

**Wiosenna Szkoła PTI
Świnoujście '98**

Tworzenie aplikacji Internetowych

**Organizowana przez
POLSKIE TOWARZYSTWO INFORMATYCZNE**

Świnoujście 11-15 maja 1998

Wiosenna Szkoła PTI
Świnoujście'98

Tworzenie aplikacji Internetowych

Organizowana przez
POLSKIE TOWARZYSTWO INFORMATYCZNE

Świnoujście 11-15 maja 1998



50-LECIE INFORMATYKI POLSKIEJ (1948 – 1998)

W końcu 1948 roku w powstającym wówczas Państwowym Instytucie Matematycznym (późniejszym Instytucie Matematycznym PAN) powołano Grupę Aparatów Matematycznych. To symboliczne wydarzenie przyjęto uznawać za początek informatyki w Polsce.- dziedziny , która spowodowała w świecie zmiany cywilizacyjne porównywane często do tych przypisywanych rewolucji przemysłowej XIX wieku.

Komitet Informatyki PAN postanowił przypomnieć tę rocznicę doceniając rolę, jaką informatyka odgrywa we współczesnym świecie , a także jako wyraz szczególnego uznania dla tych , którzy w zrujnowanej wojną Polsce wykazali zdumiewającą przenikliwość i śmiałość stawiając na rozwój zupełnie wtedy nowej dziedziny nauki i techniki.

Komitet Informatyki PAN uważa , że najważniejsze jest upowszechnienie tej * rocznicy w samym środowisku informatycznym i ogólnie w społeczeństwie. Dlatego zwraca się z apelem do organizatorów krajowych spotkań odbywających się w roku 1998 (konferencji , targów , sympozjów itp.), żeby zechcieli przypomnieć i podkreślać ich rocznicowy charakter, a może zaproponować ich ramach, jakiś szczególny rocznicowy akcent. Temu celowi ma służyć ta ulotka informacyjna i specjalnie zaprojektowany znak 50-lecia na niej umieszczony , który Komitet Informatyki rozpowszechnia i udostępnia do swobodnego wykorzystania. Komitet ma nadzieję, że np. znak, o którym mowa będzie umieszczany na różnych materiałach związanych z informatyką a ukazujących się w roku 1998 (informacje o konferencjach , materiały konferencyjne, książki itp.), a także rozpropagowany – wraz z informacją rocznicową – przez periodyki specjalistyczne i inne ogólnie dostępne środki przekazu, także poprzez Internet.

Niektóre przyszłoroczne spotkania informatyczne już zgłosiły chęć bardziej czynnego włączenia się do obchodów ustalając w porozumieniu z Komitetem jego formę. Lista takich spotkań może być dłuższa! Czekamy na podjęcie inicjatywy !

Treść tej ulotki, znak 50-lecia i inne aktualne informacje na ten temat znaleźć można pod adresem: <http://www.ipipan.waw.pl/~lat50>

Adres do korespondencji: 50 – LAT INFORMATYKI

IPI PAN

ul. Ordona 21

01-237 Warszawa

fax: 022 376564 , e-mail: lat50@ipipan.waw.pl.

KOMITET INFORMATYKI POLSKIEJ AKADEMII NAUK

SPIS TREŚCI

Zdzisław Szyjewski - Polskie Towarzystwo Informatyczne
INTERNET – nowe spojrzenie na tworzenie aplikacji

Barbara Łukasik-Makowska – A E Wrocław
Problemy projektowe tworzenia witryn internetowych

O firmie Sun Microsystems

Kamil Kurowski - RODAN-SYSTEM
Serwis Informacyjny ArsInfo

Dariusz Leonarski – Novell Polska
NetWare 5 – najwydajniejsza platforma dla aplikacji napisanych w Javie

Piotr Tucholski - CSBI S. A. Oddział Progress
Tworzenie aplikacji internetowych w systemie WebSpeed

Mariusz Chmielewski - SYSOFT
Java na poważnie

Michał Kasprzak – A T M S. A.
Bezpieczne sieci korporacyjne

INTERNET – nowe spojrzenie na tworzenie aplikacji

Zdzisław Szyjewski
Uniwersytet Szczeciński
Polskie Towarzystwo Informatyczne
Szczecin ul. Mickiewicza 66
e-mail : zszyjew@uoo.univ.szczecin.pl

W historii rozwoju każdej dziedziny życia istnieją okresy powolnego ewolucyjnego rozwoju oraz okresy skokowego wzrostu i przewartościowań dotychczasowych metod i narzędzi. Zaistnienie pewnych faktów lub odkryć powoduje, że na dotychczasowe sposoby realizacji, stosowane narzędzia, środki i wykorzystywane metody, zaczynamy patrzeć z innego punktu widzenia. Krytyczna ocena wynika z nowej sytuacji, która z jakichś powodów zaistniała i jest obowiązująca.

Sytuacje takie w relatywnie krótkiej historii rozwoju informatyki występują wyjątkowo często. Obecnie przeżywamy dynamiczny rozwój sieci globalnej INTERNET i oddziaływanie jej wykorzystania na wszystkie dziedziny życia. Pomijając wiele bardzo profesjonalnie zasadnych i dobrych rozwiązań stosowania sieci INTERNET, należy zwrócić uwagę na szerokie stosowanie jej do rozrywki. Fakt ten nie mający bezpośredniego wpływu na dzisiejsze zastosowania informatyki, tworzy potencjalną, liczną grupę przyszłych klientów zastosowań informatyki. Młodzi ludzie przyzwyczajeni do mechanizmów stosowanych w grach komputerowych, będą chcieli stosować je również w pracy zawodowej, kiedy minie czas nauki i zacznie się okres aktywności zawodowej.

Powszechne stosowanie informatyki w bardzo różnorodnych dziedzinach życia, w różnym środowisku i przez różnych użytkowników, to bardzo prawdopodobna wizja najbliższych lat. Powstaje pytanie czy obecnie stosowane metody i narzędzia wytwarzania aplikacji informatycznych sprostają temu wyzwaniu. Zastosowanie technologii sieciowych w aplikacjach informatycznych stanowiło pierwszy etap dostosowywania się twórców systemów informatycznych do nowej sytuacji. Czas aplikacji posadowionej na jednym centralnym komputerze z odpowiednio dużymi zasobami, mija bezpowrotnie. Obowiązująca zaczyna być sytuacja, gdzie aplikacja ma szeroki zasięg użytkowników o bardzo zróżnicowanym poziomie wiedzy¹. Środowisko informatyków musi być przygotowane na wytwarzanie aplikacji funkcjonujących efektywnie w sieci INTERNET.

Co jest charakterystyczne dla tej klasy aplikacji ? Obok bardzo istotnych różnic technologii informatycznej, bardzo istotną różnicą w stosunku do dotychczas wytwarzanych aplikacji jest użytkownik tej aplikacji. Użytkownik, który jest ekstremalnie różnorodny, z racji powszechności stosowania aplikacji, ponadto mniemający, że działanie aplikacji informatycznej powinno być tak samo proste i zgodne z jego oczekiwaniami, jak jest to łatwe i bezproblemowe w grach komputerowych. Nie ma rzeczy niewykonalnych, czas reakcji ma być natychmiastowy, ewentualne błędy mają być samo naprawialne i tak dalej.

¹ Nie oznacza to, że tylko takiego typu aplikacje należy teraz wytwarzać, natomiast jest to znacząca zmiana jakościowa w porównaniu z dotychczasową klasą zastosowań. Zastosowania wytwarzane obecnie mają już sprawdzone metody i narzędzia wytwarzania i nie jest to obiektem naszych zainteresowań.

Wysokie wymagania użytkownika co do funkcjonalności i elastyczności rozwiązań informatycznych idą w parze z dużą komplikacją logiki działania aplikacji i różnorodności środowiska informatycznego. Globalna sieć INTERNET połączyła różnorodne komputery z bardzo różnymi środowiskami systemów operacyjnych, oraz korzysta z bardzo różnorodnie zapisanych danych. Mnogość rozwiązań informatycznych, zintegrowanych w jednej globalnej sieci komputerowej, to problem informatyków, użytkownik aplikacji nie wnika w problemy twórców a jedynie wymaga produktu spełniającego jego wysokie oczekiwania.

Postulat systemów „otwartych” zgłaszany i realizowany od pewnego czasu w środowisku informatycznym jest częściowym rozwiązaniem problemu. Pozwala jedynie w miarę bezboleśnie łączyć w jednym systemie różne środowiska sprzętowe i operacyjne oraz wymieniać dane pomiędzy aplikacjami pracującymi w tych środowiskach. Rozproszenie zasobów wykorzystywanych w jednej aplikacji zmodyfikowało w sposób znaczący technologie obliczeń i wytwarzania systemów informatycznych. Powstały nowe narzędzia i metody wytwarzania systemów. Wykorzystanie danych przechowywanych na zdalnym komputerze i pracującym w innym niż źródłowe środowisku, stało się faktem w licznych aplikacjach informatycznych.

Wiadomo, że na każdą aplikację informatyczną obok danych składa się algorytm ich przetwarzania. Algorytm przetwarzania dotychczas był zlokalizowany centralnie i jedynie drobne operacje na danych były realizowane na zdalnych komputerach. Drobne przekształcenie danych na potrzeby aplikacji jest stosunkowo prostym zabiegiem w porównaniu z koniecznością zdalnej realizacji algorytmu obliczeniowego. Mimo unifikacji sprzętu i oprogramowania, komputery pracują na innych procesorach i mają do dyspozycji inne listy rozkazów podstawowych. Proste przeniesienie zakodowanego na jednym procesorze algorytmu na inny nie wchodzi w rachubę. Dla realizacji algorytmu (czyli procedury programowej) konieczne jest wykonanie procesu przekształcenia jej zgodnie z wymaganiami procesora, na którym ma być realizowana. Należy ponownie wykonać proces kompilacji, co nie zawsze jest procesem prowadzącym do sukcesu.

Próba rozwiązania tego problemu jest Java, język programowania, który zdobył już dużą popularność i stale zwiększa się liczba jego sympatyków. Java istnieje już od kilku lat ale jego dynamiczny wzrost popularności obserwujemy od niedawna. Częściowo wynika to z zamieszania jakie spowodowane zostało przez głównych dostawców tego języka. Głównym powodem wzrostu popularności Javy jest jednak wyjście naprzeciw problemom informatyków piszących aplikacje korzystające z sieci INTERNET. Emocje związane z wejściem tego języka przysłoniły podstawowe problemy i sposób ich rozwiązywania. Niektórzy twierdzą, że jest to jeden z wielu języków programowania i podobnie jak kolejne pojawiające się języki programowania ma swój okres dużej popularności wynikający z premii nowości. Być prorokiem w informatyce jest niezwykle trudno i niebezpiecznie, dlatego nie będziemy prognozować a jedynie uważnie obserwować dalsze losy tego języka.

Na czym polega fenomen Javy ?

Czym Java różni się od innych języków programowania ?

Java z założenia jest językiem uniwersalnym, pozwalającym na wykonanie programu napisanego w tym języku na różnych komputerach bez potrzeby zmusznych i skomplikowanych zabiegów w rodzaju kompilacji, łączenia modułów i innych procesów przystosowawczych. Cechą tą osiąga Java poprzez zainstalowanie na każdym rodzaju

komputera programu o nazwie "wirtualny komputer Java". Oczywiście na każdym konkretnym komputerze jest to inny program, który ma cechy znanego w informatyce interpretera. Pamiętamy niedawną karierę interpretera Basica, która jednak trwała relatywnie krótko ale spełniła swoją istotną rolę w wprowadzenie sprzętu klasy mikro na rynek informatyczny. Czy Java podzieli los interpreterów Basica ?

Jedną z przyczyn upadku Basica, jako jedyne go języka programowania na sprzęcie klasy mikro, było powstanie licznych dialektów tego języka. Autorzy Javy znają te doświadczenia i stąd taka walka o czystość języka. Inną cechą Javy jest tworzenie środowiska współpracy z tym językiem, co może spowodować efekt samonapędzającego się mechanizmu w opanowywaniu rynku. Powstanie komputera sieciowego (network computer czy nawet java station) tworzy środowisko sprzętowe, które wymusza stosowanie języka w zamian za wiele udogodnień i gwarancji dostarczanych w przypadku przyjęcia takiego rozwiązania. Miarą z popularnymi przeglądarkami internetowymi to kolejny krok w kierunku związania użytkownika z Javą.

Problemy związane z doбором sprzętu, środowiska systemowego i programowego, stopień wspomagania prac wytwarzania systemu to część decyzji jakie stoją przed twórcami aplikacji informatycznych. Odpowiednie wybory nasuwają rozwiązania zaszyte w przyjętej technologii ale liczne problemy pozostają i muszą być efektywnie rozwiązane przez autorów aplikacji. Nową jakość w aplikacjach informatycznych tworzy środowisko multimedialnego sprzętu dostępnego już dla większości użytkowników. Oprócz oczywistego wymogu poprawności istotną cechą staje się elegancja, estetyka rozwiązania.

Sposób prezentacji na stanowisku klienta aplikacji staje się jednym z istotniejszych problemów każdej aplikacji informatycznej. Zmysł artystyczny informatyka, umiejętność dotarcia do klienta poprzez umiejętne opakowanie dostarczanych informacji bardzo często jest czynnikiem sukcesu lub porażki. Jest to zupełnie nowe wyzwanie dla informatyków, w wielu przypadkach przekraczające możliwości dotychczas doskonałych twórców systemów informatycznych. Grafika, dźwięk, operowanie efektami animacji jest równie ważne jak poprawność algorytmu i jego szybkość a w wielu przypadkach staje się nawet ważniejsze dla użytkownika wychowanego na grach komputerowych.

W dotychczasowych doświadczeniach realizacji systemów informatycznych najczęściej mieliśmy lepszy lub gorszy kontakt z przyszłym użytkownikiem naszego produktu. Mogliśmy poznać jego oczekiwania, upodobania i w miarę możliwości zaspokoić jego potrzeby funkcjonalne i estetyczne. Obecnie coraz częściej użytkownik jest anonimowy i to w bardzo szerokim zakresie rozumienia tego słowa. Nie tylko nie potrafimy określić jego autentycznych potrzeb, zainteresowań, upodobań ale również nieznane są jego intencje i wiedza konieczna przy poprawnym korzystaniu z wytwarzanej aplikacji. Z drugiej strony powinniśmy naszą aplikację, jej wyglądem zewnętrznym, zabiegać o tego użytkownika. Konkurencja na rynku usług informacyjnych powoduje, że nasza wytwarzana aplikacja powinna być na tyle interesująca, żeby w powodzi wielu innych wyróżnić się czymś i przyciągnąć uwagę.

Zastosowania informatyki przenoszą się z gabinetów i stanowisk pracy do domu. Użytkownikiem wytwarzanej aplikacji jest normalny, anonimowy człowiek, który w zaciszu domowego pokoju wykorzystuje naszą aplikację dla rozwiązania jakiegoś własnego problemu. Ta istotna zmiana jakościowa musi mieć wpływ na proces wytwarzania na

stosowane narzędzia, środki i metody tworzenia. Czy jesteśmy do tej nowej sytuacji przygotowani?

Obecne wydanie Wiosennej Szkoły PTI Świniujście'98 ma pomóc Państwu w odpowiedzi na to pytanie. Mam nadzieję, że prezentowane wykłady zaspokoją ciekawość i udzielą odpowiedzi na podstawowe pytania związane z dalszym rozwojem informatyki w obszarze wykorzystania sieci INTERNET. Warsztaty programowania w języku Java nie tylko przybliżą ten język ale pozwolą na osobiste przekonanie się o zaletach tego narzędzia tworzenia aplikacji w INTERNECIE.

Maj 1998

Barbara Lukasik-Makowska
Akademia Ekonomiczna
we Wrocławiu

PROBLEMY PROJEKTOWE TWORZENIA WITRYN INTERNETOWYCH

1. Dla kogo powstają witryny w Internecie?

Odpowiedź na to z pozoru banalne pytanie wcale nie jest prosta i jednoznaczna. Czymże bowiem jest, lub może być, witryna internetowa w działalności określonego podmiotu gospodarczego? Czy można wyłącznie utożsamiać ją z jakąkolwiek inną, tradycyjną formą komunikacji tego podmiotu z otoczeniem? Czy należy uznać, że witryna prezentacyjna to tylko swoisty elektroniczny folder, prospekt lub nawet książka? Czy może jest to jednak specyficzny rodzaj publikacji, o potencjalnie bardzo szerokim zasięgu odbiorców, której przygotowanie stawia autorom wyjątkowe wyzwania? Jak postrzegają Internet szefowie firm?

Analiza zawartości różnych witryn dostępnych w Internecie skłania do stwierdzenia, że w wielu przypadkach istnienie Internetu (globalnej, otwartej sieci) jest przez podmioty gospodarcze bagatelizowane, a tworzenie witryn internetowych, to nadal bardziej moda, lub próba zweryfikowania własnych umiejętności, niż faktyczne projektowanie nowoczesnego narzędzia komunikacji z otoczeniem, ściśle związane z potrzebami i funkcjami określonego podmiotu gospodarczego [LUKA97]. Dominują witryny „książkopochodne”, a więc takie, gdzie poza użyciem odmiennych narzędzi prezentacji tekstu oraz wstawieniem paru (zazwyczaj przypadkowo dobranych) elementów graficznych trudno mówić o jakimkolwiek celowym posłużeniu się nową techniką. Do witryny „wrzuca” się w sposób dość losowy różne teksty historyczne i promocyjne, ubarwia się to zdjęciami o miernych walorach poznawczych, oszalała czytelność, na których tekst główny ztraca czytelność, a do tego agresywnie manifestuje „nowość”, niezależnie od faktu, że z daty ostatniej aktualizacji jasno wynika, że administrator nie zaglądał do witryny na przykład od ponad półtora roku.

Obecnie menedżerowie coraz częściej zamawiają dla swych firm witryny internetowe, lecz mają problem z ustaleniem jaki zakres informacji o swej działalności chcieli by tam umieścić. Na korespondencję przychodzącą pocztą internetową często nie ma kto odpowiedzieć, lub jedyną odpowiedzią dla korespondenta jest sugestia, aby zwrócić się do

określonego pracownika firmy telefonicznie lub tradycyjnym listem. A przecież wszyscy menedżerowie wiedzą już, że informacja jest towarem, że rynek informacji nie może być bagatelizowany, niezależnie od rodzaju uprawianej działalności [NOWA97a, NOWA97b].

2. Co powinna zawierać witryna?

Zawartość witryny ściśle wyznacza jej przeznaczenie i adresat (adresaci). Przed jej opracowaniem powinniśmy jawnie wyartykułować zarówno cel jej tworzenia, jak i adresatów, do których ją kierujemy. Okazuje się, że na pytanie postawione o cel i adresatów wielu twórców witryn nie umie udzielić jednoznacznej odpowiedzi. Zazwyczaj odpowiadają, że materiał w witrynie służy promocji i jest przeznaczony dla wszystkich, którzy się nim zainteresują. Zdajmy sobie zatem sprawę, że choć sieć Internet ma charakter publiczny, co oznacza, że do utrzymywanych w serwerach informacji mogą potencjalnie mieć dostęp dowolne osoby (oczywiście w odniesieniu do zasobów ogólnodostępnych), to w praktyce stwierdzamy dwa rodzaje użytkowników tych zasobów, a mianowicie:

- ☺ użytkowników **zorientowanych** (ukierunkowanych) - to znaczy takich, którzy świadomie przeglądają zasoby Internetu, w poszukiwaniu informacji na wybrany temat oraz
- ☺ użytkowników **przypadkowych** - którzy losowo przeglądają zasoby Internetu i „wchodzą” w dany obszar informacji, gdy zostaną czymś zainteresowani (w ich mniemaniu wejście w dany temat wyda się interesujące).

Dla pierwszych zdobycie informacji ma określoną wartość (są one użyteczne w jakimś działaniu). Można w uproszczeniu przyjąć, że użytkownikom zorientowanym bardziej zależy na łatwości dotarcia do poszukiwanej informacji, niż jakichkolwiek formach zachęty do spenetrowania danej witryny. Im w szczególności przeszkadza zarówno nadmierna objętość i sekwencyjność stron, jak i „przerośnięta” forma graficzna, opóźniająca dostęp do oczekiwanych informacji. I przeciwnie użytkowników przypadkowych bardziej zazwyczaj frapuje atrakcyjność stosowanej zachęty do poznania danego obszaru (strona tytułowa, kolor, ruch, użyte narzędzia), niż sam temat. Jeśli planujemy jakiejkolwiek przedsięwzięcie internetowe, musimy zatem uświadomić sobie, do jakich użytkowników je adresujemy. Znajdzie to swoje odbicie w różnych atrybutach (charakterystykach) witryny, a przede wszystkim:

- w zakresie i szczegółowości prezentowanych informacji,
- we wprowadzeniu akcentów „nowinkarskich” (nadażność podawania informacji),

- w formie edycyjnej i narracyjnej tekstu (język, terminologia fachowa, styl),
- w atrakcyjności szaty graficznej (ilustracje, instrumenty „zachęty”),
- w strukturze stron i mechanizmach nawigacji po informacjach zgromadzonych w witrynie (poziomy struktury, odwołania, rozwinięcia tematów),
- w odwołaniach do innych witryn internetowych (pokrewnych tematycznie),
- w dynamice witryny - częstotliwości aktualizacji prezentowanych informacji oraz zasadach przechowywania informacji archiwalnych

Wśród użytkowników zorientowanych można (i trzeba podczas prac projektowych) ponadto wyróżnić określone grupy, które posiadają zazwyczaj wspólną orientację (charakterystykę, preferencję) zawodową, branżową, regionalną, wiekową, kulturową itp. Są to po prostu segmenty rynku informacji, które musimy jawnie określić. Znajomość takich preferencji pozwala precyzyjnie ustalić **adresata(ów) witryny**, a tym samym wyznaczyć zarówno zakres informacji „dedykowanych” określonej grupie użytkowników, jej szczegółowość, wymagania aktualizacyjne, ale także wymagania językowe (styl narracji), konwencje graficzne, instrumenty zachęty itp.

Z tego samego względu należy również spróbować jednoznacznie wyartykułować jaki jest **cel(e) opracowania witryny**. Pod dyskusję można tu poddać następujące cele:

- informacyjny (prezentacja i/lub doradztwo),
- popularyzacyjny (w odniesieniu do działań, zamierzeń, przedsięwzięć, środowiska itp.),
- dydaktyczny (upowszechnienie określonej wiedzy),
- okolicznościowy (informacja wynikająca z określonego zdarzenia),
- propagandowy (propagowanie inicjatyw, lansowanie poglądów, idei, poszukiwanie sprzymierzeńców itp.),
- marketingowy (usług, produktów, osób), komercyjny,
- rozrywkowy.

Dążenie do realizacji poszczególnych celów wyznacza z jednej strony zakres i sposoby prezentacji informacji, ale także rzutuje na pozostałe przedstawione charakterystyki witryny.

3. Jakie są główne mankamenty witryn?

Lista zastrzeżeń jakie można wysuwać pod adresem wielu witryn jest niestety bardzo obszerna. Warto chyba wymienić te najczęściej spotykane, aby tym samym pomóc autorom w udoskonaleniu ich produktów. Sądzę, że wyraźnie można podzielić te zastrzeżenia na związane

merytorycznie z prezentowanymi informacjami - pod tym względem można rozpatrywać trzy grupy zastrzeżeń dotyczące zakresu, szczegółowości i aktualności prezentowanych informacji - oraz zastrzeżenia dotyczące formy czyli sposobu ich prezentacji (w tym także umiejętnego wykorzystania możliwości stosowanych narzędzi programowych). Oczywiście obserwowane problemy są częstokroć silnie powiązane, bowiem wadliwa prezentacja może sprawić, że ocena merytoryczna dostępnej informacji będzie negatywna.

Pierwsza grupa zastrzeżeń dotyczy zakresu informacji umieszczonych w witrynie, szczególnie często możemy zauważyć, że:

- prezentowana informacja nie ma właściwie swego adresata, to co przedstawiono bardziej użyteczne jest samym autorom witryny, niż jawnie wyartykułowanym odbiorcom,
- wiele z przedstawionych informacji nie wnosi żadnych nowych (poznawczych) treści,
- brak jest istotnych informacji, które wiele z osób uznałoby za istotne (potrzebne, np. godziny otwarcia placówki banku),
- autorzy witryny nie są właściwie zainteresowani gromadzeniem informacji o użyteczności opracowanej przez siebie witryny (nie gromadzą tych informacji)

Druga grupa zastrzeżeń wiąże się ze szczegółowością informacji, a podstawowymi mankamentami są:

- nieporównywalny poziom uszczegółowienia różnych grup informacji, zebranych we wspólnej witrynie (jedne z informacji są nadmiernie uszczegółowione, inne podane bardzo ogólnikowo),
- niepełne rozwinięcie struktury informacji zdefiniowanej na wyższych poziomach,
- uszczegóławianie nieodpowiednich (z punktu widzenia potrzeb odbiorców) grup informacji - brak związku między zakresem i wagnością informacji, a ich szczegółowością,
- brak wewnętrznych odwołań, pomiędzy informacjami szczegółowymi (ten aspekt dotyczy już bardziej formy).

Trzecia grupa mankamentów dotyczy aktualności informacji,

- wiele z przedstawionych informacji okazuje się być nicaktualnymi, wiele ma charakter archiwalny, często brak informacji aktualnych (np. podany numer telefonu dawno został zmieniony),
- podana data ostatniej aktualizacji zazwyczaj źle świadczy o częstotliwości weryfikacji prezentowanych informacji
- nadużywanie hasła „new” wobec braku dostatecznie częstej aktualizacji.

Zastrzeżenia dotyczące formy prezentacji to głównie:

- zbyt długie ciągi tekstowe (ponad 3 strony maszynopisu bez podziału strukturalnego),
- nadmiernie zapelniona strona (brak „światła” w tekście),
- nadmiar dużych zdjęć, włączonych do głównej strony witryny,
- zbędne „wodotryski” elementów animowanych,
- mało czytelny tekst (zbyt mała czcionka, zbyt złożone graficznie tło),
- chaotyczna szata graficzna (brak jakiegokolwiek konwencji),
- brak konsekwencji w stosowaniu standardów polskich liter,
- nadmiar elementów „w budowie” - stan ten trwa wielokrotnie bardzo długo.

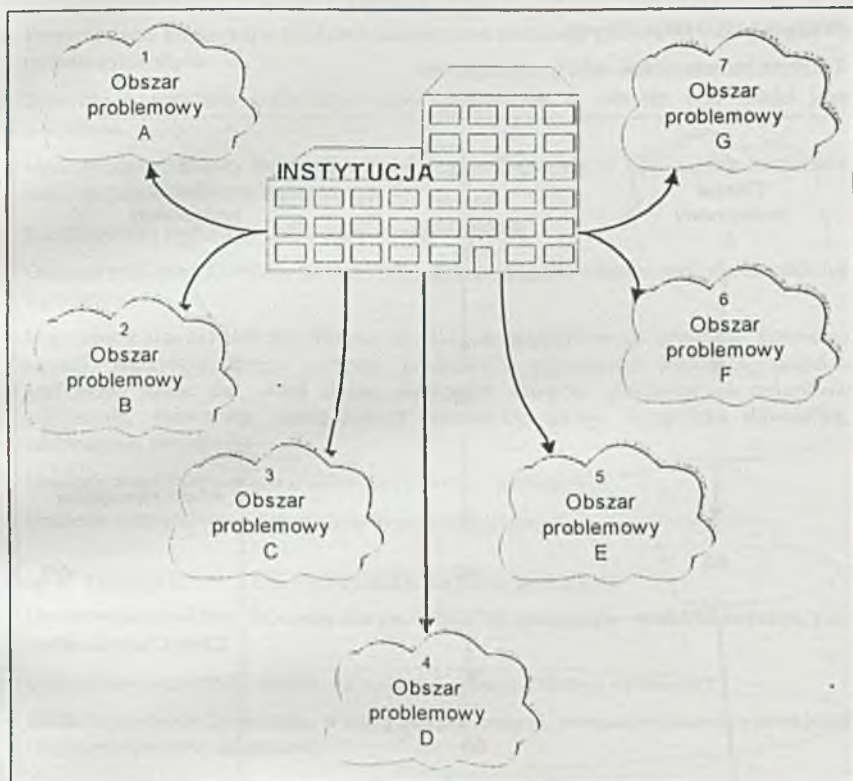
4. Jak powinna powstawać witryna?

Zakładamy, że witryna jest opracowaniem dość obszernym, które chcemy przygotować w dość krótkim czasie, tak aby prezentowane informacje nie uległy dezaktualizacji. Opracowanie powinno być zatem przedsięwzięciem zbiorowym, w którym uczestniczyć będzie określony zespół autorski, kierowany przez redaktora. Podczas całego przebiegu prac szczególnego znaczenia nabierają zarówno uzgodnienia merytoryczne (dotyczące prezentowanych problemów - informacji), jak i formalne (związane ze sposobem realizacji prac - warsztatowe). Redaktor witryny odpowiada za organizację i koordynację całości działań.

Proces tworzenia witryny ma zazwyczaj charakter iteracyjny. Wynika to z faktu, że w witrynie prezentujemy informację ze stanem aktualności na określony dzień. Ponieważ w praktyce zazwyczaj trudno jest uchwycić wszystkie (ważne lub błahę) zdarzenia, które powodują, że witryna już nie jest w pełni aktualna i dokonywać bieżących aktualizacji, należy ustalić zasadę jej okresowej, cyklicznej aktualizacji. W procesie tworzenia witryny można wyróżnić sześć etapów, a mianowicie (rys. 1) [1.UKA96]

- sformułowanie zadania,
- analiza i modelowanie witryny,
- opracowanie projektu witryny,
- przygotowanie testowej wersji witryny,
- testowanie i ocena witryny,
- włączenie witryny do zasobów sieci INTERNET.

Pierwsze trzy etapy mają charakter kreatywny. Ich przebieg, inwencja twórcza zespołu i przyjęte ustalenia, warunkują powodzenie całego przedsięwzięcia. Należy zatem poświęcić im dostatecznie dużo zaangażowania i energii. Warto zwrócić uwagę na iteracyjny charakter etapów 1 - 3, a zwłaszcza etapu analizy i modelowania. Im rozleglejsze przedsięwzięcie i liczniejszy zespół autorski, tym staranniej należy przygotować model i projekt, a wszystkie poczynione uzgodnienia, ustalenia, zalecenia, standardy itp. udokumentować pisemnie.

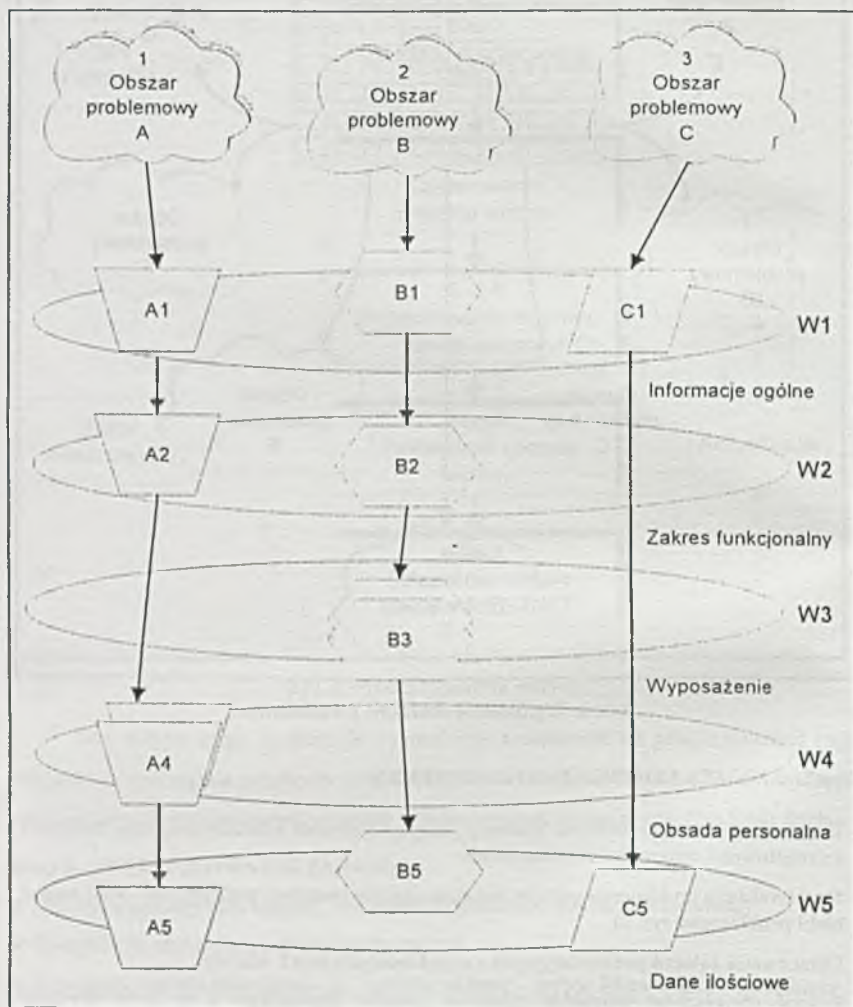


Rys. 2. Wyróżnienie obszarów problemowych

Etap 2. ANALIZA I MODELOWANIE WITRYNY

- **Burza mózgów** - wypracowanie koncepcji witryny. Ustalenie adresata(ów), celu, zakresu i szczegółowości materiałów informacyjnych.
- **Strukturalizacja problemowa witryny** (ustalone obszary i warstwy problemowe - wyróżnione bloki problemowe rys. 3).
- **Opracowanie założeń prezentacyjnych i zasad realizacyjnych witryny.**
- **Podział obowiązków** pomiędzy członków zespołu autorskiego i w razie potrzeby rozszerzenie składu zespołu.
- **Ustalenie narzędzi** do realizacji witryny oraz formy i postaci przygotowania materiałów do opracowania projektu witryny.
- **Określenie „dnia zerowego”** (stan na dzień) dla materiałów zamieszczanych w witrynie.
- **Wskazanie kategorii informacji** podatnych na dezaktualizację.

- Wyznaczenie progu poufności gromadzonych informacji
- Wstępna autoryzacja witryny.
- Ustalenie harmonogramu dalszej realizacji prac



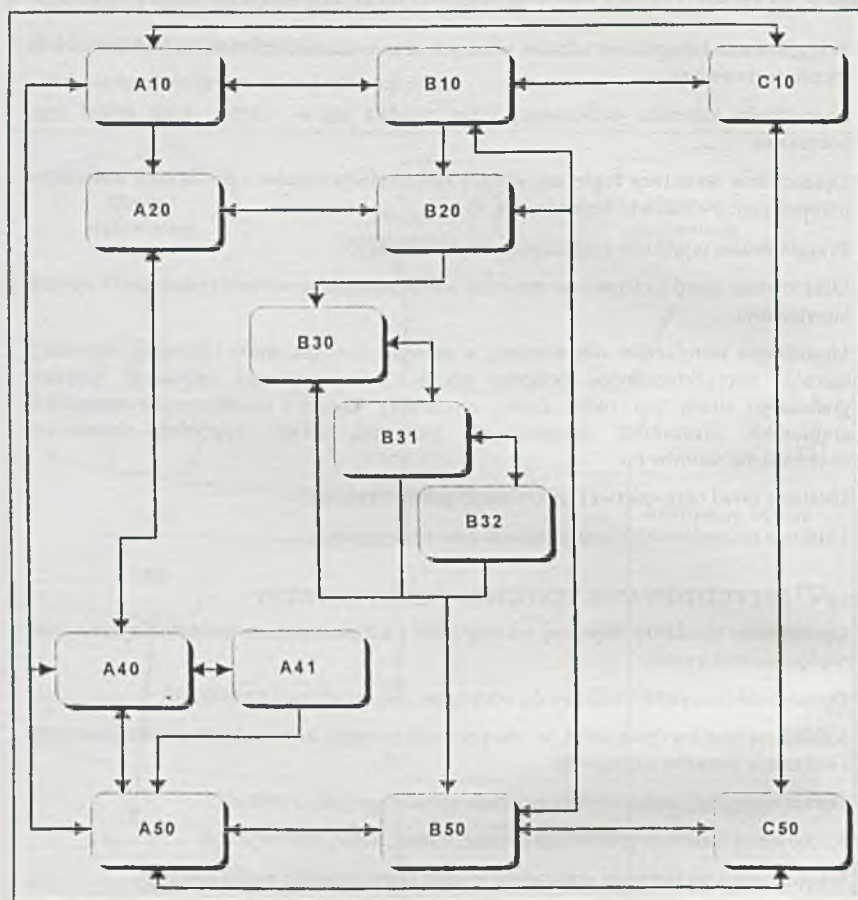
Rys. 3. Strukturalizacja problemowa witryny -
przykładowe obszary, warstwy i bloki problemowe

Etap 3. OPACOWANIE PROJEKTU WITRYNY (PREZENTACJI INSTYTUCJI)

- Przygotowanie konspektów tekstów własnych oraz wskazanie zakresu i źródeł pozyskania tekstów cytowanych.
- Specyfikacja materiału graficznego, który znajdzie się w witrynie oraz źródeł jego pozyskania.
- Opracowanie struktury logicznej witryny (segmentacja bloków i powiązania materiałów informacyjnych - makieta logiczna, rys. 4).
- Przygotowanie projektów graficznych witryny i stron(y).
- Opracowanie zasad nawigacji po materiale udostępnianym w witrynie (odwołania i słownik hiperlinków).
- Uzgodnienie standardów dla witryny, w zakresie: szczegółowości informacji, konwencji narracji, nieprzekraczalnego poziomu poufności i prywatności informacji, projektu graficznego strony (np. układ strony, wyróżniki), kategorii umieszczanych materiałów graficznych, elementów nawigacyjnych (przyciski, teksty, symbole), słownictwa, nazewnictwa, skrótów itp.
- Ustalenie zasad nazewnictwa (symbolizacji) plików i katalogów.
- Ustalenie szczegółowego harmonogramu prac realizacyjnych.

Etap 4. PRZYGOTOWANIE TESTOWEJ WERSJI WITRYNY

- Opracowanie struktury fizycznej witryny (pliki i ich powiązania - makieta fizyczna, por. rys. 5).
- Opracowanie wszystkich tekstów do witryny (w postaci plików i wydruków).
- Kolaudacja tekstów (połączenie, w miarę potrzeb reedycja, ewentualne usuwanie powtórzeń i wskazanie potrzeby uzupełnień).
- Opracowanie i/lub zeskanowanie materiału graficznego (pliki i wydruki)
- Koordynacja materiału graficznego (postać, format, kolor, inne atrybuty).
- Wkomponowanie materiału graficznego w tekst (z potrzebnymi modyfikacjami)
- Powiązanie plików w strukturalną całość (wzajemne odwołania pomiędzy segmentami) - wersja robocza witryny.
- Analiza kompletności materiału w stosunku do projektu (inventaryzacja braków, nadmiarów i odstępstw od ustaleń).
- Kontrola powołania źródeł i autoryzacji zgromadzonych materiałów.
- Korekty i uzupełnienia witryny.
- Ustalenie trybu testowania witryny i zasad ewidencjonowania własnych uwag.

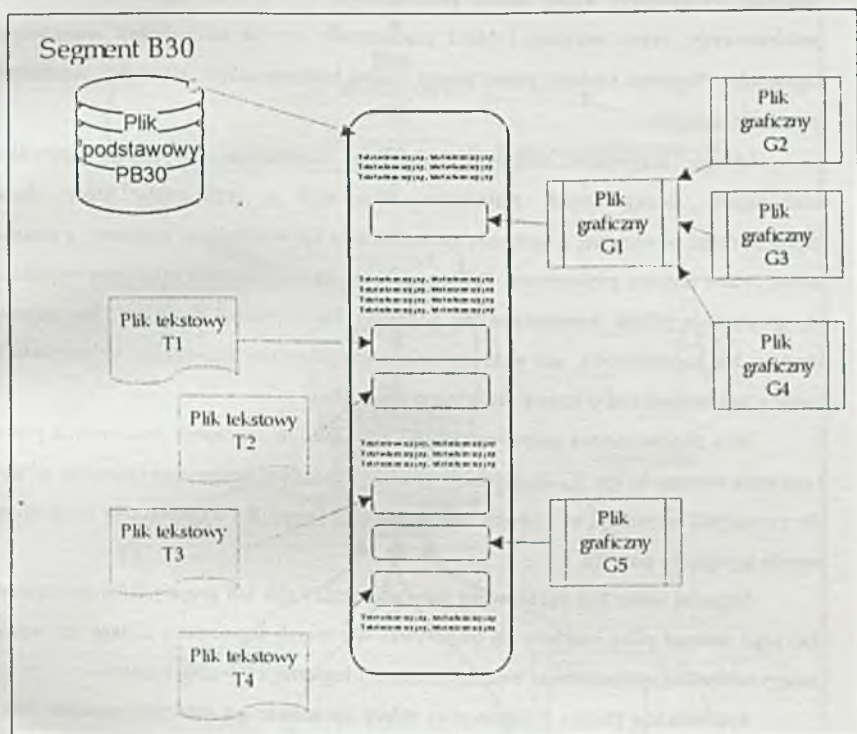


Rys. 4 Przykładowa segmentacja materiału - makieta logiczna

Etap 5. TESTOWANIE I OCENA WITRYNY

- Zweryfikowanie (przejrzanie/przeczytanie) całej witryny przez zespół autorski, wspólna ocena realizacji zamierzeń, wskazanie i uzasadnienie odstępstw od przyjętych założeń. Kontrola zachowania praw autorskich materiałów własnych i „zapożyczonych” (teksty i materiał graficzny)
- Zweryfikowanie witryny przez osoby spoza zespołu lecz związane (służbowo, formalnie, emocjonalnie) z przedmiotem prezentacji, ocena atrakcyjności, kompletności, szczegółowości, aktualności, stosowanej terminologii, poufności i prywatności dostępnych w witrynie informacji (ostateczna autoryzacja przez upoważnionego przedstawiciela Instytucji).

- Zewnętrzna ocena materiału zamieszczonego w witrynie z punktu widzenia czytelności, jednoznaczności, atrakcyjności treści i formy itp. Powinna być dokonana zarówno w odniesieniu do internetowej wersji witryny, jak i jej „treści” w postaci wydruków, przez osobę (osoby) spoza kręgu realizatorów zadania
- Korekta stwierdzonych nieprawidłowości



Rys 5 Przykładowy fragment makiety fizycznej (segment B30)

Etap 6. WŁĄCZENIE WITRYNY DO ZASOBÓW SIECI INTERNET

- Osadzenie witryny na serwerze
- Ustalenie cyklu aktualizacji witryny i osoby odpowiedzialnej za jej realizację (administrator witryny).
- Ustalenie zasad zgłaszania propozycji do modyfikacji i/lub rozbudowy witryny.

5. Jak strukturalizować materiał informacyjny?

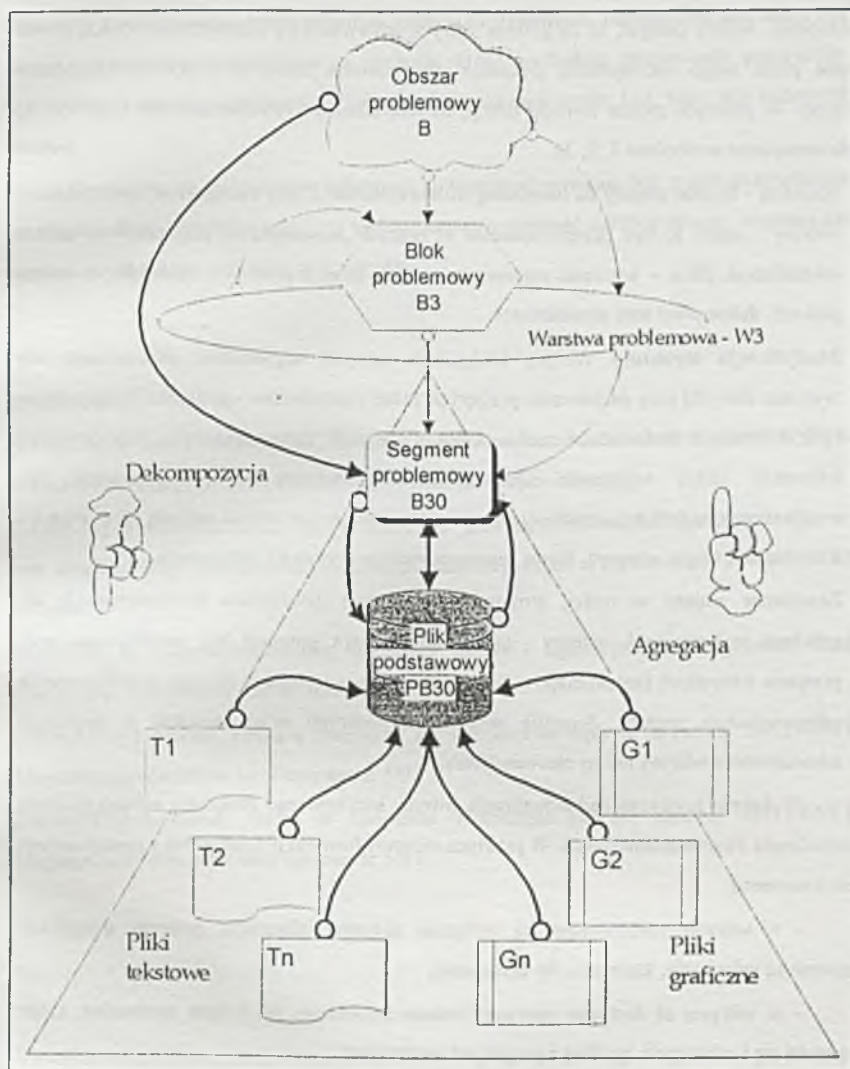
Od rozległości wyznaczonego zagadnienia, które będzie przedstawiane w witrynie, będzie zależeć ilość materiałów informacyjnych, które musimy zgromadzić i zaprezentować na stronach witryny. Jak przedstawiono powyżej najtrudniejszym zadaniem dla zespołu autorskiego jest inwentaryzacja wszystkich potrzebnych elementów. Działania takie wspomaga podejście strukturalne, które zaleca przechodzenie od ogólnie wytyczonych obszarów problemowych, przez warstwy i bloki problemowe, aż do szczegółowo wyróżnionych segmentów. Segment stanowi zatem pewną spójną logiczną całość, która jest przedmiotem prac realizacyjnych.

Istnieje oczywiście możliwość podejścia odwrotnego, to znaczy specyfikacji konkretnych, szczegółowych materiałów tekstowych i graficznych, które chcemy zaprezentować w witrynie, a następnie zszyntezowania ich w określone segmenty, a następnie nawet pewne obszary problemowe. Takie podejście jakkolwiek często stosowane w praktyce, nie gwarantuje jednak kompleksowego podejścia do prezentacji problemu. Nie zapewnia bowiem ani kompletności, ani nadzorowania szczegółowości prezentacji, nie pozwala też zazwyczaj panować nad planową aktualizacją materiałów.

Jako słuszne można natomiast przyjąć założenie, że w etapach kreatywnych procesu tworzenia witryny (etapy 1 - 3), działamy zgodnie z zasadą dekompozycji (przejście od ogółu do szczegółu) natomiast na etapach wykonawczych (etapy 4 i 5) odwrotnie czyli stosując zasadę agregacji - por. rys. 6.

Segment może być opracowany jako pojedynczy plik lub grupa plików powiązanych. Do tego samego pliku możemy się odwoływać w różnych segmentach. Z tego też względu należy oddzielnie symbolizować elementy struktury logicznej i fizycznej witryny.

Symbolizację plików (i katalogów) należy opracować po zinventaryzowaniu zakresu materiałów (zarówno tekstowych, jak i graficznych), które zidentyfikujemy opracowując strukturę logiczną witryny. W symbolizacji tej możemy (lecz nie musimy) wykorzystać ustalone wcześniej zasady symbolizacji elementów struktury logicznej. Dla ułatwienia i czytelności prac organizacyjnych w procesie tworzenia witryny korzystne jest prowadzenie dwuczęściowego identyfikowania wszystkich elementów, to znaczy stosowanie zarówno symbolizacji, jak i rozwiniętych nazw (tytułów) poszczególnych elementów. Posługiwanie się pełnymi nazwaniami jest naturalniejsze w kreatywnych etapach prac oraz w kontaktach ze zlecającym.



Rys. 6. Dekompozycja i agregacja materiału informacyjnego

6. Kiedy kończy się praca nad witryną?

Jeśli myślimy o witrynie związanej z jakąkolwiek działalnością gospodarczą, to musimy z góry założyć, że praca nad witryną nie ma końca, o ile nie kończy się ta działalność.

Aktualizacja witryny może być związana z działaniami o bardzo różnej skali złożoności. Należy przyjąć, że za gotową witrynę odpowiada jej administrator i tylko on lub osoba przez niego upoważniona dokonuje jakichkolwiek zmian w treści lub konstrukcji witryny. W praktyce można zwrócić uwagę na trzy charakterystyczne sytuacje (por. rys. 1, linie oznaczone symbolami 1, 2, 3):

1. **Korekta** - drobne zmiany na określonej stronie (stronach) przy zachowaniu całej struktury witryny - może to być przeprowadzone w postaci „kosmetycznej poprawki” w tekście (aktualizacja pliku - wymiana segmentu) i takich działań powinien cyklicznie, w miarę potrzeb, dokonywać sam administrator.
2. **Modyfikacja struktury** witryny (dołączenie nowych segmentów, eliminowanie lub wymiana starych) przy zachowaniu przyjętych zasad i standardów - pożądane byłoby aby w tych działaniach brała udział osoba, która pierwotnie przygotowywała dany fragment informacji (autor segmentu). Obowiązkiem administratora jest dopilnowanie czy zrealizowane modyfikacje struktury nie naruszyły nawigacji w całości witryny (por. etap 5 - Testowanie i ocena witryny). Nowe segmenty witryny wymagają autoryzacji.
3. **Zasadnicze zmiany** w treści, strukturze, przyjętych założeniach prezentacyjnych lub zasadach realizacyjnych witryny - taka **modernizacja** powinna być zrealizowana przy przejściu wszystkich sześciu etapów tworzenia witryny i powołaniu do tego przedsięwzięcia odpowiedniego zespołu. Sugestia modernizacji witryny może powstać z inicjatywy administratora witryny lub jej zleceniodawcy.

W świetle rozważań nad aktualizacją witryny szczególnego znaczenia nabiera problem gromadzenia danych archiwalnych. W praktyce możemy bowiem przyjąć jedną z następujących dwu konwencji:

- w witrynie prezentowane są wyłącznie aktualne informacje, podczas aktualizacji usuwane są informacje, które utraciły aktualność,
- w witrynie są dostępne zarówno bieżące informacje, jak i dane archiwalne, które gromadzi się i udostępnia zgodnie z przyjętymi ustaleniami.

Pozornie wydawać by się mogło, że jedynie pierwsze z rozwiązań znajduje praktyczne uzasadnienie. Niemniej jednak można wskazać wiele przykładów praktycznych, w których informacja archiwalna posiada szczególne znaczenie (dla niektórych użytkowników większe, niż informacja aktualna). Ilustracją takiej sytuacji może być informacja o osobach zajmujących określone, kadencyjnie obsadzone stanowisko w prezentowanej instytucji. Otóż poza

informacją na temat osoby, która aktualnie zajmuje to stanowisko, użytkownik sieci może być zainteresowany także informacjami o osobach, które tę funkcję sprawowały poprzednio. Oczywiście zakres (szczegółowość) informacji archiwalnej może być inny, niż informacji bieżącej

Zauważmy, że gromadzenie informacji archiwalnych powinno być z góry przewidziane (na etapie analizy i modelowania) i że archiwum może stanowić odrębny obszar, warstwę lub segment problemowy projektowanej witryny.

Literatura

- [LUKA96] B. Łukasik-Makowska, Założenia projektowe do tworzenia witryn prezentacyjnych w sieci INTERNET, OI-TBD Gdańsk 1996 (maszynopis powielony)
- [LUKA97] B. Łukasik-Makowska, INTERNET jako narzędzie marketingu bankowego, w: mat. konferencyjne „Zastosowania rozwiązań informatycznych w bankowości” PN nr 766 AE Wrocław 1997
- [LUKA98] B. Łukasik-Makowska, Skarbnica wiedzy - Ogólnopolski Zintegrowany Bank Edukacyjny, ComputerWorld nr 3/98
- [NOWA97a] L. Nowak, Firma w Internecie. Jak sieć Internet wpływa na biznes? INTERNET. Magazyn użytkowników sieci Internet nr 5/97
- [NOWA97b] L. Nowak, Firma w Internecie. Rekrutacja poprzez Internet INTERNET. Magazyn użytkowników sieci Internet nr 6/97



O firmie Sun Microsystems

Firma Sun Microsystems została założona w lutym 1982 roku w Kalifornii, na Uniwersytecie Stanford, a jej nazwa pochodzi od skrótu Stanford University Network. Od samego początku działalności oferta firmy była ukierunkowana na tworzenie niezawodnych systemów wspomagających przetwarzanie sieciowe. Konsekwentna realizacja założonej strategii spowodowała, że w ciągu kilku lat firma weszła do ścisłej czołówki rynku informatycznego. Pod koniec lat 80-tych firma Sun rozpoczęła ekspansję na rynki zagraniczne. W Szkocji otwarto drugi zakład produkcyjny firmy. Produkty firmy dostępne są w ponad 150 krajach na wszystkich kontynentach, przy czym sieć biur handlowych firmy obejmuje 42 kraje.

Do niedawna firma Sun Microsystems głównie znana była ze swoich stacji roboczych. Przez ostatnie kilka lat Sun umocnił swoją pozycję na rynku serwerów, a nowe rozwiązania oparte na technologii Java pozwalają firmie dotrzeć do indywidualnego użytkownika.

Misja firmy

Od momentu powstania w 1982 roku firma Sun Microsystems realizuje swoje hasło "Dopiero Sieć to Komputer", dostarczając na rynek kompleksowe rozwiązania sieciowe: niezawodne serwery, wydajne stacje robocze, macierze dyskowe, oprogramowanie oraz usługi serwisowe, pomoc techniczną i profesjonalne usługi w zakresie planowania, wdrażania i eksploatacji środowisk sieciowych.

Klasycznymi już produktami firmy Sun są niezawodne i skalowalne serwery dla przedsiębiorstw z rodziny Enterprise, wydajne stacje robocze Ultra oraz serwery internetowe Netra. Najnowsze serwery dla grup roboczych, Sun Enterprise 450, Sun Enterprise 10S oraz 5S, oferują niezwykłą funkcjonalności będąc przy tym cenowo konkurencyjny z serwerami klasy PC.

Nowy paradygmat języka Java - "aplikacja raz stworzona może pracować zawsze, wszędzie i na dowolnej platformie sprzętowej" - rewolucjonizuje pracę w sieciach komputerowych przedsiębiorstw. Strategia Java Computing oferuje przedsiębiorstwom bezpieczeństwo, łatwość tworzenia i wdrażania aplikacji oraz zgodność pomiędzy platformami, obniżając jednocześnie całkowite koszty operacyjne oraz koszt jednego stanowiska pracy.

Pozycja na rynku światowym

Sun Microsystems to firma ogólnoświatowa, która około 50% obrotów osiąga poza Stanami Zjednoczonymi. Firma zatrudnia ponad 23,000 osób na całym świecie, z czego w europejskich ośrodkach firmy zatrudnionych jest 5000 pracowników, czyli ponad 20% całego zespołu Suna.

Wśród stale wzrastających wskaźników finansowych firmy na szczególną uwagę zasługują obroty, które w roku finansowym 1997 wzrosły o ponad 20% i wyniosły 8,6 mld dolarów, przy czym obroty na jednego pracownika osiągnęły poziom 398,000\$. Sun Microsystems znalazł się na 203 pozycji listy największych światowych przedsiębiorstw *Fortune 500*. Oznacza to "awans" o 19 pozycji w porównaniu z poprzednim rokiem finansowym. Warto podkreślić, że nakłady na badania i rozwój osiągnęły w firmie Sun Microsystems poziom blisko 1 miliarda dolarów i stanowią 9.6% całkowitych obrotów firmy.

Sun zajmuje na europejskim rynku pierwszą pozycję pod względem obrotów ze sprzedaży systemu operacyjnego UNIX (z 27% udziałem w rynku), jest także liderem w sprzedaży stacji roboczych (41.1%) oraz usług dla systemu UNIX (34%). Sun zajmuje natomiast drugie miejsce pod względem przychodów ze sprzedaży serwerów UNIX-owych - gdzie udział firmy w rynku przekracza 20%.

Najnowsze dane finansowe za II kwartał roku finansowego 1997/98 jeszcze bardziej podkreślają silną pozycję Suna na rynku. Obroty wyniosły prawie 2,5 mld dolarów, co oznacza wzrost w porównaniu z poprzednim rokiem finansowym o 18%. Zwraca uwagę 25-procentowy wzrost zysku po opodatkowaniu, który z drugim kwartale roku finansowego 97/98 wyniósł 223.2 mln dolarów.

Sun na rynku polskim

W Polsce firma obecna jest od 1990 roku, kiedy to oficjalnie sprzedano pierwszy serwer firmy Sun. W 1993 roku otwarto biuro handlowe firmy w Warszawie, a w 1996 roku firma została spółką prawa polskiego zarejestrowaną pod nazwą Sun Microsystems Poland Sp. z o.o.. Ważnym wydarzeniem w polskiej historii firmy było niedawne otwarcie, we wrześniu 1997 roku, drugiego polskiego biura przedstawicielskiego w Gliwicach.

W Polsce firma Sun rozwija się bardzo dynamicznie, osiągając w latach 1994-1997 średni roczny wzrost obrotów na poziomie ponad 50%. Według badań IDC, Sun zajmuje pierwsze

miejsce w sprzedaży stacji roboczych UNIX z udziałem w rynku na poziomie 45%, zaś 30-procentowy udział w rynku serwerów sytuuje firmę na miejscu drugim.

Do największych polskich klientów firmy Sun należą Telekomunikacja Polska S.A., Era GSM, Polkomtel, Netia, Polska Telewizja Kablowa, Zakłady Samochodowe Jelez, FIAT Auto Poland i Daewoo, bank BGŻ, Browary Wielkopolski LECIH, Polska Nafta i Gaz, większość polskich uczelni wyższych i wiele, wiele innych firm i instytucji.. W sektorze administracji państwowej do klientów firmy Sun należą między innymi Urząd Wojewódzki w Warszawie, Urząd Miejski w Gdyni, Państwowa Agencja Radiokomunikacyjna i Krajowa Komisja Wyborcza.

Rozwiązania Sun Microsystems dla Internetu i intranetu

W ciągu ostatnich kilkudziesięciu lat w zasadniczy sposób zmienił się sposób funkcjonowania sieci komputerowych przedsiębiorstw. Od modelu przetwarzania opartego na modelu mainframe, poprzez przetwarzanie klien/server przemysł informatyczny doszedł do modelu przetwarzania sieciowego. Firma Sun Microsystems posuwa się o krok dalej, promując rozwiązania sieciowe w których zawartość, a nie technologia zaczynają odgrywać kluczową rolę. Sun Microsystems nazwał taki model Webtone, czyli sygnał sieciowy. Dzięki niemu dostęp do sieci staje się powszechnie dostępną usługą, a technologia staje się dla użytkownika czymś niewidzialnym, co ułatwia korzystanie z sieci.

Takie środowisko opiera się na sieciach korporacyjnych, które wykorzystują technologię globalnej sieci Internet. Od momentu firma Sun Microsystems była liderem na rynku rozwiązań dla Internetu, a obecnie została liderem rozwiązań intranetowych. Sun Microsystems oferuje także rozwiązania pozwalające na stworzenie niezawodnej i bezpiecznej infrastruktury dla transakcji elektronicznych *electronic commerce*.

Jednym z klasycznych produktów firmy Sun Microsystems jest oprogramowanie serwera internetowego Netra, działającego w środowisku operacyjnym Solaris. W ofercie firmy Sun Microsystems dla usługodawców Internetu oraz dla firm wdrażających technologię internetową jest także:

- Środowisko operacyjne Solaris, wybrane w 1997 roku „najlepszym UNIXowym systemem operacyjnym” przez pismo *Network VAR*, zapewnia niezawodną obsługę usług Internetowych, zapewniając jednocześnie doskonałą skalowalność. Solaris zawiera także bogaty zestaw narzędzi do zarządzania siecią.
- Sun Bandwidth Allocator to rozwiązanie dla usługodawców Internetu, zapewniające kontrolę i optymalizację ruchu w sieciach opartych na protokole IP. W efekcie następuje redukcja wąskich gardeł sieci oraz obniżają się rachunki klientów providerów Internetu.
- Sun Internet Mail Server zapewnia przesyłanie wiadomości w sieciach intranet, przy czym wykorzystywać można wszystkie standardy poczty elektronicznej.
- Sun WebServer umożliwia stworzenie serwisu WWW i łatwą dystrybucję informacji w sieciach intranet.
- Solstice to pakiet narzędzi pozwalający na kompleksowe zarządzanie rozproszonymi środowiskami sieciowymi. W skład pakietu wchodzi narzędzia do sporządzania kopii

zapasowych, do zarządzania rozproszonymi zasobami dyskowymi. Zarządzanie siecią możliwe jest centralnie, z jednego terminala.

- Sun Directory Services pozwala na stworzenie rozproszonych, skalowalnych i łatwych do przeszukiwania systemów katalogowych.
- SunScreen to rodzina produktów zapewniających bezpieczeństwo w sieciach komputerowych przedsiębiorstw, oferujące kontrolę zdalnego dostępu, potwierdzanie autentyczności, kodowanie, przy czym zabezpieczony jest także serwer, na którym oprogramowanie pracuje.

Obecnie Sun Microsystems coraz intensywniej wkracza na rynek rozwiązań teleinformatycznych, które integrują sieci komputerowe i z sieciami Internet i intranet.. Nowe urządzenia, które już teraz są dostępne na rynku, pozwolą na jednoczesne przeglądanie Internetu, przysyłanie elektronicznych listów i rozmowy telefoniczne. Najnowsze technologie, w tym aplikacje oparte na języku Java firmy Sun Microsystems, pozwalają dodać usługi telefoniczne do istniejących sieci komputerowych.

Firma Sun Microsystems jest twórcą języka Java, który stał się standardem w Internecie oraz wewnętrznych sieciach intranet. Środowisko Java pozwala na tworzenie aplikacji, które mogą być uruchamiane na dowolnej platformie sprzętowej. Jednocześnie język programowania Java jest niezwykle zwięzły i łatwy dla programistów – także temu zawdzięczamy jej popularność.

Java wczoraj i dziś

Wprowadzenie

Jest wręcz niewiarygodne, że zaledwie 10 lat temu wizja firmy Sun Microsystems "Dopiero Sieć to Komputer" wydawała się tak nieprawdopodobna i utopijna, że najwięksi nawet entuzjaści informatyki zaledwie wzruszali na nią ramionami. Ale nie przeszkodziło to firmie Sun Microsystems realizować tę wizję. W tej chwili, u progu kolejnego tysiąclecia, wizja Suna staje się jeszcze bardziej realna dzięki Javie i strategii Java Computing.

Java przełamała tradycyjne bariery. Nie ma już znaczenia, kto jest producentem danej platformy sprzętowej i programowej, kto jest producentem języka programowania. Java jest uniwersalną platformą i uniwersalnym językiem. Do tej pory twórcy oprogramowania musieli spędzać ogromne ilości czasu i angażować wielkie środki aby pisać programy dla wielu

systemów. Teraz mogą pisać programy dla wielu platform jednocześnie, co zwiększa wydajność i obniża koszty.

Java jest wszechobecna. Została zaakceptowana przez miliony użytkowników i wiele setek tysięcy firm szybciej niż jakakolwiek inna technologia w historii informatyki. Java działa wszędzie i zawsze. Uzyskawszy aprobatę organizacji ISO, przed firmą Sun Microsystems otworzyła się możliwość standaryzacji Javy. Uzyskanie standardu ISO oznaczać będzie, że Java stanie się również formalnym standardem dla rozproszonych środowisk informatycznych.

Przyjrzyjmy się więc dokładniej co Java oferuje użytkownikom, przedsiębiorstwom oraz biznesowi.

Ale jak to się zaczęło?

Na początku lat dziewięćdziesiątych niewielka grupka inżynierów Suna, z inżynierem d/s oprogramowania Patrickiem Naughton'em na czele zauważyła, że coraz więcej urządzeń z jakich korzystamy w naszych codziennych zajęciach - toster, kuchenka mikrofalowa, zegarek, magnetowid, radio... - to urządzenia cyfrowe, sterowane coraz to bardziej skomplikowanymi mikroprocesorami. Producenci zaś nie myślą o tym, aby urządzenia były ze sobą zgodne. Ich zadaniem jest zbudowanie sprzętu najbardziej funkcjonalnego, który najlepiej spełnia potrzeby klientów. Każdy z nich bazuje na innym dostawcy sprzętu, systemów operacyjnych i aplikacji. Pytanie jakie e postavili sobie inżynierowie brzmiało: co zrobić, aby te urządzenia mogły dzielić się informacjami? Jak zapewnić bezpieczeństwo, niezawodność i jak rozwiązać problemy z siecią?

Inżynierowie zbadali budowę szeregu urządzeń i zmodyfikowali ją tak, aby urządzenia mogły się ze sobą komunikować i przenosić między sobą informacje. Taka budowa urządzeń zmniejszyłaby problemy producentów i rozszerzyła możliwości konsumentów w zakresie integrowania urządzeń. Jednakże istniejące, tradycyjne języki programowania okazały się nieprzydatne do takiego przedsięwzięcia - były zbyt skomplikowane, a programy wymagały dużych mocy obliczeniowych. Inżynierowie postanowili stworzyć coś nowego: prostego i nowoczesnego. Tak narodziła się Java.

Jednakże dopiero sieć WWW stała się	... dni temu	co się stało?
prawdziwym przełomem w historii Javy. Java	1000	Wprowadzenie języka Java
ożywiła martwą grafikę stron Internetu.	900	Pierwsze narzędzia
Szybko okazało się, że Java dysponuje	800	Zainicjowanie działania JavaSoft
potencjałem daleko większym, niż	700	Platforma Java jest częścią każdego SO
przewidywali to jej ojcowie-założyciele.	600	200,000 programistów w Javie
Doskonałość Javy leży po części w jej	500	Wprowadzenie JDK 1.1
prostocie. Ale Java daje także programistom	400	10,000 uczestników na JavaOne '97
niezależność od platformy i obniża koszty	300	Ponad 450,000 programistów w Javie
tworzenia aplikacji. Użytkownicy korzystają	100	Ponad 750,000 programistów w Javie
z łatwej możliwości uaktualniania wersji programu i otrzymują możliwość pracy w	30	14,000 uczestników na JavaOne '98
bezpiecznym i stabilnym środowisku.		<u>1000 dni Javy!</u>

z łatwej możliwości uaktualniania wersji programu i otrzymują możliwość pracy w bezpiecznym i stabilnym środowisku.

Potencjał Javy okazał się jednak jeszcze większy. Język został zaadaptowany do tworzenia aplikacji sieciowych, które mogą być uruchamiane w ramach całej sieci. Wyceliniowało to praktycznie koszty dystrybucji oprogramowania wśród użytkowników sieci, co pozwala firmom przeznaczać więcej środków i czasu na tworzenie podstawowych produktów.

W ten sposób Java stała się otwartą, uniwersalną platformą dla przetwarzania sieciowego, która obejmuje całą organizację i wszystkie urządzenia - od telefonów, aż po aplikacje o najwyższej niezawodności.

Co to tak naprawdę jest Java?

Najkrócej rzecz ujmując środowisko Java pozwala na tworzenie aplikacji, które mogą być uruchamiane na dowolnej platformie sprzętowej. Jednocześnie jednak choć język programowania Java jest niezwykle zwięzły i łatwy, to miał on ogromny wpływ na sieć Internetu - oraz na ogólne funkcjonowanie przedsiębiorstw.

Na początku więc Java ożywiła strony WWW, dodając do nich dynamiczne, interaktywne aplikacje. Szybko jednak, dzięki powstaniu wielu rozszerzeń API, te proste aplikacje zmieniły się w kompleksowe systemy, które wykorzystują nowy paradygmat nazywany Java Computing. Dzięki strategii Java Computing zarządzanie i kontrola sieci staje się znacznie łatwiejsza. Dystrybucja oprogramowania dokonywana jest bardzo szybko i praktycznie bez ponoszenia kosztów. Przedsiębiorstwo otwiera sobie drogę na rynek transakcji elektronicznych *electronic commerce*. Spadają koszty operacyjne. Aplikacje i dane dostępne są wszędzie w ramach sieci.

Java bazuje na prostej przesłance, że wszystkie mikroprocesory powinny porozumiewać się we wspólnym języku, że wszystkie produkty oparte na procesorach powinny móc ze sobą łatwo i harmonijnie współpracować oraz dzielić się danymi. Już dziś zmieniło to sposób pracy w Internecie (kiedy użytkownik otrzymuje dostęp do informacji niezależnie od tego na jakiej platformie pracuje), zaś teraz zmienia sposób działania produktów użytkowych i przedsiębiorstw.

Dzięki Javie nie trzeba rezygnować z obecnie posiadanych platform. Java jest przecież niezależna od platformy. Java wychodzi też poza przedsiębiorstwo, czyniąc dane i aplikacje dostępnymi poprzez komputery przenośne, interaktywne kioski, telewizję kablową, telefony komórkowe i wiele, wiele innych urządzeń. Przedsiębiorstwa i użytkownicy, twórcy oprogramowania i ich użytkownicy znaleźli tysiące zastosowań dla Javy - część z nich zostanie przedstawiona w dalszej części tekstu. Wszyscy jednak odnoszą korzyści dzięki podniesieniu niezawodności, bezpieczeństwa oraz oszczędności czasu i środków.

Czym jest Java dla programistów?

Java to o wiele więcej niż tylko język programowania. Java to kompleksowe środowisko obejmujące wnętrze i otoczenie firmy. Java Computing to także filozofia pracy w sieci. W skład środowiska Java wchodzi narzędzia programistyczne, system operacyjny, oprogramowanie dla serwera, klient dla komputerów sieciowych i standardowych komputerów aż po narzędzia zarządzania siecią, pomoc techniczną, szkolenia, usługi konsultingowe oraz usługi integracyjne.

Dzięki tym kompleksowym narzędziom zarówno tworzenie aplikacji w Javie, jak i wdrażanie i utrzymanie środowisk Java jest niezwykle proste. Kluczowymi elementami środowiska Java Computing są:

- Java Development Kit (JDK) - środowisko programistyczne które znacznie ułatwia tworzenie aplikacji w Javie,
- JavaOS - prosty system operacyjny, który pozwala na uruchamianie aplikacji Java na szeregu urządzeniach: od komputerów sieciowych po telefony komórkowe.
- Java Virtual Machine - podstawowe oprogramowanie, wbudowane w system operacyjny komputera, lub uruchamiane w istniejącym systemie (Windows, Macintosh czy UNIX), które pozwala na uruchamianie aplikacji Java.

- Java API - rozszerzenia języka Java stanowiące interfejs pomiędzy aplikacjami Java, a różnymi usługami takimi jak przeszukiwanie bazy danych czy systemy zabezpieczeń.
- Java Beans - szeroki wybór bezpiecznych, skalowalnych komponentów, pracujących na różnych platformach, które służyć mogą do budowy aplikacji.
- Java WorkShop - narzędzia do tworzenie aplikacji w Javie.
- Szkolenia Javy - kompleksowy cykl szkoleń, multimedialnych kursów prowadzonych przez instruktorów, które uczą jak tworzyć aplikacje Java. Szkolenia obejmują także proces wydawania certyfikatów potwierdzających biegłość w Javie.
- Solaris i stacje robocze Ultra - to najlepsze, najszybsze i najwydajniejsze środowisko dla programistów. Umożliwiają stworzenie jednolitego i łatwego do przenoszenia kodu.

Java oferuje szeroki wybór możliwości tworzenia oprogramowania. Na rynku dostępnych jest ponad 300 narzędzi, umożliwiających tworzenie aplikacji od podstaw, można zakupić ponad 100 aplikacji, które dostosowywać można do potrzeb firmy. Jednocześnie szybko rozwija się rynek narzędzi, umożliwiających szybkie tworzenie aplikacji poprzez budowę ich z gotowych komponentów JavaBeans.

Nowy paradygmat, że aplikacja raz napisana może być uruchamiana zawsze, wszędzie i na dowolnej platformie sprzętowej sprawia, że całkowicie zmienia się idea tworzenia oprogramowania. Dawniej programiści pisali programy dla konkretnej kombinacji sprzętu i systemu operacyjnego - a więc program napisany dla Windows nie pracował pod UNIXem. Java burzy te bariery. Teraz programiści tworzą jednolity kod, który działa na wszystkich platformach. Oznacza to ogromne oszczędności czasu i środków.

Czym jest Java dla użytkownika?

Internet daje nam nieograniczone zasoby pozwalające na dzielenie się informacjami, współpracę i przeprowadzanie transakcji w sieci. Java pozwala użytkownikom i twórcom aplikacji koncentrować się na meritum ich pracy, a nie na infrastrukturze.

Użytkownik otrzymuje dostęp do tych samych aplikacji oraz do swoich danych z dowolnego miejsca i z dowolnego urządzenia - czy to w przedsiębiorstwie, czy też poza nim. Aplikacje nie działają przecież na konkretnej maszynie, ani nawet platformie. Użytkownicy nie muszą być więc twarde przypisani do konkretnego stanowiska pracy.

Użytkownicy mają dostęp do aplikacji z dowolnego urządzenia: komputera PC, Macintosh, stacji roboczej i serwera UNIX, komputera sieciowego, asystenta osobistego (PDA), telefonu

komórkowego oraz wszelkich innych urządzeń, pod warunkiem, że wyposażono je w przeglądarkę Javy.

Java zmienia cały obraz przedsiębiorstwa, zmienia także sposób pracy jego personelu. Dotychczasowy model sieci polegał na występowaniu "wysp", na których oddzieleni byli użytkownicy i ich systemy, a z których możliwy był dostęp do danych i aplikacji. Java Computing realizuje jedną z idei firmy Sun Microsystems - ideę sygnału sieciowego WebTone. WebTone to stała dostępność sieci komputerowej, aplikacji i danych. Java unifikuje "wyspy" sieci w jedną całość tak, że dostęp do sieci nie ma barier czasowych i geograficznych.

Tak więc użytkownicy nie muszą zajmować się infrastrukturą sprzętową i technologią programową stojącą za aplikacjami. Dzięki Javie możliwy jest elastyczny dostęp do aplikacji, bez konieczności administrowania systemem.

Czym jest Java dla przedsiębiorstwa?

Nie jest przesadą stwierdzenie, że strategia Java Computing zmienia sposób działania światowego biznesu. Java łączy elementy scentralizowanego i rozproszonego przetwarzania sieciowego.

Korzyści scentralizowanego przetwarzania to niski koszt, wysoka niezawodność, zdalne administrowanie oraz scentralizowany system pomocy technicznej. Systemy rozproszone oferują użytkownikom wydajne komputery na biurkach, które umożliwiają szybkie wykonywanie aplikacji. Java łączy te dwa podejścia i umożliwia zastosowanie każdego z nich tam, gdzie najlepiej będzie spełniało swoje zadanie. Oznacza to nie tylko oszczędności w zakresie inwestycji w systemy komputerowe, ale oferuje też bardziej elastyczną architekturę. Przedsiębiorstwo może wybrać "odchudzone", niedrogie komputery sieciowe, które jedynie pobierają i wykonują applety Javy, podczas gdy obliczenia i przechowywanie danych odbywa się na serwerze. Jednakże w innych miejscach przedsiębiorstwa można zainstalować stacje robocze i komputery PC, które będą posiadały moc do przeprowadzania lokalnych obliczeń, czy projektowania graficznego.

Java to także łatwiejszy i bardziej elastyczny dostęp do istniejących systemów, takich jak bazy danych typu mainframe. Za pomocą rozszerzeń API, programiści mogą budować

aplikacje, które będą miały pełny dostęp do istniejących zasobów przedsiębiorstwa. W ten sposób możliwy jest dostęp do danych w ramach korporacyjnej sieci intranetowej, jak i autoryzowany dostęp z zewnątrz, z sieci Internet. Otwarcie na dostęp zewnętrzny oznacza zaś lepsze wykorzystanie informacji, lepsze słuźenie klientom i obniżanie kosztów.

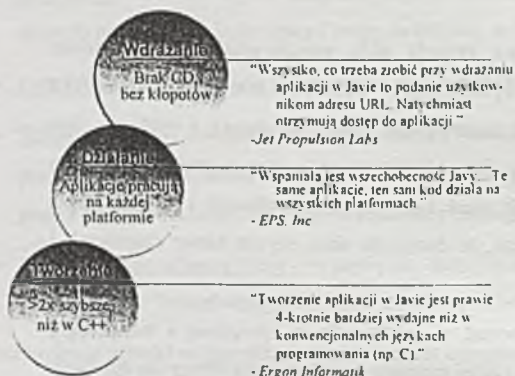
Java Computing rozwiązuje

jeszcze jeden problem. Środowiska PC oferowały elastyczność i niezależność użytkowników, jednak z biegiem czasu były coraz trudniejsze i droższe do administrowania. Według badań Gartner Group stanowisko PC kosztuje firmę 10,000\$ rocznie, wliczając w to jedynie koszty administrowania. Java natomiast, umożliwia zdalne administrowanie i scentralizowany system pomocy. Pozwala to znacznie łatwiej zarządzać środowiskiem sieciowym. Jednakże prawdziwe korzyści osiąga się podczas uaktualniania oprogramowania. Administrator wykonuje upgrade raz - na serwerze, a nie na każdej pojedynczej maszynie. Oszczędności idą tu już w setki tysięcy dolarów. Dlatego też koszt administrowania stanowiskiem komputera sieciowego nie przekracza 1.200\$ rocznie.

Czym jest Java dla biznesu?

Java nie jest przypisana do przedsiębiorstwa, gałęzi przemysłu, kraju czy kontynentu. Java to technologia faktycznie transnarodowa. Rozwiązania wykorzystujące strategię Java Computing powstają na całym świecie.

Implikacje wprowadzenia Javy do transakcji elektronicznych to rozszerzenie bazy sprzedaży, reklamy i promocji oraz transakcji na cały świat. Ożywione, interaktywne dane, dostępne gdziekolwiek w sieciowym świecie przybliżają firmę do klientów. Dane, które mogą być centralnie i szybko aktualizowane i rozprowadzane wśród klientów dają gwarancję pełnej satysfakcji klienta.



Java w urządzeniach powszechnego użytku

Oryginalnie Java zaprojektowana była jako język sterujący urządzeniami. Dlatego też ma pewne szczególne cechy, które bardzo dobrze sprawdzają się w urządzeniach powszechnego użytku.

- bezpieczeństwo - brak bezpośredniego dostępu do pamięci i tworzenie bezpiecznego, tworzy przy tym bezpieczne, zamknięte środowisko w którym działa aplikacja.
- przenośność kodu - możliwe jest wielokrotne użycie tego samego kodu, Java działa przy tym na różnych platformach sprzętowych
- oszczędność pamięci - dzięki dynamicznemu tworzeniu i wczytywaniu nowych klas urządzenie nie musi posiadać rozbudowanej pamięci,
- zdalna obsługa - możliwe jest zdalne wprowadzanie poprawek do oprogramowania oraz dynamiczne linkowanie i wczytywanie aplikacji.

Pare gadżetów...

Obok poważnych rozwiązań Java to także niezliczone pseudo-zabawki. Wszystkie te rozwiązania, choć dziś służą jedynie rozrywce, kiedyś staną się podstawą do stworzenia nowych urządzeń, bez których nie będziemy potrafili wyobrazić sobie pracy w przedsiębiorstwach.

Wyobraźmy sobie pierścień, który pozwala na otwieranie zamków w drzwiach, taki "klucz" zaprezentował Scott McNealy w czasie Internet Associate Symposium w Berlinie. Sama technologia jest fascynująca, ale idea tkwi głębiej: jest to zapowiedź przyszłości, w której Java będzie zintegrowana z większością urządzeń.

Telefon też może zostać zintegrowany z siecią Internet. Takie telefony, nazywane Webphone, pozwalają przeglądać sieć WWW, czytać notowania giełdowe, informacje o pogodzie, jak również obsługiwać zwykłą pocztę głosową. Wszystko to możliwe jest wyłącznie dzięki Javie.

Jeszcze jeden gadżet - bezprzewodowy komputer wyposażony w przeglądarkę HotJava. Taki komputer nie ma urządzeń peryferyjnych, dysku twardego, dyskietek. Działa wyłącznie w sieci oferując przeglądanie sieci WWW oraz dostęp do aplikacji Javy.

"Wydaje mi się, że najważniejsze jest, że wiedza o Javie wykroczyła już daleko poza grono zapaleńców, programistów i guru informatyki..." -
podkreśla Scott McNealy, dyrektor generalny firmy Sun Microsystems.

"Java przeniosła się na poziom przedsiębiorstw, gdzie przyczynia się do wzrostu efektywności i pomaga w codziennym prowadzeniu biznesu."

Serwis Informacyjny ArsInfo

Kamil Kurowski
Rodan-System Sp. z o.o.
Warszawa, ul Jagielska 50c
kamil.kurowski@rodan.pl

1. Wstęp.

Praca niniejsza powstała w związku z udziałem autora w opracowywaniu przez firmę Rodan-System Sp. z o.o. projektu przedsięwzięcia informatycznego pod nazwą SYSTEM INFORMACYJNY- ArsInfo. W momencie pisania tej pracy trwają przygotowania do uruchomienia tego projektu jako produktu komercyjnego. System jest realizowany we współpracy z firmą ATM Sp. z o.o., której zadaniem jest zorganizowanie dostępu do systemu poprzez sieć Internet, oraz dostawę sprzętu. Pozostałymi partnerami w funkcjonowaniu systemu będą agencje prasowe, redakcje czasopism, urzędy państwowe oraz inne firmy gotowe wystąpić w roli dostawców informacji (*ang. content provider*), których serwisy informacyjne dostępne będą w publicznie. Całość prac analitycznych oraz projektowo-programistycznych ArsInfo zrealizowana została przez zespół Rodan-System. Głównym elementem systemu, zapewniającym dostęp i wyszukiwanie informacji jest motor wyszukiwania (*ang. search engine*) Search'97 firmy VERITY Inc. którego firma Rodan-System jest polskim dystrybutorem.

2. Idea systemu.

Sama idea systemów baz danych dostępnych sieciowo jest na świecie znana od czasu powstania pierwszych sieci komputerowych o zasięgu rozległym i obejmujących różne organizacje. Bierze się ona z coraz większej potrzeby dostępu do informacji z różnych dziedzin życia. Zapotrzebowanie to wzrasta wraz z rozwojem technik przesyłania informacji oraz ich upowszechnienia.

Bardzo ważną i najdynamiczniej się rozwijającą się gałęzią tego przemysłu jest komputerowa obsługa informacji. Jej burzliwy rozwój pojawił się wraz z rozwojem ogólnosiwiatowej sieci komputerowej INTERNET. Drugim ważnym powodem szybkiego rozwoju komputerowego dostępu do informacji stał się gwałtowny rozwój technik multimedialnych. Spowodował on możliwość uzyskania nie tylko informacji w postaci danych czy też dokumentów, ale również obrazów i dźwięku. Zaowocowało to poważną jakościową zmianą w komputerowej obsłudze informacji. Dzięki możliwościom multimedialnym, informacje dostępne za pomocą komputera, mogą być w codziennym użytku konkurencyjne dla informacji dostępnej w postaci prasy, książek, czy też radia i telewizji.

W związku z rozwojem tych technik powstało wiele narzędzi przeznaczonych do wyszukiwania informacji w rozproszonych bazach danych dostępnych w sieci takich jak np. WAIS, GOPHER, które to ustąpiły już miejsca nowszym technikom wykorzystujących

WWW (World Wide Web) wzbogacone o najnowsze techniki aktywnej prezentacji informacji. Jak doskonale wiadomo, nastąpiła eksplozja popularności WWW, a skutkiem tego pojawiła się samoistnie, prawie nieograniczona (z punktu widzenia użytkownika) kopalnia informacji. Powszechność i prostota wykorzystania WWW jest również, jak się okazało, jego wadą. Gigantyczne ilości informacji stworzyły naturalną barierę dla efektywnego wykorzystania dostępnych danych. Stworzono setki milionów stron, z różnymi informacjami. Odnalezienie interesującej w gąszczu całego szumu informacyjnego Internetu jest niezwykle trudne. Podstawową zaletą WWW miała być stosunkowa łatwość i szybkość wyszukiwania i dostępu do danych. Jednakże wątpliwość budzi sformułowanie "stosunkowa łatwość". Odnosi się ona do łatwości w porównaniu z wertowaniem stosów książek, prasy, czy też przesiadywaniem w bibliotece lub oczekiwaniem na sprowadzenie potrzebnych wydawnictw. Nie ulega jednak wątpliwości, że łatwość ta jak i szybkość dostępu nie jest tak duża jak może się to wydawać i jakie są potrzeby użytkowników informacji. Bierze się to z olbrzymiego przyrostu ilości tej informacji. Prowadzi to do olbrzymich trudności w odszukaniu źródła informacji i lokalizacji samej interesującej informacji. Metodą rozwiązania tego problemu jest tworzenie baz danych zawierających meta-informację, czyli informację o informacji. Ta sytuacja przypomina z kolei wertowanie katalogu bardzo dużej biblioteki. Bazy meta-informacji cierpią na tą samą dolegliwość co bazy informacji, to znaczy rozrastają się do rozmiarów uniemożliwiających łatwe się nimi posługiwanie, a co więcej pojawiają się inne problemy związane z aktualnością informacji w nich zawartych. W tej sytuacji jedynymi sojusznikami użytkowników w walce z zalewem informacji zostały serwery wyszukiwawcze (*ang. Web-indexers*) automatycznie indeksujące strony WWW i pozwalające wyszukiwać pełnotekstowo ich zawartość[2].

Problemy te biorą się przede wszystkim ze słabości narzędzi wyszukiwujących informację i mogą być rozwiązane poprzez udoskonalenie metod dostępu i odzyskiwania informacji. W tej chwili prowadzonych jest wiele badań naukowych mających za zadanie identyfikację, rozpoznanie i znalezienie rozwiązania pozwalającego na ujarznienie zalewu informacji jaką jest Internet i WWW. Nie jest to zadanie łatwe, ponieważ sama natura danych znajdujących się na stronach WWW jest trudna do przetwarzania. Dane te możemy nazwać danymi półstrukturalnymi (*ang. semi-structured data*). Charakteryzują się one niskim współczynnikiem strukturalizacji, a co więcej, opis swojej struktury zawierają w sobie. Sprawę dodatkowo komplikują wszelkiego rodzaju aktywne elementy stron typu: applety w Javie, czy komponenty ActiveX, które nie dają się strukturalizować w sposób pasywny.

Problemy te rodzą refleksję co do właściwości kierunku rozwoju metod pozyskiwania informacji. W niniejszej pracy opisany jest system, który w oparciu o istniejące narzędzie informatyczne, umożliwiające aktywne działanie w czasie rzeczywistym, odwracając rolę w wyszukiwaniu informacji. Nie jest to zasadniczy krok naprzód w kierunku rozwiązania problemu, ale sposób na jego ominięcie. Pomysł ten dotyczy tylko części zasobów informacyjnych, to znaczy informacji o charakterze tekstu, niosącego w sobie pewną treść merytoryczną, która może posłużyć jako jego identyfikacja. Podejście takie jest wskazaniem możliwości wygodnego wyszukiwania informacji do czasu rozwiązania problemu automatycznego pozyskiwania informacji z publikacji w Internecie.

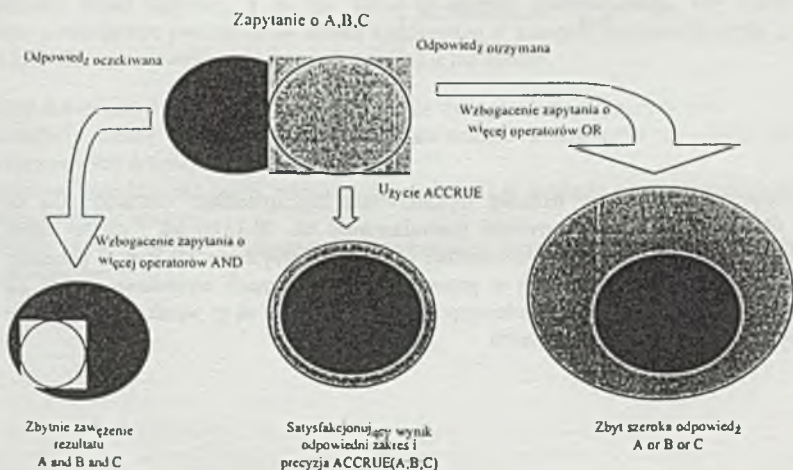
Pomysł ten polega na zastąpieniu przeszukiwania rozproszonych na WWW informacji przez umieszczenie jej w pewnej tematycznej meta bazie, tworzonej automatycznie, oraz przeszukiwaniu jej w celu znalezienia informacji na interesujące nas tematy. Proces ten jest automatyczny i polega na zdefiniowaniu w systemie własnego profilu zainteresowań i automatycznym otrzymywaniu informacji z tego zakresu, jak tylko pojawiają się one w bazie. Prowadzi to nie tylko do budowy aktywnej bazy informacji, ale również doskonale ogranicza

ilości otrzymywanej informacji do tylko istotnej. Oczywiście system tego rodzaju umożliwia również tradycyjne wyszukiwanie informacji w stałej bazie informacji.

3. SYSTEM WYSZUKIWANIA INFORMACJI.

Wraz z masowym zastosowaniem komputerów do produkowania, przechowywania i przetwarzania tekstów pojawił się nowy problem związany z ich wyszukiwaniem i efektywnym gromadzeniem. Temat ten pojawił się w literaturze informatycznej dość dawno (w latach 70-tych) w związku z systemami wyszukiwania informacji mającymi zastosowania np. w bibliotekach. W dzisiejszych czasach można jednak zauważyć nową inkarnację tego problemu, szczególnie w perspektywie wykorzystania wyszukiwania w środowisku Internetu.

Początek następnego dziesięciolecia (lata osiemdziesiąte) nie przyniósł dużych zmian w sposobie wyszukiwania informacji w zbiorach tekstowych. Dopiero pod koniec tego okresu pojawiły się systemy oparte o wyszukiwanie konceptualne (*ang. conceptual search*). Są one kontynuacją drogi rozwoju i następnym stadium systemów wyszukiwawczych opartych na logice boolowskiej i systemach statystycznych[1]. Jednym z przykładów takiego systemu i zarazem pierwszym tego typu komercyjnym systemem jest Search'97 (poprzednie wersje funkcjonowały pod nazwa Topic) firmy Verity. System ten został opracowany na zlecenie CIA. Innym, o trochę odmiennej filozofii jest system Fulcrum. Systemy te rozwinęły funkcjonalność swoich poprzedników o operator ACCRUE i budowanie tematów (topików). Używając nowego operatora i zestawu pojęć można zbudować temat odzwierciedlający nie tylko związki syntagmatyczne wyrazów występujących w tekście (związki pomiędzy wystąpieniami pojęć w ramach danego tekstu) lecz również związki paradygmatyczne (czyli związki pomiędzy pojęciami w sieci semantycznej obejmującej daną dziedzinę wiedzy). Najprościej można stwierdzić, że operator ACCRUE służy do budowy zoptymalizowanego, przy użyciu różnych metod opartych na logice rozmytej (*ang. fuzzy logic*), zapytania znanego z systemów boolowskich, złożonego z bardzo wielu operacji OR i AND. Sposób w jaki wpływa użycie operatora ACCRUE na wyniki wyszukiwania pokazane są na Rys. 1.

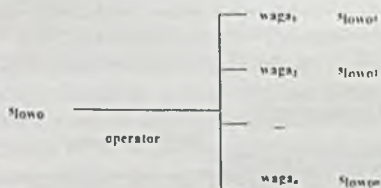


Rys. 1. Zapytanie z wykorzystaniem operatora ACCRUE.

Na Rys. 2 pokazana jest ogólna struktura tematu (topika). Operator ACCRUE możemy zdefiniować jako:

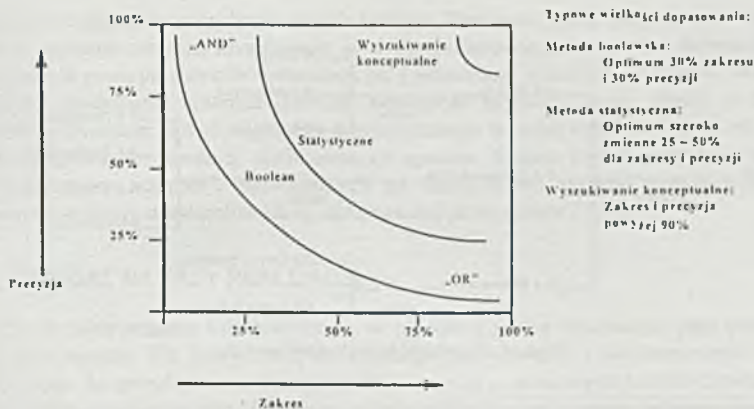
$\langle \text{ACCRUE} \rangle (T_1, T_2, \dots, T_k)$ – który sprawdza, czy tematy T_1, T_2, \dots, T_k występują w treści tekstu. Każdy z tematów może mieć przyporządkowaną wagę wyrażającą procentowy „udział” danego tematu w definiowanym znaczeniu [3].

Jest to zasadniczy krok naprzód w stosunku do systemów boolowskich i statystycznych. Wyszukiwaniu podlegają nie słowa występujące w dokumencie lecz dokument jest klasyfikowany ze względu na swoją treść (przechodzimy od wyszukiwania konkretnych wystąpień słów do globalnego spojrzenia na dokument i ustalenia jego tematu). Dodatkową zaletą takich systemów jest możliwość budowania tematów jako odpowiedniej konstrukcji drzewa innych tematów i słów. Każda gałąź takiego drzewa (składająca się ze słów, poddrzewa lub innego tematu) składająca się na definiowany temat może mieć przydzieloną wagę – liczbę z zakresu $(0;1)$ określającą w jaki sposób wystąpienie danego elementu (słowa lub gałęzi) drzewa wpływa na ostateczny wynik oceny przydatności tekstu. Ocena ta jest z zakresu $[1;0]$. Powoduje to, że znajdowane dokumenty, w naturalny sposób mogą być posortowane według otrzymanego wyniku. Oznacza to, że prawdopodobnie te dokumenty które uzyskały najwyższe noty w pełni nas satysfakcjonują i są to te których szukaliśmy. Takie podejście do konstruowania zapytań znacznie ułatwia obsługę tego typu systemów, nawet przez użytkowników mniej doświadczonych i zapewnia dużą dokładność rezultatów. Możliwość tworzenia tematów przy budowie których wykorzystano inne, już wcześniej zdefiniowane tematy prowadzi do powstawania struktur wielopoziomowych, z zagnieźdzonymi innymi tematami. Generalnie temat można określić jako pewne drzewo wiedzy. Drzewo takie (przykład na Rys. 4) jest definicją związków syntagmatycznych i paradygmatycznych co daje nam możliwość wnioskowania o temacie wyszukanego dokumentu.



Rys. 2. Ogólna struktura tematu.

Systemy wykorzystujące technikę wyszukiwania konceptualnego charakteryzują się dobrze dobranym zakresem odpowiedzi (nadmiarowość ok. 10-15%) jak i bardzo dobrą precyzją zapewniając jednocześnie komfortowe i proste wykorzystanie. Na Rys. 3 pokazano jak oba te wskaźniki kształtują się w poszczególnych systemach wyszukiwawczych: od boolowskiego, przez statystyczne do konceptualnych. Jak widać są to wyniki zdecydowanie przemawiające na korzyść tych ostatnich.



Rys. 3. Porównanie metod wyszukiwania.

Jak już zostało wspomniane wyszukiwanie konceptualne polega na szukaniu wystąpień wyrazów, lecz pewnych pojęć (tematów). Ich definicja opiera się na strukturze drzewiastej skonstruowanej ze słów oraz operatorów i wag odzwierciedlających związki między nimi. Dodatkowo tematy mogą być budowane w oparciu o już zdefiniowane pojęcia. Postępując w ten sposób ocenie podlega ocena tematu dokumentu, a nie zawartość poszczególnych słów. Przykład takiego, trochę bardziej rozbudowanego, tematu jest pokazany na Rys. 4. Jest to fragment drzewa definiującego temat „film”. Żeby nie komplikować definicji, a jednocześnie pokazać jak silnym narzędziem jest operator ACCRUE zbudowano je z wykorzystaniem tylko tego operatora. Można oczywiście stosować do rozbudowy tematu klasyczne operatory logiczne jak: AND i OR oraz wiele innych jak np. STEM (odmiana wyrazów). Na przykładzie widać również sposób w jaki korzystamy z przyporządkowania odpowiednich wag do elementów drzewa. Kolejny przykład (Rys. 5) obrazuje efekt zapytania. Jak łatwo zauważyć w odpowiedzi znaleziono słowa dotyczące całego tematu (reżyser, Wajda, scenariusz i to w różnych formach fleksyjnych), a nie tylko słowo film. Dzięki temu mamy pewność że dany tekst dotyczy sztuki filmowej, a nie np. filmu do aparatu fotograficznego, czy filmu olejowego powstającego podczas pracy silnika spalinowego w komorze spalania. Korzyści z takiego sposobu wyszukiwania możemy podsumować w ten sposób:

- Wybór dokumentów na podstawie ich tematu, a nie wystąpienia pojedynczych słów.
- Akumulacja wiedzy o różnych dziedzinach i możliwość wykorzystania tej wiedzy do budowy nowych definicji.
- Możliwość subiektywnej oceny ważności dokumentów (ze względu na ich znaczenie z punktu widzenia tematu).
- Łatwość wykorzystania zgromadzonego potencjału, nawet przez początkujących użytkowników.



Rys. 4. Przykład konstrukcji tematu.

film, Wajda, reżyser

WYSTĄPIEŁ

Wynik	Tytuł
0.99	37 nowych filmów w tym roku.
0.95	Poprawione wersje Wojen gwiazdowych
0.80	Festiwal filmowy w Berlinie

Niejący rok mimo kryzysu polskiego filmu nie był beznadziejny. Pojawiły nowe filmy, z których spora powstała z kapitału prywatnego. Obok starych mistrzów jak *Wajda*, pojawili się młodzi reżyserzy, którzy z powodzeniem zdobywają polskich widzów. Wygląda nawet, że minął odwieczny problem z dobrymi scenariuszami.

Rys. 5. Przykład wyniku wyszukiwania

Wyszukiwanie konceptualne jest w tej chwili najbardziej zaawansowaną techniką wyszukiwania pełnotekstowego stosowaną w systemach komercyjnych. Rynek coraz bardziej dostrzega siłę narzędzi wyposażonych w ten rodzaj wyszukiwania. Na ich podstawie nie tylko buduje się systemy pozwalające na przeszukiwanie dokumentów korporacyjnych, publikacji czy np. dokumentów prawnych, lecz także aktywnych agentów pozwalających na aktywne śledzenie strumieni informacji (np. serwisy informacyjne, agencyjne, pisma wchodzące itp.) i automatyczna dekretacja owych informacji łącznie z przysyłaniem ich do zainteresowanych osób, czy działów. Oprogramowanie takie pozwala na dużą automatyzację pracy w komórkach które muszą codziennie przeglądać gigantyczne ilości informacji w poszukiwaniu tych interesujących. Co więcej przy dobrym zdefiniowaniu tematu poszukiwania, mamy gwarancję, że żadna ważna informacja nam nie umknie.

4. Założenia systemu ArsInfo

4.1. WSTĘP

Przedsięwzięcie dotyczy stworzenia i eksploatacji serwisu informacyjnego polegającego na selektywnej dystrybucji informacji tekstowych lub odnośników do nich (np. prasowych, agencyjnych, oraz prawnych) wśród abonentów posiadających dostęp do Internetu. Drugim obszarem działalności systemu jest usługa polegająca na umożliwieniu wyszukiwania

informacji zawartych w stałej bazie danych systemu. Zawartość stałej bazy systemu będzie tworzona zarówno poprzez uzupełnianie jej o porcje informacji w postaci dokumentów dostarczonych przez producentów informacji jak i akumulację informacji zebranych podczas codziennej działalności systemu. Trzecim elementem systemu (w tej chwili w fazie projektowania) będzie moduł aktywnego automatycznego wyznaczania informacji w sieci WWW na podstawie wcześniej zdefiniowanych agentów. System będzie korzystał z wielu programów typu „wrapper”, pozwalających na dostosowanie danych otrzymanych od dostawców do formy strukturalizowanej, akceptowanej przez system [4][6].

4.2. PODZIAŁ NA FAZY REALIZACJI.

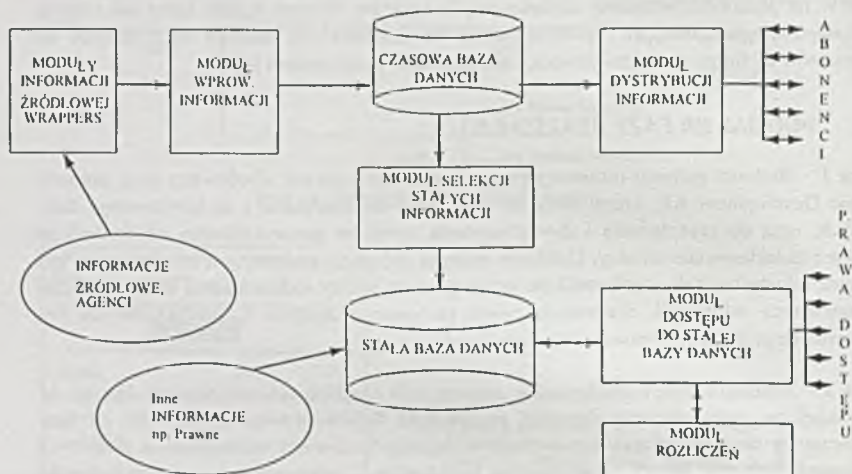
Faza 1 - Budowa systemu informacyjnego w oparciu o serwer zbudowany przy pomocy Topic Development Kit, który służy do dziennego indeksowania i archiwizowania bazy danych, oraz do przeglądania i dystrybuowania wyników poszczególnym użytkownikom poprzez zadeklarowane serwisy. Działanie systemu polega na codziennym indeksowaniu baz danych i dystrybucji do użytkowników np. za pomocą poczty elektronicznej dziennej paczki zamawianych informacji. Sterowanie całym procesem przepływu i obróbki danych jest zapewnione przez zbiór demonów napisanych w C .

Faza 2 - Zainstalowanie i uruchomienie serwera AIS (ArsInfo Internet Server). Dostęp do informacji za pośrednictwem dowolnej przeglądarki WWW, a więc system jest od razu widoczny w sieci dla tysięcy użytkowników. Istnieje możliwość udostępniania niektórych informacji darmowo, innych po wykupieniu abonamentu. Użytkownicy mogą korzystać tylko z zapytań wcześniej zdefiniowanych przez administratora systemu. Dostęp do informacji będzie rozliczany w oparciu o aplikację napisaną w SZBD CA-OpenIngres.

Faza 3 – budowa systemu automatycznych agentów w oparciu o oprogramowanie Information Agents Verity

4.3. ORGANIZACJA SYSTEMU

W tym punkcie opisane zostaną podstawowe założenia co do technicznej organizacji systemu i metody jego realizacji.



Rys. 6. Organizacja systemu.

Organizację systemu przedstawiono schematycznie na Rys. 6. Zbiory informacji tekstowych systemu obejmują informacje przejściowe zawierające wiadomości z bieżącego dnia pochodzące ze współpracujących czasopism i serwisów prasowych, oraz w przyszłości od automatycznych agentów. Te informacje przechowywane są w czasowej meta bazie danych. Otrzymywane teksty źródłowe będą poddawane wstępnej obróbce przez odpowiednie moduły obsługi informacji źródłowej i doprowadzane do homogenicznej formy akceptowalnej dla modułu wprowadzania informacji. Wybór informacji dla każdego abonenta, zgodny z jego profilem informacyjnym będzie przeprowadzony przez odpowiednią aplikację napisaną w TDK, jest to zadanie modułu dystrybucji informacji. Wyniki poszczególnych zapytań będą dystrybuowane przez odpowiednie serwisy. Wybrana część meta informacji przejściowych, uznana za przydatną dla analizy retrospektywnej zgodnie z kryteriami selekcji zdefiniowanymi w ramach modułu selekcji stałych informacji, zostanie wykorzystana dla aktualizacji stałej bazy danych. Informacje zawarte w stałej bazie danych są dostępne dla użytkowników zarejestrowanych w systemie a rozliczanie opłat będzie prowadzone w oparciu o wykorzystanie zasobów systemu (CPU, I/O, transmisja danych). Stała baza danych zawiera informacje wyselekcjonowane przez moduł selekcji informacji oraz dane inne np. prawne, marketingowe. Dostęp on-line będzie realizowany przez moduł dostępu do stałej bazy danych zrealizowany w oparciu o ArsInfo Internet Server (WWW) współpracujący z modulem rozliczeń napisanym w systemie CA-OpenIngres. Informacje źródłowe stanowią zbiory tekstowe otrzymywane codziennie z agencji prasowych, redakcji wybranych dzienników i tygodników lub jako wynik działalności agentów.

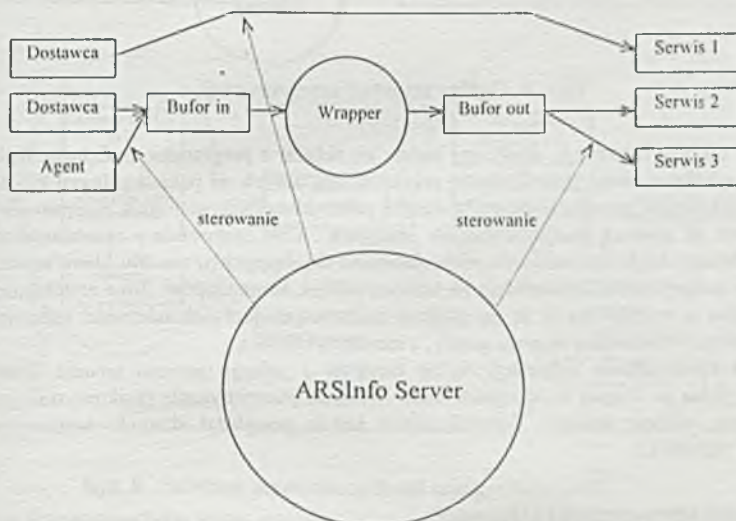
4.4. OPIS DZIAŁANIA GŁÓWNYCH MODUŁÓW.

4.4.1. Moduł obsługi informacji źródłowej.

Podstawową funkcją modułu jest lokalizacja i wstępne przygotowanie informacji wejściowej do dalszej obróbki. Dane wejściowe pochodzące z różnych źródeł takich jak dostawcy informacji, agencji, czy inne systemy informacyjne zostaną zlokalizowane, zreplikowane, oraz przygotowane do dalszej obróbki poprzez dedykowane programy typu „wrapper”.

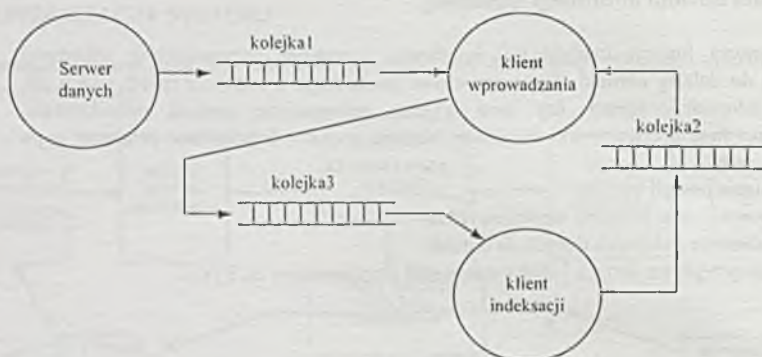
- zmianie postaci danych.
- wyczyszczenie buforów wejścia/wyjścia.
- znalezienie kolejnych danych do obróbki.

Schemat przepływu danych i sterowania został przedstawiony na Rys. 7.



Rys. 7. Schemat przepływu danych i sterowania w module obsługi informacji źródłowej.

Kluczową sprawą w systemie jest zapewnienie odpowiedniej synchronizacji i sterowania poszczególnymi jego modułami tak by efektywnie wykorzystać moc obliczeniową komputera oraz zapewnić poprawną realizację zadań. Ponieważ nie możemy założyć żadnych zależności czasowych pomiędzy zdarzeniami zachodzącymi w systemie, musimy zapewnić mechanizmy kolejowania i wstrzymywania akcji występujących w systemie. Całe sterowanie systemem w sposób naturalny możemy podzielić na trzy czasowo spójne fazy: proces otrzymywania danych z różnych źródeł, proces konwersji, proces indeksacji i dystrybucji. Dlatego sterowanie pracą systemu zostało rozdzielone pomiędzy trzy główne procesy synchronizowane takimi mechanizmami jak kolejki komunikatów, oraz semaforey. Schemat synchronizacji pokazany jest na Rys. 8



Rys. 8. Ogólny schemat synchronizacji.

Moduł obsługi informacji, źródłowej będzie się składał z programów w C które będą sterować i synchronizować jego działanie, przyjmowanie danych od poszczególnych źródeł, oraz przeprowadzanie procesu konwersji z danych półstrukturalnych na format akceptowalny przez system za pomocą specjalizowanych „wrappers”. Całe sterowanie i synchronizacja osiągnięte będzie dzięki zastosowaniu wielu demonów działających w modelu klient/serwer, niezależnie z zapewnioną komunikacją za pomocą kolejek komunikatów. Taka architektura jest niezbędna ze względu na to, że nie możemy zakładać jakiegokolwiek zależności czasowej pomiędzy przysyłaniem danych przez gazety, a ich dalszą obróbką.

Moduł wprowadzania informacji będzie korzystał z jednego procesu serwera, który cyklicznie, jedną po drugiej w odstępach czasu zadanych parametrycznie (najkorzystniejszy odstęp czasu zostanie ustalony doświadczalnie) będzie przeglądał skrzynki kontaktowe dostawców (agentów).

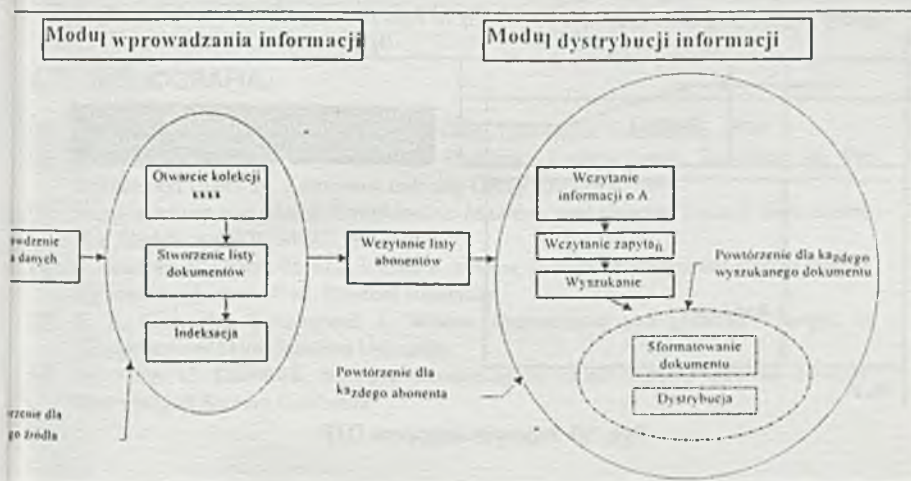
4.4.2. Moduł wprowadzania informacji.

Podstawową funkcją modułu jest przetworzenie informacji dostarczonej przez moduł informacji źródłowej w formie meta informacji tekstowych. Moduł opiera się na aplikacji napisanej z wykorzystaniem TDK firmy Verity. Aplikacja otworzy sesję, sprawdzi które meta dane są gotowe, utworzy odpowiednie kolekcje, stworzy listy plików do indeksacji, zaindeksuje pliki z list, wczyta konfigurację abonentów, wyszuka w otwartych kolekcjach dane, przekaże do serwisu, zamknie sesję i zakończy działalność.

Proces klienta będzie cyklicznie pobierał komunikaty z kolejki, przynosił dane z buforów wejściowych opisanych przez komunikat do bufora tymczasowego skąd zostaną pobrane przez odpowiedni „wrapper”, następnie poczeka na konwersję. Po zakończonej przeniesie wyniki konwersji do odpowiedniego bufora dedykowanego danemu źródłu informacji, z którego zostaną one pobrane przez proces indeksacji. W przypadku gdy komunikatem odebrany przez klienta będzie komunikat o początku indeksacji proces klient wyśle komunikat do kolejki 2, który zezwoli na indeksację. Kolejka 3 synchronizująca działanie klienta indeksacji i klienta wprowadzania jest konieczna do wstrzymania procesu konwersji przez „wrapper”, w przypadku gdy poprzednia indeksacja się jeszcze nie skończyła (bufor jest pełny). Następnie przeniesie dane do następnego bufora i wyśle komunikat do kolejki 3.

Kolejne działania to

- otwarcie sesji
- sprawdzenie które dane są gotowe do indeksacji
- otwarcie dla nich kolekcji
- stworzenie listy plików do indeksacji (oddzielnie dla każdego źródła)
- zaindeksowanie dokumentów z listy
- wczytanie konfiguracji abonentów
- wyszukanie według zapytań
- skonfigurowanie pliku do dystrybucji (wpisanie pól, zaznaczenie fragmentów, zmiana strony kodowej)
- dystrybucja



Rys. 9. Schemat działania aplikacji napisanej w TDK.

Jednym z problemów jakie trzeba rozwiązać to odpowiednie zaprojektowanie „wrapper”. Np. „wrapper” do automatycznego rozbijania plików DTP ma za zadanie m.in. dopasowanie do siebie tekstu tytułu i tekstu artykułu na podstawie informacji o względnym położeniu obu tekstów, użytych stylów oraz ewentualnie treści tych tekstów. Nie realizuje on postawionego problemu w 100% niemniej jednak uzyskuje się efekty w granicach od 50% do 90%. Tak duża rozbieżność jest podyktowana dużymi różnicami w sposobie składania gazet, czyli rozplanowywania rozkładu ramek na gazecie. Algorytm działania ma charakter czysto sekwencyjny. Złożoność poszczególnych elementów waha się od n do n^2 niemniej jednak dzięki optymalizacji stanów najczęściej występujących funkcja realizująca ten algorytm jest bardzo efektywna. Schematycznie algorytm jest pokazany na poniższym rysunku.

Kroku pierwszym następuje wczytanie informacji o analizowanych ramach, ustalenie największej ramki jaką dany tekst zajmuje oraz wstępnie określa się czy dany tekst może być tytułem (na podstawie nazwy stylu).

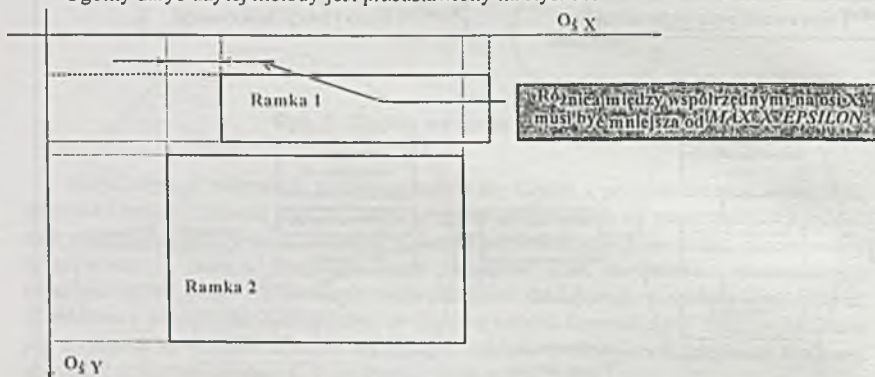
Drugi krok to sortowanie. Do tego celu została użyta funkcja standardowy algorytm *qsort*. Zostały użyte następujące kryteria sortowania: nazwa pliku źródłowego, numer strony, lewa współrzędna x , prawą współrzędna x , lewa współrzędna y . Te kryteria wynikają z

następujących przesłanek tytuł i artykuł są w tym samym pliku i na tej samej stronie, a ponad to w zasadniczej większości przypadków mają nie wiele różniące się współrzędne x-owe.

Krok trzeci. Tworzone są dwukierunkowe listy ramek które są ze sobą podobne (relacja podobieństwa w tym wypadku nie jest symetryczna, gdyż po jednej stronie występuje tytuł a po drugiej artykuł)

W kroku czwartym następuje scalenie podobnych do siebie plików. Kolejność jest ustalana według kryteriów: każdy element list musi należeć do jednej z czterech kategorii (określają one stopień bycia tytułem, bądź artykułem), a ramach kategorii według rozmiaru pliku od najmniejszego do największego).

Ogólny zarys użytej metody jest przedstawiony na Rys. 10.



Rys. 10. Algorytm wrappera DTP.

Dla osi X sytuacja jest prosta, natomiast dla osi Y, jest to szereg warunków mających wykluczyć sytuacje skrajne, czyli tytuł nie może znajdować się poniżej artykułu, ani za wysoko nad tytułem. Wykluczana jest również sytuacja w której tytuł znajduje się wewnątrz ramki, jest to związane z specyfiką składu tekstu i niejednoznacznościami powstałymi w skutek tego (nie można wykluczyć że zawarta ramka jest innym artykułem).

4.4.3. Moduł dystrybucji informacji.

Moduł będzie miał za zadanie przyjmować zlecenia od klientów, wyszukiwać artykuły według ich potrzeb i w zależności od serwisu np. wysyłać je pocztą elektroniczną w postaci plików tekstowych lub odnośników do danych.

Moduł składa się z trzech części:

- część przyjmująca zlecenia od klientów
- część przetwarzająca zlecenia na skrypty
- część szukająca i dystrybucyjna (część zintegrowana z modulem wprowadzania informacji)

Zlecenia klientów przyjmowane będą za pośrednictwem stworzonego metajęzyka i strony WWW pośredniczącej w przyjmowaniu zleceń.

Proces wyszukiwania będzie korzystał głównie z topików, i szukał według drzewa skojarzeń, podpartym algorytmem polskiej odmiany wyrazów (stemer) i polskim słownikiem wyrazów bliskoznacznych, gdy system się rozrośnie nastąpi podzielenie drzew topików i

słowników na grupy tematyczne, lub nawet stworzenia tych elementów osobno dla każdego użytkownika.

5. Podsumowanie i plany na przyszłość

Opisany tutaj system jest w końcowej fazie realizacji. Stosunkowo najmniej zaawansowaną częścią systemu są automatyczni agenci penetrujący strony WWW, oraz „wrappers” z nimi związani. Wynika to przede wszystkim z trudnościami jakie napotykały przy próbach wyciągnięcia użytecznych danych z rozproszonego środowiska WWW. O pewnych metodach rozwiązania problemu możemy przeczytać w [4][5]. Ponieważ system jest produktem komercyjnym jego przyszłość zweryfikuje i wyznaczy rynek.

6. BIBLIOGRAFIA.

- [1] Clark Brady, *The Case of Full-Text Searching*, Law Office Computing, 1994.
- [2] John Lehman, *On-Line Information Profiling: Content-Based Searching on the Information Highway*, Automated Indexing Group REPORT, 1994.
- [3] Vipul Kashyap and Marek Rusinkiewicz, *Modeling and Querying Textual Data using E-R Models and SQL*, MCC, 1997.
- [4] J. Hammer, H. Gracia-Molina, J. Cho, R. Aranha, A. Crespo, *Extracting Semistructured Information from the Web*, Stanford University.
- [5] S. S. Chawathe, S. Abiteboul, J. Widom, *Representing and Querying Changes in Semistructured Data*, Stanford University
- [6] N. Ashis, C. Knoblock, *Wrapper Generation for semistructured Internet Sources*, University of Southern California.

NetWare 5 – najwydajniejsza platforma do aplikacji napisanych w Javie

Dariusz Leonarski
NOVELL POLSKA

Novell podnosi poprzeczkę

Czołowy producent oprogramowania sieciowego czyni ogromny krok naprzód, wprowadzając na rynek NetWare 5, zawierający najszybszą na świecie wirtualną maszynę Javy, bazę danych Oracle8 i konsolę Javy

Serwerowa platforma sieci inteligentnych należy do Novella. Taka wiadomość nadeszła z dorocznej konferencji programistów, podczas której czołowa firma rynku oprogramowania sieciowego przedstawiła informacje o nowej platformie, która podnosi jeszcze wyżej poprzeczkę w zakresie wydajności, skalowalności i możliwości administrowania. Koncentrując się na platformie sieciowej Novell NetWare i wykorzystaniu przez nią systemu Novell Directory Services (NDS), Novell poinformował, że:

- W skład NetWare 5 wejdzie najszybsza na świecie maszyna wirtualna Javy, służąca do uruchamiania aplikacji i usług Javy
- Do NetWare, najszerzej stosowanej platformy sieciowej, będzie dołączana najpopularniejsza baza danych — Oracle8
- Wersja finalna systemu powinna się pojawić zgodnie z planami, w połowie tego roku. Trzecia wersja beta będzie obejmować skalowalne usługi sieciowe korzystające z otwartych standardów.

Najszybsza wirtualna maszyna Javy

NetWare będzie zawierać najszybsze obecnie środowisko Javy. Niezależne badania przeprowadzone przez Key Labs z wykorzystaniem testu wzorcowego VolanoMark potwierdziły, że NetWare jest ponad dwukrotnie szybszy od drugiego w kolejności produktu, JVM Microsoftu, działającej w systemie Windows NT Server. Testy przeprowadzono z wykorzystaniem 200 połączeń, obsługiwanych przez komputer z procesorem Intel Pentium II 300 MHz i 128

MB pamięci RAM. Wstępne analizy wskazują również, że wirtualna maszyna Javy Novella jest wielokrotnie szybsza niż implementacje Javy na najpopularniejszych systemach unixowych uruchomione na sprzęcie w tej samej kategorii cenowej.

To radykalne zwiększenie wydajności wirtualnej maszyny Javy czyni z NetWare najlepszą platformę serwerową dla aplikacji Javy, zapewniającą użytkownikom najszybszy dostęp do aplikacji i danych. Oprócz tego obsługa Javy na serwerze NetWare umożliwi programistom zaspokojenie wymagań w zakresie wydajności dużych sieci i krytycznych aplikacji. Środowisko Java Runtime Environment (JRE) 1.1.5, w skład którego wchodzi JVM, stanowi element wersji beta 3 NetWare 5 oraz opracowanego przez Novella zestawu programowania Open Solutions Architecture (OSA) SDK.

Najlepsze rozwiązanie w dwóch kategoriach

Dzięki pojawieniu się systemu NetWare 5, klienci Novella uzyskają zintegrowany system baz danych współpracujący z systemem katalogowym, charakteryzujący się doskonałą wydajnością i łatwością administrowania. Wraz z wersją Oracle8 dla pięciu użytkowników, dostarczaną z każdym systemem NetWare 5, użytkownicy uzyskają wydajną, skalowalną i zintegrowaną platformę wdrażania oprogramowania, w tym aplikacji Javy.

Udostępnienie NetWare 5 w wersji beta 3

Nowe funkcje, włączone do ostatniej wersji beta systemu NetWare 5, rozszerzają możliwości projektowania i wdrażania oraz administrowania sieciowymi rozwiązaniami informatycznymi. Funkcje te współpracują z systemem katalogowym NDS, co czyni z NetWare 5 sieciową platformę serwerową przeznaczoną dla przedsiębiorstw, która obsługuje IP i służy do administrowania w sieciach mieszanych zasobami takimi jak aplikacje, drukarki, pliki, bazy danych, intranet itd. NetWare 5 nadal obsługuje IPX, co jest szczególnie istotne dla tych klientów, którzy pracują w środowiskach mieszanych. Oto wykaz najważniejszych nowości, które pojawiły się w wersji beta 3 systemu NetWare 5:

- **ConsoleOne** — w fazie projektowania znana pod kryptonimem "Houston". Wprowadza graficzny interfejs administracyjny do konsoli serwera NetWare. Dzięki wykorzystaniu usług katalogowych NDS, ConsoleOne zapewnia administratorom możliwość elastycznego, kontekstowego administrowania całym systemem. Konsola ta jest oparta całkowicie na Javie i integruje aplety Javy oraz moduły dodatkowe, zapewniając programistom jednolite środowisko tworzenia sieciowych systemów administracyjnych. Więcej na ten temat znajduje się w dalszej części omówienia.

Rozszerzony NDS — NetWare 5 zawiera rozszerzony system katalogowania NDS Novella. Nowa wersja obsługuje protokół LDAP v.3 (Lightweight Directory Access Protocol), co zapewnia użytkownikom możliwość współpracy z innymi katalogami wykorzystującymi ten sam protokół. Nowością są też usługi katalogowania katalogów, które pozwalają administratorom wyszukiwać dowolny katalog i szybciej uzyskiwać dostęp do informacji rozproszonych po całej sieci. NetWare 5 charakteryzuje się też wyższą wydajnością pracy katalogów, a także zmniejsza ruch w sieci, co przyczynia się do zwiększenia wydajności pracy w sieciach rozległych.

Nowy system instalacji i migracji — nowa procedura konfigurowania serwera skracza czas potrzebny użytkownikom na przygotowanie swoich serwerów do zainstalowania NetWare 5. Zmodyfikowany został również kreator Novell Migration Wizard, który został włączony do wersji beta 3 systemu NetWare 5, co zapewnia użytkownikom możliwość automatycznego przejścia na ten system.

"Zainstalowaliśmy i eksploatujemy obecnie NetWare 5 w naszej sieci próbnej. Działania te stanowią fragment programu testowania wersji beta systemu NetWare 5" — stwierdził Keith G. Momberg, główny specjalista ds. sieci w firmie Lockheed Martin. "Chociaż wykorzystujemy mieszane środowisko współpracujące z protokołami IP i token ring, najważniejszymi cechami NetWare 5 są dla nas wyłączny IP oraz system składowania danych Novell Storage Services (NSS). Serwery wykorzystywane w naszej podstawowej działalności są wyposażone w układy dysków o pojemności wielu gigabajtów. Mamy nadzieję, że NSS pozwoli nam usprawnić dostęp do przechowywanych na nich danych."

Szkolenia i certyfikaty NetWare 5

Ponad 290 tysięcy osób z tytułem Certified Novell Administrator (CNA) i 135 tysięcy posiadaczy tytułu Certified Novell Engineer (CNE) będzie mogło rozszerzyć swoją wiedzę i uprawnienia na szkoleniach zakończonych uzyskaniem tytułów NetWare 5 CNA i CNE. Kursy te rozpoczną się równocześnie z wprowadzeniem na rynek nowego produktu i odbywać się będą w ponad 14500 ośrodkach szkoleniowych Novella znajdujących się na całym świecie.

Jeszcze przed rozpoczęciem kursów można poszerzyć swoje wiadomości na temat NetWare 5 na pięciogodzinnym seminarium lub jednodniowym kursie

praktycznym, które tej wiosny odbywają się w wybranych ośrodkach szkoleniowych Novella oraz biurach handlowych tej firmy na całym świecie. Aby dowiedzieć się więcej na ten temat, prosimy odwiedzić ośrodek WWW Novell Education, mieszczący się pod adresem <http://education.novell.com>.

Cena i dostępność

Wersja beta 3 systemu NetWare 5 pojawi się na całym świecie w kwietniu, w ramach otwartego programu testowania wersji beta. Trzy dyski CD z wersją beta 3 systemu NetWare 5 można otrzymać po zarejestrowaniu się w ośrodku WWW Novella (<http://www.novell.com/netware5>) i po wpłaceniu 20 USD na poczet kosztów manipulacyjnych i materiałowych.

Konsola administracyjna dla klientów i serwerów NetWare, napisana jest w 100 procentach w Javie

Firma Novell zaprezentowała produkt o nazwie Novell ConsoleOne, który ma być podstawą nowej generacji narzędzi do systemów administrowania opracowanych przez Novella i innych producentów. ConsoleOne, która w trakcie opracowywania nosiła kryptonim "Houston", jest opracowana w 100 procentach w Javie. Pozwala ona na administrowanie serwerami NetWare i komputerami-klientami w graficznym interfejsie użytkownika. Produkt ten jest oparty na standardach internetowych (Java, JNDI, IP, LDAP, SSL) i stanowi wspólną platformę, integrującą aplety Javy i moduły dodatkowe, umożliwiając programistom tworzenie oprogramowania do administrowania sieciami o ujednoliconym wyglądzie i sposobie działania. ConsoleOne wykorzystuje system Novell Directory Services (NDS), dzięki czemu zapewnia administratorom elastyczny, dynamiczny system administrowania, który obsługuje administrowanie na podstawie stanowiska, stosownie do potrzeb i zadań poszczególnych administratorów sieci.

ConsoleOne jest wyrazem strategii Novella, która polega na udostępnianiu platformy serwerowej realizującej wyrafinowane funkcje sieciowe, obejmującej również nową klasę wszechstronnych systemów do administrowania siecią wykorzystujących NDS. Java odgrywa kluczową rolę w strategii Novella, ponieważ przyspiesza opracowywanie i wdrażanie oprogramowania sieciowego, które może być dystrybuowane i udostępniane we wszystkich platformach w sieciach przedsiębiorstw, intranetach i w Internecie.

ConsoleOne jest spełnieniem obietnicy uczynienia produktów i sieci Novella najłatwiejszymi w obsłudze i w administrowaniu. Dzięki wykorzystaniu NDS i standardów, ConsoleOne stanowi niezwykle elastyczną i wyposażoną w bogate

możliwości funkcjonalne platformę, którą mogą z powodzeniem wykorzystać partnerzy Novella.

Ułatwienie administrowania siecią

Wykorzystując NDS, ConsoleOne udostępnia różne obiekty poszczególnym administratorom, w zależności od zakresu ich zadań i odpowiedzialności. Możliwości funkcjonalne pozwalają na pojedyncze rejestrowanie się, rozdzielanie zadań oraz przydzielanie ich do poszczególnych stanowisk. Administratorzy otrzymują zatem narzędzie odpowiednie do wykonywanego zadania. Poszczególne uprawnienia NDS można np. przydzielać administratorom, którym powierzone zostało administrowanie hasłami, użytkownikami, serwerami, drukarkami itp. Ponadto administratorom można przypisywać widoki sieci, w których występują wszystkie te obiekty, jednak ograniczone kontekstowo. ConsoleOne pozwala instytucjom na zapewnienie wysokiego poziomu zabezpieczenia przy administrowaniu sieciami, a równocześnie na wdrażanie elastycznych i łatwo adaptowalnych rozwiązań, spełniających wymagania administratorów i użytkowników nawet w największych sieciach.

Opracowywanie oprogramowania administracyjnego

Novell ConsoleOne stanowi platformę oprogramowania do administrowania i zarządzania w standardzie Novell OSA (Open Solution Architecture — architektura rozwiązań otwartych). OSA definiuje otwarte interfejsy i platformy, które pozwalają programistom na tworzenie łatwego w administrowaniu oprogramowania sieciowego, przygotowanego do korzystania z usług katalogowych i do pracy z Internetem.


ConsoleOne definiuje rozszerzalną architekturę, która umożliwia programistom integrację na wielu poziomach, od obsługi apletów Javy po opracowywanie modułów dodatkowych (snap-in) za pomocą interfejsów programowania (API) ConsoleOne. Dzięki interfejsom programowania ConsoleOne programiści mają możliwość korzystania z jednolitej platformy, co przyspiesza tworzenie oprogramowania. Moduły dodatkowe ConsoleOne opracowywane przez niezależnych producentów będą miały wygląd i zachowanie podobne do produktów Novella, co pozwoli na lepsze wykorzystanie umiejętności wykwalifikowanych administratorów sieci.

Dostępność

ConsoleOne zostanie najpierw udostępniona jako składnik systemu NetWare 5, a następnie dołączona do ManageWise, BorderManagera, GroupWise, NDS for NT oraz Z.E.N.works. Produkt będzie także można sprowadzić z WWW.

Powłoka i moduły dodatkowe interfejsów programowania ConsoleOne wchodzi w skład pakietu OSA SDK, który jest aktualizowany co kwartał.

TWORZENIE APLIKACJI INTERNETOWYCH W SYSTEMIE WEBSPEED



Piotr Tucholski
CSBI S.A. Oddział Progress

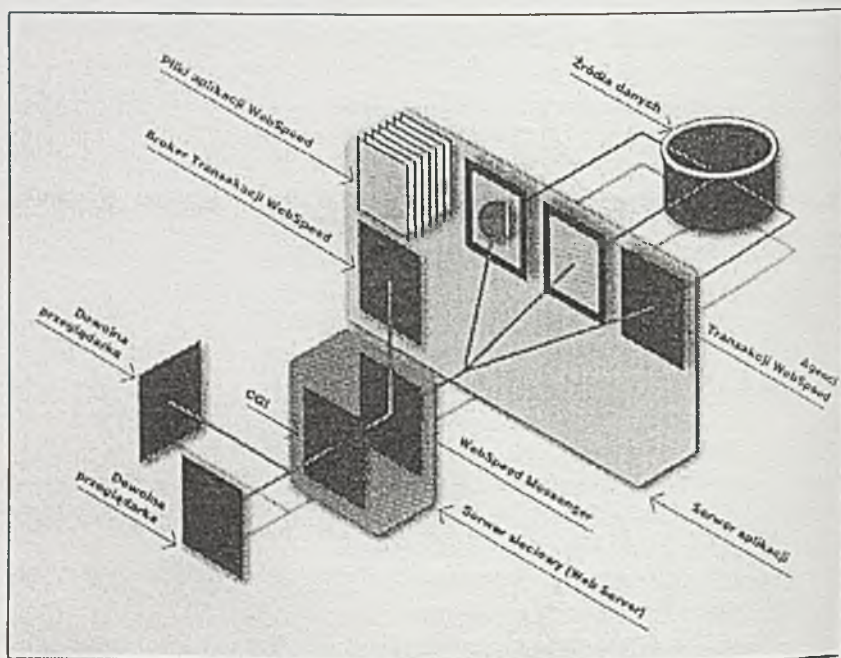
Wiosenna Szkoła PTI
Świnoujście '98

Tworzenie aplikacji internetowych w systemu WebSpeed

1. Wprowadzenie

System WebSpeed służy do tworzenia i uruchamiania aplikacji, obsługujących transakcje poprzez sieci Internet i Intranet. Programy stworzone przy pomocy tego narzędzia wyglądają dla użytkownika Internetu jak zwykła strona WWW. System WebSpeed charakteryzuje się krótkimi czasami odpowiedzi i dużą łatwością integracji z systemami zabezpieczającymi, przeglądarkami oraz serwerami sieciowymi. Cały system składa się z dwóch produktów: WebSpeed Transaction Server oraz WebSpeed Workshop.

WebSpeed Transaction Server jest modulem, działającym w systemach Unix i Windows NT. Jest on odpowiedzialny za uruchomienie i poprawne działanie aplikacji stworzonej przez użytkownika. Zapewnia spójność danych i transakcji przy jednoczesnym korzystaniu z kilku baz nawet w przypadku przerwania protokołu TCP/IP. Na rysunku 1 przedstawiono schemat działania aplikacji stworzonej i działającej przy pomocy systemu WebSpeed.



Rysunek 1.

Użytkownik wysyła żądanie wpisując określony adres URL (*Uniform Resource Locator*) na swojej lokalnej przeglądarce (*Web Browser*), która komunikuje się z serwerem sieciowym

(Web Server). Serwer sieciowy przesyła to żądanie do WebSpeed Transaction Servera, uruchamiając tzw. *WebSpeed Messenger*, który jest interfejsem pomiędzy serwerem sieciowym (standard CGI, ISAPI, NSAPI) oraz elementami WebSpeed Transaction Servera. W następnej kolejności Messenger kontaktuje się z Brokerem Transakcji (*WebSpeed Transaction Broker*), zarządzającym procesami Agentów Transakcji (*WebSpeed Transaction Agents*). Zarządzanie polega na przydzielaniu procesów agentów, służących do obsługi konkretnych żądań. Jeśli liczba agentów jest zbyt mała broker może ją zwiększyć. Agenci Transakcji wykonują poszczególne moduły stworzonej aplikacji (*Web Objects*) zapewniając im dostęp do baz danych, a moduły te obsługują żądanie użytkownika i w dynamiczny sposób generują strony HTML. Wygenerowane strony trafiają z powrotem do Messengera, a następnie do serwera sieciowego i przeglądarki.

Drugim modulem pakietu jest *WebSpeed Workshop*. Jest to złożone środowisko, służące do tworzenia i testowania aplikacji. Workshop jest zainstalowany na serwerze i dostęp do niego, a więc do tworzenia programów, odbywa się poprzez przeglądarkę internetową.

WebSpeed Workshop jest wyposażony w szereg narzędzi pomagających programiście w tworzeniu i testowaniu aplikacji, przeglądaniu struktur baz danych itd. Narzędzia te, z reguły bardzo proste w użyciu, będą przedstawione w dalszej kolejności. Rysunek 2 przedstawia *WebSpeed Workshop* i listę narzędzi (ramka z lewej strony ekranu).



Rysunek 2. WebSpeed Workshop

2. Środowisko WebSpeed Workshop

Jak już mówiliśmy, WebSpeed Workshop jest kompletnym środowiskiem programisty do tworzenia i testowania aplikacji. Dostęp do niego odbywa się poprzez przeglądarki Internetu. Środowisko to oferuje kilka różnych sposobów tworzenia programów oraz narzędzia wspomagające pracę programisty.

Rozpoczynając pracę z Workshopem, mamy do wyboru trzy przyciski:

Files - pliki

Tools - narzędzia (patrz rysunek 2)

Help - rozbudowany system pomocy on-line

2.1. Tworzenie nowych aplikacji przy pomocy Wizardów

Wybierając Files -> New File, otrzymujemy na ekranie listę wzorców i tzw. *wizardów* pomocnych przy tworzeniu nowego pliku.

Wizard to po angielsku dobra wróżka. „Prowadzi za rękę” użytkownika aż do stworzenia określonej aplikacji. Do wyboru są tutaj dwa wizardy:

- Report Wizard - pomaga utworzyć obiekt, zawierający tabelę z danymi z wybranych pól baz danych. Obiekt ten może być wyposażony w przyciski nawigacyjne oraz pole wypełnialne, do określenia warunków wyszukiwania rekordów. Przykład takiego obiektu przedstawiono na rysunku 3. Wydruk przykładu - patrz dodatek A1.

Cust-Num: Search

Cust-Num	Name	Address
4	Go Fishing Ltd	Unit 2
5	Match Point Tennis	66 Homer Ave
6	Fanatical Athletes	20 Bicep Bridge Rd
7	Aerobics valine KY	Peltolantie 2
8	Game Set Match	Box 60
9	Pihitupaan Pyora	Putkontie 2
10	Just Joggers Limited	Fairwind Trading Est
11	Keilailu ja Biliardi	Vattuniemenkuja 8 A
12	Surf Lautaveikkoset	Venemestarinkatu 35
13	Biliardi ja tennis	Urheilutie 1

Rysunek 3. Obiekt typu Report Wizard

- Detail Wizard - pomaga stworzyć obiekt, zawierający elementy do edycji rekordów ich tworzenia i kasowania. Podobnie jak Report Wizard może zawierać przyciski nawigacyjne oraz pole wypełnialne, do określenia warunków wyszukiwania rekordów (patrz rysunek 4). Wydruk przykładu - patrz dodatek A2.

Cust-Num: Search

Cust-Num:

Name:

Address:

Rysunek 4. Obiekt typu Detail Wizard

2.2. Narzędzia

Rysunek 2 przedstawia ekran Workshopa po wybraniu opcji *Tools*, czyli - Narzędzia.

Bardzo użytecznym narzędziem jest *Scripting Lab*. Służy ono do testowania fragmentów aplikacji. Użytkownik ma natychmiastową możliwość sprawdzenia działania elementów programu.

File Tools - to zestaw narzędzi do przeglądania, otwierania/zamykania, edycji, kompilacji, uruchamiania, zapisywania aplikacji itd. Poniższy rysunek 5 przedstawia pasek z narzędziami.



Rysunek 5. Narzędzia do pracy z plikami

2.3. Włączanie kodu języka SpeedScript do dokumentu HTML

Utwórzmy nowy plik typu *Blank (Simple HTML File with Blank WebSpeed Script)*. Na ekranie pojawi się pasek z podstawowymi funkcjami do tworzenia i testowania aplikacji. Do tworzenia programu przechodzi się, wybierając ikonkę *Edit*. Nowo utworzony plik zawiera już znaczniki języka HTML, definiujące część nagłówkową oraz część główną, w której można osadzić elementy języka SpeedScript pomiędzy znacznikami `<script language="SpeedScript"> ... </script>`.

Posłużymy się tutaj danymi z przykładowej bazy danych *Sports*, która zawiera m.in. tablicę *Customer* z danymi klientów (np. pola *Cust-num* i *Name*). Spróbujmy wyświetlić te dane dla klientów, których nazwa zaczyna się na literę „A”.

```
<html>
<head>
<meta name="AUTHOR" content="Your Name">
<title>WebSpeed Script</title>
</head>
```

```
<body>
<script language="SpeedScript">
```

```
FOR EACH customer WHERE customer.name BEGINS "a":
    DISPLAY {&WEBSTREAM} customer.cust-num
customer.name.
END.
</script>
</body>
</html>
```

Efekt działania powyższego programu przedstawia rysunek 6.

W systemie WebSpeed zdefiniowane są dwa strumienie wyjścia, służące do wyświetlania danych {&WEBSTREAM} i {&DISPLAY}. Efekt działania poniższych dwóch poleceń będzie identyczny:

```
{&DISPLAY} cust-num name.
```

```
DISPLAY {&WEBSTREAM} cust-num name.
```

Cust-Num Name	
37	ABC Mountain Bikes
7	Aerobics valine KY
1015	Alfa
18	Antin Metsestysase
73	Auffi Bergausrustung

Rysunek 6.

Spróbujmy teraz zmodyfikować program, tak aby te same dane były wyświetlane w tabeli, jaką można utworzyć przy pomocy odpowiednich znaczników HTML. Najistotniejsze jest tutaj rozdzielenie poleceń języka SpeedScript i tagów HTML. W tym przypadku istnieją dwa polecenia, które muszą znajdować się pomiędzy znacznikami <script...> i </script>; otwierające i zamykające pętlę FOR EACH. Nazwy pól muszą znajdować się pomiędzy znakami ` (apostrof na tym samym klawiszu co tylda), w przeciwnym wypadku na ekran zostanie wyprowadzony tekst nazw pól, a nie dane z tabeli.

W efekcie na ekranie pojawi się tabela przedstawiona na rysunku 7.


```

<html>
<head>
<meta name="AUTHOR" content="Your Name">
<title>WebSpeed Script</title>
</head>
<body>

<table border=2>
  <tr>
    <th align=left> Nr </th>
    <th align=left> Nazwa </th>
  </tr>
  <script language="SpeedScript">
    FOR EACH customer WHERE customer.name
    BEGINS"a":
  </script>
  <tr>
    <td align=left> `customer.cust-num` </td>
    <td align=left> `customer.name` </td>
  </tr>
  <script language="SpeedScript">
    END.
  </script>
</table>

</body>
</html>

```

Nr	Nazwa
37	ABC Mountain Bike
7	Aerobics valne KY
1015	Alfa
18	Antin Metsastysase
73	Auffi Bergausrustung

Rysunek 7.

2.4. Aplikacje typu HTML Mapping

Tworzenie aplikacji typu HTML Mapping jest bardziej skomplikowanym procesem od poprzedniej metody, jednakże można dzięki niej stworzyć bardziej złożone aplikacje, obsługujące zapytania podawane przez użytkownika w sposób interaktywny.

Pierwszym krokiem takiej metody jest stworzenie pliku HTML z elementami formy, zawierającej obiekty typu: pola do wprowadzania danych, przyciski, itp. Plik taki można stworzyć przy pomocy edytora Workshop albo dowolnego edytora stron HTML czy plików tekstowych. W tych dwóch ostatnich przypadkach należy przesłać utworzony plik HTML na serwer np. przy użyciu programu *ftp*.

Poniższy przykład *p-map.html* zawiera trzy pola typu fill-in : numer, nazwa i adres oraz dwa przyciski z etykietami << oraz >>. Każdy taki element elementarny posiada nazwę, przy

czym nazwa obu przycisków jest taka sama *btn*. Jest to konieczne aby w późniejszej fazie tworzenia aplikacji można było rozpoznawać, który z przycisków został naciśnięty.

```
<html>
<head>
<title>Dane klienta</title>
</head>

<body>
<h1>Dane klienta</h1>
<br>
<form method="POST" action="p-map.w">
  <pre>
    Numer: <input type="text" size="9" name="numer">
    Nazwa: <input type="text" size="20" name="nazwa">
    Adres: <input type="text" size="21" name="adres">
  <br>
    <input type="submit" name="btn" value="<<">
    <input type="submit" name="btn" value=">>">
  </pre>
</form>
</body>
</html>
```

Następnym krokiem jest utworzenie na bazie pliku *html* pliku typu *Web Object* (z rozszerzeniem *.w*). Nazwa tego pliku stanowi wartość parametru ACTION w definicji formy.

Kolejną czynnością jest tzw. mapowanie w czyli przypisanie elementom formy (HTML Field) określonych pól z bazy danych lub zmiennych (WebSpeed Field) (patrz rysunek 8).

HTML Field	HTML Type	WebSpeed Field	Field Source
<input type="text" value="adres"/>	input text	Customer.Address	<input checked="" type="radio"/> Local <input type="radio"/> DB <input type="radio"/> User
<input type="text" value="nazwa"/>	input text	Customer.Name	<input checked="" type="radio"/> Local <input type="radio"/> DB <input type="radio"/> User
<input type="text" value="numer"/>	input text	Customer.Cust-Num	<input checked="" type="radio"/> Local <input type="radio"/> DB <input type="radio"/> User

Rysunek 8. Mapowanie.

Nowo powstały plik *p-map.w* zawiera procedurę output-header, służącą m.in. do ustawiania stanu obiektów (state-less lub state-aware) oraz procedurę process-web-request, dzięki której można obsługiwać konkretne żądanie. Poniżej przedstawiono wydruk tej procedury po usunięciu komentarzy.

PROCEDURE process-web-request :

RUN output-header.

IF REQUEST_METHOD = "POST":U THEN DO:

```
    RUN dispatch IN THIS-PROCEDURE ('input-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('assign-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('display-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('enable-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('output-fields':U).
END.

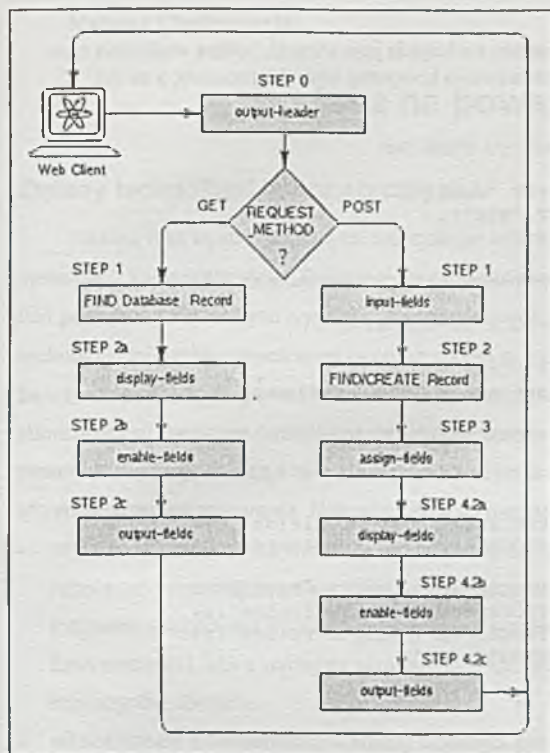
/* Form has been submitted. */

/* REQUEST-METHOD = GET */
ELSE DO:
    RUN dispatch IN THIS-PROCEDURE ('display-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('enable-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('output-fields':U).
END.
END PROCEDURE.
```

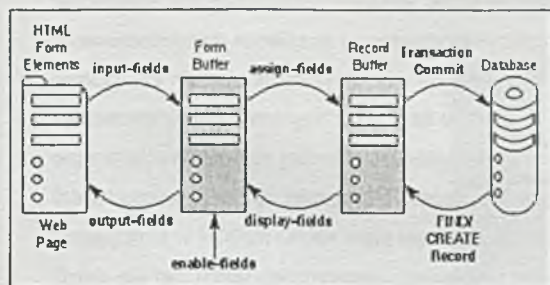
Istotne jest czy żądanie jest obsługiwane po raz pierwszy. Jeśli tak to żądanie jest przesyłane metodą GET, w przeciwnym wypadku metodą POST. Rysunek 9 przedstawia model obsługi żądania w procedurze process-web-request.

- output-header - wyprowadzenie nagłówka HTTP
- input-fields - przeniesienie danych z formy HTML do bufora pamięci w obiekcie
- assign-fields - przypisanie wartościom zmiennych i/lub pól rekordów wartości znajdujących się w buforze pamięci obiektu
- display-fields - przeniesienie wartości zmiennych i/lub pól rekordów do bufora pamięci obiektu
- enable-fields - procedura, umożliwiająca użytkownikowi modyfikację zmiennych
- output-fields - wyprowadzenie strony wraz z danymi do serwera sieciowego

Rysunek 10 ilustruje sposób przepływu danych pomiędzy elementami formy, buforem pamięci obiektu i buforem rekordów bazy danych.



Rysunek 9. Model obsługi zapytania w aplikacji typu HTML Mapping.



Rysunek 10. Model przepływu danych poprzez *Web Object*.

Do odczytu danych z elementów formy HTML służy instrukcja Get-Field.

Jako parametr podaje się nazwę elementu a odczytuje jego wartość, będącą wielkością typu znakowego (character). Poniżej przedstawiono kompletny wydruk procedury, a na rys. 11 gotowa aplikacja.

```

IF REQUEST_METHOD = "POST":U THEN DO:

    vNumer = Get-Field(INPUT "Numer").
    vbtn = Get-Field(INPUT "btn").

IF vbtn = ">>" THEN DO:
    FIND customer WHERE cust-num = INTEGER(vNumer) NO-LOCK
    NO-ERROR.
    FIND NEXT customer NO-LOCK NO-ERROR.
END.
IF vbtn = "<<" THEN DO:
    FIND customer WHERE cust-num = INTEGER(vNumer) NO-LOCK
    NO-ERROR.
    FIND PREV customer NO-LOCK NO-ERROR.
END.

/*
    RUN dispatch IN THIS-PROCEDURE ('input-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('assign-fields':U).
*/

/*
    RUN dispatch IN THIS-PROCEDURE ('display-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('enable-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('output-fields':U).
END. /* Form has been submitted. */

/* REQUEST-METHOD = GET */
ELSE DO:

    FIND FIRST customer NO-LOCK.
    RUN dispatch IN THIS-PROCEDURE ('display-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('enable-fields':U).
    RUN dispatch IN THIS-PROCEDURE ('output-fields':U).
END.
END PROCEDURE.

```

Numer: 1

Nazwa: Lift Line Skiing

Adres: 276 North Street

< >

Rysunek 11. Gotowa aplikacja

Java na poważnie

Zmiany technologii informatycznych

Rozwój firm wykorzystujących technologie informatyczne wymusza zmiany w systemach komputerowych. Zmiany technologiczne mają najczęściej jako podstawę fakt pojawienia się nowych wymagań ekonomicznych. Jednocześnie użytkownicy technologii informatycznych wiedzą coraz więcej o jej możliwościach, co powoduje, że wzrastają ich wymagania i oczekiwania. Dla organizacji, w których system informatyczny decyduje o możliwości funkcjonowania na konkurencyjnym rynku, zmiany technologiczne są ściśle powiązane z czynnikami ekonomicznymi tworzenia i utrzymania oprogramowania. Najważniejsze wymagania to:

- szybsze odpowiedzi na zmiany – użytkownicy żądają szybszego uaktualniania rozwiązań informatycznych w odpowiedzi na częste zmiany wymagań rynkowych. Potrzebują środowisk do szybkiego tworzenia aplikacji (Rapid Development Environments), które wydajnie skracają cykl ich tworzenia, oraz elastycznych magazynów danych.
- niższe koszty informatyzacji – koszty działania prawie zawsze były i są decydującym czynnikiem w budowie sukcesu organizacji. Najważniejsze jednak, aby równoważyć koszty z korzyściami jakie powinny zapewnić. W przypadku technologii informatycznych wyraźnie widać wzrost kosztów przy wyborze zaawansowanych rozwiązań i w zmaganiach z konkurencją.
- współpraca różnorodnych systemów – wymiana informacji dostęp do nich mimo zastosowania różnorodnych rozwiązań technicznych. Rzadko się zdarza, że duża organizacja dysponuje jednolitą architekturą systemów informatycznych. Infrastruktura powstaje najczęściej etapami, w ciągu wielu lat. Wybrane rozwiązania w krótkim czasie stają się niewystarczające do rosnących potrzeb. Dzieje się tak, mimo dokonywania „zakupów z zapasem” środków technicznych. Zmiany w standardach informatycznych, wzrost wymagań, ekonomiczne uwarunkowania wymuszają ewolucję przyjętych rozwiązań. Stąd kluczowym

wydaje się możliwość współdziałania oprogramowania i sprzętu z różnych okresów technologicznych.

- Niskie koszty utrzymania systemu. Rozbudowany system wymaga nie tylko zaawansowanych rozwiązań technicznych, ale również wykwalifikowanego personelu, który będzie odpowiedzialny za zapewnienie ciągłości funkcjonowania systemu, za pilnowanie poprawności konfiguracji, za dostarczanie nowych wersji oprogramowania, za konserwację sprzętu informatycznego. Wysoko wykwalifikowane kadry są kosztownym elementem firmy. Stworzenie dobrego zespołu trwa długo i kosztuje немало.

Siła języka Java

Java obiecuje realizację opisanych postulatów. Stanowi kombinację wysokiej produktywności tworzenia programów, implementacji raz napisanego kodu w wielu środowiskach, niemal zerowych kosztów administracji oprogramowania stacji klienta. Ponadto Java ma wbudowane mechanizmy obsługi sieci, co znacznie redukuje kłopoty z tworzeniem i instalacją rozproszonych systemów. Wbudowane mechanizmy komunikacji z bazami danych znacznie podnoszą możliwości tworzenia profesjonalnych aplikacji.

- Produktywność – Java wygląda jak C++, co pozwala wielkiej rzeszy programistów łatwo zmienić środowisko pracy. Jest pozbawiona mechanizmów, które w C++ stanowią problem, takich jak wskaźniki, czy wielodziedziczenie.
- Raz stworzona aplikacja może funkcjonować w wielu środowiskach – Fakt uruchamiania aplikacji w Javie wewnątrz „wirtualnej maszyny” pozwala aby program poprawnie działał na wielu platformach systemowo-sprzętowych. „Maszyna wirtualna Javy” jest wbudowana w większość przeglądarek WWW.
- Brak potrzeby administracji stacją klienta. Aplety Javy mogą być obsługiwane wewnątrz przeglądarki automatycznie, tak że administracja systemem klienta praktycznie nie istnieje. Dopóki działa przeglądarka lub inna „maszyna wirtualna Javy” tak długo kod Javy może być wykonywany. Nie potrzeba instalować żadnych bibliotek DLL czy innych komponentów. Powoduje to, że instalacja aplikacji opartych o Javę może odbywać się poprzez sieć bez żadnych dodatkowych zabiegów.

- Wbudowane mechanizmy sieciowe – Java wspiera protokoły TCP/IP i HTTP, tak że użycie sieci w ramach aplikacji nie wymaga skomplikowanego programowania tak jak w C++ i wprowadza standard na obsługę sieci. Z punktu widzenia programisty w Javie, dostęp do sieci jest tak prosty jak dostęp do lokalnych plików.
- Połączenia z bazami danych – JDBC (Java Data Base Connectivity) stanowi standard dostępu do baz danych za pomocą SQL, podobnie jak poprzez ODBC. JDBC jest zaprojektowane (inaczej niż ODBC) tak, aby możliwe było załadowanie przez sieć wszystkich komponentów niezbędnych do połączenia z bazą danych. Wszystkie te własności Javy świadczą o jej dużym potencjale przy tworzeniu i implementowaniu przemysłowych aplikacji.

Ewolucja technologii Javy

Techniki i narzędzia Javy rozwijają się niezwykle szybko. W ciągu dwóch lat narzędzia programistyczne, środowiska implementacji, biblioteki klas szybko znalazły się na rynku. Ta pierwsza generacja produktów dostarczyła tylko część z obiecywanych przez Javę zalet. Większość produktów nie była adresowana do budowania korporacyjnych systemów. Narzędzia te najczęściej były nieadekwatne do profesjonalnych zastosowań ze względu na niedoskonałość środowiska programistycznego, brak serwerów aplikacyjnych będących w stanie wykonywać logikę biznesową aplikacji wyrażoną w języku Java, niedoskonałość czy wręcz brak efektywnych sterowników JDBC łączących aplikacje w Javie ze standardowymi bazami relacyjnymi.

Środowisko programistyczne Javy

Tworzenie aplikacji w języku Java odbywa się w środowisku łączącym programowanie za pomocą wpisywania kodu z graficznym manipulowaniem komponentami aplikacji. Ewolucja pierwszych narzędzi programistycznych jest niezbędna, aby można je było zaliczyć do profesjonalnych. I tak, muszą pozwalać na tworzenie zarówno apletów, jak i obiektów funkcjonujących po stronie serwera aplikacji i realizujących logikę przetwarzania informacji (w tym transakcje); muszą organizować powstawanie złożonych systemów, a nie tylko prostych obiektów; muszą być przystosowane do pracy zespołowej nad rozległym systemem; powinny obsługiwać wiele standardów komponentów (ActiveX, JavaBeans, CORBA) ;

stanowić kompletny system z narzędziami do debugowania, połączeniami do baz danych etc.

Budowanie serwera aplikacyjnego

Serwer aplikacyjny stanowi spójną warstwę pośredniczącą pomiędzy aplikacją klienta, a bazami danych funkcjonującymi w organizacji. Poprzez zarządzanie logiką przetwarzania informacji pozwala tworzyć aplikacje klienckie odpowiedzialne jedynie za wizualizację danych, nie wymagające kosztownego utrzymania. Zagadnienie to jest szczególnie wyraźne przy złożonych, korporacyjnych systemach, gdzie koszt utworzenia aplikacji są znacznie mniejsze od kosztów jej eksploatacji. Monitory transakcyjne dostarczają technik zapewniających wydajność i skalowalność systemów. Jednak ich specyficzna dla każdego producenta natura utrudnia tworzenie uniwersalnych systemów i administrowanie nimi. Brokery obiektów CORBA stanowią dobry model inicjowania rozproszonych serwerów. Są jednak skomplikowane i często kosztowne. Ponadto nie zapewniają efektywnego dostępu do baz danych. Udostępniają raczej obiekty niż znacznie wygodniejsze wyniki obliczeń i poszukiwań.

Serwery Web stanowią wygodną warstwę pośrednią dla systemów opartych o Internet/intranet. Aplikacje oparte o udostępniania statycznych lub dynamicznych stron WWW są łatwe do stworzenia i praktycznie nie wymagają kosztownej administracji komputerem użytkownika. Niezbędna jest właściwie tylko przeglądarka internetowa. Rozwiązanie to ma jednak wady. Nie zapewnia transakcyjności, nie zapamiętuje stanu sesji użytkownika, nie zapewnia dużej wydajności (nie wykorzystuje wieloprocesorowej architektury komputerów, nie jest zoptymalizowane na dostęp do baz danych). Rozwiązaniem mogą tu być serwery transakcyjne. Ważne jednak, aby były tak zaprojektowane żeby obsługiwać tysiące użytkowników, a nie tylko kilku. Ważne, aby obsługiwały wiele modeli obiektów (w tym JavaBeans) stworzonych różnymi narzędziami. Ważne, aby pozwalały realizować nie tylko synchroniczne transakcje ale i też asynchroniczne na zasadzie „fire and forget”. I wszystko to za pomocą protokołów: TDS, IIOP, DCOM, HTTP.

Połączenia z bazami danych

Systemy informatyczne, stworzone w Javie, komunikują się z bazami danych za pomocą standardowego interfejsu – JDBC (Java Database Connectivity). Standard

ten pozwala tworzyć aplikację w Javie niezależną od typu bazy danych, z którą aplikacja będzie się komunikować. Wiele jest rodzajów rozwiązań realizujących połączenie z bazą danych w standardzie JDBC:

- Połączenia JDBC – ODBC. Połączenie to zrealizowane przez JavaSoft daje możliwość komunikacji z wieloma bazami danych. Wystarczy, że będzie dostępny sterownik ODBC, a można już z programu w Javie wymieniać informacje z bazą danych. Rozwiązanie to ma zaletę i wadę. Zaletą jest dostęp do ogromnej liczby baz danych. Istnieją sterowniki ODBC, dla prawie każdej bazy danych. Wadą jest skomplikowanie i ograniczenia funkcjonalności i wydajności. Dzieje się to dlatego, że informacje z programu w Javie docierają do bazy danych długą drogą. Najpierw wywołanie metody JDBC musi zostać skonwertowane na wywołanie ODBC, a to z kolei na wywołanie funkcji dedykowanego interfejsu bazy danych, który poprzez specjalizowany protokół sieciowy odwoła się do serwera bazy danych.
- Połączenie Javy z wywołaniami dedykowanego interfejsu bazy danych. Sterowniki takie są napisane najczęściej w kombinacji języka Java i C/C++. Wymagają one podobnie jak połączenia JDBC – ODBC zainstalowania odpowiednich bibliotek napisanych w C (dostarczonych przez producenta bazy danych). Mamy do czynienia z podobnymi kosztami eksploatacji, jak przy zwykłej aplikacji klient/serwer.
- Połączenia Javy przez serwer aplikacyjny. Aplikacja komunikuje się ze sterownikiem napisanym w Javie. Sterownik, za pomocą niezależnego od bazy danych protokołu komunikuje się z serwerem aplikacyjnym. Ten z kolei tłumaczy polecenia i zapytania na standard bazy danych i posługując się mechanizmami komunikacyjnymi specyficznymi dla konkretnej bazy danych przesyła je do serwera. Rozwiązanie takie często jest niewydajne przy obciążeniu zadaniami wielu użytkowników ze względu na ograniczenia wydajności serwera aplikacyjnego tłumaczącego wywołania do bazy danych.
- Dedykowane sterowniki napisane w Javie. Sterownik taki zamienia wywołania funkcji bezpośrednio na protokół sieciowy specyficzny dla bazy danych. Komunikacja odbywa się bezpośrednio, z pominięciem ODBC, serwerów aplikacyjnych, bibliotek komunikacji z bazą danych.

Rozwiązania Sybase

Sybase oparł swoją strategię (Adaptive Component Architecture) na komponentach budowanych w języku Java. Proponuje różne rozwiązania, które mogą w początkowym etapie uzupełnić, a w przyszłości zastąpić inne technologie. Ideą tej strategii jest budowanie komponentów odpowiedzialnych za logikę biznesową i transakcyjną aplikacji, a następnie instalowanie (implementowanie) ich we wszystkich warstwach aplikacji, zależnie od potrzeb wydajności i zastosowania. Począwszy od apletów uruchamianych w przeglądarkach internetowych i aplikacji napisanych w PowerJ (Java) czy PowerBuilderze, poprzez obiekty JavaBeans na serwerze transakcyjnym Jaguar i serwletry w serwerze WWW, a zakończywszy na obiektach w bazie danych.

PowerJ

PowerJ jest środowiskiem typu RAD do budowania aplikacji w języku Java. Sposób programowania jest bardzo zbliżony do Power++ (dawniej Optima++). Zasadniczą różnicę stanowi język programowania. (W Power++ jest to C++.) Wiele jest narzędzi ułatwiających programowanie w Javie. PowerJ jest ukierunkowany na tworzenie złożonych systemów, od których zależy funkcjonowanie organizacji. Oznacza to, że nie ogranicza się do dostarczenia narzędzi do budowania interfejsu użytkownika, ale koncentruje się na warstwie logiki aplikacji, ułatwiając tworzenie obiektów odpowiedzialnych z przekształcanie informacji zgodnie z procesami funkcjonującymi w organizacji. Narzędzie pozwala tworzyć obiekty realizujące logikę biznesową w warstwie pośredniej – w serwerze aplikacyjnym. PowerJ pozwala realizować kompletne rozwiązania, a nie tylko aplety. Pozwala na tworzenie i integrowanie stron w HTML, w JavaScript, apletów, serwletów, zapytań i połączeń do baz danych. Powstałe obiekty mogą być uruchamiane, testowane i debugowane w zintegrowanym środowisku. PowerJ pozwala pracować zespołowo nad złożonymi projektami. Jest zintegrowany z ObjectCycle – narzędziem zgodnym z Microsoft SCC do zarządzania wersjami kodu programu.

Wiele organizacji planuje w przyszłości użycie języka Java, jednak obecnie mniej niż 5% pracowników ma doświadczenie w posługiwaniu się tym językiem. PowerJ, a w szczególności system pomocy, zestaw kreatorów, graficzne techniki

programowania zostały tak zaprojektowane, aby programista, który poznał podstawy Javy nie musiał spędzać wiele czasu na opanowywaniu nowego środowiska pracy.

PowerJ obsługuje dwa standardy Javy: JDK 1.10 i JDK 1.02. Zawiera mechanizmy połączenia z bazą danych – JDBC 1.21. Pozwala budować biblioteki .JAR, .ZIP, .CAB; tworzyć komponenty dla serwerów Web (CGI, NSAPI, ISAPI, Dynamo). Razem z narzędziem jest dostępna biblioteka obiektów JavaBeans i ActiveX. (Aplikacje stworzone w Javie za pomocą PowerJ mogą zawierać obiekty ActiveX.)

Jaguar CTS

Sposoby budowania programów zmieniają się z latami. Od kodu maszynowego poprzez różne języki, najpierw strukturalne, potem obiektowe aż po składanie systemu z gotowych komponentów. Jeśli głębiej się zastanowić, to programiści używają coraz wyższego poziomu abstrakcji przy wyrażaniu procesów, które oprogramowują. Pozwala to znacznie szybciej budować systemy, pomijając szczegóły implementacji komponentów. Niewielu jest takich, którzy tworzą kompilatory, bazy danych, piszą algorytmy sortowania, wyszukiwania. Najczęściej oprogramowują gotowe komponenty, tworzą zapytania do baz danych. Generalnie zajmują się tylko tym co specyficzne dla danego systemu i tym co stanowi adaptację istniejących standardowych komponentów do potrzeb powstającego oprogramowania. Programowanie za pomocą komponentów wymaga środowiska, w którym będą uruchamiane. Jaguar stanowi takie środowisko. Jest jednocześnie serwerem transakcyjnym i miejscem gdzie mogą funkcjonować i wymieniać informacje obiekty wielu standardów: JavaBeans, ActiveX, CORBA, C++. Komunikacja w ramach jednej sesji może odbywać się jednocześnie za pomocą wielu protokołów: HTTP, DCOM, IIOP, TDS. Jaguar jest środowiskiem do tworzenia zaawansowanych systemów w architekturze wielowarstwowej, posiada wewnętrzną wielowątkową, wieloprocessorową architekturę zapewniającą skalowanie wraz ze wzrostem wymagań stawianych serwerowi. Zapewnia za pomocą mechanizmu kolejowania (dbQueue) asynchroniczne wykonywanie transakcji, bezpieczeństwo poprzez szyfrowanie, autentyfikację, cyfrowe podpisy (RSA, Kerberos, X509). Jest dostępny na platformy Sun Solaris, HP-UX, NT. Dzięki temu pozwala budować zaawansowane aplikacje internetowe (NetOLTP).

jConnect

jConnect jest sterownikiem obsługującym połączenie do baz danych ze środowiska Javy. Jest napisany tylko w języku Java. Tłumaczy odwołania do bazy danych bezpośrednio na protokół sieciowy TDS bezpośrednio przesyłany do bazy danych. Zapewnia bezpośredni dostęp do baz danych Sybase, a za pomocą DirectCONNECT do innych źródeł. Nie wymaga specjalnej instalacji. Może być załadowany (ok. 200 kB) razem z apletem do przeglądarki internetowej i natychmiast uruchomiony.

Java w bazie danych

Sybase udostępnia architekturę języka Java w najnowszej generacji swoich relacyjnych baz danych, rodzinie Adaptive Server. Takie rozwiązanie otwiera nowe poziomy zaawansowania i otwartości w programowaniu dla serwera oraz nowy wymiar elastyczności przy tworzeniu i uruchamianiu aplikacji. Jednocześnie dostarcza otwarty, obiektowo – relacyjny system zarządzania bazą danych.

Do tej pory logika w bazie danych opierała się na SQL-owych procedurach wbudowanych, natomiast logika biznesowa aplikacji poza systemem DBMS była implementowana w językach 3GL (C/C++ lub Java) lub 4GL (PowerBuilder, Visual Basic). Java w bazie danych usuwa barierę pomiędzy programowaniem dla strony klienta i serwera. Umożliwia wielokrotne wykorzystanie raz napisanego kodu i uruchamianie go w dowolnej warstwie aplikacji, w tym na serwerze, uzupełniając pewne niedociągnięcia i niedogodności SQL.

Chcąc zapewnić ciągłość i otwartość swoich rozwiązań, Sybase współpracuje z JavaSoft, komiteetem standardu ANSI SQL oraz konsorcjum JSQL w celu ustalenia standardu wykorzystania Javy w bazach danych.

Sybase uważa, że Java jest również kluczem do poprawy produktywności programowania dla serwerów bazodanowych i rozszerzenia ich możliwości, poprzez zapewnienie:

- spójnego, zaawansowanego języka programowania dla wszystkich warstw architektury;
- naturalnej implementacji;
- wbudowanych mechanizmów bezpieczeństwa;
- podejścia obiektowego.

SQL nadal pozostaje doskonałym środkiem do manipulacji i definicji danych. Java zastąpi natomiast SQL-owe procedury wbudowane zapewniając szereg nowych udogodnień. Klasy Javy raz zainstalowane na serwerze mogą być wielokrotnie uruchamiane i debugowane dzięki wbudowanej maszynie wirtualnej. Dostęp z Javy do SQL zapewnia wewnętrzny interfejs JDBC. Na serwerze może być uruchamiany prekompilowany kod pisany w JSQL, klasy Javy tworzone za pomocą popularnych narzędzi RAD (PowerJ) i standardowe komponenty JavaBeans. Klasy Javy nie muszą być budowane specjalnie dla serwera.

Java na poważnie

Wracając do tytułu tego artykułu zastanówmy się jaką rolę odegra Java w rozwoju systemów informatycznych. Wiele poważnych firm informatycznych inwestuje w środowisko tego języka duże środki. Opisane produkty Sybase stanowią część dokonań firm informatycznych, jakie można spotkać próbując znaleźć rozwiązanie problemów z utrzymaniem systemów, ich szybką adaptacją do zmieniających się wymagań, realizacją transakcji w Internecie. Ich sukces rynkowy, tak jak i innych narzędzi związanych z Javą zależy od tego, jak wiele poważnych organizacji zdecyduje się na tę technologię.

Pojawiające się pytania, czy Java zastąpi inne języki programowania, czy pozostanie jako narzędzie programowania apletów wzbogacających wygląd stron WWW, czy model komponentów JavaBeans wyznaczy trwały standard programowania, czy JDBC stanie się standardowym interfejsem dostarczającym z każdą liczącą się bazą danych. Na te wszystkie pytania trudno jest dać prostą odpowiedź i jednocześnie mieć stuprocentową pewność. Burzliwy rozwój Internetu, zmienił w ciągu ostatnich lat kierunki rozwoju informatyki. Nikt nie uważa Internetu za chwilową modę, za kolejny wynalazek bez dużego znaczenia dla cywilizacji. Wszyscy uważają, że techniki internetowe nie mogą być pominięte przy wyborze technologii tworzenia systemów. Stąd najprawdopodobniej język Java nadal będzie odgrywać istotną rolę. Czy wyprze inne języki programowania? Raczej nie. Tak jak COBOL przetrzymał programowanie w C/C++, architekturę klient/serwer tak samo obecne techniki programowania będą nadal funkcjonować. Stanie się to dlatego, że istnieje wiele systemów, których przerabianie na nową technologię tylko dlatego, że jest nowocześniejsza nie jest ekonomicznie uzasadnione. Natomiast nowe systemy, szczególnie te, które poprzez swój związek z Internetem, będą realizować swoją

misję w nowoczesnych organizacjach, będą oparte o język i środowisko Javy. Wykorzystanie Javy nie ograniczy się do budowania interfejsu użytkownika, ale poprzez model obiektów takich jak JavaBeans pozwoli budować serwery aplikacyjne odpowiedzialne za większość przetwarzania informacji. Tutaj może najlepiej sprawdzić się uniwersalność Javy. Raz napisany kod może funkcjonować na różnorodnych platformach, komponenty można tak dobierać, aby dostosować system do indywidualnych wymagań. Wszystko to minimalnym nakładem kosztów na czynności administracyjne, konfiguracyjne, a z zapewnieniem skalowalności i bezpieczeństwa. Mając wątpliwości czy Java to właściwe środowisko do tworzenia poważnych systemów warto przyjrzeć się narzędziom dedykowanym do tego środowiska. Tematy dokumentacji z nimi dostarczonej, moduły odpowiedzialne za zaawansowane techniki programowania, gotowe komponenty wyraźnie pokazują, że są to rozwiązania profesjonalne i dedykowane do tworzenia poważnych systemów.

Bezpieczne sieci korporacyjne

Michał Kasprzak
ATM S.A.
ul. Grochowska 21a
04-183 Warszawa
michal@atm.com.pl

Bezpieczeństwo infrastruktury sieciowej

Zapewnienie bezpieczeństwa danych w rozległej sieci korporacyjnej jest zadaniem bardzo trudnym. Na ogół komplikuje je różnorodność wykorzystywanych technologii transmisji danych obejmujących dzierżawione połączenia kablowe, kanały cyfrowe, czy publiczne sieci pakietowe, jak np. Frame Relay. Zapewnienie kryptograficznej ochrony danych przesyłanych taką siecią często wiąże się z koniecznością dokonania kompleksowej analizy potrzeb, zagrożeń, poziomu ryzyka i uwarunkowań organizacyjno-ekonomicznych. Dopiero taka analiza pozwala wybrać technologię zabezpieczeń. Zasadniczo szyfrowanie może następować na poziomie poszczególnych łączy danych, albo poszczególnych pakietów.

Szyfrowanie połączeń

To podejście polega na stosowaniu modułów szyfrujących i deszyfrujących dla każdego łącza punkt-punkt, czy to dzierżawionego fizycznego połączenia, czy wirtualnego kanału cyfrowego. Jest to technika godna polecenia w przypadku infrastruktury bazującej na urządzeniach, które po dodaniu pewnych modułów programowych zapewniają szyfrowanie łączy. Takie możliwości nie zawsze jednak istnieją, albo wiążą się ze spowolnieniem pracy urządzeń sieciowych. W takich sytuacjach konieczne jest wyposażenie każdego połączenia w parę dodatkowych urządzeń szyfrujących.

Podstawową zaletą tego rozwiązania jest przezroczystość zabezpieczeń kryptograficznych z punktu infrastruktury WAN. Wdrożenie takiego zabezpieczenia może następować stopniowo i nie wymaga skoordynowania z odpowiednimi zmianami systemu przekazywania pakietów w sieci rozległej.

Szyfrowanie pakietów

W tym podejściu szyfrowanie realizowane jest przez dodatkowe urządzenia umieszczone pomiędzy siecią lokalną każdego węzła i routerem dostępowym. Przy takim rozwiązaniu wszystkie dane przekazywane elementom aktywnym sieci rozległej są już w formie zaszyfrowanej.

Podstawową zaletą tego podejścia jest rozdzielenie funkcji szyfrowania i fizycznej transmisji danych. Dzięki temu wszelkie zmiany w infrastrukturze WAN, jak na przykład zmiana technologii transmisji danych na pewnych relacjach, nie muszą pociągać za sobą konieczności wymiany urządzeń szyfrujących.

Podstawową wadą tego rozwiązania jest konieczność modyfikacji schematu routowania pakietów w sieci rozległej, gdyż szyfrowanie na poziomie pakietów nie jest rozwiązaniem w pełni przezroczystym.

Ochrona przed zagrożeniami wewnętrznymi

W przypadku dużych instytucji zatrudniających wielu pracowników należy poważnie rozważyć problem wewnętrznych zagrożeń. Z tego względu wskazanym jest stosowanie, nawet w ograniczonym zakresie, systemów monitorowania ruchu w sieci (ang.: *intrusion detection tool*) służących do wykrywania prób ataków na systemy komputerowe. W ramach działań prewencyjnych wskazanym jest używanie narzędzi do automatycznej kontroli poprawności konfiguracji urządzeń sieciowych i systemów komputerowych.

Bezpieczny intranet i poczta korporacyjna

Zagwarantowanie bezpieczeństwa danych przekazywanych w poczcie elektronicznej i serwisach opierających się na systemie WWW w dużym przedsiębiorstwie wymaga stosowania kompleksowego systemu zabezpieczeń. Rozwiązanie to powinno obejmować:

1. system mocnego uwierzytelnienia użytkowników, bazujący na połączeniu kart identyfikacyjnych (ang. *token cards*) lub inteligentnych kart mikroprocesorowych (ang. *smart cards*) z poufnymi hasłami dostępu; system taki wymaga zastosowania centralnego serwera uwierzytelniającego;
2. serwer WWW spełniający szereg wymogów dotyczących bezpieczeństwa, jak wsparcie standardowych rozwiązań kryptograficznej ochrony danych, możliwość rejestracji zdarzeń, audyt, odpowiednio skuteczny mechanizm kontroli praw dostępu do poszczególnych zasobów;
3. rozwiązania dla klientów – przeglądarki WWW pozwalające na mocną ochronę kryptograficzną

- połączeń w standardzie HTML bazującą na protokole SSL albo SSH;
4. serwer pocztowy gwarantujący bezpieczeństwo przekazywanych przesylek i odpowiedni poziom niezawodności serwisu;
 5. oprogramowanie do obsługi poczty elektronicznej oferujące możliwość zarówno mocnego szyfrowania korespondencji, jak i kryptograficznego potwierdzenia tożsamości nadawcy – podpisów elektronicznych;
 6. serwer certyfikatów pozwalających jednoznacznie przypisać kryptograficzne kody wykorzystywane do szyfrowania i podpisów cyfrowych z konkretną osobą;
 7. system inteligentnych kart mikroprocesorowych (*ang. smart cards*), umożliwiających szyfrowanie i elektroniczne podpisy bez narażania użytkowników na niebezpieczeństwo kradzieży prywatnych kodów kryptograficznych;
- w rozwiązaniu tym odszyfrowanie poczty bądź elektroniczne podpisanie dokumentu wymaga wylegitymowania się posiadaniem osobistej karty mikroprocesorowej i znajomością kodu dostępu do karty.

Podsumowanie

Powszechnie wykorzystywane protokoły transmisji danych oraz rozwiązania aplikacyjne nie zapewniają należytego poziomu bezpieczeństwa powierzonym im informacjom.

Celem tego referatu było zademonstrowanie, że istnieje szereg technik, które pozwalają istotnie podnieść poziom bezpieczeństwa danych w sieciach korporacyjnych. Posłużono się przykładami kryptograficznych zabezpieczeń łączy dalekobieżnych oraz technik ochrony poczty elektronicznej i zasobów intranetowych. Rozwiązania technologiczne są dostępne już dziś, pozostaje tylko umiejętnie je wykorzystać.

PTI*Zarząd Główny**ul. Żelazna 87**00-879 Warszawa**tel./fax (0-48-22) 652-32-59*

CZŁONKOWIE WSPIERAJĄCY POLSKIE TOWARZYSTWO INFORMATYCZNE

*IBM Polska Sp. z o.o.**02-652 Warszawa, Wiśniowy Business Park, ul. 1 Sierpnia 2a**OCE Poland**02-232 Warszawa, ul. Lopuszańska 53**oraz**Centralny Ośrodek Informatyki Górnictwa S.A., 40-065 Katowice, ul. Mikołowska 100**Fabryka Wtryskarek „PONAR-ŻYWIEC” S.A., 34-300 Żywiec, ul. Stolarska 21**ICL Poland Sp. Z o.o., 01-199 Warszawa ul. Leszno 21**InfoViDE Sp. z o.o., 02-119 Warszawa, ul. Pruszkowska 17**Ster-Projekt Sp. z o.o., 02-652 Warszawa, ul. Magazynowa 1**Zakłady Przemysłu Barwników „BORUTA” S.A., 95-100 Zgierz, ul. Struga 30**International Business Systems, 51-124 Wrocław, ul. H Kamieńskiego 57***OPIEKA PRASOWA****PC KURIER**

