Więcej aplikacji typu mashup

Poniżej znajdują się przykłady aplikacji typu mashup, z których można czerpać inspirację:

- Popurls (*popurls.com*) — kolekcja adresów URL z popularnych witryn,
- Housingmaps.com (*www.housingmaps.com*) — wyświetla na mapie pozycje domów z witryny Craigslist,
- Keggy (*us.keggy.com*) — witryna agregująca wiadomości z różnych źródeł i personalizująca ją pod kątem określonego czytelnika,
- Alkemis (*local.alkemis.com*) — agreguje i umieszcza na mapie wszystkie rodzaje danych, na przykład zdjęcia i obrazy z kamer w wybranych miastach,
- Gametriping.com (*www.gametriping.com*) — kolekcja zdjęć satelitarnych i zdjęć z serwisu Flickr ze stadionami baseballowymi.