

Cel i zakres pracy dyplomowej.....	8
1. Internet i usługa WWW.	9
1.1. Uwagi wstępne	9
1.2. Czym są: Internet i World Wide Web ?	9
1.3. Historia Internetu.....	10
1.4. Stan obecny i prognozy na przyszłość.....	11
1.5. WWW jako technologia informatyczna.....	12
1.5.1. Protokół komunikacyjny	12
1.5.2. Język HTML	12
1.5.3. Przeglądarki (browsery)	14
1.5.4. Serwery WWW	21
1.5.4.1. Uwagi wstępne	21
1.5.4.2. Istota działania	21
1.5.4.3. Platformy systemowe	22
1.5.4.4. Używane oprogramowanie.....	23
1.5.4.5. Przegląd oferowanego oprogramowania.....	24
1.5.5. Generacja dokumentu HTML (w kontekście łączenia usługi WWW z bazą danych).....	28
1.5.5.1. Wstęp	28
1.5.5.2. Generacja statyczna.....	28
1.5.5.3. Generacja dynamiczna z użyciem skryptu CGI (Common Gateway Interface)	29
1.5.5.4. Generacja dynamiczna z użyciem skryptu CGI - mapowanie katalogów	31
1.5.5.5. Generacja dynamiczna z użyciem CGI i zewnętrznych programów ..	31
1.5.5.6. Generacja dynamiczna z użyciem CGI i oprogramowania motoru bazy danych.....	32
1.5.5.7. Generacja dynamiczna z użyciem CGI i API motoru bazy danych ..	33
1.5.5.8. Generacja dynamiczna z użyciem API serwera WWW i API bazy danych	35
1.5.5.9. Rozwiązania alternatywne.....	38
1.5.6. Buforowanie informacji - W3cache	38
1.5.6.1. Opis ogólny	38

1.5.6.2. W3cache w Polsce.....	39
1.5.6.3. W3cache na świecie	41
1.5.7. Bezpieczeństwo.....	42
1.5.8. Przesyłanie dźwięku i obrazu poprzez sieć Internet.....	45
1.5.9. Nowa koncepcja usług telekomunikacyjnych	45
1.5.10. Wyszukiwanie informacji	47
1.5.10.1. Charakterystyka usługi	47
1.5.10.2. Opis funkcjonalny usługi.	49
1.6. Ekonomia Internetu i WWW	51
1.6.1. Rys historyczny: pierwsza walka o rynek	51
1.6.2. WWW jako najbardziej dochodowy rynek reklam	53
1.6.3. Usługi finansowe.....	56
1.6.3.1. Home banking	56
1.6.3.2. Inteligentne karty identyfikacyjne	59
1.6.4. Handel internetowy	61
1.6.5. WWW w statystykach.....	63
1.6.5.1. Internet w Polsce	63
1.6.5.2. Internet na świecie.....	66
1.7. Przegląd zastosowań	70
1.7.1. Alta Vista - największy serwis informacyjny.....	70
1.7.1.1. Ilość przetwarzanych danych.	70
1.7.1.2. Wydajność a udostępniane funkcje.	71
1.7.1.3. Konfiguracja sprzętowa.....	72
1.7.1.4. Bezpieczeństwo.....	72
1.7.1.5. Mechanizmy optymalizacji transmisji	74
1.7.2. Deja news - największy system obsługi news.....	76
1.7.2.1. Ilość przetwarzanych danych.	76
1.7.2.2. Wydajność a udostępniane funkcje.	76
1.7.2.3. Konfiguracja sprzętowa.....	76
1.7.3. Yahoo - najbardziej znany serwis informacyjny	77
1.7.4. TOW - polski system obrotu papierami wartościowymi.....	78
1.7.4.1. Opis ogólny	78
1.7.4.2. Konfiguracja sprzętowa.....	80

1.7.5. Panorama Firm - polski system „yellow pages” (USWest Polska).....	80
1.7.5.1. Opis ogólny	80
1.7.5.2. Konfiguracja sprzętowa.....	82
1.7.6. HAFAS - system informacji kolejowej	82
1.7.6.1. Opis ogólny	82
1.7.6.2. Konfiguracja sprzętowa.....	84
2. Analiza systemu BIS	85
2.1. Założenia i uwagi wstępne	85
2.2. Definicja problemu (Problem Statement).	85
2.3. Specyfikacja wymagań.....	85
2.4. Model obiektowy.....	88
2.4.1. Uwagi i założenia wstępne.....	88
2.4.2. Ewolucja modelu.....	88
2.4.2.1. Model wstępny	89
2.4.2.2. Model uwzględniający zróżnicowanie użytkowników systemu ze względem na prawa zapisu	91
2.4.2.3. Model poszerzający opis klasy <i>publikacja</i> o kategorie <i>przedmiot</i> i <i>słowa kluczowe</i>	92
2.4.2.4. Model zakładający istnienie zróżnicowanie poziomu uprawnień do tablic.....	95
2.4.2.5. Model poszerzający opis klasy <i>Publikacja</i> o nowe atrybuty i klasy opisujące.....	97
2.4.3. Model końcowy etapu analizy.....	98
2.4.4. Komentarz do procesu modelowania	99
2.5. Słownik danych	100
2.5.1. Klasy.....	100
2.5.2. Związki.....	102
2.5.3. Atrybuty.....	104
2.5.4. Synonimy	107
2.5.5. Inne.....	107
2.6. Model dynamiczny	107
2.6.1. Scenariusze zdarzeń i odpowiadające im ścieżki zdarzeń	107
2.6.1.1. Uwagi wstępne	107

2.6.1.2. Scenariusz I : Czytelnik dokonuje odczytu informacji o publikacji	108
2.6.1.3. Scenariusz II : Pisarz wpisuje nową publikację	108
2.6.1.4. Scenariusz III : Pisarz wpisuje nową publikację decydując się na utworzenie nowego typu publikacji.	109
2.6.1.5. Scenariusz IV : Administrator dokonuje wprowadzenia nowego użytkownika	110
2.6.1.6. Scenariusz V : Administrator wykonuje dump bazy danych (back-up)	111
2.6.1.7. Scenariusz VI : Administrator dokonuje przywrócenia stanu bazy danych z pliku ze zarchiwizowanymi danymi.....	112
2.6.1.8. Scenariusz VII : Administrator dokonuje usunięcia publikacji.....	113
2.6.1.9. Scenariusz VIII : Pisarz próbuje usunąć publikację, której sam nie wprowadził.....	114
2.6.1.10. Scenariusz IX : Użytkownik próbuje połączyć się z systemem BIS, podczas zakłóceń łączności.....	115
2.6.1.11. Scenariusz X : Administrator usuwa zbędną dziedzinę.	115
2.6.1.12. Scenariusz XI : Administrator próbuje usunąć Administratora. ...	116
2.6.2. Przykładowe diagramy przejść stanów	117
2.6.2.1. Uwagi wstępne	117
2.6.2.2. Zarys systemu.....	118
2.6.2.3. Implementacja współbieżności: pierwsze podejście	118
2.6.2.4. Implementacja współbieżności: drugie podejście	119
2.6.2.5. Diagram przepływu zdarzeń.....	119
2.7. Model funkcjonalny	127
2.7.1. Identyfikacja danych wejściowych i wyjściowych.....	127
2.8. Diagram przepływu danych.....	129
2.8.1. Opis funkcji.....	129
2.8.1.1. Sprawdzenie domeny klienta WWW	129
2.8.1.2. Odczytanie formy przez serwer WWW	130
2.8.1.3. Modyfikacja domen.....	130
2.8.1.4. Przygotowanie komunikatu o błędzie	130
2.8.1.5. Przygotowanie odpowiedzi dla użytkownika.....	130
2.8.1.6. Aktualizacja danych	131
2.8.1.7. Rozpoznanie żądania użytkownika	131

2.8.1.8. Realizacja żądania użytkownika	131
2.9. Podsumowanie procesu analizy	131
3. Projekt systemu BIS.....	133
3.1. Koncepcja systemu.....	133
3.2. Podział na podsystemy	134
3.3. Przydział podsystemów do procesorów	135
3.4. Problem budowy interfejsu: Java, ActiveX, czy API-CGI ?.....	136
3.5. Zachowanie systemu w warunkach granicznych.....	138
3.5.1. Inicjalizacja	138
3.5.2. Shutdown	138
3.5.3. Sytuacje krytyczne.....	138
3.6. Model obiektowy a postać relacyjna	139
3.6.1. Uwagi wstępne	139
3.6.2. Diagram ERD.....	140
3.6.3. Problem zmiany struktury bazy danych w trakcie działania systemu	141
3.7. Strategia zapewniania bezpieczeństwa.....	143
3.8. Współbieżny dostęp do bazy danych	146
3.8.1. Określenie punktów systemu, w których istnieje dostęp współbieżny ..	146
3.8.2. Charakterystyka systemu - specyfika zapytań realizowanych ze stron WWW.	148
3.8.3. Problem utraconej aktualizacji.....	150
3.8.4. Problem niespójnej analizy	151
3.8.5. Problem archiwizacji w trakcie działania systemu.	152
3.8.6. Błędne usuwanie na skutek współbieżnej modyfikacji.....	153
3.8.7. Podsumowanie - przyjęta strategia zapewniania współbieżności	153
3.9. Konstrukcja interfejsu	154
3.9.1. Możliwości oferowane przez HTML	154
3.9.1.1. Uwagi wstępne	154
3.9.2. Przekazywanie informacji do użytkownika	155
3.9.2.1. Parametryzowanie tekstu.....	155
3.9.2.2. Wyświetlanie grafiki	156
3.9.2.3. Filmy i animacje inne niż gify animowane	156
3.9.3. Wprowadzanie danych	156

3.9.3.1. Formy (formularze).....	156
3.9.3.2. Lista rozwijana (pull-down).....	157
3.9.3.3. Lista przewijana (wybór opcji).....	157
3.9.3.4. Pola wprowadzania danych.....	158
3.9.3.5. Specjalne pole wprowadzania danych - typ password.....	158
3.9.3.6. Pole wyboru (ang. checkbox).....	159
3.9.3.7. Przycisk wyboru opcji (ang. radiobutton).....	160
3.9.3.8. Przyciski SUBMIT i RESET.....	160
3.9.4. Prototypowanie form dla systemu BIS.....	161
3.9.5. Ocena jakości interfejsu.....	163
3.10. Wymagania nakładane na zasoby.....	163
3.10.1. System operacyjny.....	163
3.10.2. Procesor.....	163
3.10.3. Pamięć operacyjna.....	163
3.10.4. Karta sieciowa.....	164
3.10.5. Internet lub inna sieć komputerowa.....	164
3.10.6. Pamięć masowa.....	164
3.10.7. Inne.....	164
3.10.8. Monitor.....	164
3.11. Wybór komponentów systemu.....	164
3.11.1. Komponenty sprzętowe.....	164
3.11.2. System operacyjny.....	165
3.11.3. Motor bazy danych.....	166
3.11.4. Serwer WWW.....	167
3.11.5. PHPFI - Interfejs API serwera WWW (rozszerzenie poleceń dostępnych w HTML).....	168
3.12. Komentarz końcowy procesu projektowania.....	169
4. Implementacja.....	170
4.1. Uwagi wstępne.....	170
4.2. Wybrane problemy implementacyjne.....	170
4.2.1. Dynamiczne przydzielanie nazw elementom interfejsu.....	170
4.2.2. Ochrona przed błędnym kasowaniem wynikającym ze współbieżnego dostępu.....	173

4.2.3. Zmiany struktury bazy danych podczas pracy systemu.....	174
4.2.4. Bezpieczeństwo formularzy - metody POST i GET	174
4.2.5. Ukrywanie parametrów przekazywanych przez zmienne	175
4.2.6. Nadawanie identyfikatorów	176
4.2.7. Ochrona procesu archiwizacji danych.....	176
4.2.8. Ochrona dostępu - implementacja haseł.....	177
4.2.9. Śledzenie zmian w bazie danych.....	177
5. Testowanie systemu.....	179
5.1. Zbieżność z założonym projektem (stopień rozwiązania problemu)	179
5.2. Koszt systemu	179
5.2.1. Uwagi wstępne	179
5.2.2. Opcja minimalna (oszczędnościowa) - sprzęt używany.....	180
5.2.3. Opcja minimalna (oszczędnościowa) - sprzęt nowy	180
5.3. Ocena zastosowanej metodologii	181
5.4. Bezpieczeństwo.....	181
5.5. Wydajność.....	182
5.6. Współbieżność	182
5.7. Odporność na błędy i awarie. Odtwarzanie danych.	183
5.8. Modyfikowalność.....	183
5.9. Ocena końcowa systemu	184
6. Przyszłość systemu BIS.....	185
6.1. Transakcje	185
6.2. System doradzania oparty na analizie stanu bazy danych.....	185
6.3. Rozszerzenie badania wstawianych elementów.....	185
7. Literatura.....	187

Zakres i cel pracy dyplomowej

Celem pracy jest zaprojektowanie i implementacja systemu umożliwiającego gromadzenie i udostępnianie danych bibliograficznych z wykorzystaniem sieci Internet. System ten ma być przykładem możliwości budowy interfejsu bazy danych opartego o język HTML. Powinien on umożliwiać przechowywanie informacji o:

- różnych rodzajach publikacji (książki, czasopisma, artykuły, prace naukowe itd.),
- autorze publikacji,
- miejscu (adres sieciowy, adres prywatny itd.) i formie (plik, ksero itd.) w jakiej jest udostępniana publikacja
- osobie udostępniającej publikację.

System powinien umożliwiać wykonywanie operacji dopisywania, usuwania, modyfikacji oraz wyszukiwania informacji. Dostęp do systemu powinien być kontrolowany - niezbędne jest zapewnienie identyfikacji użytkownika, który w zależności od posiadanych uprawnień, może (lub nie) wykonać wybraną operację. System powinien również zapewniać wielu użytkownikom jednoczesny odczyt i modyfikację przechowywanych informacji, w sposób nie powodujący niebezpieczeństwa utraty spójności danych.

Produkt powinien cechować się odpowiednią jakością, mierzoną za pomocą określonych uprzednio kryteriów. Powinien on także charakteryzować się niskimi kosztami instalacyjnymi i eksploatacyjnymi.

Część teoretyczna pracy powinna obejmować dwa zagadnienia:

- a. dokumentację projektu BIS: zapis (dokumentację krok-po-kroku) procesu analizy, projektowania i implementacji systemu, komentarz krytyczny do ich przebiegu, szkic rozwiązań alternatywnych,
- b. osadzenie produktu w aktualnej sytuacji na rynku usług WWW: przegląd aktualnie stosowanych technologii, opis historii i zastosowań sieci Internet (ze szczególnym uwzględnieniem usługi WWW), ocenę aktualnego stanu WWW w Polsce i na świecie

1. Internet i usługa WWW.

1.1. Uwagi wstępne

Poniższa część pracy ma charakter informatyczno-ekonomiczny. Jej zadaniem jest szeroko rozumiane przedstawienie Internetu (ze szczególnym uwzględnieniem usługi World Wide Web) jako przyszłego środowiska działania systemu BIS. Autorzy pracy postawili przed sobą zadanie nie tylko naszkicowania technologii i produktów działających obecnie na tym rynku, ale również podjęli się próby opisu Internetu z punktu widzenia marketingowego (dynamika wzrostu rynku, ocena wyników najpoważniejszych zaangażowanych firm, aspekt finansowy itp.). Jest to według nas ważne przy próbie oceny naszego produktu - dla jego pełniejszego zrozumienia i odpowiedzi na pytanie, czy jego idea i techniki przez nas zastosowane mają przyszłość w szybko zmieniającym się świecie informatyki.

1.2. Czym są: Internet i World Wide Web ?

Nazwa Internet dotyczy wspólnego protokołu, komputerów i innych zasobów stanowiących jedną globalną sieć komputerową. Nazwa ta odnosi się również czasem do wszystkich sieci rozległych, w których stosowanym protokołem transmisji jest protokół TCP/IP, a także zasobów które ta sieć udostępnia. Termin ten bywa też odnoszony do środowiska osób, które rozbudowują i używają wspólną sieć komputerową (definicja na podstawie [3]).

Główną przyczyną bardzo silnego wzrostu zainteresowania tą siecią stała się usługa WWW (World Wide Web). Nazwa ta obejmuje dane przesyłane za pomocą protokołu HTTP (Hyper Text Transfer Protocol), używanego głównie do przekazywania komponentów multimedialnych (dokumentów składających się z często z grafiki statycznej, sekwencji filmowych i dźwiękowych itp.). Choć usługa WWW jest tylko jedną z wielu usług oferowanych za pomocą Internetu (DNS, telnet, email, ftp ...), w ostatnim czasie obserwujemy równouprawnianie się pojęć „Internet” i „WWW”. Dzieje się tak dlatego, że przeglądarki WWW (ang. browsery - oprogramowanie pozwalające na interpretację dokumentów HTML¹) pozwalają na pełną integrację usług w jeden system - można za ich pomocą np. zainicjować sesję ftp, czy nawiązać połączenie „telnet”. U podstaw błyskawicznego rozwoju usługi

WWW legł także fakt, że posługiwanie nią jest bardzo proste i intuicyjne. W odróżnieniu od pozostałych usług nie wymaga ona znajomości skomplikowanych poleceń, czy zasad posługiwania się nimi. Jediną wymaganą od użytkownika znajomością jest zasada posługiwania się myszą lub innym urządzeniem wskazującym (dotyczy to nowoczesnych przeglądarek z interfejsem graficznym), oraz zasada tworzenia i wybierania odnośników (ang. hyperlink).

1.3. Historia Internetu

Internet jako sieć globalna wywodzi swoje pochodzenie od prostych połączeń typu komputer-komputer, realizowanych jeszcze za pośrednictwem taśm, kart perforowanych i innych mediów, które można określić jako „off-line”. Połączenia takie powstawały już w latach sześćdziesiątych. Następnie pojawiły się proste sieci komputerowe, choć ich zastosowania były początkowo bardzo ograniczone z uwagi na problemy techniczne (sieć przestawała działać, gdy jeden z jej węzłów przestawał działać lub był wyłączony, trudno było połączyć więcej niż kilka jednostek obliczeniowych, niska była prędkość transmisji itp.).

Mimo pierwszych trudności, na jakie natrafiała rodząca się technologia, zainteresowały się nią czynniki wojskowe USA. Z ich inicjatywy, około roku 1970, agencja rządowa ARPA (Advanced Research Projects Agency, część Departamentu Obrony USA) uruchomiła pierwszy fragment eksperymentalnej sieci zwanej ARPAnet. Miała ona spełniać następujące postulaty:

- sieć musi działać pomimo uszkodzenia wielu jej komponentów i linii połączeniowych
- sieć musi być w stanie połączyć wiele różnych typów komputerów
- sieć powinna samodzielnie kierować przepływem informacji (dobierać trasy przekazywania wiadomości w oparciu o informacje o uszkodzonych lub przeciążonych łączach)
- do sieci można podłączać komputery lub mniejsze sieci lokalne

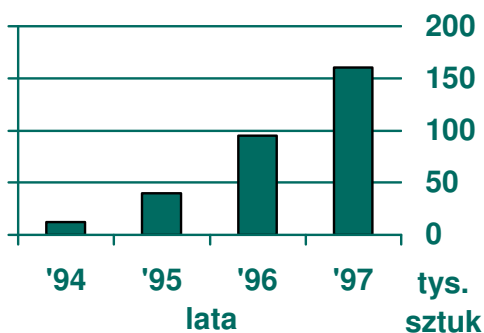
Pod koniec lat 70-tych, wzrost sieci, spowodowany udostępnieniem jej dla ośrodków uniwersyteckich i firm prowadzących działalność związaną z obronnością był już tak szybki, że standardy, na których ona bazowała przestały być

¹ Hyper Text Markup Language - nazwa języka opisu dokumentów przesyłanych za pomocą protokołu HTTP

wystarczające. Stało się to przyczyną przejścia z technologii Package Switch Nodes na standard TCP/IP. Zmiana ta miała na celu wykorzystanie dwóch zalet protokołu TCP/IP: praktycznego braku ograniczeń na wielkość sieci² i prostoty jego implementacji nawet na prostych systemach komputerowych. Kilka lat później Narodowa Fundacja Nauki (National Science Foundation) opracowała i wdrożyła własną sieć komputerową opartą na modelu sieci ARPA. Siecią tą była NSFnet, która dzięki sporym inwestycjom (poprawa jakości i przepustowości łączy) po pewnym czasie wchłonęła sieć ARPA (jednocześnie dla celów militarnych została wydzielona osobna sieć MILnet). Sieć NSF i powstałe w latach 80-tych komercyjne sieci Usenet, Bitnet, CompuServe i America Online stały się po umożliwieniu połączeń między nimi początkiem tego, co dziś znamy pod nazwą „Internet”.

1.4. Stan obecny i prognozy na przyszłość

Szacuje się, że o ile w roku 1996 było 31 milionów użytkowników Internetu korzystających z poczty elektronicznej (w skali całego świata), to w roku 2000 liczba ta sięgnie 70 milionów. Badania przeprowadzone w Polsce pokazują, że w roku 1996 mieliśmy prawie 100 tysięcy komputerów podłączonych do Internetu i 300-500 tysięcy użytkowników korzystających stale z poczty elektronicznej. Rynek ten jest potencjalnie bardzo chłonny w związku z szybko rosnącą liczbą użytkowników komputerów (w roku 1995 sprzedano w Polsce 364 tysiące komputerów).



Rysunek 1 : Liczba komputerów podłączonych do Internetu w Polsce

² Warto wspomnieć, że do niedawna uznawany za wystarczający na długie lata system identyfikacji komputerów w sieci Internet oparty o adresy IP okazuje się ostatnio zbyt mało „pojemny”. Dynamika procesu rozszerzania sieci sprawia, że brane są pod uwagę możliwości przededefiniowania standardu adresu IP (np. przez jego rozszerzenie o 2 bajty).

1.5. WWW jako technologia informatyczna

1.5.1. Protokół komunikacyjny

Do wymiany dokumentów webowych używany jest protokół komunikacyjny HTTP (Hyper Text Transfer Protocol). Umożliwia on przesyłanie tych dokumentów, a także określenie parametrów przesyłania (np. czy dane powinny być zapisywane w pamięci podręcznej - cacheowane, czy też nie, kiedy powinno nastąpić odświeżenie zawartości danej strony itp.). Dokładniejsze informacje na ten temat można znaleźć w sieci Internet, np. pod adresem www.w3.org. W powszechnym użyciu są wersje 0.9 i 1.0 tego protokołu.

1.5.2. Język HTML

Język HTML jest z punktu widzenia formalnego językiem opisu strony, podobnie jak np. postscript. Określa on, w jaki sposób mają być interpretowane poszczególne części dokumentu. Jest to możliwe za pomocą tzw. tagów - elementów sterujących w tekście, których ogólny format jest następujący (niektóre elementy języka nie wymagają tagu kończącego - np. wypunktowanie listy , paragraf tekstu <P> itp.):

```
<tag-start>tekst</tag-end>
```

I tak, np. sekwencja:

```
Cześć tego tekstu została napisana  
pismem<B>pogrubionym</B> a część - <I>pochyłym</I>.
```

Zostanie zinterpretowana przez przeglądarkę jako:

Cześć tego tekstu została napisana pismem **pogrubionym** a część - *pochyłym*.

Warto tu zwrócić uwagę na charakterystyczną cechę: brak przełamania tekstu po słowie „napisana”. Język HTML ignoruje dodatkowe spacje i znaki CR wstawiane pomiędzy słowami tekstu. Jeśli dodać do tego fakt, że nie istnieje możliwość wyjustowania do obu marginesów, łatwo zdać sobie sprawę, jak prostym i w gruncie rzeczy ograniczonym językiem jest HTML, a co za tym idzie - jak trudno na jego

bazie zbudować estetyczny i prosty w obsłudze interfejs. Problem ten omówiony jest w rozdziale poświęconym implementacji interfejsu - patrz 3.9.

O sile i elastyczności HTML nie stanowią jednak jego możliwości określania parametrów tekstu. Najważniejszą cechą tego języka jest możliwość stosowania odnośników (ang. link), pomiędzy częściami dokumentu. Odnośniki te obrazują logiczne zależności pomiędzy częściami dokumentu (np. stosując w tekście słowo „Einstein” możemy powiązać je z dokumentem zawierającym opis życiorysu uczonego lub streszczenie teorii względności). Ważne jest przy tym to, że multimedialność języka pozwala na osadzanie nie tylko tekstowych obiektów - mogą to być także obrazy, sekwencje filmowe a nawet zapis audio.

Idea języka opartego o system logicznych odnośników pomiędzy częściami dokumentu jest stosunkowo stara. Pierwszy model takiego języka zaproponowano już w roku 1947 - a więc na długo przed prawdziwym początkiem ery informatycznej. Praktyczna implementacja - HTML pojawił się stosunkowo niedawno - na bazie języka, który był użyty w celu poprawy wydajności pracy zespołu badawczego w jednym z amerykańskich instytutów. Rozwój idei doprowadził do wykształcenia wersji 2.0, która weszła na rynek jako język przeglądarek WWW. Aby było to możliwe, opracowano system lokalizacji dokumentu w sieci globalnej, dzięki czemu stosowane odnośniki mogą wiązać nie tylko części tego samego dokumentu, ale prowadzić do dowolnego innego obiektu znajdującego się na jednym z komputerów podłączonych do sieci. System ten nosi nazwę URL (Universal Resource Locator) i bazuje na adresach następującej postaci:

`http://www.tpsa.com.pl/actual/cenniki.html`

- `http://` - nazwa będąca określeniem rodzaju usługi - Hyper Text Transfer Protocol (za pomocą przeglądarek WWW można obecnie otwierać sesje ftp, telnet czy gopher)
- `www.tpsa.com.pl` - to adres komputera w sieci Internet³

³ Dokładniej - jego nazwa symboliczna, która jest odwzorowywana w adres numeryczny (np. 123.45.23.123) za pomocą usługi DNS (Domain Name Service), opartej o system tzw. serwerów nazw (nameservers). Serwery te udostępniają dane zawarte w rozproszonej bazie danych, której struktura (a także protokół wymiany informacji pomiędzy serwerami nazw) jest zdefiniowany w RFC 920, 1034, 1035

- `/actual/` - nazwa katalogu, w którym umieszczony jest plik, do którego żądamy dostępu. Ponieważ zwykle serwery WWW, udostępniające te pliki pozwalają na określenie miejsca w strukturze fizycznej plików, od której rozpoczynamy szukanie (HTML document root), a więc rzeczywistym położeniem pliku może być np. `/usr/local/etc/httpd/htdocs/actual/`
- `cenniki.html` - nazwa lokalizowanego pliku HTML. Standardowo przyjmuje się rozszerzenie `html` (`htm` w starszych wersjach Windows), choć nie jest to konieczne. Obiektem wskazywanym za pomocą URL może być także plik graficzny (np. rysunek `gif` lub `jpg`), zapis sekwencji video (`*.mpg`, `*.avi`) itp.

W tym miejscu należy dodać, że z uwagi na pewne wady adresu URL (wskazuje on dokument z dokładnością do komputera - jeśli ten sam dokument znajduje się na innej maszynie, posiada też inny adres), trwają prace nad wypracowaniem bardziej efektywnej metody lokalizacji. Przewidywany adres ma być znany pod nazwą URN (Universal Resource Name), ale projekt nie jest zbytnio zaawansowany.

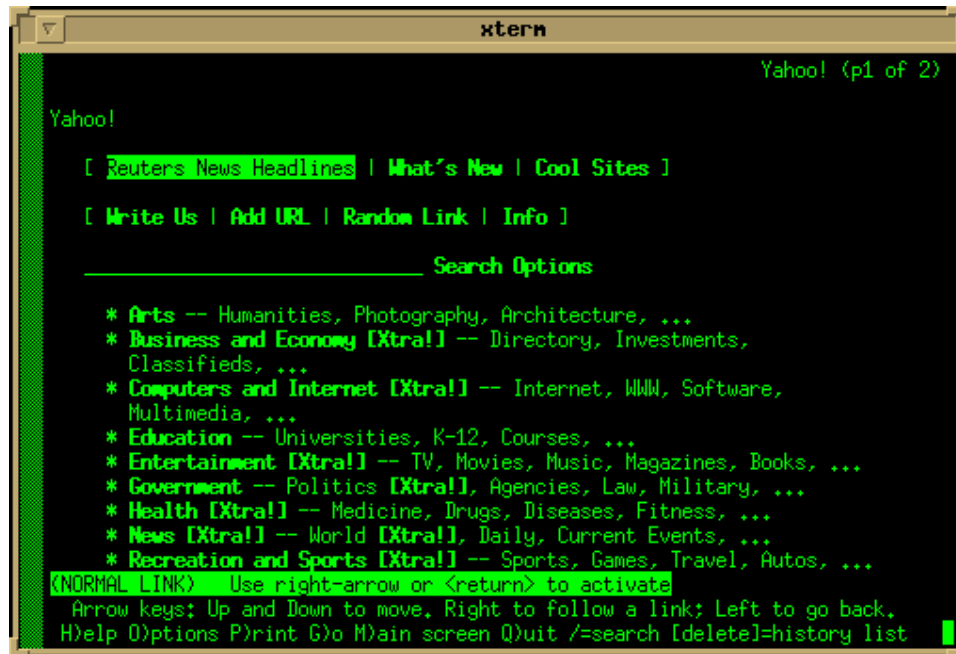
Język HTML w wersji 2.0 został stosunkowo szybko uznany jako standard i otrzymał normę ISO. Prace nad standardem HTML 3.0 są jednak ciągle w toku⁴, a proponowane rozwiązania (istnieje stała lista dyskusyjna przeznaczona dla osób, które chciałyby partycypować w projekcie HTML 3.0) są testowane w przeglądarce Arena - jedynym oficjalnym browserze HTML 3.0.

Błyskawiczny rozwój sieci Internet i usługi WWW spowodował, że wiodące firmy starały się opracować własne rozwiązania. Netscape i Microsoft zaproponowały dla swoich przeglądarek rozwiązania podobne, choć różniące się w kilku szczegółach (p. 2.5.3). Dla tych rozwiązań stosowano różne nazwy: HTML 2.0 bis, HTML 2.0+, HTML 2.0 enhanced itp. W końcu przyjęła się popularna nazwa HTML 3.0 (choć w istocie brak jest podstaw do jej stosowania), o czym może świadczyć chociażby liczba podręczników tego języka dostępna na rynku.

1.5.3. Przeglądarki (browsery)

Historycznie, jedną z pierwszych przeglądarek był Lynx - program unixowy, pracujący w środowisku tekstowym. Umożliwiał on interpretację języka HTML na popularnych terminalach `vt100` i podobnych. Realizowane były jedynie tagi HTML

2.0⁵, ponieważ był to jedyny obowiązujący standard. Programy tekstowe, realizujące ten standard nazywa się czasami przeglądarkami I-szej generacji.

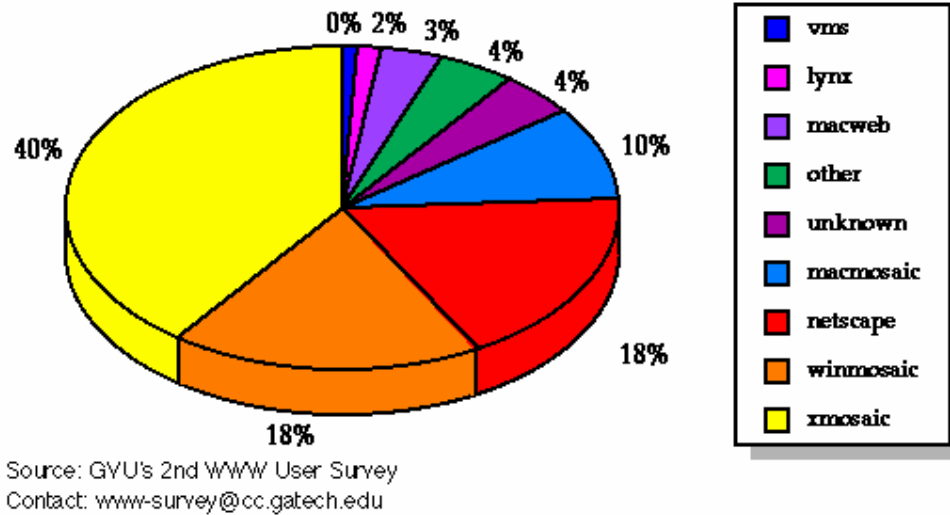


Rysunek 2: Przeglądarka I generacji: Lynx

Możliwości, jakie otwierały się w związku z rozwojem języka HTML doprowadziły w początku lat 90-tych do wyprodukowania pierwszej przeglądarki pracującej w środowisku graficznym (XWindow). Była to pierwsza przeglądarka II generacji - NCSA Mosaic. Stanowiła ona pierwszy krok w stronę multimedialności - możliwe stało się osadzanie w dokumentach HTML plików graficznych (w formacie gif) za pomocą tagu (image source).

⁴ W sieci Internet można znaleźć wersje „draft” specyfikacji

⁵ Obecne wersje Lynxa - np. dostarczane wraz z dystrybucjami Linuxa (Debian, RedHat) - interpretują już większość tagów HTML 3.0



Rysunek 3: Wyniki badań statystycznych dotyczących przeglądarek (z roku 1994)

NCSA Mosaic przyczynił się do gwałtownego rozwoju usługi WWW, a idąca w parze z tym wzrastająca liczba stron WWW i umieszczanie na nich efektywnych, ale stosunkowo dużych rysunków stało się dużym problemem, ponieważ istniejące przeglądarki nie posiadały wbudowanych mechanizmów tzw. inteligentnego ładowania dokumentów HTML. Dokument przed wyświetleniem musiał być załadowany ze wszystkimi jego elementami, co mogło sprawiać kłopoty w warunkach dużego ruchu w sieci.

Wchodząca w tym czasie na rynek firma Netscape Communications postanowiła rozwiązać ten problem, oferując przeglądarkę nowej klasy - Netscape Navigator 1.0. Potrafiła ona dostosować się do struktury wczytywanego dokumentu HTML, żądając najpierw przesłania tekstu (który zajmuje objętościowo znacznie mniej miejsca niż nawet niewielkie obrazki), a potem grafiki. Możliwe stało się też wykorzystanie rysunków formatu jpg, zajmującego znacznie mniej miejsca niż format gif (wiąże się to z zastosowaniem kompresji z utratą danych opartej o rozwinięcie Fouriera):



rysunek.gif (45 Kb)



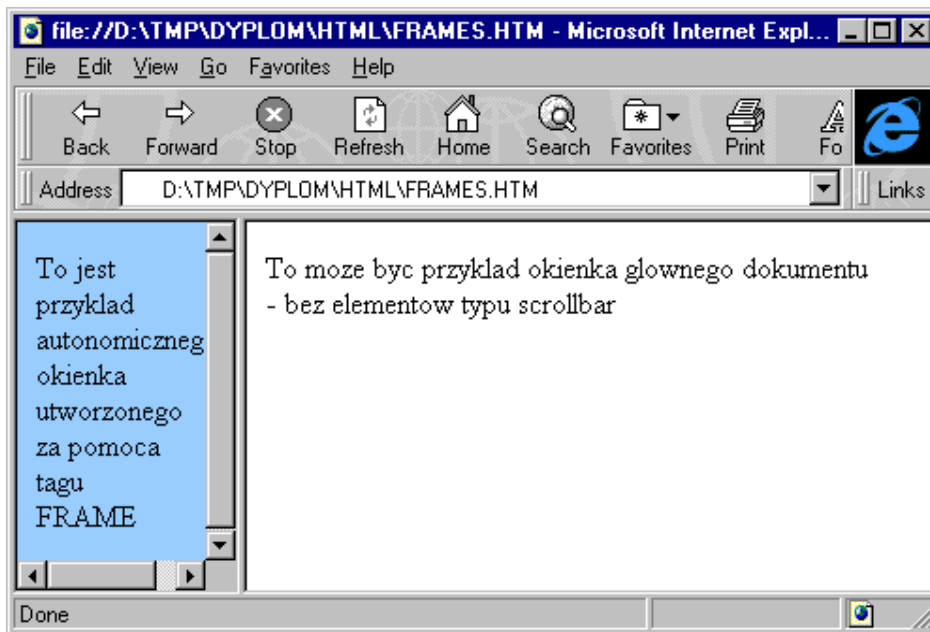
rysunek.jpg (13 Kb)

Rysunek 4: Porównanie jakości i wielkości rysunków w formatach gif i jpg

W Netscape Navigatorze poszerzono także obsługę starego formatu graficznego gif umożliwiając wyświetlanie najnowszej jego wersji (tzw. format gif89), który pozwalał na określenie koloru, który miał być interpretowany jako przezroczysty (tzw. transparent gif). Dodatkowo, wprowadzono nowe tagi do języka HTML - np. możliwość zdefiniowania tła strony dokumentu (tłem mógł być także rysunek).

Netscape Navigator zjednał sobie przychylne opinie większości użytkowników WWW, stając się niepodzielnie panującą przeglądarką. Kilka miesięcy później, Microsoft, który do tej pory nie wydawał się zainteresowany rywalizacją na polu usług Internetowych i skupiał swoją uwagę na promowaniu środowiska Windows, zdecydował się na próbę obalenia dominacji Netscape Communications na rynku. Jego produkt - Internet Explorer nawiązywał do możliwości Navigatora, choć w kilku punktach różnił się od niego. Powodem tych różnic był brak formalnej specyfikacji języka HTML 3.0 - rozszerzenia zaproponowane w Navigatorze były jedynie propozycją firmy Netscape. Sytuacja panująca w okresie tzw. III generacji przeglądarek spowodowała podział wśród użytkowników WWW. Ci, którzy mieli dostęp do środowiska Windows i używali oprogramowania Microsoft, zaczęli wykorzystywać udostępniane przez niego tagi, nieinterpretowane przez Netscape Navigatorsa. Przyjął się zwyczaj umieszczania na stronach HTML stopek typu „best viewed by Netscape” czy „page designed for MS IE” - w zależności od preferencji autorów dokumentu.

Netscape Communications odpowiedziało wyprodukowaniem Netscape Navigatora 3.0, który udostępnił kolejne możliwości - np. dzielenie ekranu przeglądarki na autonomiczne części - tzw. frames.



Rysunek 5: Interpretacja ramek (frames) w oknie MS Internet Explorera 4.0

Nie zahamowało to kontrofensywy marketingowej Microsoftu, który stosunkowo szybko opracował nową wersję Internet Explorera - IE 3.0. Od tej pory datuje się zmierzch hegemonii Netscape i wyrównywanie się możliwości przeglądarek. W tej chwili rynek jest podzielony, choć nadal Microsoft znajduje się na drugiej pozycji, co wynika z dwóch faktów: przyzwyczajenia użytkowników do interfejsu Navigatora i szerszej gamy systemów operacyjnych, na której dostępny jest produkt Netscape.

Dla twórców stron HTML do dziś istotną sprawą jest, jakiej przeglądarki używa klient. Choć funkcjonalnie przeglądarki są tożsame (praktycznie wszystkie tagi są rozpoznawane przez oba produkty), to występują pewne różnice w szczegółach - np. listy pull-down wyglądają trochę inaczej, co stwarza dość poważne problemy przy budowie interfejsu w języku HTML. Problem ten jest szerzej omówiony w części dokumentacyjnej projektu BIS, ponieważ zetknęliśmy się z nim podczas budowy naszego systemu. Warto też wspomnieć, że na skutek różnic w obsługiwanych paletach barw, rysunek, który ładnie wygląda pod Internet

Explorerem (wypełnione kolorem powierzchnie są „pełne”), może być zupełnie inaczej odebrany przez Netscape Navigatora (pojawiają się charakterystyczne punkty wynikające z zastosowania procedur symulacji kolorów⁶).

Przedstawiamy poniżej porównanie funkcjonalne dwóch największych rywali na rynku przeglądarek - Netscape Navigatora 3.0 Gold i MS Internet Explorera 3.0. Są to obecnie najpopularniejsze i najlepsze programami w tej klasie (choć pojawił się już IE 4.0⁷ i pierwsze wersje testowe IE 5.0). Obie przeglądarki cechuje możliwość obsługi aplikacji graficznych i multimedialnych, interaktywność oraz duża elastyczność - można je w prosty sposób dostosowywać do indywidualnych potrzeb i upodobań. Wiele funkcji dawniej realizowanych przez procedury rozszerzające zostało obecnie zintegrowane, co pozwala na szybsze i bardziej intuicyjne nawigowanie w sieci WWW. Oba programy są stabilne, pewne w działaniu oraz dysponują kompletnym zestawem funkcji i możliwości. W takiej sytuacji wskazanie najlepszego produktu jest sprawą bardzo trudną. Dlatego też ograniczymy się do ich porównania.

Cecha przeglądarki	Microsoft Internet Explorer 3.0	Netscape Navigator 3.0
udział w rynku	około 30%	około 65%
obsługa aplikacji graficznych i multimedialnych	tak	tak - LiveVideo (.avi), Live Audio (dźwięk), Live3D (VRML)
możliwość wyłączenia grafiki	tak	nie
ściągnięcie tekstu przed grafiką	tak	tak
definiowanie haseł dostępu do wybranych	tak	nie

⁶ tzw. drżenie - ang. dithering

⁷ tzw. Internet Explorer Suite, zawierający również przeglądarkę news i system obsługi poczty elektronicznej. IE 4.0 Suite wysyła listy email w postaci HTML, co wydaje się być rozwiązaniem dość kontrowersyjnym, z uwagi na dużą liczbę użytkowników stosujących oprogramowanie poczty nie potrafiące interpretować tagów HTML (np. pine, eudora itp.). Na pomysł zastosowania przeglądarki do obsługi news wpadła zresztą jako pierwsza firma Netscape Communications.

ośrodków WWW		
obsługa poczty i wiadomości	tak	tak
systemy oceny jakości dokumentów	PICS (Platform for Internet Content Specification), RSAC (Recreational Software Advisory Council)	nie
dostępne wersje	Windows (3.x, 95, NT), Macintosh	Windows (3.x, 95, NT), Macintosh, UNIX, OS/2
kompilator Java (typu JIT- just-in-time),	tak	tak
JavaScript	tak	tak
VBScript	tak	nie
obsługa MPEG-1, OLE	wbudowane	zewnętrzne
ActiveX	tak	nie ⁸
LiveContent	nie	tak
kontrola apletów Javy	nie	tak, za pomocą Live-Connect
rozszerzenia HTML	arkusze stylów CSS Level 1 (Cascading Style Sheets)	znaczniki: Multicol, Spacing
wymiana danych w czasie rzeczywistym	NetMeeting	CoolTalk

Tabela 1: Porównanie cech Netscape Navigator 3.0
i MS Internet Explorer 3.0

Warto także wspomnieć o istniejących do dziś przeglądarkach tekstowych - przykładem może być tu Lynx. Oprogramowanie to może z powodzeniem rywalizować z Netscape lub MS IE, nie tylko w systemach na których użycie przeglądarki graficznej jest niemożliwe (np. Linux bez systemu XWindow), ale także

⁸ Najnowsze wersje Netscape (Navigator 4.0 i 4.01, Communicator) posiadają już wbudowany moduł obsługi ActiveX

wszędzie tam, gdzie nieduża moc procesora powoduje, że użycie „zaawansowanej” przeglądarki poważnie obciąża system.

Na koniec, prezentujemy rozwój kolejnych generacji przeglądarek na tle ostatnich lat:

IV						Navigator 3.0 Netscape Gold Explorer 3.0
III					Navigator 2.0 Explorer 2.0	
II			NCSA Mosaic	Netscape Navigator 1.1		
I	Text only	Lynx				
	1991	1992	1993	1994	1995	1996

Rysunek 6 : Generacje przeglądarek WWW

1.5.4. Serwery WWW

1.5.4.1. Uwagi wstępne

Poniższe rozważania były pisane dla serwerów pisanych dla systemu Unix, dlatego pojawiają się w nich pojęcia i terminy charakterystyczne dla tego systemu (proces, prawa roota itp.), ale wynikające z nich konkluzje odnoszą się również dla innych platform operacyjnych (Windows NT, Windows 95, Windows 3.x⁹).

1.5.4.2. Istota działania

Serwer WWW jest procesem, który zajmuje się obsługą zleceń przychodzących na określony port komunikacyjny. Zwykle jest to port 80, ale ponieważ procesy wykorzystujące go muszą być uruchomione przez administratora systemu (wymóg ten nakładany jest na wszystkie porty o numerach niższych niż 1024), czasami spotyka się rozwiązania wykorzystujące port 8080. Należy wówczas dopisywać numer portu do adresu URL:

<http://www.domain.com/sample.html>

⁹ Istniały również specjalne serwery pisane dla tego systemu, choć nie zapewniał on pracy wielowątkowej!

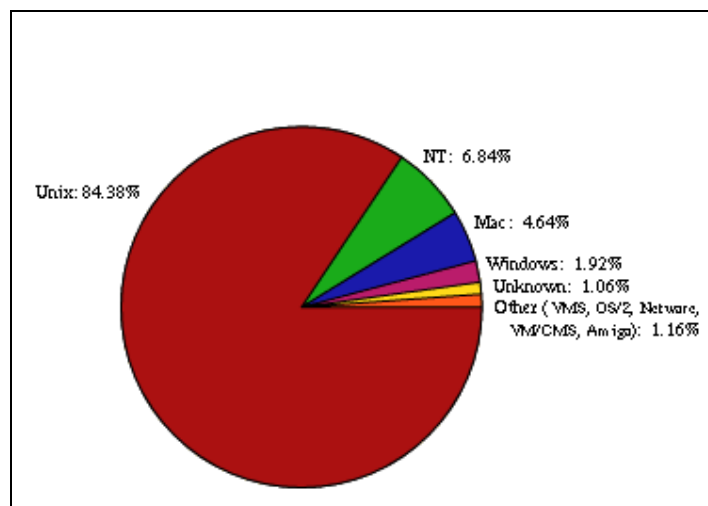
<http://www.domain.com/sample.html:8080>

1.5.4.3. Platformy systemowe

Poniższe dane zawierają przegląd systemów operacyjnych, których używają serwery WWW na całym świecie. Dane te były zebrane na początku 1997 roku. Dane pochodzą z wyników badań firmy Crawler (wyszukiwanie informacji WWW).

Platforma systemowa serwerów HTTP	Udział procentowy
Unix	84,38%
NT	6,48%
Macintosh	4,64%
Windows	1,92%
Nieznany	1,06%
Inne (VMS, OS/2, NetWare, VM/CMS)	1,16%

Tabela 2: Przekrój systemów WWW pod względem zastosowanej platformy systemu operacyjnego



Rysunek 7: Systemy operacyjne w zastosowaniach WWW

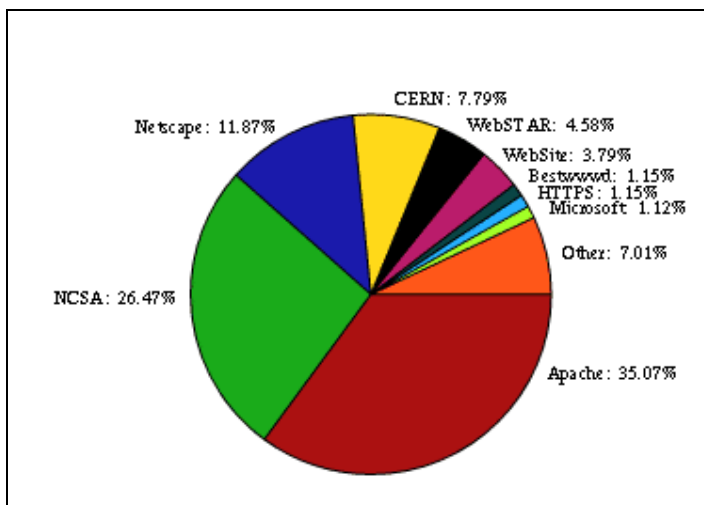
Mimo zdecydowanej ofensywy firmy Microsoft, nadal na rynku dominuje platforma systemu operacyjnego Unix. Istnieją dwa główne powody takiej sytuacji techniczny (informatyczny) i ekonomiczny (marketingowy). Pierwszy to fakt, iż system Unix był od samego początku pomyślany jako system wielowątkowy, idealnie nadający się do obsługi współbieżności. System Windows stał się systemem współbieżnym dopiero w wersji 95 (ale jego niestabilność predestynuje do roli platformy sprzętowej raczej wersję NT). Drugi powód to błędna polityka firmy Microsoft, która nie doceniła możliwości, jakie niesie ze sobą rynek Internetu i usługi WWW. Microsoft przez długi okres nie prowadził aktywnej działalności badawczej i promocyjnej (patrz uwagi nt. historii Internetu), co spowodowało, że znakomita większość rynku została zajęta przez systemy unixowe.

1.5.4.4. Używane oprogramowanie

Poniższe dane zawierają przegląd serwerów WWW, używanych na całym świecie. Dane te były zebrane na początku 1997 roku - patrz uwaga w poprzednim rozdziale.

Oprogramowanie serwerów HTTP	Udział procentowy
Apache	35,07%
NCSA	26,47%
Netscape	11,87%
CERN	7,79%
WebSTAR	4,58%
WebSite	3,79%
Bestwwwd	1,15%
HTTPS	1,15%
Microsoft	1,12%
Inne	7,01%

Tabela 3: Przekrój rynku oprogramowania serwerów WWW



Rysunek 8: zróżnicowanie rynku oprogramowania

Zwraca uwagę, że ponad 80% to oprogramowanie, które jest dostępne w wersji shareware lub z ograniczoną licencją bezpłatną (np. Netscape, który jest bezpłatny dla celów niekomercyjnych i uczelnianych). Najpopularniejszy zaś serwer to Apache - produkt dostępny jako freeware. Dominacji tych serwerów sprzyja fakt dużej przewagi systemów unixowych w przekroju systemów WWW.

Większość z podanych wyżej serwerów jest cały czas unowocześniana, aby nadążyć za nowymi technologiami i rozwojem rynku. Typowym przykładem jest wspomniany Apache, o którym informacje można znaleźć w sieci Internet pod adresem <http://www.apache.org>. Istnieją listy dyskusyjne użytkowników tego oprogramowania, na których wymieniane są nowe idee, proponowane kierunki rozwoju. Podobną strategię wybrali twórcy Netscape'a i kilku innych serwerów.

1.5.4.5. Przegląd oferowanego oprogramowania

Usługa WWW powstała stosunkowo niedawno - pierwsze jej projekty powstały na początku lat 90-tych. Pierwsze serwery, o bardzo ograniczonych możliwościach, pojawiły się niedługo potem. Błyskawiczny rozwój rynku zaowocował pojawieniem się nowych produktów, zarówno komercyjnych, jak i shareware oraz edukacyjnych. W tej chwili liczba gotowych rozwiązań sięga tysięcy. Poniżej zamieszczamy przykładową listę oferowanych serwerów, wraz z krótką ich charakterystyką, uzyskaną z danych serwera informacyjnego yahoo.com.

nazwa serwera	oferowane funkcje
I/Net Web Server AS/400	pierwszy i jak dotąd jedyny serwer pracujący pod systemem operacyjnym AS/400, jest zalecany przez firmę IBM, posiada możliwości aliasingu, wbudowane strony przykładowe, dostęp do bazy danych, dynamiczne indeksowanie, obsługuje image mapping, IFS, języki narodowe, skrypty, SSI (server-side includes), multimedialne obiekty USF, katalogi użytkowników i Javę.
ZBServer - http://www.zbserver.com/price.html	cehuje go prosty interfejs, bezpieczeństwo na poziomie użytkowników / grup użytkowników, makra, wbudowany serwer usługi gopher, szybkość działania, stabilność, możliwość realizacji skryptów DOS/ Win CGI i Javy
Hades WWWDemon (Novotech Communications) - http://205.241.169.43	praca w systemach Windows 95 i NT, modułowa budowa, dodatkowe obiekty typu plug-in, serwery: httpd (WEB), htttpp (Proxy), smtpp (Mail), telnetd (telnet), fingerd (finger) ftpd (ftp) i talkdd (talk)
Web Form Express - http://www.pacificnet/~walkerc/wfc.htm	serwer pracujący w trybie 32bitowym Windows (Win 95 i Win NT), posiada wbudowany dostęp do baz danych Access 7.0 i ODBC (w tym dostęp do serwerów SQL), makra, wbudowany interfejs OLE, bezpieczeństwo oparte o system nadawania praw użytkownikom i ich grupom, URL Security, wbudowane image-maps, liczniki. Kompatybilny z obsługą map stosowanych w serwerach CERN
Folk Web Servers v. 1.01 http://ww.ilar.com/folkweb.htm	serwer WWW dla systemu Windows 95, Windows NT 3.5 lub lepszych, pełna implementacja protokołu HTTP 1.0, protokołów GET, POST i HEAD, kompatybilny z mapami imagemaps w wersji NCSA, posiada wbudowaną bazę danych, obsługę Windows CGI 2.1, wbudowany interfejs GUI wykorzystywany w celu

	konfiguracji i zarządzania, aliasing, rozszerzenia MIME, kontrolę dostępu do stron, autoryzację użytkowników i grup użytkowników, filtrację adresów IP, automatyczną archiwizację plików dzienników (logów), możliwość konfiguracji on-line, kompilator perl, obsługuje standard logu NCSA, pełna dokumentacja
Apache versions 1.0.5 and 1.1b2 http://www.apache.org	serwer unixowy, dostępny pełny kod źródłowy, eksperymentalny serwer proxy, skrypty oparte na plikach, podtrzymuje połączenia typu Keep-Alive Persistent, posiada konfigurowalne środowisko pisania skryptów CGI, rozszerzoną obsługę plików graficznych, metafile standardu CERN, rozszerzoną obsługę user-dir, moduł obsługi stanu, przekierowywanie plików, minimalny DNS, dodatkowe ikony, bezpłatny
Communication Builder for Win95/WinNT http://www.aristosoft.com/ifact/prod2.htm	konfigurowalny interfejs GUI, szybką obsługę cacheowania informacji, multiple-domain, live-chat, CGI w wersji 1.1, HTTP 1.0, obsługę ISMAP, MIME, CERN/NCSA format logs, szyfrowanie za pomocą RSA
CyberPresence © Secure SSL Internet Server	serwer dla 32bitowych platform windowsowych, realizuje protokół SSL (2.0), zapewnia szyfrowanie i autoryzację, zintegrowany system wyszukiwania zbiorów tekstowych o podwyższonej prędkości szukania, podtrzymywanie virtual hosts, szablony nowych domen i nowych stron domowych, autoindeksowanie katalogów, wspólne logi, autentyzacja zgodna z basic-authentication, multiportowe hosty wirtualne, podtrzymuje skalowalność względem architektury sprzętowej, optymalizację działania, rozszerzenia MIME, formy,

	gniazda (sockets) do kodu jądra (kernel), przekierowania, zintegrowany system obsługi imagemaps
MDMA: Multithreaded Demon for Multimedia - http://sunsite.unc.edu/mdma-release/mdma.html	dedykowany dla systemu Solaris, optymalizowany pod względem szybkości działania, obsługuje wielowątkowość, dynamicznie rozszerzalne MDMA, monitorowanie systemu
Net Web Server http://www.fireants.com/web.shtml	multi-homing, animacje generowane po stronie servera, skrypty CGI, image-mapping, rozszerzone procedury dzienników, licencjonowanie użytkowników,
Spinner Server http://spinner.infovav.se/	SunSparc 5/110, Solaris 2.4 z minimalną pamięcią 4MB, licencja GPL, protokół komunikacyjny HTTP 0.9/1.0, bezpłatny, mała objętość kodu, szybki, do 2000 odwołań na minutę, konfigurowalne rozszerzenia plików, prosty w obsłudze interfejs GUI, object-oriented, potrafi słuchać jednocześnie na kilku portach, podtrzymuje virtual-hosts, multi interfejs ethernetu
SPRY corporate windows NT Webserver and Safety Web Server http://server.spry.com/docs/prodinfo/server/serverps.html	podtrzymuje protokół SSL, wbudowany prosty w obsłudze interfejs GUI, standard MIME, GDI 1.1, HTTP 1.0, driver ODBS do baz danych, konfigurowalny moduł kontroli dostępu użytkownika do stron html, skompilowaną obsługę bibliotek dynamicznych dll, logowanie do plików ASCII, CERN lub ODBC, proxy caching, skrypty w języku Perl 5, oznakowany „designed for Microsoft Back Office”, Hot Metal Pro, zdalna administracja, virtual hosts, kontrola użytkowników na poziomie użytkownik / grupa / IP
SpyGlass Technology http://www.spyglass.com/products/server.html	wieloplatformowy serwer dla Win 95, Win NT, unix, SunOS, Solaris, HP-UX, AIX, DEC OSF/1, niewielki rozmiar kodu, szybki, wbudowany protokół SSL

tthttpd TinyTurboThrotting HTTPD http://www.acme.com/software/tthttpd	Solaris 2.X, SunOS 4.1.X, BSD/OS 2.X Linux 1.2.X, OSF/1, mały i prosty kod, przenośność między platformami, optymalizacja wielkości kodu
Tiny HTTPD http://www.inka.de/~bigred/sw/tinyhttpd.html	bezpłatny, dostępny kod źródłowy, demon napisany w języku Perl

Tabela 4: Przegląd wybranych produktów WWW i oferowane przez nie możliwości

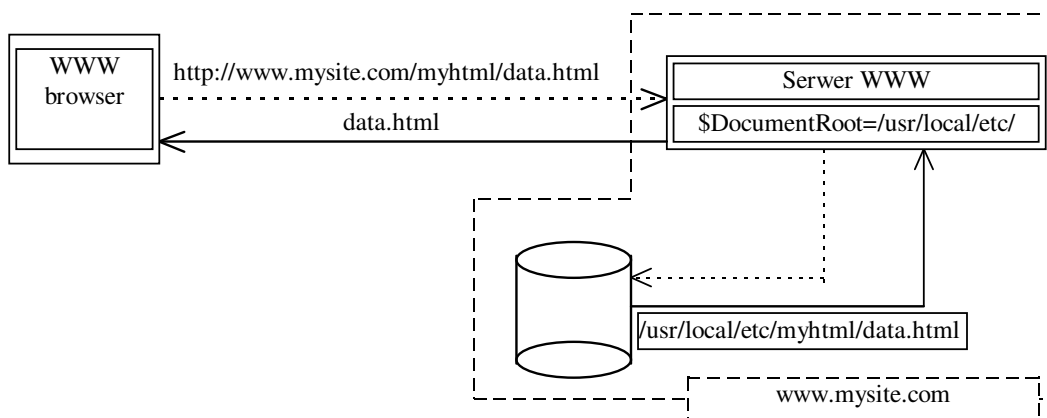
1.5.5. Generacja dokumentu HTML (w kontekście łączenia usługi WWW z bazą danych)

1.5.5.1. Wstęp

Podstawą do zorganizowania współpracy pomiędzy serwerem WWW a bazą danych jest określenie sposobu, w jaki można wstawiać wyniki zapytań kierowanych do bazy danych do dokumentów HTML. Poniżej przedstawiamy zarys metod tworzenia dokumentu HTML - począwszy od najprostszej, konwencjonalnej techniki statycznej, aż po generację „in fly” zastosowaną w projekcie BIS.

1.5.5.2. Generacja statyczna

Dokument HTML jest tworzony na podstawie odpowiedniego pliku zapisanego na dysku (położenie pliku jest określone w żądaniu odbieranym przez serwer WWW), w pokazany niżej sposób (zmienna systemowa serwera - DocumentRoot zawiera ścieżkę do katalogu, w którym przechowywane są pliki HTML). Sposób ten nazywany jest generacją statyczną, ponieważ postać dokumentu jest stała i nie może ulec zmianie (bez zmiany pliku HTML). Ponadto, użytkownik (korzystający z przeglądarki) ma możliwość odczytania pełnej postaci pliku HTML (t.j. takiej, w jakiej jest on przechowywany na dysku).



Rysunek 9: Generacja statyczna dokumentów HTML

1.5.5.3. Generacja dynamiczna z użyciem skryptu CGI (Common Gateway Interface)

Metoda ta została opracowana, kiedy okazało się, że często istnieje potrzeba przekazania do użytkownika danych, których nie możemy określić w momencie pisania dokumentu HTML. Przykładem może tu być np. liczba użytkowników pracujących w systemie, gdy nastąpiło żądanie dokumentu HTML. Aby to zrealizować, konieczne jest odwołanie się do shella i skorzystanie np. z polecenia `who`. Serwery WWW zaczęły więc udostępniać możliwość wywoływania programów w języku shella - nazwanych skryptami cgi. Okazało się jednak, że niesie to bardzo poważny problem, związany z bezpieczeństwem systemu, ponieważ programy te były uruchamiane przez proces serwera, co utrudniało ich kontrolę. Najbardziej znanym problemem była możliwość uzyskania praw root'a w następujący prosty sposób: wystarczyło utworzyć we własnym katalogu skrypt napisany w języku powłoki, ustawić jego prawa tak, aby był wykonywalny dla wszystkich i uruchomić go za pomocą przeglądarki www. Ponieważ proces serwera www bardzo często pracuje jako proces użytkownika root, nasz skrypt uzyskiwał prawa administratora. Błąd ten dotyczył np. serwerów NCSA 1.2 i NCSA 1.3 - w owym czasie najpopularniejszych na rynku (!). Od tamtej pory w strukturze katalogów serwera www pojawił się katalog `cgi-bin` (nazwę tę można w wypadku większości serwerów konfigurować). Pozwalając na pisanie w tym katalogu tylko wybranym osobom (w praktyce zwykle prawo zapisu tam ma jedynie webmaster, czyli administrator usługi WWW na danym komputerze) w poważny sposób

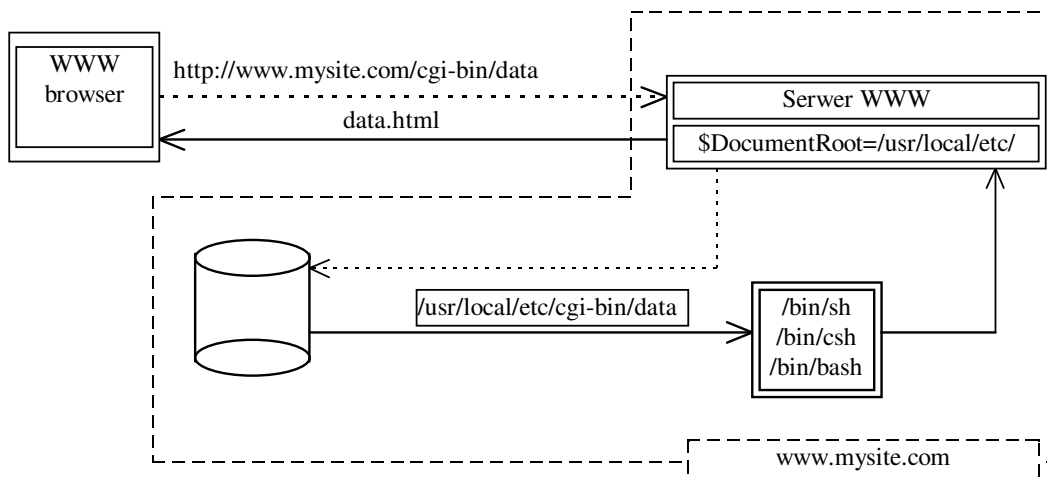
podniesiono bezpieczeństwo systemu (administrator i webmaster mieli na bieżąco wgląd do treści plików), zwłaszcza, że twórcy serwerów NCSA usunęli błąd, który polegał na możliwości realizacji odwołań do linków symbolicznych umieszczonych w katalogu cgi-bin do skryptów znajdujących się poza tym katalogiem.

Metoda ta, choć w dużym stopniu rozszerzająca możliwości usługi WWW, daleka była od optymalnej. Przede wszystkim, nie było możliwe łączenie zwykłych plików HTML ze skryptami cgi (np. przez wywołanie skryptu cgi z pliku html). Powodowało to określone trudności, zwłaszcza dla użytkownika, który musiał wpisywać w adresie URL frazę /cgi-bin/. Adresy stawały się „sztuczne” i trudniejsze do zapamiętania, co miało znaczenie w przypadku produktów rynkowych, a ponadto zniknęły możliwości ich „eleganckiego” zapisu, związanego z polskim brzmieniem domeny:

www.wyszukiwanie-adresow.com.pl./adresy/polska/gdansk/cgi-
bin/znajdz?tel=324356

(ostatnia część adresu to przekazanie parametru do skryptu cgi-bin)

Metoda skryptów CGI, zaproponowana dla serwerów WWW pracujących w systemie Unix (NCSA server, Apache itp.), przyjęła się powszechnie i dlatego zwykło się stosować terminy z niej pochodzące również dla serwerów Windows (tzw. win-cgi, których rolę pełniły odpowiednio napisane programy exe lub biblioteki dynamiczne dll).



Rysunek 10: Generacja dynamiczna z użyciem skryptu CGI (Common Gateway Interface)

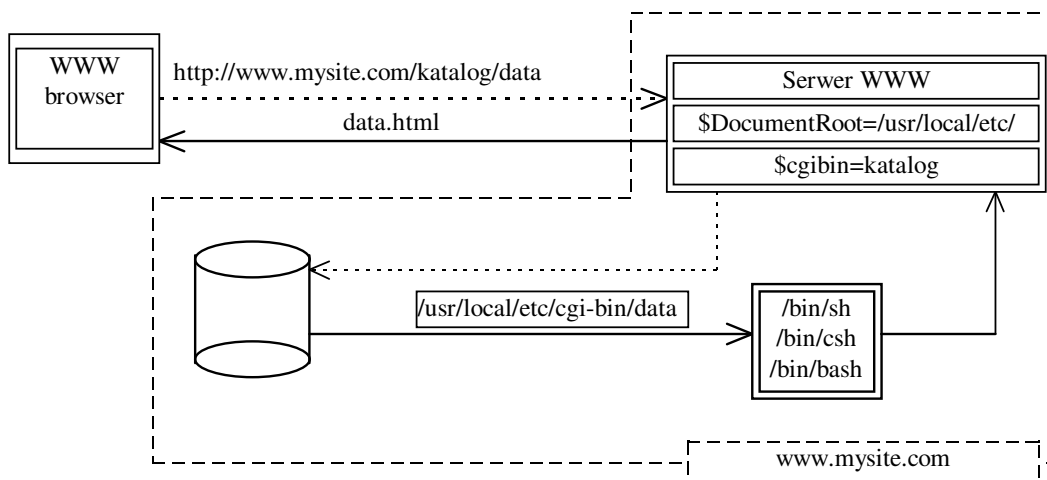
1.5.5.4. Generacja dynamiczna z użyciem skryptu CGI - mapowanie katalogów

Wspomniane w poprzednim punkcie problemy spowodowały, że pojawiło się tzw. mapowanie katalogów - lokalizacja pliku w adresie URL nie musiała odzwierciedlać już fizycznego położenia pliku, ponieważ każdy element ścieżki był przetwarzany przez serwer przed wyszukaniem pliku - np.

URL: `www.mysite.com/katalog/data`

był mapowany na fizyczny katalog

`/usr/local/etc/cgi-bin/` i plik `data`



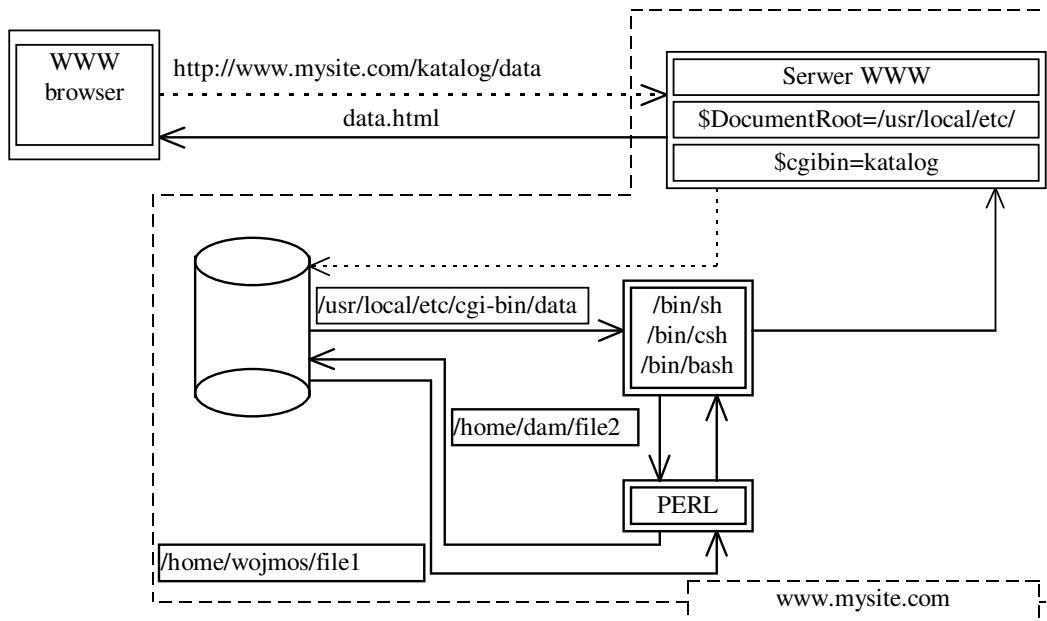
Rysunek 11: Generacja dynamiczna z użyciem skryptu CGI -
mapowanie katalogów

1.5.5.5. Generacja dynamiczna z użyciem CGI i zewnętrznych programów

Metoda ta była oparta na wypracowanym wcześniej mechanizmie skryptów CGI, ale tym razem chodziło o możliwość wywołania dodatkowych programów zewnętrznych. Dzięki temu możliwe było zastosowanie odpowiednich narzędzi przetwarzania plików tekstowych (np. awk, perl), co ułatwiło powstanie pierwszych prostych baz danych z interfejsem WWW (pierwszymi takimi bazami danych były liczniki odwołań do poszczególnych stron HTML, realizowane jako pliki tekstowe generowane na podstawie logów dostępu). W pierwszych koncepcjach projektu BIS

(czerwiec 1996) baza danych miała być realizowana właśnie ten sposób - później zaproponowaliśmy wykorzystanie programów napisanych w języku C, jako interfejsu API do Informixa, aby ostatecznie (styczeń 1997) zdecydować się na interfejs PHP/FI.

Warto tu także wspomnieć, że opisana metoda dała wygodny dostęp do plików znajdujących się poza katalogami znajdującymi się w drzewie serwera WWW.

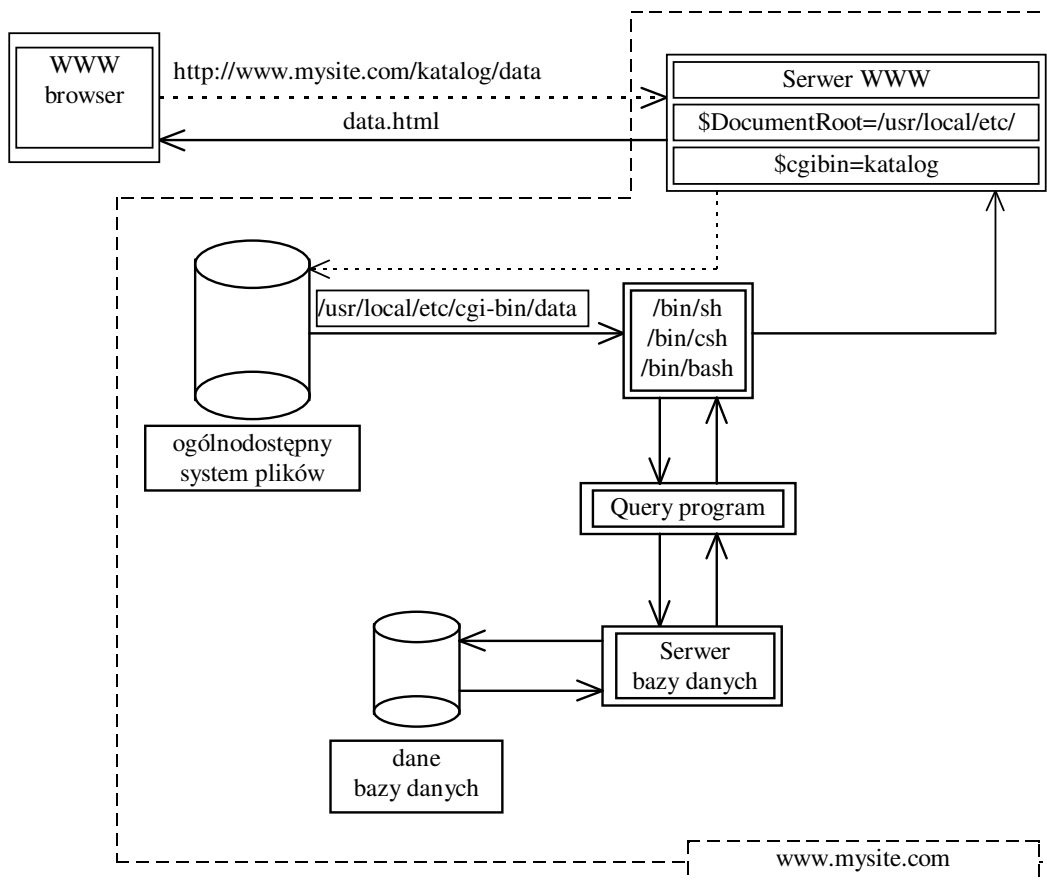


Rysunek 12: Generacja dynamiczna z użyciem CGI i zewnętrznych programów

1.5.5.6. Generacja dynamiczna z użyciem CGI i oprogramowania motoru bazy danych

Metoda ta nie była właściwie nowym rozwiązaniem - zastosowano w niej po prostu wywołanie określonego typu programów - realizujących zapytania do bazy danych. Jedyną nowością był zastosowanie motoru bazy danych, co spowodowało rozdzielenie systemu na dwa autonomiczne podsystemy, których realizacją zajmowały się dwa serwery (dzięki temu możliwe było zastosowanie dwóch komputerów o mniejszej mocy obliczeniowej - np. 2 x i386DX33, zamiast 486DX50). Pewnym minusem było trudniejsze wykorzystanie plików, ponieważ ich

obsługę trzeba było opierać na programach shella, lub wywoływanych z niego (niepotrzebna komplikacja pisanych skryptów).

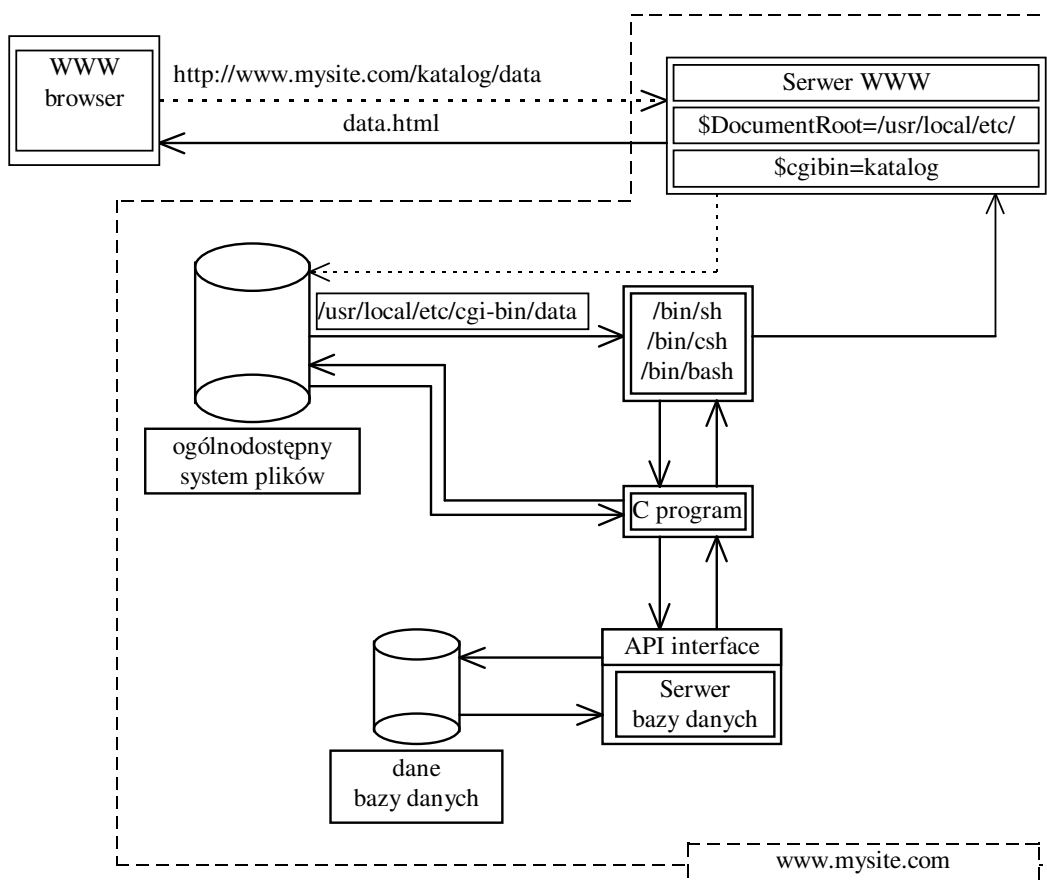


Rysunek 13: Generacja dynamiczna z użyciem CGI i oprogramowania motoru bazy danych

1.5.5.7. Generacja dynamiczna z użyciem CGI i API motoru bazy danych

Metoda ta stanowi rozwinięcie metody opisanej w poprzednim punkcie. Zamiast wykorzystywać programy z otoczenia serwera bazy danych (query programs), program obsługi wywołuje funkcje wewnętrznego interfejsu API, napisane zwykle w języku C (są one dostarczane w postaci modułów i plików nagłówkowych *.h). Metoda ta jest znacznie wygodniejsza niż opisana poprzednio, ponieważ programista może sam budować nowe funkcje, rozszerzając możliwości i dostosowując API do własnych potrzeb. Nie traci on również wygodnego dostępu do plików. Z punktu widzenia formalnego, API to część kodu serwera rozszerzająca

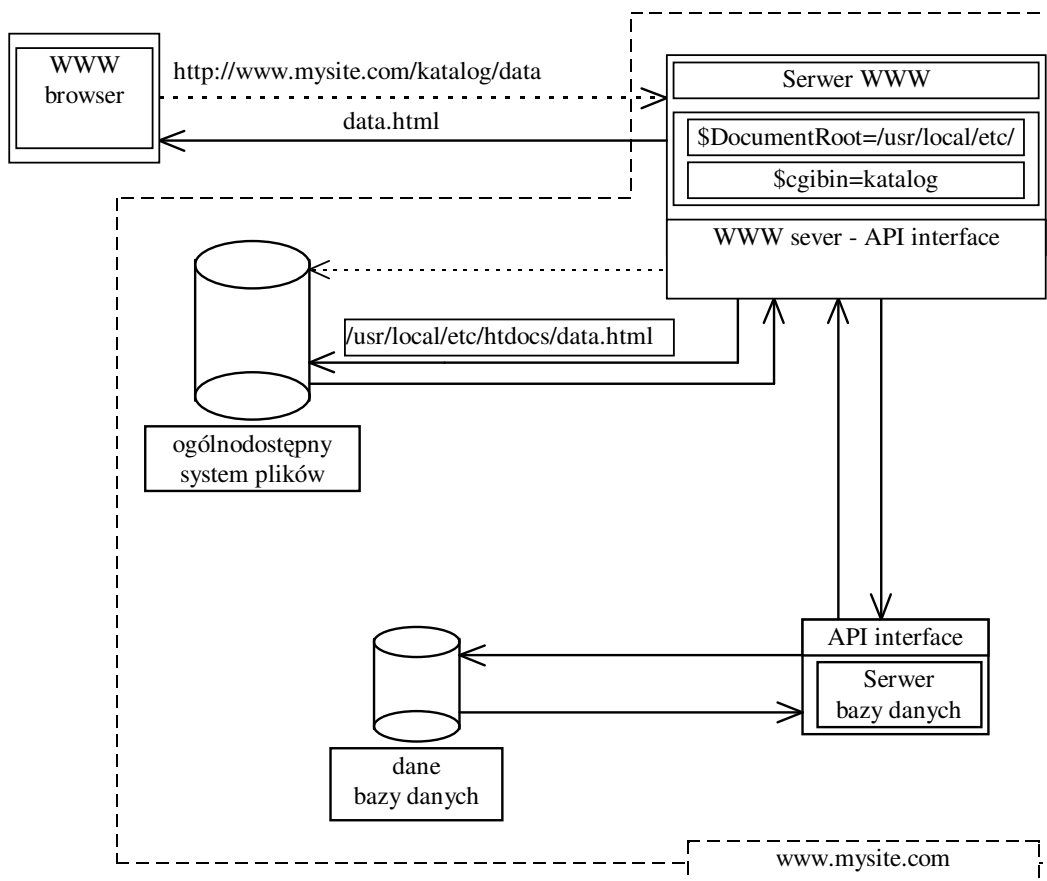
jego możliwości, może on być rodziwy lub firmowy. Tworzy się go specjalnie po to, aby zapewnić jak najwyższą wydajność serwera w czasie współpracy z inną aplikacją czy bazą danych. Kolejną zaletą jest bardzo niskie zapotrzebowanie na zasoby systemu. Programowanie API jest jednak znacznie trudniejsze niż tworzenie aplikacji CGI. Tworzenie aplikacji z API wymaga specjalistycznych technik programistycznych takich jak wielowątkowość, synchronizacja procesów, programowanie protokołu i obsługa błędów. Istnieje wiele odmian API: ISAPI (Internet Server API) firmy Microsoft, NSAPI (Netscape Server API) firmy Netscape czy WSAPI firmy O'Reilly.



Rysunek 14: Generacja dynamiczna z użyciem CGI i API motoru bazy danych

1.5.5.8. Generacja dynamiczna z użyciem API serwera WWW i API bazy danych

Jest to najbardziej zaawansowana - i do dziś stosunkowo jeszcze rzadko spotykana - metoda generacji dokumentów „w locie” (ang. in fly). Polega ona na rozszerzeniu zestawu komend HTML za pomocą rozbudowy wewnętrznego parsera, stanowiącego integralną część serwera WWW. Dzięki temu uzyskujemy możliwość bezpośredniego odwoływania się do interfejsu API motoru bazy danych z poziomu HTML. Warto tu także wspomnieć, że nie jest konieczne użycie zewnętrznych skryptów, ani programów z otoczenia motoru bazy danych.



Rysunek 15: Generacja dynamiczna z użyciem API serwera WWW i API bazy danych

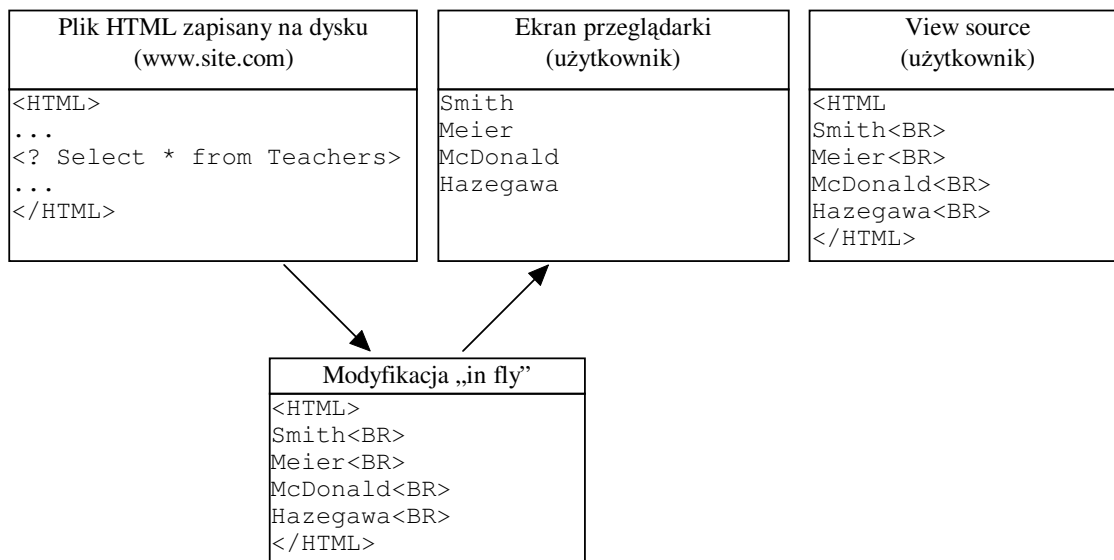
Należy także wskazać, że użytkownik nie ma dostępu do zawartości zapisanego na dysku pliku HTML - jest ona modyfikowana „in fly”, co powoduje, że po wywołaniu polecenia „view source” w przeglądarce uzyskujemy jedynie wynik

końcowy (jest to specyficzne dla generacji dynamicznej). Idee tą wyjaśniają poniższe rysunki:

Plik HTML zapisany na dysku (www.site.com)	Wygląd ekranu przeglądarki (użytkownik)	View source (użytkownik)
<pre><HTML> ... Smith
 Meier
 McDonald
 Hazegawa
 ... </HTML></pre>	<pre>Smith Meier McDonald Hazegawa</pre>	<pre><HTML> ... Smith
 Meier
 McDonald
 Hazegawa
 ... </HTML></pre>

Rysunek 16: Etapy generacji statycznej dokumentu

Rysunek pokazuje klasyczną, statyczną drogę dokumentu HTML. Na dysku serwera WWW zapisana jest postać ukazana z lewej strony (przykład zawiera listę nazwisk nauczycieli). Po przesłaniu pliku do przeglądarki klienta, lista ta ukazuje się na ekranie, zaś po wywołaniu „view source”, mamy dostęp (oczywiście typu „read”) do postaci pierwotnej. Jeśli któryś z nauczycieli zostaje zmieniony lub dodany, należy zaktualizować plik HTML. Niestety, aktualizacja ta musi odbyć się w trybie off-line (wyłączenie serwera WWW), ponieważ bezpośrednie odwołanie (zapis) do tego pliku mogłoby spowodować błędy (w sytuacji, gdy ktoś w tej samej chwili będzie chciał skorzystać z pliku za pomocą usługi WWW).



Rysunek 17: Etapy generacji dynamicznej dokumentu

Generacja dynamiczna umożliwia zmianę zawartości dokumentu HTML po jego odczycie z dysku, a przed wysłaniem do odbiorcy (przeglądarki). Dla użytkownika operacja ta jest całkowicie przezroczysta, otrzymuje on taką samą informację jak w przypadku poprzednim. Zalety tego rozwiązania to przede wszystkim umożliwienie wprowadzania zmian na bieżąco do zawartości dokumentu HTML - w rozpatrywanym przykładzie (wprowadzamy nowego nauczyciela) wystarcza wykonanie odpowiedniego zapytania do bazy danych (insert). W przypadku próby jednoczesnego dostępu do danych (zapytanie serwera WWW i modyfikacja) serwer bazy danych przeprowadzi w całości modyfikację, a następnie zrealizuje drugie żądanie (jest ono wstrzymane - serwer WWW czeka z wysłaniem dokumentu HTML na zakończenie modyfikacji „in fly”). Drugą zaletą tego rozwiązania jest ukrycie przed użytkownikiem mechanizmu, w jaki generujemy nasze pliki HTML - zapewnia to lepszą ochronę naszych danych i konkurencyjność naszych produktów (jeśli stosujemy np. specjalne algorytmy, które polepszają skuteczność naszej bazy danych, nasza konkurencja nie może skopiować naszych rozwiązań¹⁰).

¹⁰ Do tej pory usługa WWW nie była przed tym zabezpieczona - stąd efektywne rozwiązania stron domowych (np. ładne tło, ciekawe rysunki, nowatorskie rozwiązania plastyczne) były natychmiast kopiowane przez tłumy idących na łatwiznę autorów WWW (próba apeli znanych jako akcja „bądź

1.5.5.9. Rozwiązania alternatywne

Jednym z ważniejszych problemów, z jakimi wiązało się wykorzystanie metody CGI, był brak oszczędności zasobów - każdorazowe otwarcie strony i wywołanie skryptu powodowało generowanie nowego procesu. W przypadku relacyjnych baz danych, wiązało się to dodatkowo z otwieraniem coraz to nowych połączeń.

Podjmuje się ostatnio próby ulepszenia CGI - A. Montefusco (analityk i programista IBM Semea) zaproponował (patrz [19]) pakiet tzw. TGI (Task Gateway Interface) - aplikacji działających w systemie w sposób ciągły, lecz spełniających usługi tylko na żądanie. Ma to dodatkowo ułatwiać obsługę programów współbieżnych, ponieważ łatwiejsze staje się zarządzanie komunikacją międzyprocesową (skrypty CGI „terminują” się po wykonaniu swojego zadania).

1.5.6. Buforowanie informacji - W3cache

1.5.6.1. Opis ogólny

Wszyscy użytkownicy WWW stykają się problemem wolnej transmisji spowodowanym przeciążeniem łącz. Nawet bardzo szybkie medium transmisyjne może zostać przeładowane ogromną ilością informacji, co spowoduje spadek wartości prędkości transmisji nawet poniżej 0,1 Kb/s. Okazuje się jednak, że zauważalną poprawę sytuacji można uzyskać poprzez odpowiednie skonfigurowanie przeglądarki. Chodzi tu o wybranie tzw. serwera w3cache. Serwer tego typu monitoruje pracę użytkowników. Każde wywołanie strony WWW jest wprowadzane do serwera. Na podstawie tworzonej statystyki serwer sam ładuje do swoich buforów najczęściej odwiedzane strony. Każdy adres URL, który zostanie podany przez użytkownika trafia do serwera w3cache, który sprawdza, czy aktualnie strona ta znajduje się w buforze. Jeśli tak jest, to przesyła ją natychmiast do użytkownika (korzystając z jej stanu zapamiętanego w buforze). Pozwala to na skrócenie czasu dostępu do strony a jednocześnie na zmniejszenie ruchu w sieci dzięki skróceniu trasy przebiegu informacji.

Serwer w3cache pozwala również na dostęp do dwóch innych usług nie związanych z zapisem do cache'u. Pierwsza z nich to usługa „Security Proxy”

oryginalny” polegająca na umieszczaniu w stopkach stron HTML próśb o poszanowanie oryginalności pomysłów nie zdała egzaminu w praktyce).

pozwalająca na uzyskanie zabezpieczenia transmisji z poziomu serwera. Serwer ten kontroluje użytkownika, treść transmisji oraz miejsce z którego dana informacja jest ściągana (lub docelowe miejsce, do którego jest przesyłana) i uniemożliwia osobom trzecim podłączenie się do transmisji, co bez serwera w3cache jest dość proste. Ponadto serwer sprawdza adres i weryfikuje użytkownika otrzymującego informacje z danymi użytkownika od którego pochodzi żądanie ściągnięcia danych. Druga usługa dotyczy użytkowników, którzy korzystają z sieci chronionej przez „firewall host” lub „firewall router”. Chodzi tu o „SOCKS host”, czyli adres komputera na którym znajduje się oprogramowanie niezbędne do przejścia przez „ścianę ogniową” („firewall”). Tego typu informacje powinien udzielać zainteresowanym administrator danej sieci, gdyż nie są one objęte standaryzacją.

1.5.6.2. W3cache w Polsce

Polska nie odbiegła od światowych trendów i utworzyła sieć w3cache w tym samym czasie co inne państwa Europy. Sieć ta obejmuje serwery w3cache w większych miastach. Składa się ona generalnie z serwerów publicznych i dwóch serwerów lokalnych. Zasada ich działania jest taka sama, ale serwery lokalne obsługują tylko osoby posiadające konta w sieci, pod którą dany serwer podlega, albo tylko wymienione przez administratora domeny [„http://w3cache.icm.edu.pl.”](http://w3cache.icm.edu.pl) - „Serwery W3cache”.

Serwer	Rejon	Sprzęt
SunSITE.icm.edu.pl.	Polska	Sun SS1000E 6 CPU, 384 MB RAM

Tabela 5: Serwery w3cache na poziomie kraju.

Serwer	Rejon	Sprzęt
w3cache.pcz.czest.pl	Częstochowa	HP 9000, 2 CPU, 384 MB RAM
w3cache.agh.edu.pl.	Kraków	Sun SS20, 2 CPU, 128 MB RAM
w3cache.pk.edu.pl.	Kraków	SGI Challenge L, 2CPU, 512 MB RAM
w3cache.task.gda.pl.	Gdańsk	SGI Challenge XL, 2 CPU, 128 MB

		RAM
w3cache.us.edu.pl.	Katowice	Sun SS1000E, 2 CPU, 128 MB RAM
w3cache.man.lodz.pl.	Łódź	---
w3cache.man.poznan.pl.	Poznań	SGI Challenge L, 6 CPU, 128 MB RAM
w3cache.boa.uni.torun.pl.	Toruń	Sun SS20, 2 CPU, 96 MB RAM
w3cache.man.torun.pl.	Toruń	Sun SS1000E, 2 CPU, 256 MB RAM
w3cache.icm.edu.pl.	Warszawa	Sun SS1000E, 6 CPU, 384 MB RAM
w3cache.pwr.wroc.pl	Wrocław	Sun SS1000, 4 CPU, 256 MB RAM

Tabela 6: Serwery w3cache na poziomie większych miast.

Powyższa tabela przedstawia spis wszystkich serwerów w3cache działających na poziomie miast, a więc obsługujących największe grupy użytkowników. W pierwszej kolumnie tabeli znajduje się adres serwera (który należy podać podczas konfiguracji przeglądarek). Wszystkie serwery udostępniają swe usługi poprzez port 8080. Podczas wyboru serwera należy kierować się nie tylko jego odległością geograficzną, ale również dostępnymi między nimi połączeniami.

Serwer	Domena	Sprzęt
w3cache.amu.edu.pl	amu.edu.pl.	DEC
w3cache.twins.pk.edu.pl	pk.edu.pl.	DEC AXP, 160 MB RAM
wwwcache.icm.edu.pl	icm.edu.pl	Sun SS20, 96 MB RAM
w3cache.icm.edu.pl	mimuw.edu.pl. chem.uw.edu.pl. biogeo.uw.edu.pl. nencki.gov.pl. geo.uw.edu.pl. ibb.waw.pl. biol.uw.edu.pl. slcj.uw.edu.pl.	Sun SS1000E, 384 MB RAM
w3cache.amwaw.edu.pl	amwaw.edu.pl.	SGI

w3cache.sggw.waw.pl	sggw.waw.pl. kei.sggw.waw.pl.	Sun SS20, 64 MB RAM
w3cache.wsub.waw.pl.	wsub.waw.pl.	---

Tabela 7: Serwery w3cache na poziomie sieci lokalnych.

W powyższej tabeli znajdują się informacje o serwerach w3cache, które obsługują wyłącznie użytkowników z wybranych domen. Obsługują one głównie duże grupy użytkowników na uczelniach. W ostatniej kolumnie znajduje się informacja określająca sprzętową konfigurację komputerów udostępniających usługę w3cache (najważniejsze informacje to typ komputera, zasobach pamięci operacyjnej i dyskowej).

1.5.6.3. W3cache na świecie

Zestawienie projektów dotyczących w3cache na świecie „<http://w3cache.icm.edu.pl>.- „W3Cache Project” (dane na 08.01.96).

Kraj	Projekt	Liczba serwerów	Rodzaj
Austria	Ebone	1	---
Francja	Renater	4	ogólnokrajowy, nieformalny
Niemcy	DE-Cache	20	ogólnokrajowy
Wielka Brytania	SunSITE	1	eksperymentalny
Wielka Brytania	MCC	---	---
Włochy	INFN	1	---
Szwecja	KTH	2	eksperymentalny
USA	NLANR	6	ogólnokrajowy, badawczy
Afryka	---	---	komercyjny
Dania	SunSITE	1	---
Nowa Zelandia	NZGate	---	komercyjny

Finlandia	---	---	eksperymentalny
Rumunia	---	---	eksperymentalny
Rosja	---	---	komercyjny
Australia	Ichthus	---	---
Wielka Brytania	Hensa	1	---
Holandia	Lagoon	1	---

Tabela 8: W3cache na świecie

Powyższa tabela przedstawia zestawienie światowych projektów związanych z usługą w3cache. Niestety, w większości przypadków niemożliwe okazało się ustalenie nawet przybliżonych liczb serwerów, których projekty te dotyczą. Pomimo to można zauważyć duże zainteresowanie państw europejskich takich jak Niemcy czy Francja oraz krajów takich jak USA i Rosja.

1.5.7. Bezpieczeństwo

Zagrożenia bezpieczeństwa są szczególnie ważne w dzisiejszym zglobalizowanym świecie informatycznym. W momencie podłączania systemu do ogólnodostępnej sieci komputerowej (np. do Internetu) potencjalne niebezpieczeństwo włamania radykalnie wzrasta (maleje zaś prawdopodobieństwo odkrycia kto włamał się do naszego systemu). Nie należy jednak zapominać, że nadal poważne zagrożenie tkwi w aspektach „niesieciowych” (statystyki FBI szacują te zagrożenia na ponad 60% ogółu włamań). Zagadnienia bezpiecznej pracy sieciowej sprowadzają się do ochrony trzech punktów:

- terminala maszyny pracującej w sieci,
- sieci wewnętrznej (Intranet) przedsiębiorstwa (zwykle chronionej przez firewall)
- otoczenia sieci wewnętrznej (z perspektywy sieci globalnej)

Idealna strategia oceny wiarygodności ochrony dostępu bazuje na znalezieniu i usunięciu potencjalnych luk przed podłączeniem komputera do sieci Intranet. W praktyce jednak okazuje się, że w sporej części przedsiębiorstw (zwłaszcza małych i średnich firm branż nieinformatycznych) testowanie systemów odbywa się już po uruchomieniu systemu (a nawet po podłączeniu do Internetu).

W dziedzinie usług WWW sytuacji takiej sprzyjała szybkość z jaką rozwijał się ten rynek - produkty (także profesjonalne !) często były nie do końca sprawdzone pod względem wiarygodności. Najbardziej jaskrawym tego przypadkiem była możliwość uzyskania praw administratora przez każdego użytkownika systemu Unix na którym pracował serwer WWW NCSA 1.2 lub NCSA 1.3 (w rozpatrywanym okresie serwery te były jednymi z nielicznych istniejących i pracowały na znaczącej części komputerów).

Luki w zapewnieniu bezpieczeństwa stron WWW to do dziś popularne „dziury” systemów informatycznych. Wynika to często z niekompetencji ludzi obsługujących te serwisy (utworzyć serwer WWW i proste strony w języku HTML jest bardzo łatwo). Jako przykład można tu podać „włamania” polskich hackerów na strony domowe Sejmu i b. premiera Waldemara Pawlaka. W chwili obecnej na rynek wchodzi specjalistyczne programy przeznaczone do automatycznej analizy systemu komputerowego pracującego z użyciem WWW pod kątem zagrożeń. Przykładem takiego oprogramowania jest ScanWWW¹¹. Okazuje się, że istniejące w tej chwili systemy zabezpieczeń nie są instalowane i używane poprawnie, co sprawia, że wiarygodność ochrony systemu jest tylko pozorna. Poniżej podajemy kilka przykładów metod używanych przez tzw. hackerów do ominięcia zabezpieczeń:

- „CGI attack” - typowy atak dokonywany przez system WWW. Bazuje on na nieprawidłowym działaniu lub wykorzystaniu skryptów cgi. Nieumiejętna konfiguracja serwera WWW przez początkujących webmasterów powoduje, że atak taki jest bardzo skuteczny.

- Podsluch haseł („net sniffing”). Opiera się na fakcie, że w standardowych protokołach używanych w sieci Internet dane (a więc i hasła dostępu) są przekazywane w postaci niezakodowanej, co umożliwia odpowiednim programom filtrowanie treści pakietów.

- „Source porting”: reguły filtrowania w „ścianach ogniowych” (firewall) są często odnoszone do numerów portów. Maszyna ze stosem TCP/IP posiada 65535

¹¹ Inna nazwa to Internet Scanner SAFESutie - oprogramowanie sprawdzające systemy WWW pod kątem bezpieczeństwa. Pakiet działa w środowiskach Win NT 3.51, Sun OS 4.1.3, HP-UX 9.05, IBM AIX 3.2.5, Linux 1.2.x (wymagane aktualizacje jądra w postaci „łat” - patches), Linux 2.0.x - oprogramowanie to zostało bliżej opisane w artykule „Wszechstronna analiza bezpieczeństwa sieci” - patrz „Software” 6/96. Autor, Robert Głowczyński jest pracownikiem Instytutu Bezpieczeństwa Sieciowego.

numerów portów. Niektóre z nich posiadają konkretne zastosowania - np. port poczty email ma numer 25. Wiele systemów firewall pozwala na otwarcie sesji ftp (dla której przeznaczona jest port 20), więc potencjalny włamywacz może zmodyfikować telnet tak, aby połączenie inicjowane było na port przewidziany dla usługi ftp.

- „Source routing” wykorzystuje dostępną w protokole IP opcję pozwalającą na zdefiniowanie sposobu routingu pakietów co pozwala na obejście wielu reguł filtracji.

- „SOCKs attack”. SOCKs to bogata w różnorakie funkcje biblioteka firewall. Jest to narzędzie pozwalające m.in. skonfigurować system nadając usługom i użytkownikom określone prawa. Złe jego użycie (najbardziej znany błąd to nadanie praw określonym użytkownikom bez równoczesnego zabronienia dostępu dla innych) umożliwia proste przejście przez firewall.

- „IP spoofing” to technika wykorzystująca wzajemne relacje zaufania hostów. Jeżeli komputer A „ufa” komputerowi B pozwalając np. na logowanie się bez hasła, to „napastnik” może próbować podmiany pakietów. W tym celu, do generowanych przez siebie pakietów dopisuje on jako adres nadawcy adres „zaufanego” hosta. Metoda ta bywa także nazywana metodą przewidywania numerów TCP, ponieważ wymaga wstawienia numeru kolejnego pakietu w ciągu. Tej techniki włamań używał m.in. znany hacker Kevin Mitnick

- „direct RPC scanning” to poszukiwanie, którym usługom RPC przyporządkowane poszczególne porty komunikacyjne. Zwykle blokada tzw. „portmappera” (port 111, usługa katalogowa) pozwala uniknąć wykrycia funkcji RPC, ale stosując szukanie bezpośrednie można odnaleźć potrzebną usługę.

- „stealth scanning”, czyli „ciche przeszukiwanie” to próby nawiązywania połączenia (w celu „wybadania” maszyny, na którą prowadzony jest atak) prowadzone na bardzo niskim poziomie protokołu transmisyjnego. Osiąga się dzięki temu brak alarmu np. podnoszonego w oparciu o śledzenie klasycznych połączeń do maszyny (w takie alarmy, doskonale chroniące przed atakiem typu „brute force” wyposażone są powszechnie używane systemy zabezpieczeń, np. szeroko znany SATAN).

1.5.8. Przesyłanie dźwięku i obrazu poprzez sieć Internet.

Zmiany technologiczne w dziedzinie komputerów osobistych i zwiększająca się popularność Internetu umożliwiły rozwój oprogramowania i usług związanych z przesyłaniem dźwięku i obrazu przez sieć Internet. Należy także podkreślić zainteresowanie Internetem jakie wykazuje szeroko pojęty business, traktujący globalną sieć jako inwestycje w rynek reklamowy i usługowy. Owocuje to poważnym dokapitalizowaniem tej dziedziny informatyki, co stymuluje dalszy, trwały i szybki rozwój.

Podstawy technologiczne:

- komputery PC są standardowo wyposażane w szybki procesor i kartę dźwiękową (w tej chwili można przyjąć, że przynajmniej połowa sprzedawanych obecnie komputerów posiada kartę dźwiękową i dostatecznie szybki procesor, który umożliwia obróbkę dźwięku)
- w przyszłym roku standardem staną się komputery z procesorami zawierającymi instrukcje multimedialne (MMX)
- standardowe modemy mają prędkość transmisji **28.8 kbps** lub większa
- firmy Netscape (Cool Talk) i Microsoft (Net Meeting) rozpowszechniają darmowe oprogramowanie do prowadzenia rozmów i konferencji przez Internet
- zaawansowane techniki kompresji i redukcji mowy umożliwiają co najmniej **4-krotną** redukcję zajmowanego pasma bez pogarszania jakości mowy (do ok. 17 kbps jak w GSM)

Wszystko to powoduje, że przesyłanie dźwięku poprzez Internet staje się coraz bardziej popularne. Najpoważniejsze w tej chwili problemy w rozwoju tej technologii to:

- brak ogólnie akceptowanych standardów
- brak możliwości rezerwacji pasma pomiędzy użytkownikami
- nadal stosunkowo duże obciążenie dla łącz

1.5.9. Nowa koncepcja usług telekomunikacyjnych

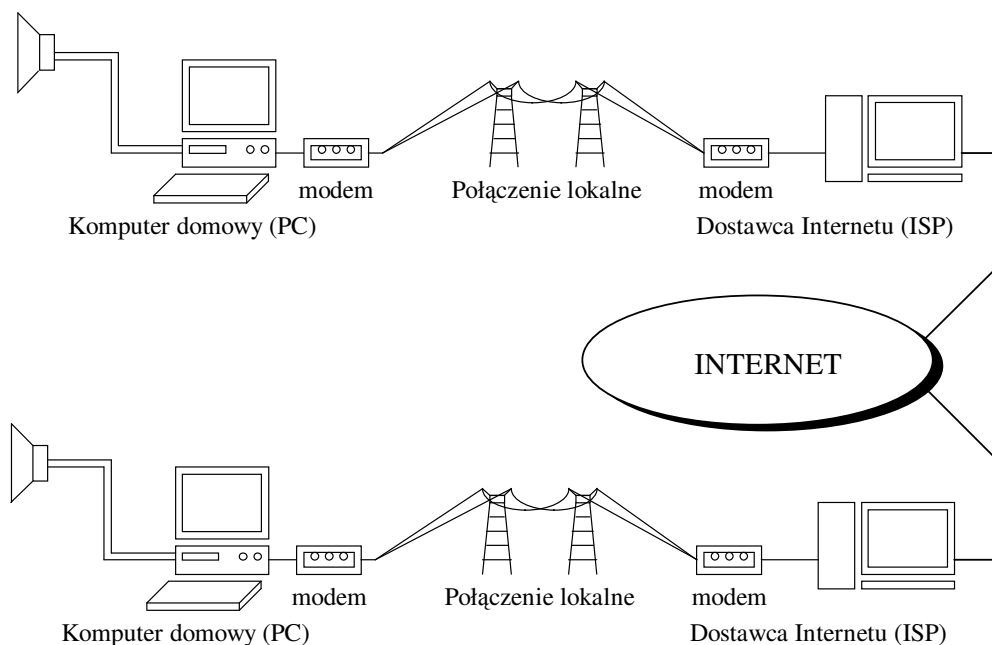
W ostatnich latach, dzięki postępowi technologii mamy do czynienia z zupełnie nowym zjawiskiem: Internet i oprogramowanie stosowane do jego

wykorzystania zaczyna migrować na rynki zdominowane do tej pory przez klasycznych potentatów. Typowym przykładem może być tu rynek mediów prasowych - pojawiająca się w Internecie informacja jest szybsza, ciekawsza w formie (techniki animacji i wykorzystanie dźwięku) i konkurencyjna np. w stosunku do wydawnictw papierowych (czasopisma, gazety codzienne). Inny, podobny przykład to rynek telewizyjny. W tej sytuacji większość koncernów prasowych i telewizyjnych stara się uczestniczyć w nowokreującym się rynku WWW. Bieżące serwisy informacyjne na stronach WWW prowadzą np. BBC, NBC oraz Time, a w Polsce - w postaci elektronicznych wersji gazety - np. Rzeczpospolita i Wprost).

Sieć Internetu niesie też wielkie zagrożenie dla interesów operatorów sieci telekomunikacyjnych. Korzystając z oprogramowania obróbki sygnału dźwiękowego (jest ono dostępne m.in. w sieci w postaci shareware i komercyjnej¹²) można w prosty sposób zrealizować połączenie międzynarodowe bez ponoszenia związanych z tym opłat (użytkownik rozmawiający z domu za pośrednictwem Internetu np. z Polski do Japonii płaci tylko za połączenie z lokalnym dostawcą Internetu¹³). W podobny zresztą sposób usługi Internetowe wypierają z rynku tradycyjne faxy - przesyłanie dokumentów przez Internet jest tańsze, bardziej wydajne i bezpieczniejsze (dane można szyfrować).

¹² np. pakiet oprogramowania Intela

¹³ Ang. Internet Service Provider (ISP)



Rysunek 18: Uzyskiwanie tanich połączeń międzynarodowych

1.5.10. Wyszukiwanie informacji

1.5.10.1. Charakterystyka usługi

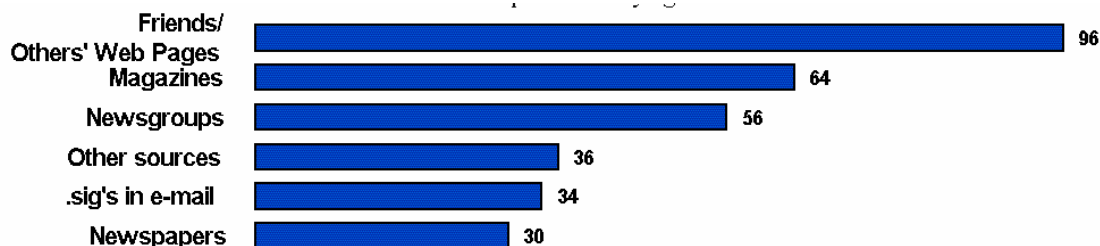
Praktycznie niemożliwe jest oszacowanie (nawet w przybliżeniu) liczby stron udostępnianych w Internecie. Jest to ogromne źródło informacji. Wyszukanie interesujących nas danych w takich warunkach nie jest sprawą prostą. Dlatego też w wielu punktach sieci udostępniane są narzędzia zwane wyszukiwaczami informacji (ang. „search engines”). Jest to w chwili obecnej jedna z najpopularniejszych usług w sieci WWW. Najbardziej znanymi w tej dziedzinie są m.in: Yahoo, AltaVista, Lycos, RedHot, WebCrawler, Infoseek. Wciąż jednak powstają nowe, bardziej efektywne i wygodniejsze w obsłudze narzędzia. Zadaniem wyszukiwarki jest udostępnienie użytkownikowi informacji o lokalizacji stron zawierających informacje na podany przez niego temat. Jej działanie jest ściśle związane z wykorzystaniem baz danych, które przechowują i udostępniają zgromadzone informacje. We wszystkich znanych miejscach sieci WWW udostępnianie tego rodzaju informacji jest bezpłatne (dostawca tego typu usług czerpie natomiast poważne zyski z akcji reklamowych - patrz 2.7.2).

Jednym z poważnych problemów wiążących z zarządzaniem wyszukiwarką jest dbanie o aktualność informacji. O ile bowiem tworząc „webowy site”

użytkownik zwykle wysyła informacje o tym fakcie do większych serwerów gromadzących informacje o sieci, to zamykając go lub przenosząc robi to stosunkowo rzadko. Efektem jest nagminne występowanie przestarzałych adresów na stronach wyszukiwarek. Niektóre serwery próbują sobie z tym radzić co jakiś czas automatycznie łącząc się z adresami pobranymi z bazy danych, ale rozwiązanie to nie zapewnia 100% skuteczności (sewer może wykryć tylko brak strony pod wskazanym adresem, co nie zadowala nas, ponieważ pod tym adresem może znajdować się strona o zupełnie innej tematyce - stworzona na miejsce dawnej), a po za tym generuje zbędny duży ruch w sieci, przeciążając łącza.

Inny, niezmiernie ważny problem to „przeciążenie informacją”. W chwili obecnej, w epoce burzliwego rozwoju WWW istnieją setki tysięcy stron dotyczących określonych tematów. Zadanie wyszukiwarce ogólnego pytania mija się praktycznie z celem, ponieważ po uzyskaniu np. 500.000 adresów URL, które łączą się z interesującą nas dziedziną nie jesteśmy ani o krok dalej w kierunku zlokalizowania interesującej nas strony. Rozwiązanie tej sytuacji stanowi zupełnie nowy problem w dziedzinie baz danych - filtrowanie zbyt obszernych odpowiedzi. Ostatnio zaproponowano (w stosunku co prawda tylko do wyszukiwania grup list dyskusyjnych, ale problem jest podobny) zastosowanie sztucznej inteligencji - sieci neuronowych, które próbują zorientować się „za nas” w zawartości listy dyskusyjnej. Sieć taka uczy się na podstawie najczęściej czytanych przez użytkownika artykułów, jaka tematyka go interesuje (badane jest m.in. występowanie i częstotliwość pojawiania się określonych słów i zwrotów).

Opisując zagadnienie wyszukiwania informacji nie sposób nie wspomnieć o preferencjach użytkowników. Okazuje się, że są oni stosunkowo odporni na „reklamę bezpośrednią”, dużym zaufaniem obdarzają natomiast porady swoich przyjaciół (za: Michigan Business School WWW Survey Results, http://www.cc.gatech.edu/gvu/user_surveys/).



1.5.10.2. Opis funkcjonalny usługi.

Wszystkie wyszukiwarki informacji podają w ten lub inny sposób adresy stron WWW dotyczących wybranego przez nas tematu. Zwykle do tego celu udostępniany jest prosty interfejs pozwalający użytkownikowi na podanie nazwy interesującej go dziedziny, zdefiniowanie zakresu poszukiwań (Usenet, adresy e-mail, Web) oraz wybór sposobu wyświetlania odpowiedzi.

Search and Display the Results

Search with Digital's Alta Vista [[Advanced Search](#)]

Rysunek 19: Interfejs WWW: proste wyszukiwanie danych

Po zdefiniowaniu tematu można wybrać przycisk „Submit”, co spowoduje rozpoczęcie wyszukiwania interesujących nas informacji.

Find all listings containing the *keys* (separated by space)

Search Yahoo! Usenet Email Addresses

Find only new listings added during the past

Find listings that contain

- At least one of the *keys* (boolean **or**)
- All *keys* (boolean **and**)

Consider *keys* to be

- Substrings
- Complete words

Display listings per page

Rysunek 20: Interfejs WWW: wyszukiwanie zaawansowane

Odpowiedź uzyskujemy zwykle bardzo szybko (opóźnienia w praktyce są spowodowane istniejącym ruchem w sieci). Większość narzędzi oprócz tych podstawowych funkcji udostępnia również znacznie bogatsze możliwości konfiguracji.

Format uzyskiwanych odpowiedzi zależy od użytkownika. Możemy zażyczyć sobie wyświetlenia informacji w "porcjach" np. po 25 odnośników. Możemy również wybrać okres czasu powstania danej strony, zdecydować czy chcemy, aby wyszukiwanie obejmowało fragmenty podanych słów czy tylko całe słowa. Jednak po dość ogólnym sformułowaniu tematu prawdopodobnie uzyskamy odpowiedź zawierającą np. 100.000 adresów stron na których znajduje się podany przez nas wyraz. Dlatego też warto podawać kilka nazw dokładniej charakteryzujących dziedzinę a następnie wybrać sposób interpretacji podanych wyrazów. Chodzi tu o zdecydowanie czy muszą być wszystkie nazwy w jednym dokumencie czy wystarczy, aby znalazła się jedna.

W zależności od firmy udostępniającej tego typu interfejs można spotkać się również z innymi opcjami ułatwiającymi wyszukiwanie informacji. Może to być udostępnienie katalogu zgromadzonych danych co pozwala na zawężenie obszaru poszukiwań i w efekcie szybsze dojście do fachowych danych.

<ul style="list-style-type: none"> • Arts and Humanities Architecture, Photography, Literature... • Business and Economy [Xtra!] Companies, Investments, Classifieds... • Computers and Internet [Xtra!] Internet, WWW, Software, Multimedia... • Education Universities, K-12, College Entrance... • Entertainment [Xtra!] Cool Links, Movies, Music, Humor... • Government Military, Politics [Xtra!], Law... • Health [Xtra!] Medicine, Drugs, Diseases, Fitness... 	<ul style="list-style-type: none"> • News and Media [Xtra!] Current Events, Magazines, TV, Newspapers... • Recreation and Sports [Xtra!] Sports, Games, Travel, Autos, Outdoors... • Reference Libraries, Dictionaries, Phone Numbers... • Regional Countries, Regions, U.S. States... • Science CS, Biology, Astronomy, Engineering... • Social Science Anthropology, Sociology, Economics... • Society and Culture People, Environment, Religion...
--	---

Rysunek 21: Interfejs WWW: katalog przedmiotowy (<http://www.yahoo.com>)

Oprócz opcji umożliwiających sprecyzowanie poszukiwanej informacji zwykle możemy dodać nowy adres URL („Add URL”). Pozwala to na dodanie do bazy danych odnośnika np.: do własnej strony WWW. Po wybraniu odpowiedniej opcji musimy tylko wypełnić prostą „ankietę” zawierającą pytania dotyczące nowo dodawanego odnośnika. Twórcy tego typu oprogramowania nie zapominają również o udostępnieniu szczegółowych informacji dotyczących obsługi wyszukiwarki zarówno dla zaawansowanego użytkownika jak i „nowicjusza”. Jak zwykle można ją uzyskać wybierając opcję „Help” (czasem „About”).



Rysunek 22: Interfejs WWW: menu w postaci banneru

1.6. Ekonomia Internetu i WWW

1.6.1. Rys historyczny: pierwsza walka o rynek

Wydaje się, że w pierwszych latach swojego istnienia rynek Internetu jako potencjalne miejsce do zarabiania pieniędzy został bardzo niedoceniony. Można tak twierdzić zwłaszcza w kontekście usług WWW - ponieważ po okresie dominacji niekomercyjnego serwera i klienta NCSA Mosaic tylko jedna firma zdecydowała się na zaangażowanie na tym polu. Była to firma Netscape Communications, która wyprodukowała pierwsze wersje najbardziej obecnie znanego na świecie przeglądarki. Trzeba sobie zdawać sprawę, że mimo, iż Netscape przejął część specjalistów z NCSA, jego położenie nie było łatwe - potencjalna konkurencja (Mosaic, czy nawet tekstowy Lynx) nie była nastawiona na zysk (były to de facto projekty uniwersyteckie) i jej software był bezpłatny. Netscape postanowił więc drastycznie podnieść jakość usług - wprowadził do swojego oprogramowania nowy, bardzo atrakcyjny ze względu na stopień kompresji danych format graficzny JPEG, umożliwił „inteligentne wczytywanie” stron http (najpierw tekst, potem grafika, niezależnie od rozmieszczenia na stronie itp.). Warto także podkreślić, że praktycznie wszystkie programy Netscape były (i są) dostępne w wersji bezpłatnej - dla ośrodków akademickich i zastosowań niekomercyjnych. Strategia ta miała na celu pobudzenie rynku usług WWW i dotarcie do szerokich mas klientów.

Dodatkowy zysk polegał na tym, że dominując na rynkach akademickich, Netscape „przyzwyczajal” klientów do własnych rozwiązań, co miało potem owocować, gdy kończący uczelnie studenci odchodzili do własnych firm, wyznaczając im kierunki rozwoju i często decydując o polityce informatyzacji. Pierwszy etap wypełniania rynku WWW zakończył się totalną dominacją browserów Netscape.

Rozwój rynku internetowego został wyraźnie zlekceważony w firmie Microsoft, która zajęta była wówczas walką o rynek komputerów osobistych i pracami nad systemem Windows 95. Wypuszczenie na rynek pierwszych wersji Microsoft Internet Explorer było pierwszym sygnałem, że firma Gatesa nie zamierza jednak oddać potężnego rynku bez walki. Netscape docenił niebezpieczeństwo i odpowiedział kolejnymi wersjami swoich browserów, wprowadzając nowe rozwiązania, niedostępne w produktach Microsoftu. Dużym ułatwieniem dla Netscape była sytuacja, w której rynek, zdominowany przez produkty jednej firmy uważał nowinki proponowane w jej wyrobach za propozycję standardu języka HTML. Pracująca nad standaryzacją HTML 3.0 i nowymi elementami języka grupa programistyczna tworząca oficjalną przeglądarkę Arena HTML3.0 praktycznie nie liczyła się. Microsoft jednak przyjął politykę przejmowania rozwiązań technicznych Netscape - i tak ramki (frames), które pojawiły się w przeglądarce Netscape 2.0, pojawiły się też w Explorerze 3.0. Drugi etap walki o rynek zakończył się wyrównaniem szans - Netscape musiał przyjąć dużego konkurenta na swoim rynku i pogodzić się z faktem utraty absolutnej supremacji.

Powstaje pytanie, jakie będą dalsze losy obu firm. Walka konkurencyjna nie wygasła bowiem, zwłaszcza że Microsoft nastawił się na potężną inwestycję - jej Internet Explorer jest oprogramowanie bezpłatnym, a jego zapowiadane nowe wersje 4.0 i 5.0 mają być w pełni zintegrowane ze środowiskiem Windows 97. W praktyce oznaczać to ma podobno możliwość uruchamiania programów z poziomu przeglądarki i zapewne wykorzystanie mechanizmów OLE. Dodatkowym atutem jest możliwość wykorzystania możliwości środowiska, a więc np. korzystania z wbudowanych w Windows metod obsługi animacji i filmów (pliki *.avi) itp. Poważną wadą strategii Microsoft jest brak zainteresowania szerszym rynkiem - jego software jest dostępny głównie w wersji Windows-PC. Tę szansę stara się wykorzystać Netscape, wchodząc na rynek szerokim frontem oprogramowania

przeznaczonego na systemy Unix-PC (Linux), SGI, Mac itp. Istnieją też oczywiście „przyczółki” na platformach Windowsowych np. Netscape for Windows.

Jak na razie Netscape wyszedł zwycięsko z rywalizacji, obroty firmy wzrosły o 329% i wyniosły 100 mln USD (dla porównania: rok wcześniej saldo wyniosło 23 mln !). Rok 1995 firma zamknęła zyskiem netto 7.7 mln USD. Mimo wielu prognoz o zbliżającym się upadku firmy (na skutek konkurencji Microsoftu), prezes firmy Jim Barskdale uważa, że firma rokuje duże nadzieje zwłaszcza związane z wejściem na rynek korporacyjny.

85% dochodów Netscape jest związanych ze sprzedażą oprogramowania (59% - browsery, 25% - serwery, 6% - inne). Mimo takiej struktury zysku, Netscape jest także poważnym uczestnikiem rynku reklam, na którym w zeszłym kwartale zarobił około 7 mln USD.

Dla porównania zamieszczamy wyniki finansowe innych firm:

Unisys Corp. - zysk w ostatnim kwartale 14.2 mln USD (poprzednio starta 32.2 mln USD w skali roku !). Przychód - 1.63 mln.

Bayan Company N.V. - zysk netto w ciągu 1 kwartału - 9.3 mln USD (3.1 mln rok temu), przychód 97.7 mln, co było 80% wzrostem w stosunku do analogicznego okresu roku ubiegłego.

Microsoft - przychód 2.3 mld USD (zbliżony do wyników roku poprzedniego). Zysk - 614 mln (wyniki obejmują całość działalności, a więc również sprzedaż systemów Windows 95, Office itp.)

Silicon Graphics - strata 22 mln USD, przychód 766 mln USD. Opóźnienie spowodowane oczekiwaniami klientów na nowe produkty (interpretacja przedstawicieli firmy).

1.6.2. WWW jako najbardziej dochodowy rynek reklam

Indywidualne strony WWW, nie stanowią skutecznej reklamy, jak wynika z obserwacji częstości ich odczytów są one odwiedzane od kilku do kilkuset razy w tygodniu. Jest to spowodowane to po części istnieniem ogromnej ilości takich stron, jak również zbyt małym upowszechnieniem informacji o tych stronach.

Główny rozwój reklamy wiąże się jednak z umieszczaniem odnośników do innych stron, w serwisach, które są najczęściej odwiedzane przez użytkowników sieci Internet. Na pierwszym miejscu należy umieścić serwisy związane z wyszukiwaniem informacji. Firmy zainteresowane reklamą, chcą umieszczać tam swoją reklamę, ze względu na bardzo dużą liczbę użytkowników odwiedzających te miejsca. Przykładowe ilości odwołań do najpopularniejszych stron prezentuje poniższa tabel. Dane te dotyczą jednego tygodnia.

Firma	Ilośćostępów	Szacowna liczba różnych użytkowników
Netscape	30,000,000	3,000,000
Yahoo	9,452,579	1,400,000
ESPNNet SportsZone	8,500,000	1,975,701
InfoSeek Corp	6,000,000	910,000
Pathfinder	4,800,000	1,400,000
Playboy	4,723,957	1,141,112
HotWired	3,000,000	428,571
Microsoft	3,000,000	280,000

Tabela 9: Liczba odwołań do wybranych stron WWW

Istnieje pewien problem z interpretacją tych danych, spowodowany faktem, że firmy te rejestrują jedynie liczbę dostępów do ich stron. Z liczby tej nie wynika liczba różnych użytkowników, którzy trafiają na ich strony (ten sam użytkownik może wielokrotnie korzystać z danego site)¹⁴. Dodatkowo należy brać pod uwagę, że dane te są brane na podstawie oficjalnych raportów firm, a firmy te są zainteresowane - z oczywistych względów marketingowych - podawaniem jak najwyższych liczb.

¹⁴ Ponadto istnieje jeszcze kilka „szczegółów technicznych” - m.in. fakt, że w protokole http, każde odwołanie do obiektu osadzonego w dokumencie HTML generuje w logu zapis nowego połączenia. W ten sposób, użytkownik łączący się ze stroną, na której jest wiele małych rysunków (np. tzw.

Najpopularniejsze serwisy są w większości bezpłatne, co sprawia, że trafia do nich bardzo duża liczba użytkowników Internetu. Korzystają z tego operatorzy tych usług, oferując chętnym miejsce na płatną reklamę. Płatna reklama na takich stronach polega na umieszczeniu tak zwanych "bannerów" reklamujących zwykle inne strony WWW, lub określone produkty. Przykłady takich reklam prezentuje rysunek:



Rysunek 23: Przykłady reklam w postaci bannerów

Koszty umieszczenia tych niewielkich reklamowych obrazków są bardzo wysokie. Poniższa tabela prezentuje przykładowe cenniki niektórych firm:

Miejsce	Cena	Dzienny ruch
Netscape	od \$10 000 do \$30 000 miesięcznie	400 000 osób/dzień
Yahoo	\$20 000 miesięcznie	300 000 osób/dzień
HotWired	\$15 000 miesięcznie	250 000 osób/dzień

Tabela 10: Koszt prostej reklamy w WWW (banner)

Zainteresowanie tą formą reklamy wzrasta jednak bardzo silnie, po pierwsze ze względu na ciągle wzrastającą liczbę użytkowników sieci Internet (potencjalni klienci), jak również ze względu na zwiększający się ruch w sieci. Reklama tego typu przynosi też nowe niedostępne dla dotychczasowych mediów możliwości. Użytkownik zainteresowany reklamą, przechodzi bezpośrednio na strony WWW danej firmy, tak więc zamieszczenie małego obrazka na często odwiedzanej stronie, jest ekwiwalentem dołączenia całego katalogu produktów danej firmy do kolorowego czasopisma

„bulletów”, pełniących rolę wypunktowania listy) generuje za każdym razem wielokrotne żądanie dostępu do tej strony.

1.6.3. Usługi finansowe

1.6.3.1. Home banking

Home banking to jedna z najbardziej ekscytujących perspektyw rozwoju systemów bankowych. Jej idea to wyeliminowanie konieczności czasochłonnych kontaktów klienta z bankiem, operacji na fizycznych pieniądzach i pośrednictwa finansowego. Do niedawna home banking pozostawało w świecie science-fiction, dziś obserwujemy narodziny tego obiecującego rynku.

Home banking nie może jednak powstać z niczego. Wymaga on, jak każdy rodzaj działalności odpowiedniego zaplecza. W tym przypadku o sukcesie home bankingu decydują następujące elementy:

- istnienie i dostępność medium komunikacyjnego - dotychczas nie było praktycznie środka komunikacyjnego dostępnego powszechnie dla każdego użytkownika. Podstawą bowiem home bankingu musi być powszechność środków dostępu - i oczywiście jego atrakcyjność cenowa. Dwa ewentualne środki dostępu do odbiorcy, które praktycznie każdy mógłby mieć to telewizja i telefon. Pierwszy z nich nie udostępnia jednak możliwości komunikacji w obie strony („interaktywności” - a propozycje takiej telewizji dopiero się pojawiają), drugi - jest ograniczony do łączności głosowej. Dopiero pojawienie się globalnej sieci komputerowej zlikwidowało tę niedogodność. Wykorzystanie łącz telefonicznych (dial-up networking) umożliwia dotarcie do powszechnego odbiorcy, a względna taniość koniecznego zaplecza technicznego nie stanowi bariery (zresztą są one już bardzo popularne - w USA posiada je 40% wszystkich gospodarstw domowych). Ważny jest także aspekt możliwości interakcyjnych, obliczeniowych (szyfrowanie i utajnianie danych), a także potencjalna możliwość integracji z istniejącymi obecnie strukturami informatycznymi banków.
- łatwość i prostota obsługi - tworzone i stale rozwijane oprogramowanie udostępnia coraz bogatsze narzędzia pozwalające na wykonywanie skomplikowanych zadań nawet użytkownikom nie mającym dotąd żadnego doświadczenia. Element ten, z pozoru błahy, może mieć jednak bardzo istotne znaczenie w przełamywaniu barier światopoglądowych i przy wdrażaniu home bankingu do szerszego użytku - bo choć z jednej strony potencjalny klient banku może czuć się speszony „bezdusznocią” maszyny, to jednak komputerowy kasjer

może być zawsze miły, uprzejmy i gotowy służyć nam 24 godziny na dobę. Warto tu wspomnieć o sukcesie, jaki odniosły bankomaty !

- wiarygodność usługi - zwłaszcza w kontekście odporności na włamanie. Należy nie tylko uwzględnić rzeczywisty poziom zabezpieczeń, ale i wziąć poprawkę na potencjalny opór klientów obawiających się zastosowania nowej technologii. Przykładowo - 20% klientów w USA wyraża swoją zdecydowaną wątpliwość w ochronę swoich interesów w „wirtualnym banku”, a przecież poziom techniki w Stanach i jej integracja z codziennym życiem stoją na znacznie wyższym poziomie niż w Europie. Zresztą w naszym regionie nawet poważne instytucje finansowe dostają przysłowiowej „gęziej skórki” na sam dźwięk słowa „sieć”. Znamienny jest fakt, że ostatnio firma oferująca reklamę banku na stronie WWW w sieci Internet spotkała się z odmową, motywowaną ... potencjalnym niebezpieczeństwem ingerencji w system bankowy.
- Niezmiernie ważny jest również system ekonomiczno-prawny. Konieczne jest określenie warunków, na jakich przekazy elektroniczne mają moc prawną, a także zabezpieczenie prawnego zaplecza działalności sektora bankowego. Przykładem takiego prawa, które wpłynęło na globalizację usług bankowych w Stanach Zjednoczonych, była ustawa likwidująca ograniczenie działalności banków do jednego stanu. Obecnie pierwszy bank internetowy (patrz niżej) czeka na uregulowania w zakresie obsługi kart kredytowych, usługi maklerskie, obsługę rozliczeń ubezpieczeniowych i rachunków bankowych dla przedsiębiorstw. Konieczne jest także objęcie internetowych usług bankowych odpowiednim systemem ubezpieczeń - należy tu wypracować odpowiednią politykę zarówno na szczeblu Państwa, jak i strategii współpracy z poszczególnymi instytucjami ubezpieczeniowymi.

W chwili obecnej działa już pierwszy internetowy bank wirtualny. Nie posiada on fizycznych kas, jest w pełni z informatyzowany. Jego założenia i zaplecze techniczne zostało pokazane i omówione w styczniu 1996 w Palace Hotel, New York. Podstawą działania Security First Network Bank jest sieć Internet - jako medium transmisyjne pomiędzy systemem bankowym a terminalami użytkowników końcowych. Zgodnie z prognozami, bank rozwija się bardzo szybko, czemu sprzyja specyfika amerykańskiego rynku klienta (bank otwarty jest 24 godziny na dobę, realizując on-line wszelkie zlecenia klientów, co minimalizuje czasochłonność

operacji finansowych i pozwala reagować z oszołamiącą prędkością na zmieniające się warunki rynkowe). Nie bez znaczenia jest też fakt, że w Stanach sprzedano w zeszłym roku więcej komputerów niż odbiorników telewizyjnych.

Uruchomienie banku możliwe było już wcześniej jak zapewnia jego prezes, James Mohan, ale czekano z otwarciem na wyniki raportu Instytutu Badań Technicznych Stanu Georgia dotyczącego oceny wiarygodności zabezpieczeń. W nowym banku zastosowano nowatorskie rozwiązania techniczne, wielowarstwowe zabezpieczenia, których podstawę stanowi specjalny system operacyjny o nazwie Trusted Operation System. Jego podstawą było oprogramowanie opracowane w firmie Hewlett-Packard, a sam system zabezpieczeń został wykonany przez Secure Ware, czołową firmę kryptograficzną USA (firma ta od ośmiu lat zabezpiecza systemy komputerowe Departamentu Obrony USA - m.in. o jakości jej usług świadczy fakt, że jej oprogramowanie chroni trasy przelotu bombowców strategicznych). Do komunikacji z Internetem służą techniki wypracowane w firmie Netscape, zaś sam system pracuje w oparciu o stacje HP 9000 z procesorem PA-RISC.

Wzmiankowany raport wykazał, że skonstruowany system bankowy wykazał się bardzo wysoką odpornością na włamanie - eksperci komputerowi (dysponujący wiedzą i zapleczem technologicznym Massachusetts Institute of Technology) nie byli w stanie przełamać jego zabezpieczeń mimo wsparcia najnowocześniejszej techniki (warto podkreślić, że w całym świecie informatycznym, za wystarczający stopień bezpieczeństwa przyjmuje się ochronę, dzięki której koszt złamania zabezpieczenia jest porównywalny lub wyższy niż zyski z jego przejścia).

Warto podkreślić, że bank internetowy cieszy się już tak dużym zaufaniem, że jego wkłady są ubezpieczone przez Federalny Fundusz Ubezpieczenia Wkładów Bankowych USA.

Innym bankiem zorientowanym na usługę home banking jest Bank of Scotland, który rozpoczął działalność w 1985 r. Oferuje on system o nazwie HOBS, który wykorzystuje małe urządzenia o nazwie Screen Phone - są to ekrany z wbudowaną klawiaturą, podłączane do linii telefonicznej. W porównaniu z koncepcją Internetowego banku w USA cechuje je większa prostota, przenośność (choć w świecie laptopów i radiowych połączeń z Internetem ten element zaczyna powoli tracić na znaczeniu). Jako protokół transmisyjny wykorzystano videotex, ten

sam co we francuskim Minitelu. Głównym powodem, dla którego bank europejski zdecydował się na wprowadzenie takiego systemu był fakt, że analiza gospodarcza wykazała, że wprowadzenie systemu będzie tańszym i skuteczniejszym rozwiązaniem niż otwieranie nowych przedstawicielstw

Koszt korzystania z systemu HOBS:

4 GBP - abonament miesięczny

3 GBP - dzierżawa miesięczna Screen Phone

95 GBP - zakup Screen Phone

Istnieje również specjalna wersja systemu dla odrębnej kategorii użytkowników jaka są klienci instytucjonalni.

W Polsce również systemy tele i home banking wykazują dużą dynamikę, mimo niedoskonałości infrastruktury technicznej i braku uregulowań prawnych. Czynnikiem obciążającym jest też u nas zła opinia klientów (zwłaszcza stereotypy rozpowszechnione wśród klientów indywidualnych) o systemie bankowym i nieufność wobec zaawansowanej technologii. Wyżej wspomniany powody są przyczyną, dla której w Polsce usługę home banking świadczy się dla klientów instytucjonalnych.

Oprócz świadczenia zwykłych usług finansowych, sektor bankowy wykorzystuje Internet do szeroko pojętej informacji i reklamy. Reklama ta to z jednej strony wykorzystanie serwisów informacyjnych - a także „prasy” internetowej, a z drugiej prowadzenie własnych serwisów WWW. Serwis taki zapewnia klientom informację o działalności banku, podaje aktualne stopy procentowe, przeliczniki walut, pomaga w marketingu itp. Banki w Polsce nie mają jeszcze w pełni rozwiniętej struktury WWW i korzystają często z usług providerów internetowych i web-publisherów.

1.6.3.2. Inteligentne karty identyfikacyjne

Kiedy mowa o transakcjach finansowych dokonywanych za pomocą sieci - a zwłaszcza sieci globalnej - na pierwszy plan wysuwa się zawsze zagadnienie wiarygodności takiego przekazu. Jest to sprawa istotna z wielu względów. Pomijając już absolutnie oczywisty problem potencjalnego nielegalnego wykorzystania rachunku drugiej osoby i kradzieży jej pieniędzy, należy wspomnieć o barierze psychologicznej, która jest często niedoceniana w szacunkach dotyczących Internetu.

O ile bowiem nawet sprawdzona i „wiarygodna” technologia budzi w początkowym okresie stosowania szereg wątpliwości, o ileż większe opory wzbudzi na pewno globalne wprowadzenie wirtualnego handlu - do niedawna dziedziny pisarzy s-f. Warto też podkreślić, że niewielkie firmy nie są zainteresowane działalnością na niezbadanych obszarach, zwłaszcza takich, na których ryzyko niepowodzenia (związane w tym przypadku z ewentualnymi niedoskonałościami zabezpieczeń operacji finansowych) jest stosunkowo wysokie.

Ponieważ z drugiej strony, rynek związany z wirtualnym handlem wydaje się być wysoce obiecujący, kilka firm-potentatów zdecydowało się zainicjować różnorodne projekty związane z zapewnieniem takiego zaplecza technicznego, które umożliwiłoby wprowadzenie globalnego handlu opartego na przekazie informacji (w tym finansowej). Z punktu widzenia strategii tych firm wydaje się to uzasadnione: po ustanowieniu warunków do szerokiej wymiany handlowej w sporą część rynku (a więc i zysków) będą mogli zagarnąć prekursorzy takiej działalności (jako ci, którzy zdobyli już rozgłos i posiadają odpowiednie doświadczenie). Podobnie postąpiła kiedyś firma Netscape Communications, inwestując w rozwój oprogramowania WWW.

Z punktu widzenia wiarygodności przekazu informacji pożądane jest, aby istniał sposób na zabezpieczenie przesyłanych przez Internet zleceń wypłat przed różnym rodzajem obcej ingerencji. Do najważniejszych należą:

- zmiana informacji w medium transmisyjnym (A wysłała do banku zlecenie przelania \$100 na konto B, ale na skutek działania C do banku dociera zlecenie przekazu \$200).
- fałszowanie zleceń (A nie wysłała zlecenia do banku, ale zlecenie takie wysłała B, podpisując je jako A).
- podsłuch informacji (nie ingerujemy czynnie w treść, zadowolając się biernym podsłuchem - zagadnienie to jest wbrew pozorom nie mniej istotne niż dwa uprzednio wspomniane i zahacza o problemy szeroko pojętej tajności informacji i prywatności danych)

Z punktu widzenia sieci i jej użytkowników istotne jest, aby rozwiązanie tych problemów nie wymagało daleko idących modernizacji istniejących urządzeń, było niezależne od płaszczyzny sprzętowej i systemowej i akceptowane przez „internetową większość”. Próba znalezienia takiego rozwiązania są podjęte ostatnio

przez Microsoft (przy współpracy innych potentatów - jak choćby Hewlett-Packard) prace nad rozwojem tzw. Smart Cards (inteligentne karty identyfikacyjne). Projekt ten powstaje w połączeniu z rozwojem CryptoAPI 2.0, narzędzia służącego do uwierzytelniania (certyfikacji) aplikacji. Pakiet CryptoAPI ma być przeznaczony do ogólnego i szerokiego stosowania (propozycja wytworzenia standardu), dlatego jest możliwe włączenie go do oprogramowania tworzonego nie tylko na bazie C/C++, ale i Visual Basic i Javy.

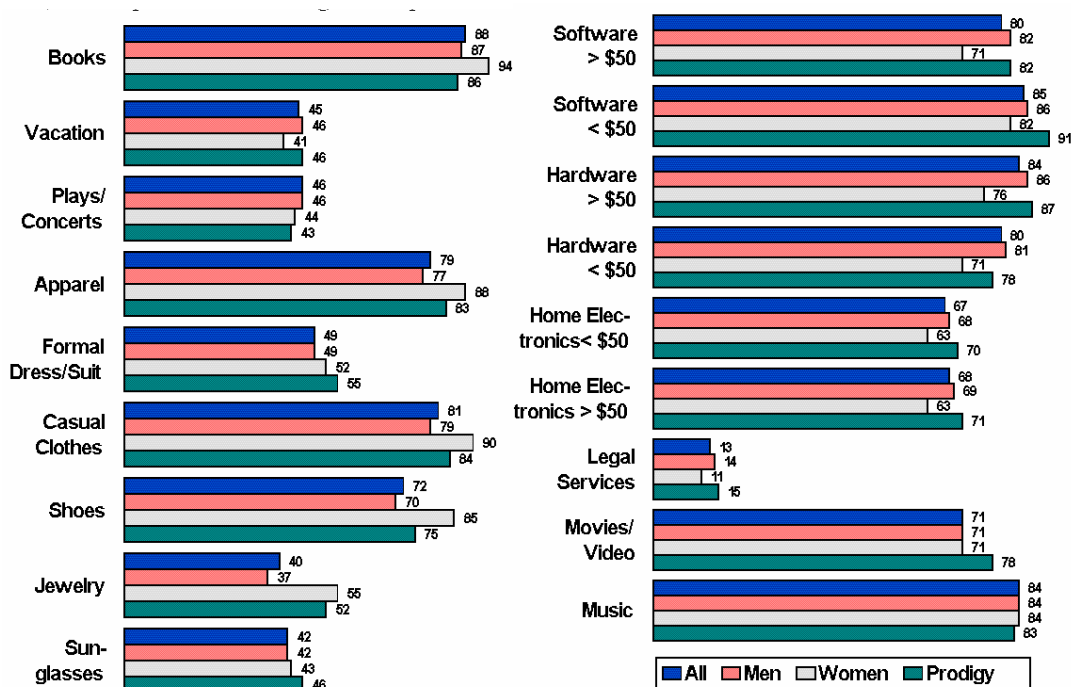
1.6.4. Handel internetowy

Powstanie Internetu i jego rozwój otworzyło nowe pole działalności, dotąd nie mające rynku zbytu. Istnieje wiele firm, których działalność i wytwórczość opiera się wyłącznie na Internecie. Są to przede wszystkim firmy oferujące sprzęt elektroniczny wykorzystywany do pracy z Siecią. Duży zysk osiągnęły firmy produkujące i handlujące modemami - jak choćby Zyxel, czy Zoom. Osobną działkę stanowią firmy software'ową, począwszy od firm dostarczających oprogramowanie podstawowe - serwery WWW i browsery (ten rynek wydaje się jednak być praktycznie bardzo trudny dla nowych inicjatyw - zwłaszcza po pojawieniu się bezpłatnych serwerów dla wielu platform - takich jak Apache, często wykorzystywany nawet w zastosowaniach komercyjnych), poprzez firmy promujące wszelkiego rodzaju plug-iny, a kończąc na firmach kryptograficznych. Warto także wspomnieć o szerokim spektrum firm zainteresowanych bazami danych - np. Hughes Software oferująca kilka wersji motoru relacyjnej bazy SQL o możliwościach co prawda daleko uboższych niż Oracle czy Informix, ale za to po atrakcyjnie niskich cenach - rzędu 150USD ! Internetem zainteresowani są także potentaci - Borland, Oracle, Informix, których każdy produkt jest już zorientowany na WWW. Wydaje się, że otwiera się przyszły rynek aplikacji pisanych pod kątem wykorzystania w sieci Internet.

Powstawanie sklepów internetowych jest przejawem ogólnej tendencji do komercjalizacji Internetu. Komercjalizacja ta jest samonapędzającym się niejako procesem, ponieważ przyciąga ona rezerwy kapitału, których siła przekłada się na opracowywanie i wprowadzanie nowoczesnych rozwiązań technicznych, co z kolei sprawia, że Internet staje się jeszcze bardziej atrakcyjnym terenem penetracji dla firm poszukujących nowych terenów działalności.

Owe wirtualne sklepy to połączenie virtual reality z home banking. Ich idea jest następująca: klient powinien mieć możliwość obejrzenia towaru, znalezienia na jego temat pełnej informacji (specyfikacje techniczne, osiągi, wymiary itp.) nie ruszając się z domu, za pomocą swojego terminala. Kiedy już zdecyduje się na zakup, przekazuje zlecenie do pewnego systemu finansowego (bank lub system obsługi kart kredytowych) za pomocą tego samego terminala, finalizując transakcje.

Poniżej przedstawiamy wyniki badań ekonomicznych, dotyczących preferencji w zakupach pewnych towarów z użyciem sieci Internet (wyniki te udostępniła Szkoła Zarządzania i Businessu Michigan USA, <http://www.cc.gatech.edu>):



Rysunek 24: Preferencje klientów sklepów internetowych

Podstawowe problemy związane z usługą handlu internetowego w Polsce to:

- konieczność zapewnienia informacji dla klienta - najlepsza metoda - przekaz wizyjny jest stosunkowo obciążający dla sieci.
- cena usługi - w krajach, gdzie łączność telefoniczna jest stosunkowo droga, usługi takie będą się rozwijały wolniej niż np. w stanach zjednoczonych, gdzie strategia operatora, polega nie na czerpaniu dochodów z połączenia, ale z usług realizowanych dzięki temu połączeniu
- zapewnienie wiarygodności przekazu poufnych danych

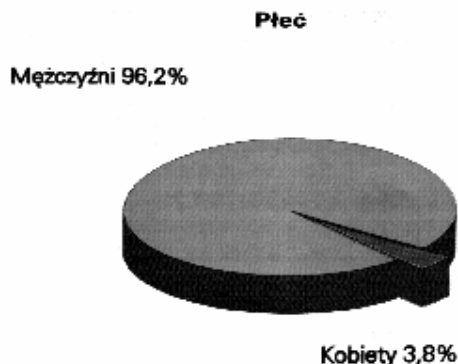
- udostępnienie sposobu na przekazywanie pieniędzy (numer karty kredytowej ?)
- nieufność klientów wobec nowej formy zakupów

Generalnie w chwili obecnej w Polsce obserwuje się tworzenie sklepów w połowicznej formie - na razie towary są jedynie prezentowane, podobnie jak w teleshopach (TV POLSAT), zwłaszcza, że Internet ma nad tele marketingiem znaczną przewagę: towar jest prezentowany na stałe, a nie w godzinach emisji, a ponadto koszty takiej prezentacji są znacznie niższe niż reklama w TV.

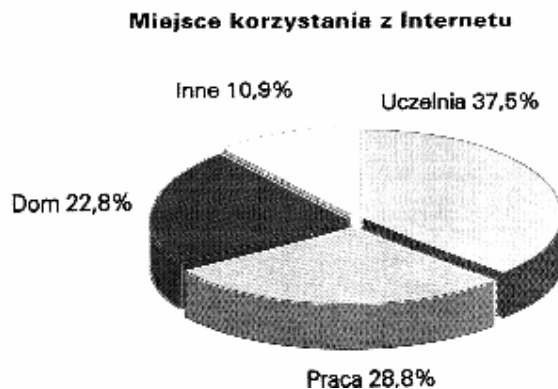
1.6.5. WWW w statystykach

1.6.5.1. Internet w Polsce¹⁵

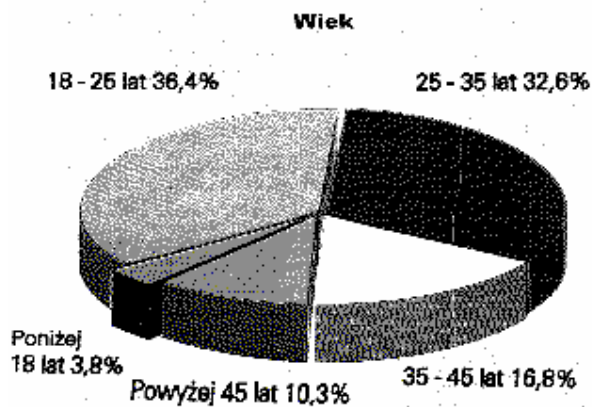
Rozwój Internetu w Polsce jest zjawiskiem bardzo gwałtownym. Sprzyja mu towarzyszący rozwój gospodarczy kraju, stopniowe podwyższanie się zamożności mieszkańców i wzrost liczby sprzedawanych komputerów. Główna bariera to wysoki koszt połączeń telefonicznych (drogie jest nawet połączenie z lokalnym dostawcą Internetu) - co wyraźnie widać po charakterystyce miejsc, z których używamy Internetu - ponad 65% to komputery, z których połączenie jest opłacane przez „kogoś innego” (uczelnia, zakład pracy). Przekroje demograficzne pozwalają naszkicować charakterystykę polskiego użytkownika Internetu - jest to zwykle student (lub uczeń liceum), młody pracownik uczelni, lub osoba pracująca w businessie (często świeżo zatrudniony absolwent szkoły wyższej). Jest to zjawisko interesujące z socjologicznego punktu widzenia - tworzenie się generacji ludzi wychowanych na nieskrępowanym dostępie do informacji (nota bene wskazuje się zarówno cechy pozytywne, jak i negatywne tego zjawiska)



Rysunek 25: Statystyka polskiego Internetu - zróżnicowanie względem płci



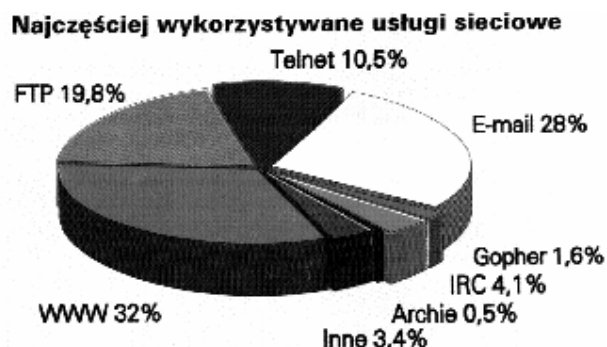
Rysunek 26: Statystyka polskiego Internetu - zróżnicowanie względem lokalizacji komputera



Rysunek 27: Statystyka polskiego Internetu - zróżnicowanie względem wieku użytkowników

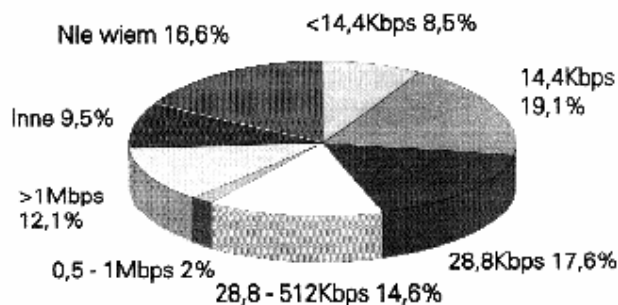
Jednocześnie, w polskim Internecie, obserwuje się wyraźną tendencję do wzrostu procentowego udziału usługi WWW w ogólnej liczbie użytkowanych usług (rok temu liczba ta była o kilka procent niższa!). Jest to nurt zauważalny na całym świecie, tym bardziej, że WWW integruje w sobie praktycznie wszystkie wcześniejsze usługi.

¹⁵ Dane i wykresy pochodzą z wyników przeglądu statystycznego opublikowanego w kilku numerach Internetu na początku 1997 roku.



Rysunek 28: Statystyka polskiego Internetu - usługi sieciowe

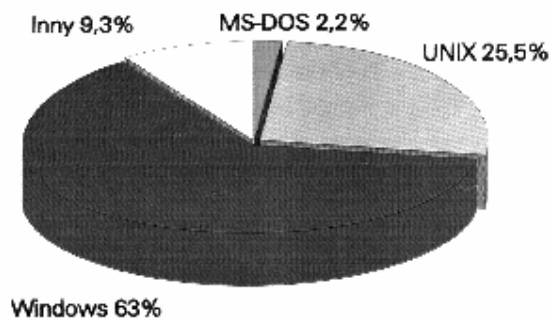
Charakterystyczną cechą polskiego „domowego” użytkownika Internetu jest wolniejszy niż w Europie zachodniej modem. Wiąże się to zarówno z oszczędnością (długie użytkowanie modemu bez zmiany na lepszy), jak i z technicznymi niedomaganiem polskich łączy telekomunikacyjnych (brak central cyfrowych, wysoki współczynnik zaszumienia linii). Widać to wyraźnie na wykresie - po odrzuceniu szybkich linii (uczelnie) okazuje się, że dominują modemy 14.4 kbps (w krajach Unii Europejskiej - 28.8kbps).



Rysunek 29: Statystyka polskiego Internetu - przepustowość łączy do Internetu

W polskim Internecie na rynku systemów operacyjnych króluje system Windows. Wiąże się to zarówno z rozpowszechnieniem komputerów osobistych w domach i biurach, jak i łatwym dostępem do oprogramowania (duża ilość nielicencjonowanych kopii). System Unix to w przeważającej większości efekt używania systemów unixowych (Linux, HP, Sun) na uczelniach. Jak widać inne platformy systemowe (np. OS/2) praktycznie nie liczą się na rynku przeglądarek (ok. 10%) - jest to zresztą tendencja ogólnoświatowa.

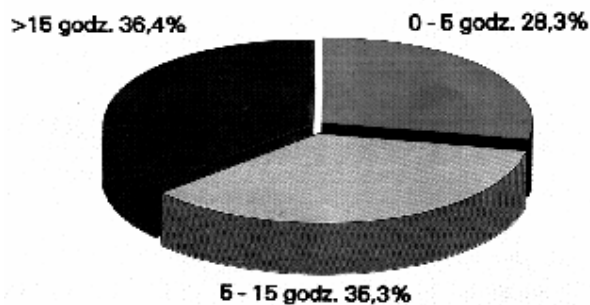
Systemy operacyjne używane do połączenia się z Internetem



Rysunek 30: Statystyka polskiego Internetu - systemy operacyjne użytkowników

Najbardziej frapujący jest diagram prezentujący ilość czasu spędzaną „w sieci”. Okazuje się, że (biorąc pod uwagę, że z Internetu korzysta się głównie w dni robocze¹⁶) trzecia część użytkowników korzysta z Internetu ponad 3 godziny dziennie !

Czas spędzony na korzystaniu z Internetu (tygodniowo)



Rysunek 31: Statystyka polskiego Internetu - czas pracy w sieci

1.6.5.2. Internet na świecie¹⁷

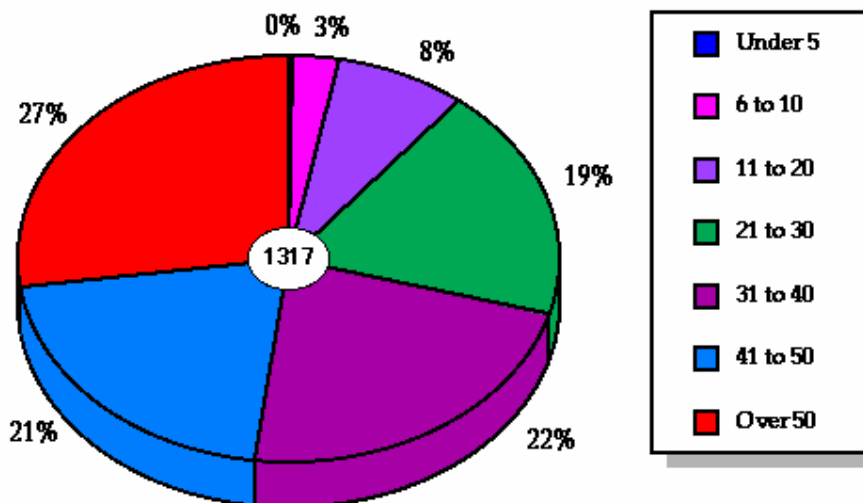
Okazuje się, że Polska, pomimo pewnego zacofania technologicznego, nie odbiega specjalnie od najbardziej „zinternetowanych” krajów świata. Różnice są jedynie ilościowe, co jest zrozumiałe, zważywszy na to, że np. w Stanach szeroki dostęp do sieci - nie tylko dla ośrodków uniwersyteckich - rozpoczął się kilkanaście lat temu.

¹⁶ Z uwagi na preferowane miejsca, skąd następuje łączność: praca i uczelnia.

¹⁷ Informacje dotyczą badań przeprowadzonych w USA - najbardziej „zinternetowanym” kraju świata. Dane opublikowała Michigan Business School, USA

Poniżej prezentujemy wykres obrazujący ilość godzin spędzonych miesięcznie w sieci Internet w środowiskach uniwersyteckich. Okazuje się, że ponad połowa respondowanych spędzała średnio ponad godzinę dziennie przeszukując zasoby sieciowe !

Hours in Front of Computer

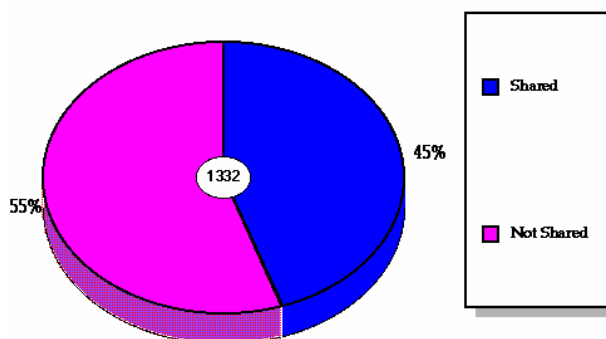


Rysunek 32: Internet na świecie: liczba godzin spędzanych na przeszukiwaniu sieci w miesiącu

Warto też wskazać, że prawie 50% użytkowników współdzieli komputer sieciowy z innymi użytkownikami. W Polsce stosunek ten jest oczywiście znacznie lepszy z uwagi na fakt, że łączymy się z Internetem głównie z pracy lub uczelni (na świecie rozwojowi Internetu sprzyja polityka operatorów sieci telefonicznych, zmierzająca do zmniejszenia opłat za połączenia lokalne¹⁸).

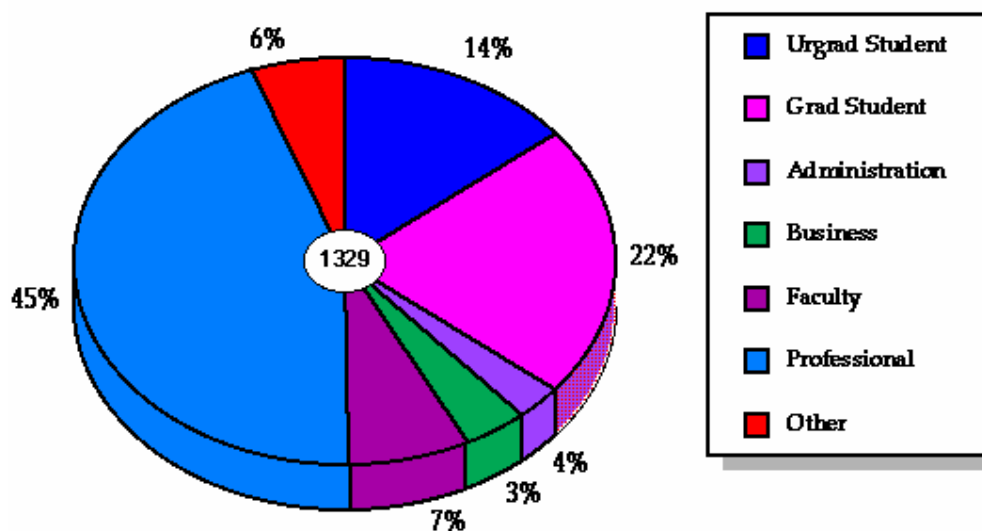
¹⁸ Bywa nawet, że połączenia lokalne bywają bezpłatne (tzw. usługa podstawowa) - dochód czerpany jest za połączenia pow. pewnej odległości i za usługi realizowane z wykorzystaniem telefonu (w Polsce w tym systemie realizuje się np. połączenia Audio-Tele, w których zyskiem z opłaty dzieli się TP S.A. i TVP).

Machine Use



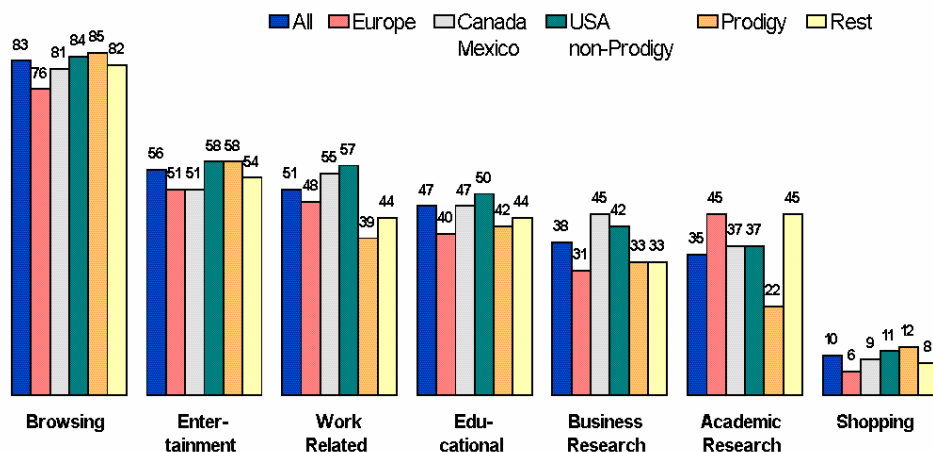
Rysunek 33: Internet na świecie - użytkowanie komputerów sieciowych

Occupation

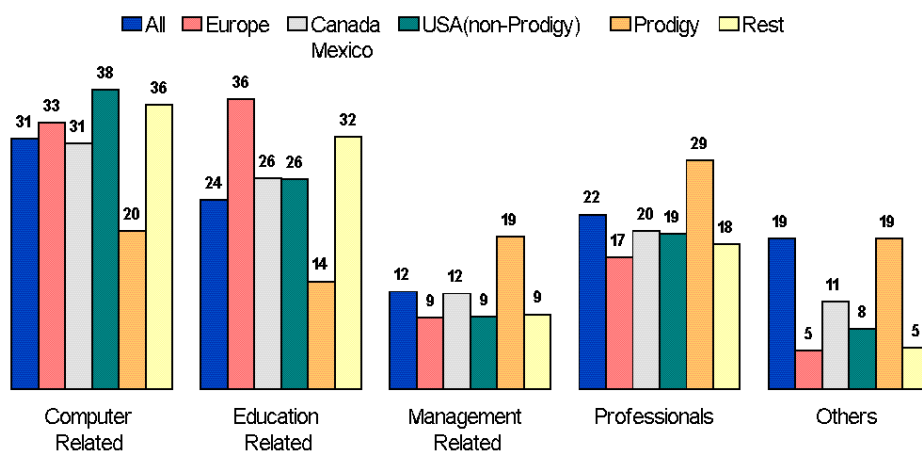


Rysunek 34: Internet na świecie - profil użytkownika

Ciekawy jest również wykres obrazujący preferencje użytkowników usługi WWW. Zwraca uwagę cecha różnicująca Europę na tle innych regionów - stosunkowo wysoki współczynnik przeszukiwań sieci w celu uzyskania danych dla prac naukowych i jednocześnie mniejsze zainteresowanie internetowymi zakupami. Europejska tendencja do „edukacyjnego” użytkowania sieci wiąże się z odpowiednio większą obecnością przedstawicieli uczelni wśród ogółu „internautów” (patrz rys.36)

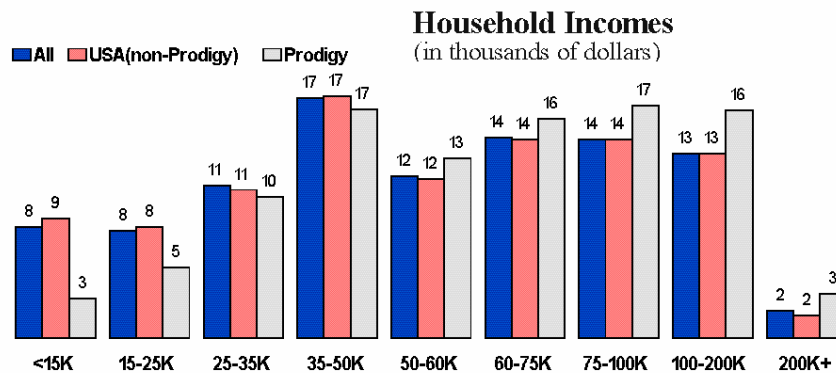


Rysunek 35: Internet na świecie - profil użytkownika (2)

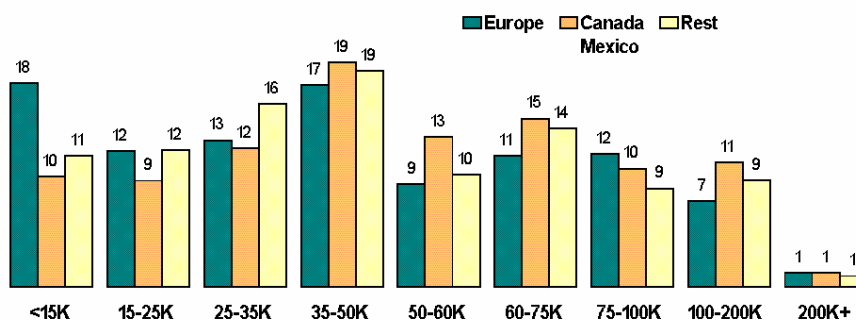


Rysunek 36: Internet na świecie - profil użytkownika WWW

Na zakończenie krótkiego statystycznego przeglądu Internetu prezentujemy wykres przedstawiający strukturę użytkowników sieci w zależności od uzyskiwanych dochodów. Zwracają uwagę różnice rezultatów uzyskanych w USA i w pozostałych regionach - np. w Europie.



Rysunek 37: Internet na świecie - przychody użytkowników (USA)



Rysunek 38: Internet na świecie - przychody użytkowników (poza USA)

1.7. Przegląd zastosowań

1.7.1. Alta Vista - największy serwis informacyjny

1.7.1.1. Ilość przetwarzanych danych.

System **AltaVista** umożliwia uzyskanie adresów stron zawierających poszukiwane przez nas informacje. Wyszukiwanie odbywa się na podstawie podanych haseł. System operuje na 15 bilionach zaindeksowanych słów udostępniając około 30 milionów stron znajdujących się na 275.600 serwerach oraz 4 miliony artykułów z 14.000 list nowości z Usenetu. Tygodniowo strona główna systemu jest odczytywana 23 miliony razy. System ten jest aktualnie dostępny na platformach Digital UNIX dla procesorów Alpha. „Alta Vista” oficjalnie działa od 15 grudnia 1995.



Rysunek 39: Ekran Alta Visty w okienku Netscape Navigatora 3.0 Gold

1.7.1.2. Wydajność a udostępniane funkcje.

Na większość zapytań użytkowników system odpowiada w ułamku sekundy. Jest to związane z wydzieleniem osobnego komputera tylko do tego celu. Przeszukiwaniem Internetu zajmuje się „Scooter data collector”, który jest obecnie

najszybszym systemem przeszukującym sieć WWW. Przeszukuje około 3 miliony stron WWW dziennie. W czasie swoich poszukiwań nie powoduje przełamania zabezpieczeń systemowych typu firewall jak również nie przeszukuje żadnych stron wyłączonych z publicznego dostępu przez webmastera. Posiada wbudowany algorytm, który zapewnia minimalne obciążanie przeszukiwanego komputera nie powodując znaczącego spadku jego wydajności. Tworzeniem indeksów zajmuje się „Net indexer 2”. Potrafi on utworzyć w ciągu godziny indeksy do 1GB znalezionych przez Scootera słów na stronach WWW. Nie dubluje istniejących indeksów, a jednocześnie prowadzi statystyki dzięki, którym na początku listy tematycznej znajdują się odnośniki do najczęściej odczytywanych i najbardziej użytecznych danych.

1.7.1.3. Konfiguracja sprzętowa.

Alta Vista to pięć serwerów Digital opartych na 64-bitowym procesorze Alpha. Pierwszy z nich nazywany jest „Alta Vista” to ten, z którym normalnie mamy do czynienia. Jego zadaniem jest odbierać pytania i polecenia użytkowników. Serwer ten ma 4GB dysk twardy i 256MB pamięci operacyjnej. Drugim serwerem jest „Web Indexer”, który zajmuje się konstruowaniem indeksu słów występujących na stronach WWW. Posiada 210GB macierz dyskową (indeks całej światowej sieci Web zajmuje dziś 30 GB), 6GB pamięci operacyjnej i 10 procesorów. Kolejny komputer nazwano „Scooter”. Ma on 30GB macierz dyskową, 1GB pamięci. Pełni rolę „pająka”, który podąża po odsyłaczach stron WWW, poszukując kolejnych stron i „podając” je komputerowi budującemu indeks. Maszyna indeksująca nowości ma tylko 13GB dysk i zaledwie 196MB pamięci. Ostatni komputer będący serwerem nowości ma 24GB macierz dyskową i 160MB pamięci. Przechowuje na dyskach pełny tekst wszystkich zaindeksowanych artykułów UseNet.

1.7.1.4. Bezpieczeństwo

Kontrola dostępu do danych jest realizowana za pomocą firmowego firewalla (ściany ogniowej) - „Alta Vista Firewall” (posiadającego certyfikat NCSA). Firewall daje bardzo wysoki poziom bezpieczeństwa, gdyż uniemożliwia bezpośrednie połączenie sieci prywatnej z siecią Internet. Wszystkie dane muszą najpierw pozytywnie przejść przez filtr, którym jest właśnie firewall. Udostępnia on możliwość kontroli dostępu do najpopularniejszych usług takich jak: poczta

elektroniczna (SMTP), transfer plików (FTP), telnet, nowości Usenetu (NNTP), World Wide Web czy finger. Dodatkowo zapewnia również możliwość obsługi niestandardowych usług poprzez wbudowanie mechanizmów protokołu TCP. Odpowiednie skonfigurowanie oprogramowania typu firewall pozwala na realizację polityki bezpieczeństwa w firmie.

Po zainstalowaniu oprogramowania „Alta Vista firewall” wszystkie usługi są początkowo niedostępne. Administrator świadomie podejmuje decyzje związane z udostępnianiem usług i ich kontrolą. Kontrola oparta jest na możliwości automatycznego generowania (zdefiniowanych przez administratora) raportów lub też na tworzeniu kompletnego logu pracy firewalla. W logu tym zawarte będą informacje o wszystkich decyzjach podjętych przez firewall, próbach zewnętrznej ingerencji wraz z możliwością dokładnego ustalenia lokalizacji z której ta próba została podjęta. Firewall śledzi każde połączenie: zapamiętuje godziny, daty rozpoczęcia i zakończenia, ustala nazwę komputera z którego zostało nawiązane połączenie oraz identyfikator użytkownika, ilość danych przesyłanych w obu kierunkach, próby nieautoryzowanej ingerencji w system plików. Dla lepszej kontroli można również ustalić tworzenie raportów dziennych, tygodniowych lub miesięcznych dla każdej z usług oddzielnie. Raporty te mogą dotyczyć aktualnego stanu bezpieczeństwa, konfiguracji sieci, konfiguracji mechanizmów autoryzacji użytkowników (sposób sprawdzania tożsamości użytkownika zwykle oparty na identyfikatorach i hasłach), czy też konfiguracji filtrowania ruchu pakietów dla każdej z usług.

„Alta Vista firewall” posiada wbudowany system kontroli włamań działający w czasie rzeczywistym. Oznacza to, że system potrafi zidentyfikować stan związany z zagrożeniem bezpieczeństwa w chwili gdy ktoś próbuje „włamać się” a nie już po przełamaniu zabezpieczeń. Jednocześnie w takim przypadku system potrafi poinformować ustalone osoby za pomocą poczty czy pagera o próbie włamania do systemu. Informacja o włamaniu to tylko połowa sukcesu. Firewall potrafi również automatycznie podjąć walkę z włamywaczem. Najczęściej spotykanymi metodami są: podniesienie poziomu dostępu do danych dla komputera z którego nastąpiło połączenie, lub zamknięcie przepływu danych związanych z daną usługą bądź też odcięcie całkowite połączeń z zewnątrz. Firewall potrafi wykryć próby zewnętrznych ingerencji (ze strony Internetu) a także z wewnątrz sieci prywatnej odrzucając

wybrane pakiety danych (lub nie przepuszczając je dalej) z jednoczesnym zapisem informacji o incydencie do logu i poinformowaniem ustalonych osób o zaistniałej sytuacji.

Jednym z najważniejszych działań jest kontrola autoryzacji użytkowników. Podstawowym wymogiem bezpieczeństwa jest uniemożliwienie dostępu użytkownikom nieupoważnionym tzn. nie znającym odpowiednich haseł. Identyfikator (login) i hasło użytkownika nie mogą być przesyłane przez sieć w postaci zwykłego ciągu znaków, gdyż stanowiło by to potencjalne źródło niebezpieczeństwa. Firewall posiada wbudowane mechanizmy kontroli dostępu szczególnie dla takich usług jak telnet czy ftp. Jednak to nie wszystko. Określa się również poziomy dostępu użytkowników do wybranych danych. Im ważniejsze i poufniejsze dane tym mniejszy krąg osób uprawnionych do jego odczytu czy modyfikacji a więc i wyższe wymagania autoryzacyjne (posiadanie dodatkowych uprawnień, znajomość dodatkowych haseł itd.).

Oprócz ataków bezpośrednich ze strony użytkowników „Alta Vista firewall” potrafi również wykryć większość znanych wirusów komputerowych. „Alta Vista firewall” jest dostępny w wersjach na różne platformy systemowe: Windows NT, BSD/OS i Digital UNIX. Można go w łatwy sposób skalować tak, aby pracował dla małej firmy jak również i wielkiej korporacji.

Liczba stanowisk	Cena
do 50	3 995 \$
do 200	7 995 \$
nieograniczona	14 395 \$

Tabela 11: AltaVista Firewall - warianty oferty cenowej

1.7.1.5. Mechanizmy optymalizacji transmisji

„Alta Vista Tunnel” to oprogramowanie komunikacyjne zapewniające wymianę informacji pomiędzy komputerami znajdującymi się w sieci prywatnej i poza nią. Zapewnia ono możliwości efektywnego kodowania danych, autoryzacji użytkowników i kontroli integralności danych. Współpracuje z każdym oprogramowaniem typu firewall pozwalając na tworzenie wirtualnych sieci

prywatnych (VPN). Oznacza to, że użytkownicy sieci lokalnych (LAN) mogą swobodnie łączyć się i bezpiecznie wymieniać informacje z użytkownikami sieci rozległych (WAN) bez konieczności budowy dodatkowych połączeń fizycznych. „Alta Vista Tunnel” do autoryzacji wykorzystuje technologię wymiany 512 bitowego klucza publicznego zgodnie z algorytmem RSA. Pozwala to na „sprawdzenie drugiej strony” przed rozpoczęciem sesji wymiany danych.

Po pomyślnej autoryzacji oprogramowanie „Alta Vista Tunnel” rozpoczyna przesyłanie danych w postaci pakietów zakodowanych przy użyciu tajnego 128 bitowego klucza RC4 co powoduje, że dane są nierozpoznawalne dla innych. Tylko adresat wiadomości potrafi przetworzyć dane z powrotem do czytelnej postaci. Dla zwiększenia bezpieczeństwa „Alta Vista Tunnel” co 30 minut wymienia poprzez tunel klucze kodujące. Wbudowany mechanizm MD5 pozwala na sprawdzanie poprawności przesyłanych danych, wykrywanie błędów powstałych w wyniku transmisji czy celowego działania z zewnątrz. Zastosowanie oprogramowania „Alta Vista Tunnel” nie powoduje konieczności modyfikacji aplikacji, gdyż przetwarza ono dane przesyłane do sieci przez te aplikacje. Oprogramowanie to pozwala na bezpieczne połączenie się z serwerami firmowymi z dowolnego miejsca sieci bez konieczności dzierżawienia linii a co za tym idzie jest bardzo tanie. Oprogramowanie „Alta Vista Tunnel” działa na różnych platformach systemowych: Windows 95, Windows NT Intel, Digital UNIX, BSD/OS i FreeBSD.

Wersja oprogramowania	Liczba równoległych tuneli	Cena
Personal Edition	1	99 \$
Workgroup Edition (UNIX, BSD/OS)	50	995 \$
Workgroup L Edition (UNIX)	200	1 995 \$
Workgroup L Edition (BSD/OS)	100	1 995 \$
Workgroup XL Edition (UNIX)	512	2 495 \$
Workgroup XL Edition (BSD/OS)	N/A	2 495 \$

Tabela 12: AltaVista Tunnel - oferty cenowe

1.7.2. Deja news - największy system obsługi news

1.7.2.1. Ilość przetwarzanych danych.

Deja-news jest największą bazą danych nowości na świecie, która powstała w 1979 roku. Umożliwia dostęp do ponad 120GB danych Usenetu (publiczny serwis informacyjny gromadzący dane z 15.000 tematycznych list dyskusyjnych, szacunkowa ilość odbiorców to około 22 miliony ludzi). W bazie danych znajduje się około 80 milionów artykułów, aktualnie zapewnia się możliwość odwołań do informacji dostępnych od 1995 roku (planowana jest rozbudowa tak, aby udostępniać jednocześnie wszystkie dane od 1979 roku).

1.7.2.2. Wydajność a udostępniane funkcje.

System zapewnia obsługę **każdego** zapytania w ciągu maksymalnie 2 sekund. Uzyskuje się to poprzez zastosowanie szybkich algorytmów wyszukiwawczych (a nie superkomputerów) wspomaganych wielopoziomą strukturą indeksów (indeks wskazuje na lokalizację danych w obrębie pamięci masowej co umożliwia znalezienie żądanych danych bez konieczności odczytu innych wiadomości). Dane są zorganizowane hierarchicznie dzięki czemu hasła będące uszczegółowieniem wybranej dziedziny znajdują się w dokładnie ustalonym miejscu bazy danych.

System prowadzi również statystyki dotyczące częstościostępów do informacji zawartych w plikach o podanym adresie. Statystyka ta pozwala na weryfikację aktualności danych i jest wykorzystywana również przy automatycznej selekcji ważności informacji (dane odczytywane częściej są dla użytkownika ważniejsze i dlatego też będą przesyłane do użytkownika wcześniej niż pozostałe - w pierwszej kolejności). System umożliwia również przesyłanie poczty do wybranej przez użytkownika listy nowości wraz z możliwością tworzenia zakładek, sygnatur i list adresów najczęściej używanych.

1.7.2.3. Konfiguracja sprzętowa.

Twórcy systemu podkreślają, że nie stosują w swoim systemie kosztownych superkomputerów, a mimo to uzyskują wysoką wydajność systemu. Szczegółowe informacje dotyczące stosowanego sprzętu nie są udostępniane, jednak znane są inne dane mogące pomóc w jego szacunkowej ocenie.

- W celu przyspieszenia jednoczesnej obsługi wielu użytkowników dane przechowywane są w kilku kopiach jednocześnie (a więc stosowane są macierze dyskowe lub dyski w kilku osobnych komputerach).
- Indeksy są uaktualniane kilka razy dziennie. Biorąc pod uwagę wielkość indeksów oraz częstość ich aktualizacji można stwierdzić czynność tę wykonuje wydzielony komputer o dość dużej mocy obliczeniowej.

1.7.3. Yahoo - najbardziej znany serwis informacyjny

Adres yahoo.com jest znany praktycznie każdemu użytkownikowi sieci Internet. Przez długi czas był to adres największego serwisu informacyjnego WWW (obecnie ilością danych przewyższa go tylko Alta Vista), do dziś jest pierwszy na liście najbardziej popularnych (klasyfikacja oparta o liczbę zapytań zadawanych w ciągu miesiąca). Interfejs Yahoo jest bardzo prosty w obsłudze - na pierwszej stronie znajduje się tylko jedno pole wprowadzania danych (oczywiście możliwe jest także wydawanie skomplikowanych zapytań, ale są do tego przeznaczone specjalne formularze, dostępne po wybraniu odpowiedniego odnośnika). Serwis ten jako pierwszy rozwiązał problem klasyfikacji odnośników, udostępniając kategorie znaczeniowe.

Wpis nowych danych jest dostępny dla wszystkich użytkowników Internetu, ale odbywa się w trybie „off-line”: chcąc umieścić informację o nowej stronie domowej (ang. homepage, termin często stosowany w odniesieniu do udostępnianych w sieci plików HTML) wysyłamy krótki jej opis do administratorów serwisu. Do tego celu posługujemy się specjalnie przygotowanym formularzem¹⁹. Aktualizacja (lub informacja o jej odrzuceniu) następuje zwykle w czasie dwóch - trzech tygodni (!).

¹⁹ Dane strony te nie są strukturalizowane (jest to prosty słowny opis). W przeciwieństwie do naszego systemu BIS nie jest możliwa zmiana ich typu. Dalsze informacje na temat różnic w dozwolonych formatach danych w obu systemach zawarte są w części poświęconej analizie.



Rysunek 40: Interfejs Yahoo - pierwsza strona (<http://yahoo.com>)

1.7.4. TOW - polski system obrotu papierami wartościowymi

1.7.4.1. Opis ogólny

Rynek finansowych usług internetowych istnieje również w Polsce, czego przykładem może być zrealizowana Tabela Ofert Wierzytelności

(www.itnet.com.pl.), która pozwala autoryzowanym klientom na dostęp do informacji na temat wiarytelności, ich cen i handlu nimi. Z uwagi na brak mechanizmów przesyłania informacji o przekazach pieniężnych TOW nie ma w tej chwili możliwości prowadzenia zdalnych zakupów.

Netscape: Tabela Ofert Wierzytelności

File Edit View Go Bookmarks Options Directory Window Help

Location:

What's New? What's Cool? Destinations Net Search People Software

Szukaj ofert sprzedaży wiarytelności

Podaj dane dotyczące oferty, która chciałbyś znaleźć - im więcej parametrów zostanie uwzględnionych tym przeszukiwanie będzie dokładniejsze.

Kod oferty	<input type="text"/>
Firma/Nazwisko i imie dłużnika	<input type="text"/>
Data wprowadzenia oferty: <small>w formacie dd/mm/yyyy</small>	po <input type="text"/> przed <input type="text"/>
Branża wg EKD	<input type="text"/>
Województwo	<input type="text" value="WSZYSTKIE"/> bialskopodlaskie bialostockie bielskie
Forma prawna dłużnika	<input type="text" value="WSZYSTKIE"/> osoba fizyczna spolka cywilna spolka z o.o.
Szukaj wiarytelności objętej postępowaniem układowym lub ugodowym	<input type="radio"/> Tak <input type="radio"/> Nie <input checked="" type="radio"/> Bez znaczenia
Ilość wyświetlanych rekordów	<input type="text" value="WSZYSTKIE"/>

Rysunek 41: Tabela Ofert Wierzytelności - interfejs w interpretacji przeglądarki Netscape 3.0 Gold

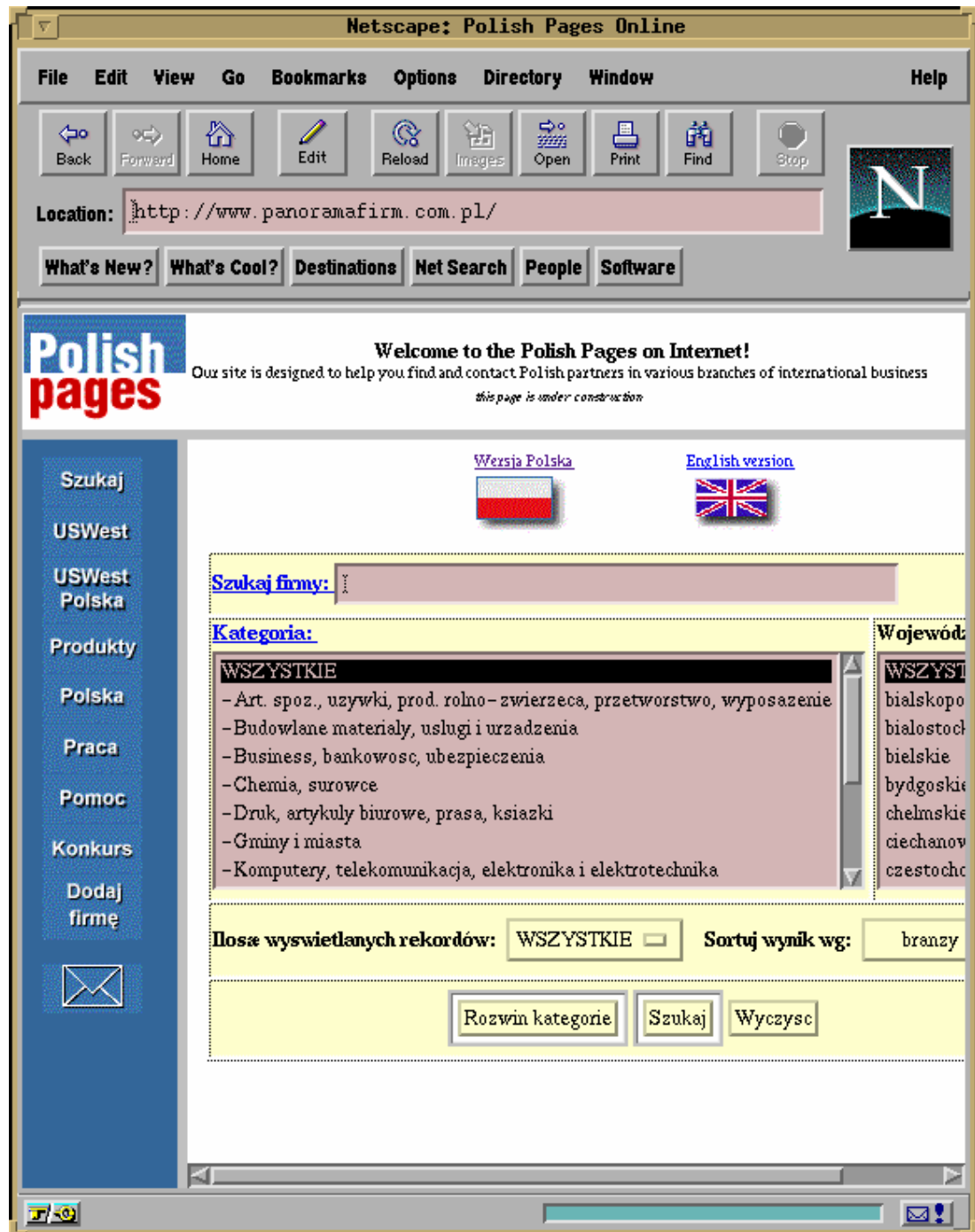
1.7.4.2. Konfiguracja sprzętowa

System pracuje na dedykowanym komputerze klasy Pentium (133MHz, 64MB RAM), który zapewnia zarówno obsługę serwera WWW, jak i bazy danych (mSQL Hughes Software). Dostęp do sieci Internet odbywał się za pomocą zwykłego łącza stałego dzierżawionego od Telekomunikacji Polskiej S.A., obecnie - stosowaną technologią jest Frame Relay.

1.7.5. Panorama Firm - polski system „yellow pages” (USWest Polska)

1.7.5.1. Opis ogólny

Nazwą Yellow Pages przyjęto określać systemy oferujące informację na temat firm, ich działalności, lokalizacji itp. Z punktu widzenia informatyka, systemy te to najczęściej duże bazy danych typu „read-only” (użytkownicy nie mają praw zapisu do nich). Przykładem takiego systemu jest działający od niedawna (początek 1997 roku) w Polsce serwis dostępny pod adresami www.panoramafirm.com.pl i www.uswest.com.pl. Pracuje on w oparciu o serwer Apache, moduł interfejsu PHP/FI bazę daną mSQL (Hughes Inc.).



Rysunek 42: Panorama Firm w Internecie

Na przykładzie powyższego rysunku widać wyraźnie problemy, jakie pojawiają się na skutek różnic pomiędzy MS Internet Explorerem a Netscape Navigatorem. Podczas prac nad internetową Panoramą Firm USWEST zażyczył

sobie²⁰ optymalizacji interfejsu pod kątem Internet Explorera (w większości biur, a więc u potencjalnych klientów dominuje system Windows). Z tego powodu pełne okno Netscape nie obejmuje całego formularza, konieczne jest niewygodne przesuwanie się za pomocą paska przewijania lub dostosowanie rozmiarów okna aplikacji.

Interfejs Panoramy Firm stworzono w oparciu o ramki (frames), dzielące ekran na trzy pola. Ramki górna i boczna mają ustawiony atrybut NOScroll, więc nie posiadają pasków przewijania. Taka kompozycja jest dość często stosowana, ponieważ niezależnie od wyświetlanej w głównym oknie strony, użytkownik zawsze ma dostęp do klawiszy menu (po lewej) i wizytówki serwisu (często miejsce to jest wykorzystywane na dodatkowe, płatne reklamy).

1.7.5.2. Konfiguracja sprzętowa

System pracuje na bazie komputera COMPAQ Pentium Pro 200 MHz, 128 RAM, 1.2 GB HDD. Jako system operacyjny zastosowano Linux w dystrybucji Slackware.

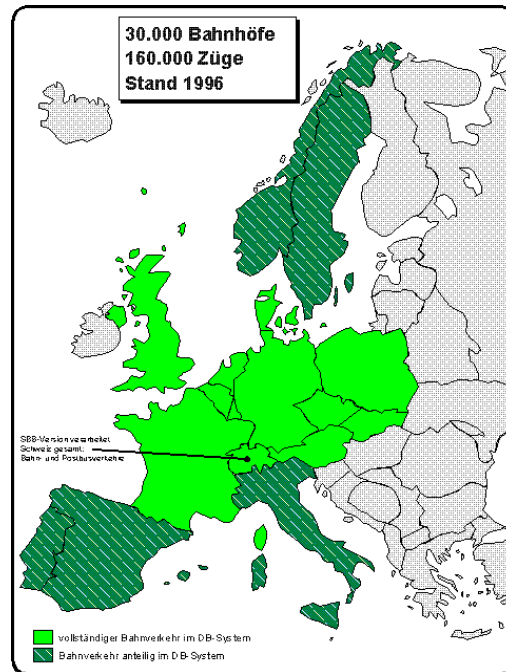
1.7.6. HAFAS - system informacji kolejowej

1.7.6.1. Opis ogólny

System HAFAS <http://www.hacon.de/>, służy do wyszukiwania połączeń kolejowych na terenie Niemiec i dużej części Europy. System ten gwarantuje obsługę zapytań w czasie krótszym niż 1 sekunda. Dziennie obsługiwane jest do 1.000.000 zapytań (ze strony Deutsche Bahn AG i HAFAS). Pełna wersja programu (bazy danych) zajmuje około 10MB.

System HAFAS cieszy się dużą popularnością, zarówno na terenie Niemiec, jak i w pozostałych krajach Unii Europejskiej. Nie bez znaczenia pozostaje tu prosty i wygodny w obsłudze interfejs, który jest szczególnie atrakcyjny dla osób nie mających dotychczas dużego kontaktu z elektronicznymi mediami.

²⁰ Należący do USWEST system był wykonywany i pracuje w siedzibie firmy Itnet, jednego z gdańskich dostawców Internetu



Rysunek 43: Reklamówka programu HAFAS



Rysunek 44: HAFAS: Połączenia kolejowe

Korzystając z systemu HAFAS możemy się również szybko dowiedzieć o połączeniach kolejowych w Polsce. Dostępna jest również informacja o głównych szlakach kolejowych.

Porównując system niemiecki z analogicznym polskim (który pracuje bez interfejsu graficznego), okazuje się, że - niestety - znajdowanie połączeń nie jest najsilniejszą stroną rodzimego oprogramowania. Podczas kilku eksperymentów okazało się, że system Hafas znalazł szybsze połączenia międzynarodowe (z Niemiec do jednego z mniejszych miast Polski) niż polski system połączeń kolejowych uwzględniający połączenia międzynarodowe.

1.7.6.2. Konfiguracja sprzętowa

W systemie HAFAS zastosowano komputer klasy Pentium 133MHz z 2GB dyskiem twardym SCSI, 64MB RAM. System operacyjny to OS/2 Warp Connect. Jako serwer WWW użyty został serwer Apache.

2. Analiza systemu BIS

2.1. Założenia i uwagi wstępne

Proces analizy i projektowania oparto na metodologii Rumbough, opisanej w [2]. Dokumentacja zawiera również elementy nie należące formalnie do tej metodologii, jak specyfikację wymagań, opis testów itp.

System będzie dostępny za pomocą usługi WWW w sieci Internet. Będzie on stosowany w placówkach edukacyjnych, jest więc możliwe stosowanie oprogramowania licencjonowanego z opcją dla jednostek uczelnianych i wersji „eduware”. Jedną z cech implementowanego systemu powinna być jego konkurencyjność wyrażająca się m.in. w niskich kosztach i krótkim czasie realizacji²¹.

2.2. Definicja problemu (Problem Statement).

Potrzebny jest system komputerowy, umożliwiający gromadzenie i udostępnianie informacji o publikacjach, dostępny w sieci WWW. System taki musi mieć współdziałać z popularnymi przeglądarkami typu Netscape Navigator, Microsoft Internet Explorer, dając użytkownikowi prosty i wygodny w obsłudze interfejs, zbliżony do stosowanych powszechnie rozwiązań. W procesie produkcji należy uwzględnić, że produkt będzie pracował w warunkach uczelnianych, co wymaga stosowania tanich rozwiązań sprzętowych i programowych.

Wszyscy użytkownicy systemu powinni mieć możliwość wyszukiwania informacji o publikacjach, niektórzy także dodatkowo - możliwość zapisu i uaktualniania informacji. Należy także przewidzieć dodatkowe możliwości dla osoby opiekującej się systemem (Administratora), takie jak tworzenie kopii zapasowych (back-up'ów), śledzenie ilości odwołań do usługi i strojenie bazy danych w trakcie użytkowania.

2.3. Specyfikacja wymagań

Lp.	wymaganie	opis
-----	-----------	------

²¹ Stawiamy się w pozycji firmy, której zadaniem jest zrealizować pewien projekt (począwszy od stadium analizy), którego efektem ma być określony produkt (w naszym przypadku - system bibliograficzny).

1	środowisko projektowe	dowolne (ew. wskazania mogą dotyczyć przyszłego przewidywanego systemu pracy)
2	środowisko pracy systemu	jeden z systemów operacyjnych pracujących w katedrze ZZI WETI PG - proponowany Unix w wersji dla Sun Sparc Microsystems lub w wersji dla komputerów PC (Linux)
3	interfejsy zewnętrzne	system należy projektować wykorzystując interfejs udostępniany przez przeglądarki webowe (Netscape Navigator, Microsoft Internet Explorer, Lynx itp.)
4	opis wyników i raportów dla użytkowników	raport (z działania systemu) dla użytkowników nie jest przewidywany w obecnie wytwarzanej wersji oprogramowania
5	źródła danych	użytkownik systemu
7	magazyny danych	nie ma nałożonych wymagań
8	słowniki danych	będą wytworzone w trakcie tworzenia oprogramowania, zostaną dołączone do dokumentacji projektu
9	specyfikacja funkcjonalna	jest zawarta w opisie problemu (definicja problemu)
10	wymagania wydajnościowe	podane w dokumentacji projektu (definicja problemu)
11	obsługa sytuacji wyjątkowych	ręczne usuwanie problemów przez administratora
12	priorytety projektowe	konkurencyjność produktu (niski koszt), wykonanie w odpowiednim czasie (limitowanym przewidywanym terminem pracy dyplomowej - początek lipca 1997)
13	priorytety implementacyjne	interfejs WWW, baza danych (drobniejsze kwestie jak optymalizacja bazy danych, elementy graficzne interfejsu mogą być zrealizowane w następnych wersjach)

14	przewidywane modyfikacje i rozbudowy systemu	integracja systemu z innymi serwisami uczelnianymi WWW ²²
15	kryteria akceptacji	wypełnienie założeń, przeprowadzenie testów funkcjonalnych i wydajnościowych, ukończenie dokumentacji, zatwierdzenie przez promotora
16	testy funkcjonalne	zostaną udokumentowane po ich przeprowadzeniu (dalej w niniejszym opracowaniu) - ich celem ma być ocena w jakim stopniu oprogramowanie realizuje rozwiązanie postawionego problemu
17	testy wydajnościowe	zostaną udokumentowane po ich przeprowadzeniu (dalej w niniejszym opracowaniu)
18	wskazówki dla projektantów	skorzystać z obiektowej metodologii analizy i projektowania Rumbough
19	wymagania wydajnościowe	system powinien realizować podstawowe zapytania (takie wyszukiwanie publikacji itp.) w odpowiednio krótkim czasie t.j. średnio nie dłużej niż 1 sek., nie uwzględniając opóźnienia wynikającego z trudności przepustowych łącza transmisyjnego
20	wymagania opcjonalne ²³	system powinien cechować się pewną przenośnością, pozwalającą na instalację zarówno w prostych środowiskach sprzętowych opartych na komputerach klasy PC, jak i bardziej zaawansowanych (np. HP, Sun itp.); w analizie należy założyć średnie obciążenie serwera WWW, na którym pracuje system, wynikające z

²² Inne rozszerzenia projektu - wypracowane podczas jego wytwarzania - opisano dalej w niniejszej dokumentacji

²³ Wypracowane w trakcie definiowania wymagań systemu, jako niekonieczne w obecnej wersji oprogramowania.

		możliwości stosowania go również do innych celów; zastosować rozwiązania pozwalające na prace użytkownikom terminali tekstowych, używających przeglądarek takich jak Lynx; grupa administratorów, powiązanych relacją zwierzchnictwa (hierarchia)
21	inne wymagania	niski koszt wytworzenia i utrzymania, modyfikowalność, zastosowanie obiektowej metodologii modelowania Rumbaugh

Tabela 13: Formalna specyfikacja wymagań systemu BIS

2.4. Model obiektowy

2.4.1. Uwagi i założenia wstępne

Tworząc model obiektowy korzystaliśmy z metodologii OMT, generując podstawowe klasy na bazie opisu słownego problemu. Następnie, stopniowo poprawialiśmy model, opierając się na wnioskach z dyskusji (w „zespole analityków”²⁴), własnym doświadczeniu oraz konsultacjach technicznych. Ostateczne zmiany wprowadziliśmy już w trakcie implementacji, co było wynikiem ewolucji idei naszego oprogramowania i postępu technologii (co pozwoliło na zastosowanie nowych koncepcji).

2.4.2. Ewolucja modelu

Zmiany wprowadzone do modelu to zarówno rozszerzenia, jak i zawężenia koncepcji. Rozszerzenia wynikały przede wszystkim z konieczności rozwiązywania problemów, których istnienie stwierdzono dopiero w procesie prac nad systemem, a także z wypracowania nowych rozwiązań (wprowadzonych do systemu lub pomyślanych jako potencjalne kierunki rozwoju). Ograniczenia to rezultat zmiany priorytetów (np. nadawanie praw użytkownikom do wszystkich tablic okazało się mało wydajne i kosztowne w sensie implementacyjnym i wydajnościowym) a także wynik testów (zwłaszcza w zakresie budowy interfejsu, którego prototyp okazał się

²⁴ Autorzy pracy, promotor, konsultant, członkowie projektu DARIF (zarządzanie informacją w przedsiębiorstwach w oparciu o bazy danych i WWW) mający styczność z tworzeniem systemu BIS podczas pobytu autorów na praktyce w Niemczech.

zbyt rozbudowany i nieczytelny dla przeciętnego użytkownika z uwagi na konieczność wpisania zbyt wielu danych). Przedstawiony poniżej proces ewolucji modelu wyraźnie pokazuje zmiany koncepcji naszego oprogramowania.

2.4.2.1. Model wstępny

Punktem wyjścia dla tego modelu jest bardzo proste ujęcie naszego systemu, oparte na niżej pokazanej strukturze (patrz rys. 45) ukazującej zależności pomiędzy podstawowymi obiektami. *Użytkownik*²⁵ (naszego systemu) korzysta z danych o *Publikacjach*, przynależnych do pewnych wydzielonych kategorii (*Dziedzin*).

Podjęcie to było na tyle uzasadnione, że istniała konieczność opracowania wstępnego modelu, obrazującego bardziej powiązania między klasami niż konkretną budowę danych. Nadto należy wskazać, że większość problemów pojawiło się w trakcie prac nad systemem - pojawiły się także nowe idee (zwłaszcza jako wynik obserwacji aktualnych trendów rynkowych i nowych rozwiązań stosowanych w produktach podobnego typu).

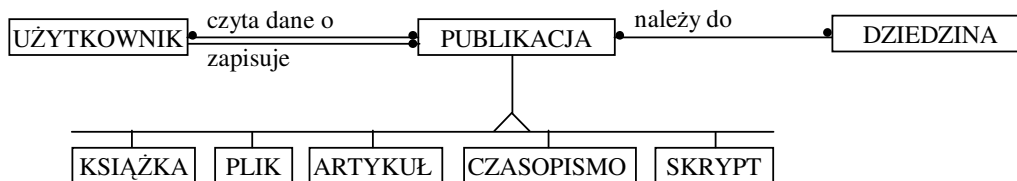


Rysunek 45: Model obiektowy systemu BIS - wersja I

Pierwsza rozbudowa tego modelu miała na celu zamodelowanie następujących aspektów:

- krotności relacji (zwłaszcza w odniesieniu do *Dziedzin* - publikacje mogą należeć do wielu z nich)
- modyfikacje danych: *Użytkownik* może także zapisywać dane o *Publikacjach*
 - nie wszyscy *Użytkownicy* mają takie same prawa do zmiany danych tablic
 - *Publikacje* mogą mieć różny typ (postać). Powinniśmy rozróżniać np. książki od czasopism, ponieważ określamy je danymi o innej strukturze (np. czasopismo posiada swój numer, a książka - ilość wolumenów).

²⁵ *Pismem pochylonym* oznaczyliśmy nazwy klas z modelu obiektowego. Wystąpienia obiektów oznaczane są pismem normalnym.

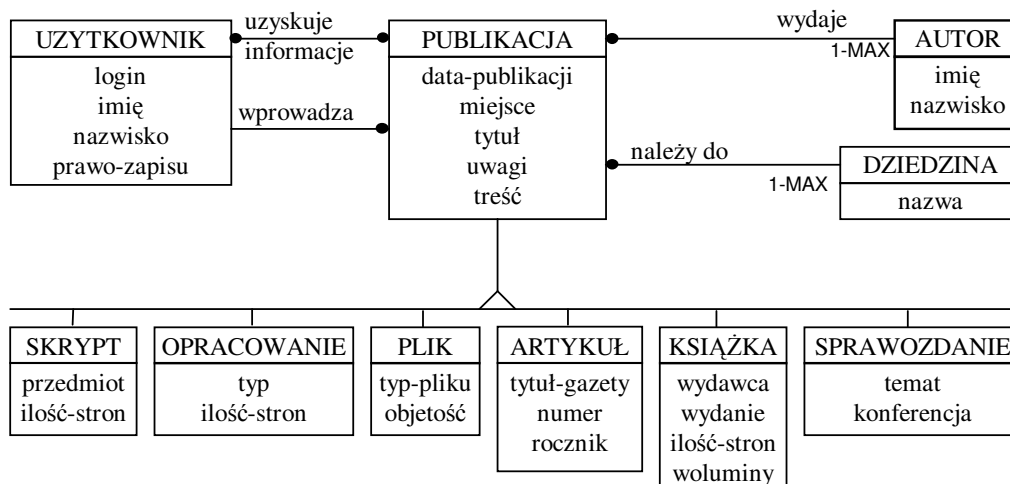


Rysunek 46: Model obiektowy systemu BIS - wersja II

Podczas definiowania atrybutów, okazało się, że celowe będzie wyodrębnienie nowej klasy *Autor*, zamiast rozbudowy klasy *Publikacja*. Za tym rozwiązaniem przemawiają względy wydajnościowe: przewidujemy, że w trakcie implementacji, klasa *Publikacja* stanie się często przeszukiwaną tablicą, a jedną z cech systemu, który chcemy stworzyć jest niewielkie obciążenie dla środowiska operacyjnego, jakie nasz system ma stwarzać (w związku z tym zapewne zastosujemy prosty motor bazy danych o dużej wrażliwości na uwarunkowania optymalizacyjne). Wadą jest konieczność zarządzania kolejną tablicą - należy o tym pamiętać z uwagi na specyfikę projektu: będziemy wykorzystywać język HTML, który nie został stworzony z myślą o wykorzystaniu do operowania na bazach danych²⁶. Na tym etapie założyliśmy także dla uproszczenia, że publikacja może zostać przypisana do pewnej maksymalnej liczby dziedzin. Rozwiązanie to, z oczywistych względów nieoptymalne, zostało wykorzystane, aby uprościć proces modelowania na wczesnym etapie.

A oto wynik końcowej fazy tworzenia modelu wstępnego:

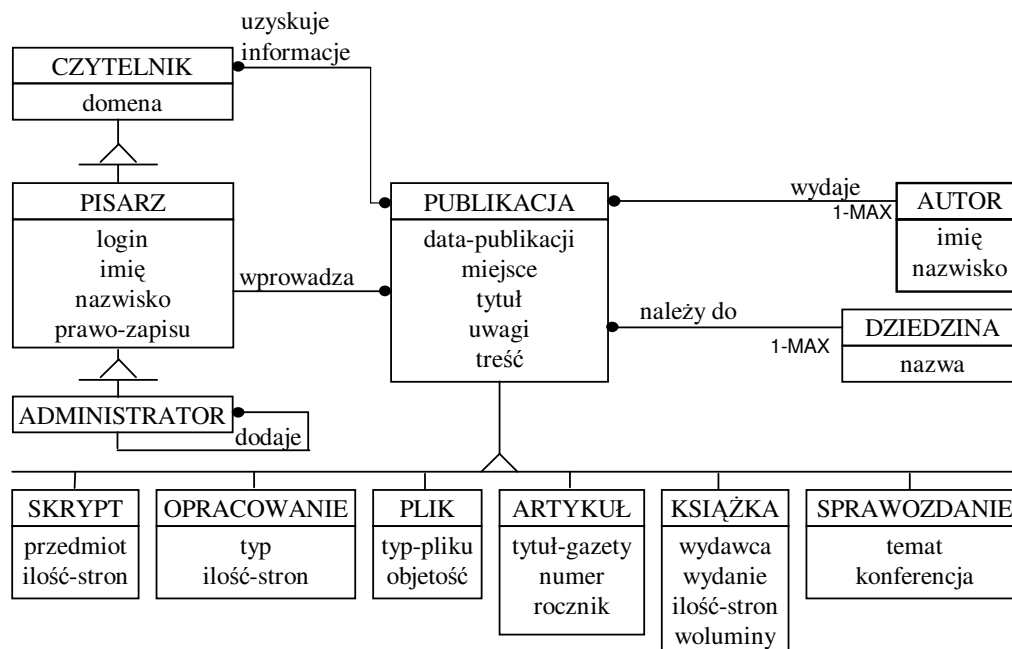
²⁶ Model ten był tworzony w czasie, gdy praktycznie nie było jeszcze baz danych współpracujących z przeglądarkami WWW - należało bardzo poważnie liczyć się z ograniczeniem dostępnych narzędzi i języków programowania.



Rysunek 47: Model obiektowy systemu BIS - wersja III

2.4.2.2. Model uwzględniający zróżnicowanie użytkowników systemu ze względu na prawa zapisu

Model ten wprowadza pojęcia *Pisarza*, *Czytelnika* i *Administradora*, jako uszczegółowienie dawniejszego *Użytkownika*. Zróżnicowanie to umożliwi nadanie niektórym użytkownikom praw zapisu do bazy danych, a także nadania dodatkowych praw administratorskich (np. możliwość wykonania back-upu czy usunięcia pewnych dziedzin, uznanych za niepotrzebne). Początkowo rozważaliśmy jedynie pojedynczego Administratora, korzystając jednak z doświadczeń z pracą w Informixie, uznaliśmy że warto zaimplementować mechanizm hierarchii Administratorów. Założyliśmy jednocześnie, że Administratorem może być jedynie jeden z pisarzy. Czytelnikiem może być każdy użytkownik sieci Internet. Jest on związany ze swoją domeną (adres internetowy komputera, z którego nawiązywana jest łączność), co daje możliwość kontroli dostępności naszej usługi (np. ograniczenie dla komputerów z domeny eti.pg.gda.pl.). Dalszą rozbudową tego modelu stało się wprowadzenie idei grup uprawnień.



Rysunek 48: Model obiektowy systemu BIS - wersja IV

2.4.2.3. Model poszerzający opis klasy *publikacja* o kategorie *przedmiot* i *słowa kluczowe*.

Etap ten zakończył się wykształceniem dojrzałej formy klasyfikacji publikacji. Początkowe szeregowanie jedynie do *Dziedzin* okazało się mało elastyczne, zwłaszcza w kontekście poziomów szczegółowości. Przykładowy problem: czy bardziej celowe jest tworzenie ogólnej kategorii „sztuka”, czy raczej bardziej zawężające pole szukania „malarstwo”. A może należałoby zdefiniować kategorię „impresjonizm” ? Warto także wspomnieć, że należy mieć na uwadze postawę przyszłych „pisarzy” - tym bardziej, że przewidujemy nadanie im uprawnień do tworzenia nowych kategorii - każdy z nich będzie zapewne uważał, że jego publikacja jest na tyle istotna, że warto przyznać jej kategorię bardzo uszczegółowioną (osoba pisząca pracę naukową np. o owadach może uznać, że kategoria „owady blaszkoskrzydłe” ma podobną wagę co np. „renesans”).

Wątpliwości te skłoniły nas do przyjęcia trójpoziomowej hierarchii klasyfikacji. Wydaje się nam, że większa ilość poziomów mogłaby niepotrzebnie skomplikować proces szeregowania publikacji, zaś mniejsza - nie rozwiązuje wyżej opisanego problemu.

W rezultacie, publikacja jest opisywana przez następujące kategorie: dziedzina, przedmiot i słowo kluczowe. Ich stopień abstrakcji jest następujący:

kategoria	opis
dziedzina	bardzo ogólne, opisujące zwykle dziedzinę nauki do której należy publikacja, typowe przykłady: „matematyka” czy „informatyka”
przedmiot	średni poziom ogólności, stanowi uszczegółowienie w zakresie konkretnej dziedziny; zawęża obszar zainteresowania w konkretnej gałęzi nauki; typowe przykłady to np. „grafika komputerowa”, „rachunek różniczkowy”
słowo kluczowe	pojedyncze słowo opisujące tematykę publikacji, zwykle odnoszące się do bezpośrednio do jej treści; typowe przykłady: „raster”, „bitmapa”, „dywergencja”, „gradient”

Tabela 14: Klasyfikacja publikacji do kategorii

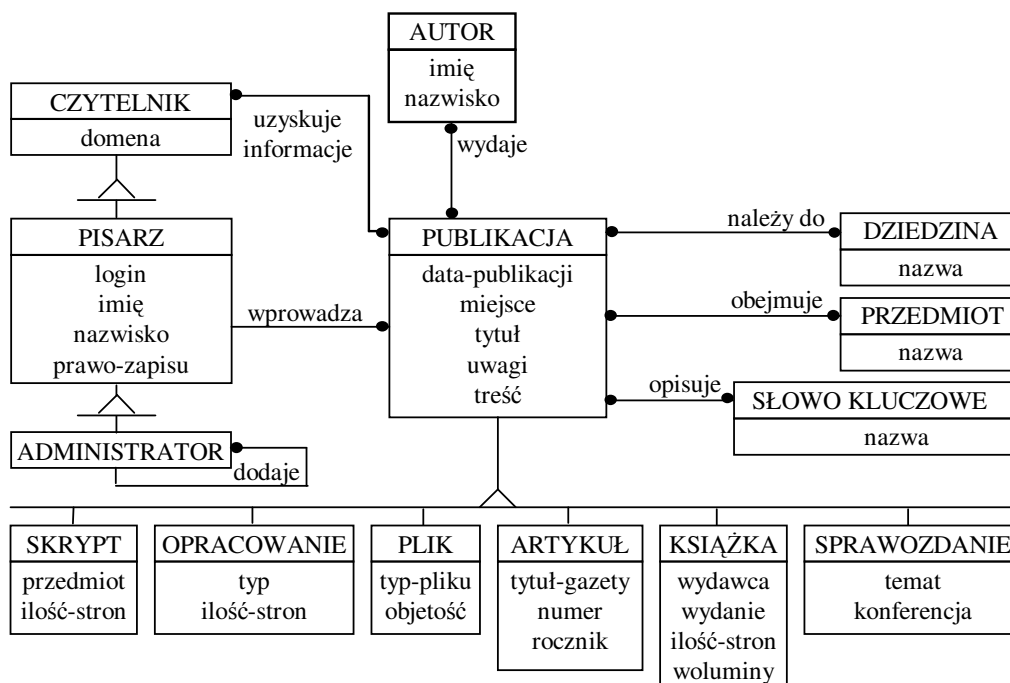
Metoda klasyfikacji publikacji ma kluczowe znaczenie dla systemu BIS. Użytkownicy muszą intuicyjnie rozumieć zakres abstrakcji poszczególnych kategorii. Wydaje się nam, że proponowane przez nas rozwiązanie spełnia ten warunek - każdą publikację można opisać w prosty sposób, unikając niejednoznaczności i kłopotów z określeniem zakresu pojęć. Przykłady dobrze dobranych opisów to:

Lp.	dziedziny	przedmioty	słowa kluczowe	tytuł publikacji
1.	informatyka	grafika komputerowa	bitmapa, raster, bmp, gif, pcx	„Kodowanie obrazów cyfrowych we współczesnych systemach komputerowych”
2.	informatyka	algorytmika	graf, komiwojażer, drzewo, cykl	„Problem komiwojażera i inne ciekawe zagadnienia teorii grafów w ujęciu algorytmicznym”
3.	informatyka	akustyka	fonem, digitalizacja,	„Wykorzystanie sieci neuronowych do

			neuron, sieć	konstrukcji systemów rozpoznawania mowy”
4.	informatyka	modelowanie	obiekt, analiza, metodologia	„Metodologia OMT w zastosowaniach”
5.	informatyka	modelowanie	model, „rich picture”	„Introduction to SSM”
6.	biologia	genetyka	białko, szok, odporność	„Białka szoku termicznego i ich wpływ na system immunologiczny”
7.	biologia	oceanobiologia, ekologia	plankton, ocean, fotosynteza	„Zróżnicowanie fitoplanktonu w morzach południowych, raport komisji Oceanografii przy ONZ”
8.	sztuki piękne	muzyka	Wagner, opera, romantyzm	„Die Welt von die Nibelungen”
9.	sztuki piękne	rzeźbiarstwo	surrealizm, modernizm	„Surrealizm a modernizm - szkic krytyczny”
10.	matematyka	rachunek prawdopodobieństwa	Poisson, rozkład, doświadczenie, rozrzut	„Static model of Poisson's experiment”

Tabela 15: Przykłady klasyfikacji publikacji

Na podstawie powyższych rozważań przyjęliśmy kolejną wersję modelu obiektowego:



Rysunek 49: Model obiektowy systemu BIS - wersja V

2.4.2.4. Model zakładający istnienie zróżnicowanie poziomu uprawnień do tablic

W tym miejscu wykształciła się praktycznie ostateczna postać naszej strategii bezpieczeństwa i kontroli dostępu. Najmniejsze prawa (tylko odczyt do publikacji²⁷) mają czytelnicy, identyfikowani tylko po dziedzinie komputera, z którego łączą się z systemem. W przypadku nałożenia ograniczeń na system w postaci dopuszczenia tylko użytkowników wybranej domeny (domen) rozwiązanie to może uszczuplać prawa użytkowników eti.pg.gda.pl., którzy z pewnych powodów (wyjazd, demonstracja na innych komputerach) decydują się łączyć z innych dziedzin, ale za przyjęciem tego rozwiązania są następujące czynniki:

- jest raczej wysoce prawdopodobne, że tego typu ograniczenia nie będą stosowane (system będzie otwarty - w sensie praw odczytu - dla szerokiego grona użytkowników)

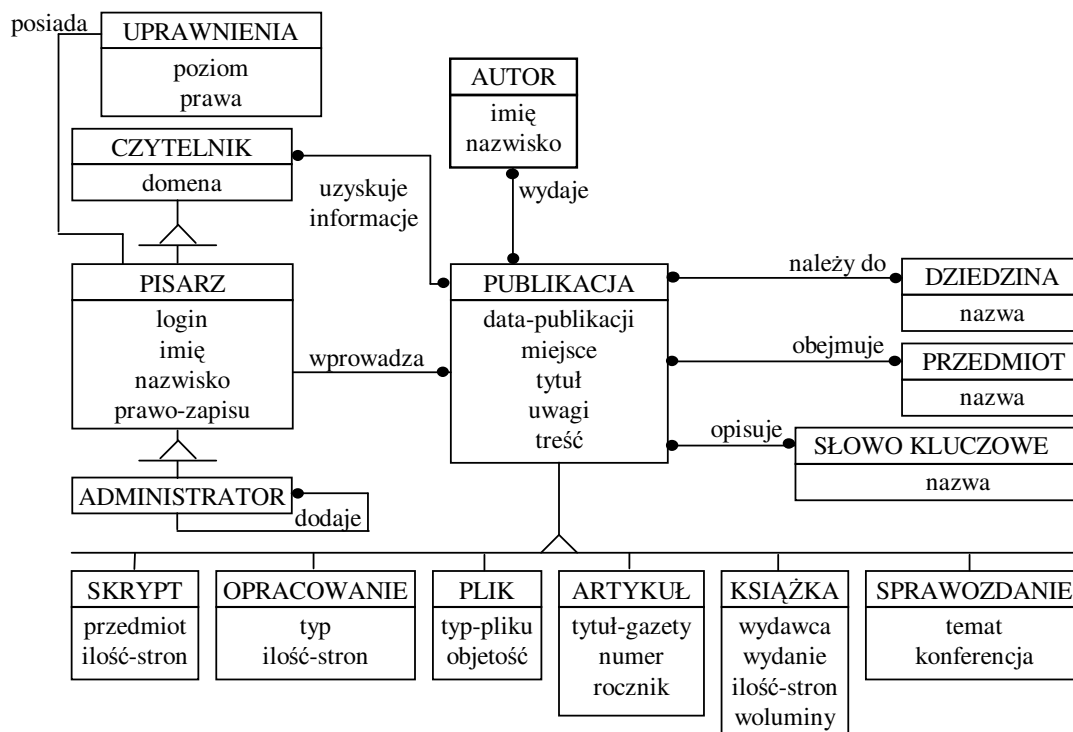
- nie jest możliwe automatyczne śledzenie loginów użytkowników (natomiast można śledzić w ten sposób ich dziedziny - zapewnia to protokół HTTP).

²⁷ Oraz do obiektów potomnych takich jak artykuł czy książka

- ew. rozwiązanie alternatywne: wpisywanie hasła użytkownika, tak jak to ma miejsce w przypadku prawa pisarzy (i w większości komercyjnych serwerów www i news) jest czasochłonne w implementacji i stanowi zdecydowane utrudnienie dla korzystającego z systemu, w praktyce odstręczające potencjalnych użytkowników.

Pisarze oczywiście podlegają znacznie szerszej kontroli niż czytelnicy. Oprócz sprawdzania domeny istotna jest znajomość identyfikatora i hasła (nadawanego użytkownikom naszego systemu) oraz posiadanie odpowiednich uprawnień do wykonania określonych operacji np.: dodania nowej publikacji. Uprawnienia pisarza identyfikowane są poprzez atrybut *poziom*. Pozwala to określić poziom dostępu, czyli przynależność do jednej z *grup uprawnień* np. pisarz, administrator (ale można również zdefiniować nowy poziom o nazwie np. pisarz-książek). Poziom uprawnień decyduje o faktycznych możliwościach pisarza np.: pisarz będzie mógł dodawać dowolne nowe publikacje, a pisarz-książek tylko publikacje będące książkami. W ten sposób możemy elastycznie nadawać i odbierać uprawnienia użytkownikom lub też tworzyć nowe grupy użytkowników. Uprawnienia są określane poprzez liczby określające dostęp do poszczególnych tablic. Liczby te to odwzorowanie bitowe praw odczytu (read), zapisu (write), kasowania (delete) i modyfikacji (update).

Zdecydowaliśmy się również na zdefiniowanie stanowiska administratora, który będzie miał możliwość przekazywania uprawnień innej osobie, która automatycznie staje się również administratorem. W tym przypadku nowo wprowadzony administrator nie może skasować informacji o swoim zwierzchniku. Podobna reguła dotyczyć będzie publikacji czy pisarzy wprowadzonych przez administratorów oraz publikacji wprowadzonych przez pisarzy. Administrator może usunąć wybraną publikację (np.: z powodu przestarzałości) jak również i samego pisarza. Jeśli pisarz ten wpisał do systemu więcej publikacji to administrator może usunąć wszystkie lub pozostawić część z nich stając się jednocześnie osobą, która wpisała te informacje (ponosi za nie odpowiedzialność).

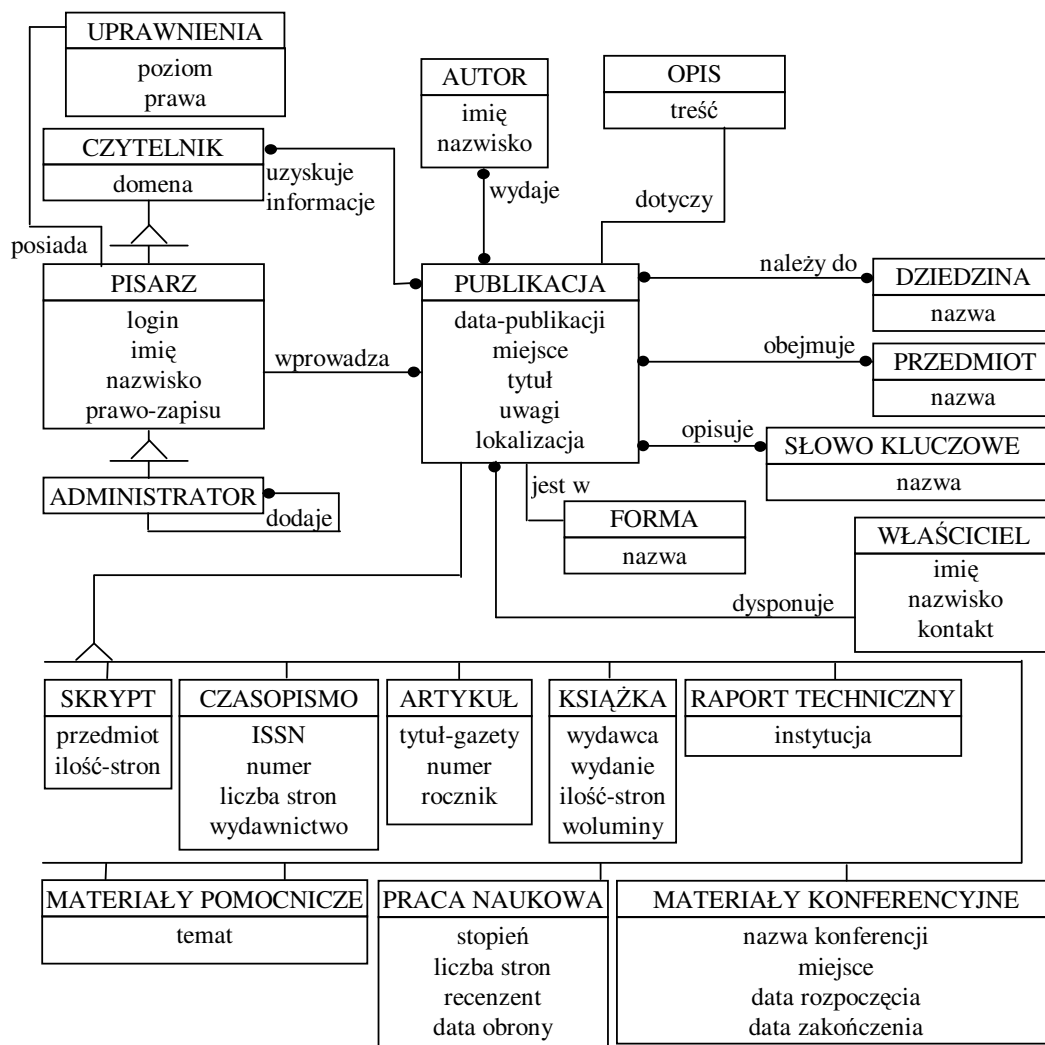


Rysunek 50: Model obiektowy systemu BIS - wersja VI

2.4.2.5. Model poszerzający opis klasy *Publikacja* o nowe atrybuty i klasy opisujące.

Na tym etapie poprawiono dość prowizoryczny podział publikacji. Wśród nich znalazł się: skrypt i opracowanie oraz plik. Istnienie tego ostatniego typu nasunęło nam myśl, że właściwie jest to pewna *Forma* (może to być również ksero, książka w twardej oprawie itd.), a więc niejako atrybut wszystkich publikacji. Wynikła stąd konieczność utworzenia obiektów opisujących fizyczne własności publikacji, co pokazuje następny diagram. Przy okazji zauważyliśmy brak wielu ważnych typów opisujących publikację, które dodaliśmy i uszczegółowiliśmy dopiero w tej fazie. Zdecydowaliśmy również, że warto wyróżnić atrybut treść (oznaczający krótką charakterystykę zawartości publikacji) i stworzyć dla niego odrębną klasę, o nazwie opis, powiązaną z klasą publikacja. Wprowadziliśmy również nowy atrybut o nazwie lokalizacja. Opisuje on miejsce w którym znajduje się opisywany egzemplarz publikacji. Może to być ścieżka dostępu do katalogu na serwerze katedralnym, adres strony WWW na serwerze uczelnianym. Wprowadziliśmy również dodatkową klasę opisującą właściciela opisywanej kopii publikacji. Jest to spowodowane tym, że atrybut lokalizacja może również

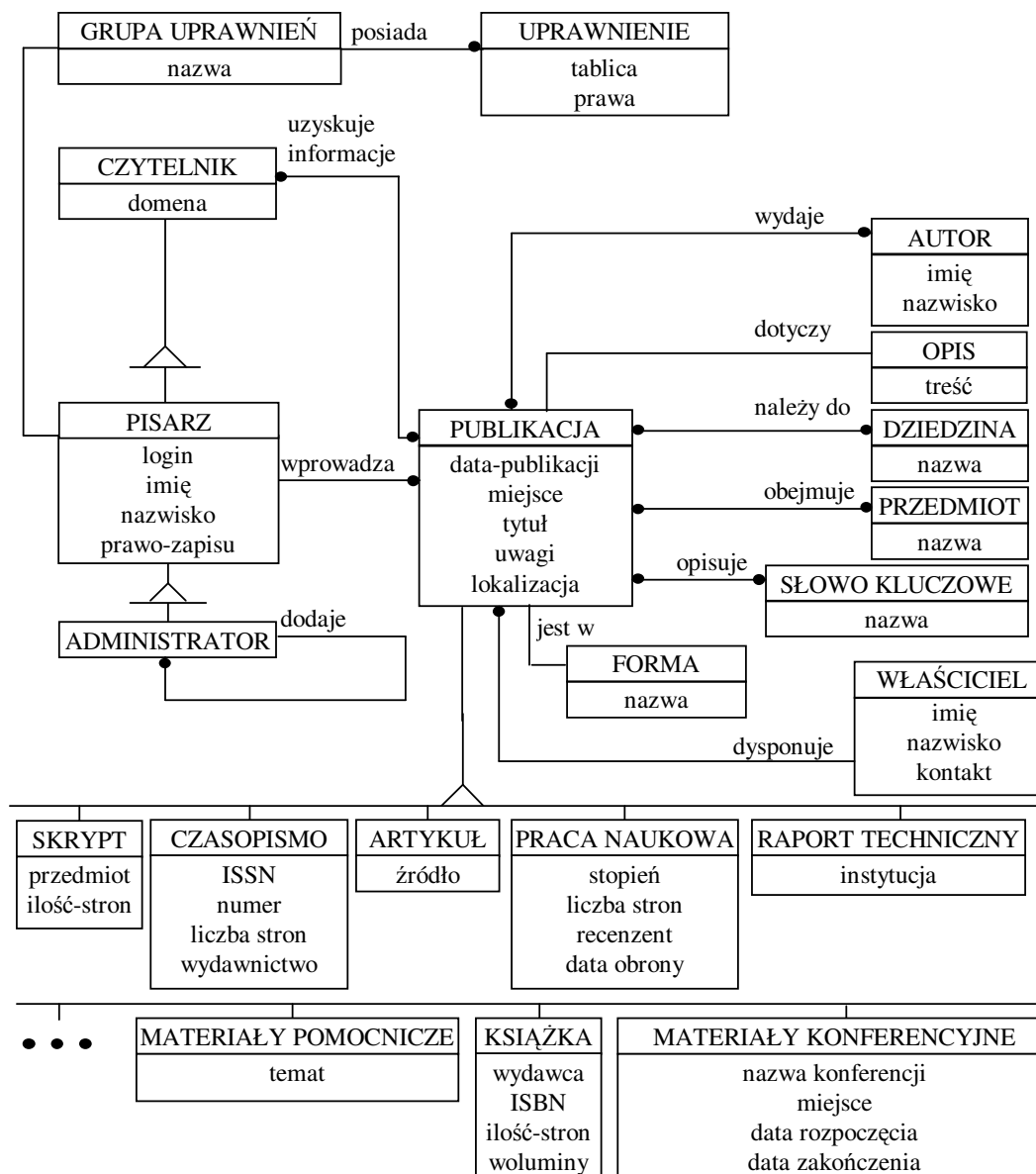
wskazywać np. na numer pokoju na uczelni. W takim przypadku warto również znać bliższe dane tej osoby i adres bądź telefon kontaktowy.



Rysunek 51: Model obiektowy systemu BIS - wersja VII

2.4.3. Model końcowy etapu analizy

W wersji ostatecznej znalazły się pewne zmiany, których znaczenie było raczej czysto „kosmetyczne” i nie wpłynęło na ogólną koncepcję systemu. Zmiany te to m.in. wprowadzenie klas *Uprawnienia* i *Grupa uprawnień*, oraz wypracowanie ostatecznej postaci atrybutów klasy *publikacja* i jej potomków.



Rysunek 52: Model obiektowy systemu BIS - wersja końcowa

2.4.4. Komentarz do procesu modelowania

W procesie modelowania specyfika naszego projektu nie wpłynęła w poważny sposób na wyniki analizy. Należy także wskazać na to, że niektóre elementy pojawiły się w modelu dopiero po przeprowadzeniu projektowania (iteracja modelowania). Jest to praktyczne potwierdzenie tezy, że nie wszystkie problemy i niuanse mogą być rozwiązane bez możliwości powtórzenia już wykonanej analizy. Jednocześnie kilkakrotnie stanęliśmy przed sytuacją wymagającą podjęcia decyzji:

- wprowadzić poprawki do uzyskanych wcześniej wniosków (zalety: nie buduje się „sztucznych” założeń, rozwiązuje się problemy ogólne, wady: konieczność przebudowy modeli, których zmiana może rzutować na inne dziedziny budowanej aplikacji, także i te, które do tej pory były już gotowe)

- budować rozwiązania zastępcze, w celu rozwiązania doraźnych problemów (zalety: szybki czas realizacji, wady: problemy z zarządzaniem rosnącą liczbą „uzupełnień”, rozwiązywane problemy szczególne)

Praktyka projektu BIS wykazała znaczną wyższość pierwszej metody. Widać to wyraźnie w złożonym procesie ewolucji modelu obiektowego, a także w zmianach koncepcji systemu.

2.5. Słownik danych

Niniejszy punkt obejmuje wersję słownika wypracowaną na etapie analizy obiektowej, przed przystąpieniem do fazy implementacji.

2.5.1. Klasy

nazwa klasy	opis
Administrator	operator systemu, osoba (grupa osób) zarządzająca spójnością i wydajnością systemu. Dbą o usuwanie zbędnych publikacji, kategorii itp. Każdy z administratorów jest jednocześnie pisarzem. Administratorzy są uszeregowani w strukturę drzewiastą zwaną hierarchią, z głównym administratorem znajdującym się w korzeniu, tworzoną na podstawie relacji wprowadzania do systemu. Prawa administratorskie są ograniczane „idąc w dół”, tzn. Administrator niższy nie może usunąć „wyższego”.
Artykuł	jeden z typów publikacji, całość lub część artykułu zamieszczonego w czasopiśmie lub gazecie. Forma ta nie wymaga istnienia macierzystej gazety (jako publikacji) w bazie danych.
Autor	osoba lub osoby posiadające prawa autorskie do publikacji, lub osoba widniejąca w Komitecie redakcyjnym.
Czasopismo	jeden z typów publikacji, gazeta lub inny periodyk.
Czytelnik	osoba korzystająca z systemu informacji bibliograficznej. W

	praktyce - użytkownik sieci Internet, korzystający z odpowiedniej przeglądarki, łączący się z adresem serwisu WWW współpracującego z Systemem.
Dziedzina	jedna z kategorii (patrz też przedmiot i słowo kluczowe), w których można zakwalifikować publikację. Jest to pojęcie najbardziej ogólne, opisujące szeroki zakres przynależności publikacji. W porównaniu z klasyczną biblioteką, dziedzina odpowiada grupie tematycznej w katalogu przedmiotowym. Dobrymi przykładami pojęć są: „matematyka”, „informatyka”, „biologia”.
Forma	określa fizyczną postać publikacji: ksero, oryginał, video, taśma magnetofonowa itp.
Grupa uprawnień	grupa użytkowników, posiadających określone praw do poszczególnych tablic
Książka	jeden z typów publikacji, charakteryzuje się posiadaniem ISBN (jako wydawnictwo)
Materiały konferencyjne	jedna z form publikacji, obejmuje każdy materiał powiązany z konferencją naukową, dotyczy zwłaszcza sprawozdań i zbiorów referatów
Materiały pomocnicze	jeden z typów publikacji, należą tu m.in. notatki wykładowe, pomocniki i „quick-reference cards” do oprogramowania, instruktaże i niektóre materiały szkoleniowe
Opis	krótki, kilkudzaniowy skrót treści publikacji, zawierający podstawowe informacje na temat problemów, których dotyczy dane wydawnictwo.
Pisarz	użytkownik systemu posiadający oprócz prawa czytania, także prawo wstawiania nowych publikacji i usuwania tych, które sam wstawił. Posiadanie uprawnień pisarza jest warunkiem uzyskania statusu administratora.
Praca naukowa	jeden z typów publikacji, określenie odnosi się do wszystkich prac będących podstawą uzyskania stopnia naukowego (prace dyplomowe, magisterskie, doktorskie itp.)

Przedmiot	jedna z kategorii (patrz też przedmiot i słowo kluczowe), w których można zakwalifikować publikację. Jest to pojęcie o średnim stopniu abstrakcji, ma sens zawężenia pola szukania w obrębie wybranej dziedziny. Zwykle określa jeden z kierunków konkretnej gałęzi nauki. W porównaniu z klasyczną biblioteką pełni rolę jednej z pozycji szczegółowego katalogu przedmiotowego.
Publikacja	wydawnictwo, o którym informację ma przechowywać system BIS, może występować w jednym z kilku typów (takich jak książka, skrypt, raport techniczny itp.), charakteryzuje się pewnymi cechami (np. formą - plik, ksero, oryginał itp.), posiada autora właściciela itp. Publikacje są klasyfikowane do kategorii.
Raport techniczny	jeden z typów publikacji
Skrypt	jeden z typów publikacji, obejmuje wydawnictwa naukowe dotyczące przedmiotów nauczanych w placówkach edukacyjnych, skrypty, zbiory zadań i ćwiczenia
Słowo kluczowe	jedna z kategorii (patrz też przedmiot i słowo kluczowe), w których można zakwalifikować publikację. Jest to wyraz lub związek frazeologiczny opisujący daną publikację (treść lub temat). Słowo kluczowe jest najbardziej wąską znaczeniowo kategorią.
Uprawnienie	prawa do danej tablicy, opisujące możliwe operacje (WDU - write, delete i update; R - read, jako prawo należne wszystkim użytkownikom nie jest zapisywane)
Właściciel	osoba posiadająca daną publikację (mogąca ją udostępnić). Zwykle bezpośredni właściciel lub inna osoba odpowiedzialna za udostępnianie.

Tabela 16: Słownik danych: opis klas

2.5.2. Związki

związek	klasy łączone	opis
czyta dane o	użytkownik, publikacja	- nazwa dla związku wypracowanego w początkowych fazach modelowania; związek ten łączył klasy użytkownik i publikacja. Nie uwzględniał możliwości zapisywania (dodawania, modyfikacji lub usuwania) nowych publikacji. W późniejszych etapach zastąpiony związkiem uzyskuje informacje.
Dodaje	administrator, administrator	- modeluje strukturę administratorów, ponieważ wprowadzanie kolejnych administratorów pozwala ułożyć ich w hierarchię drzewiastą, co pozwala określić ich wzajemne prawa (administrator A, który wprowadził administratora B, może go usunąć, ale nie odwrotnie)
dysponuje	właściciel, publikacja	- związek właściciela i publikacji. Modeluje sytuację, w której właściciel publikacji ma prawo wypożyczyć ją potrzebującym itp. Typowa sytuacja to np. posiadanie przez pracownika PG publikacji w pokoju i udostępnianie jej studentom.
jest w	publikacja, forma	- związek odnoszący się do występowania publikacji w pewnej formie. Np. książka (publikacja) C. Clark'a „Szyfr RSA” może być przechowywana na PG w postaci elektronicznej - pliku postscript.
należy do	publikacja, dziedzina	- związek ten w początkowych modelach łączył klasy publikacja i dziedzina, modelując klasyfikowanie publikacji do określonych dziedzin. W kolejnych wersjach, po dodaniu nowych kategorii, uzyskał podobny status do związków „obejmuje” i „opisuje”
obejmuje	publikacja, przedmiot	- związek podobny do związku „należy do”, opisuje przynależność publikacji do kategorii przedmiot
opisuje	publikacja,	- związek podobny do związku „należy do”, opisuje

	słowo kluczowe	przynależność publikacji do kategorii słowo kluczowe
posiada	uprawnienia, pisarz	- związek, który pojawił się w momencie wypracowania koncepcji uprawnień (grup uprawnień); pisarz posiada uprawnienia (czyli należy do grupy uprawnień) do tablic bazy danych, będących publikacjami. Początkowo, po wypracowaniu koncepcji uprawnień zamierzaliśmy nadawać uprawnienia do wszystkich tablic bazy danych, ale okazało się to zbędne - prawa dotyczą tylko tablic związanych bezpośrednio z publikacjami.
uzyskuje informacje	czytelnik, publikacja	- związek modelujący zależność czytelnika i publikacji. Czytelnik korzysta z systemu BIS uzyskując dane o żądanych publikacjach.
wydaje	autor, publikacja	- opisuje związek autora i publikacji W pierwszych wersjach modelu związek ten był klasy „1 do MAX”, gdzie MAX było pewną ustaloną stałą - wynikało to z początkowego założenia, że maksymalna liczba autorów danej publikacji będzie stała. Po przyjęciu decyzji o wprowadzeniu tablic łączących, związek ten przyjął postać „wiele do wiele”.
zapisuje	użytkownik, publikacja	- w pierwszych wersjach modelu opisywał możliwość wprowadzania nowych publikacji (a także modyfikacji danych), zastąpiony związkiem wprowadza

Tabela 17: Słownik danych: opis związków

2.5.3. Atrybuty

nazwa atrybutu	klasa	uwagi
data rozpoczęcia	materiały	data oficjalnego początku konferencji

	konferencyjne	
data zakończenia	materiały konferencyjne	data oficjalnego zakończenia konferencji
data-publicacji	publikacja	podaje datę ukazania się publikacji
domena	czytelnik	nazwa komputera użytkownika przesyłana do serwera
imię	autor	imię autora
imię	pisarz	imię pisarza
imię	właściciela	imię właściciela
instytucja	raport techniczny	określają wydawnictwo
ISBN	książka	kod widniejący w stopce redakcyjnej
ISSN	czasopismo	kod widniejący w stopce redakcyjnej
kontakt	właściciel	określa sposób, w jaki można zadawać pytania o publikację; adres email, telefon itp.
liczba stron	czasopismo	wielkość liczoną w stronach
liczba stron	książka	wielkość liczoną w stronach
liczba stron	praca naukowa	wielkość liczoną w stronach maszynopisu
liczba stron	skrypt	wielkość liczoną w stronach
login	pisarz	identyfikator, za pomocą którego przeprowadzana jest autoryzacja
miejsce	materiały konferencyjne	miejsce odbycia się konferencji - najczęściej miasto lub państwo
miejsce	publikacja	miejsce wydania
nazwa	dziedzina	np. „matematyka”, „informatyka”, „biologia”
nazwa	forma	np. plik, oryginał, ksero itp.
nazwa	przedmiot	np. „grafika”, „wektory”, „genetyka”
nazwa	słowo kluczowe	np. „jpg”, „genom”, „bushido”
nazwa konferencji	materiały konferencyjne	zawierający pełną nazwę konferencji, np. „V konferencja Genetyki Molekularnej Genom 97”

nazwisko	autor	nazwisko autora
nazwisko	pisarz	nazwisko pisarza
nazwisko	właściciel	nazwisko właściciela
numer	artykuł	numer gazety, z której on pochodził w danym roczniku - np. artykuł „Makrowirusy” zamieszczony w PC Kurierze 5/97 ma numer 5
numer	czasopismo	numer w danym roczniku - np. PC Kurier 5/97 ma numer 5
przedmiot	skrypt	dotyczy przedmiotu, z którym związany jest skrypt np. zbiór zadań K. Kolki z fizyki, używany w nauczaniu na Politechnice Gdańskiej miałyby w tym polu przedmiot „fizyka”
recenzent	praca naukowa	zawierający imię i nazwisko recenzenta
rocznik	artykuł	- rocznik gazety, z której on pochodził - np. artykuł „Makrowirusy” zamieszczony w PC Kurierze 5/97 ma rocznik 1997
stopień	praca naukowa	określający jej poziom: np. praca magisterska, doktorska itp.
temat	materiały pomocnicze	podający zakres tematyki materiałów
treść	opis	jest jego „zawartością” - krótka charakterystyką publikacji
tytuł	publikacja	jedna z podstaw szukania
tytuł gazety	artykuł	podający z jakiej gazety (lub innego periodyka) został zaczerpnięty artykuł
uwagi	publikacja	komentarz
woluminy	książka	podający liczbę woluminów wchodzących w jej skład (np. 2 - dla publikacji dwutomowej)
wydanie	książka	numer kolejny wznowienia np. II
wydawca	książka	określa wydawnictwo np. Yourdon Inc.

wydawnictwo	czasopismo	podaje właściciela np. „Wydawnictwo Lupus”
źródło	artykuł	określa pochodzenia artykułu (tytuł gazet, numer itp.)

Tabela 18: Słownik danych: opis atrybutów

2.5.4. Synonimy

nazwa	synonim
postać publikacji	inaczej: typ; określa własności fizyczne publikacji, np. książka, czasopismo itp.

Tabela 19: Słownik danych: opis synonimów

2.5.5. Inne

nazwa	opis
kategoria	nazwa stosowana w odniesieniu do sposobu, w jaki opisujemy publikacje; oznacza rodzaj publikacji, jej zaklasyfikowanie. Początkowo termin ten obejmował jedynie dziedziny, potem także słowa kluczowe i przedmioty.
Klasyfikowanie	określanie przynależności publikacji do kategorii (tj. dziedzin, słów kluczowych i przedmiotów)
hierarchia administratorów	struktura drzewiasta określająca kolejność wprowadzania administratorów do systemu i powiązania pomiędzy nimi.
typ publikacji	jedna z postaci publikacji - np. książka, artykuł, czasopismo lub materiały konferencyjne

Tabela 20: Słownik danych: informacje dodatkowe

2.6. Model dynamiczny

2.6.1. Scenariusze zdarzeń i odpowiadające im ścieżki zdarzeń

2.6.1.1. Uwagi wstępne

W poniższych scenariuszach przyjęto dla uproszczenia, że system informacji bibliotecznej BIS pracuje w całości na jednym komputerze (fizycznie). Dla ułatwienia rozważań wyróżniono jako dwie osobne części system BIS i jego bazę danych.

2.6.1.2. Scenariusz I : Czytelnik dokonuje odczytu informacji o publikacji

Czytelnik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

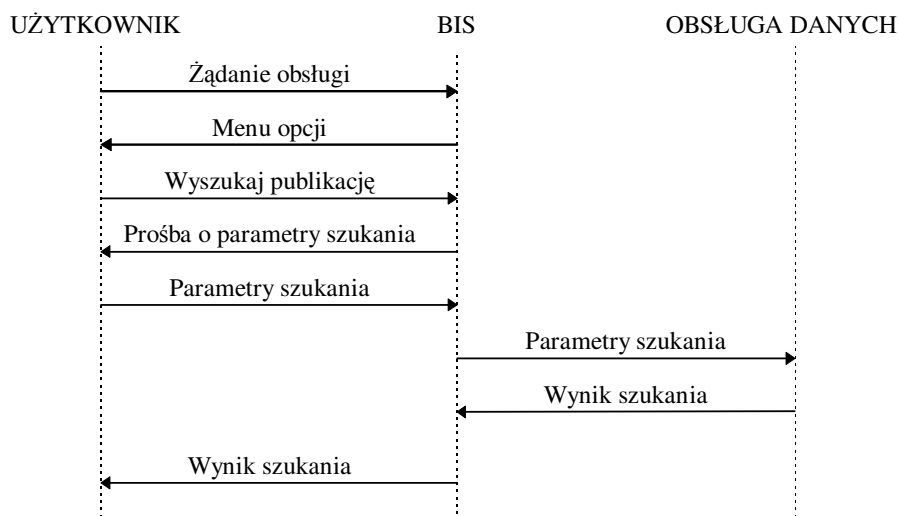
Czytelnik wybiera opcję wyszukiwania publikacji.

BIS prosi użytkownika o podanie informacji o publikacji.

Czytelnik wprowadza znane mu dane.

BIS wyszukuje w swojej bazie danych pełną informację na temat publikacji.

BIS podaje użytkownikowi informację na temat znalezionej publikacji.



Rysunek 53: Ścieżka zdarzeń odpowiadająca scenariuszowi I

2.6.1.3.Scenariusz II : Pisarz wpisuje nową publikację

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję wpisywania nowej publikacji.

BIS prosi o podanie loginu i hasła uprawniającego do zapisu.

Użytkownik podaje wymagane informacje.

BIS sprawdza, czy podany login i hasło znajdują się na liście uprawnionych (pisarze).

BIS umożliwia pozytywnie zweryfikowanemu pisarzowi wybranie typu publikacji.

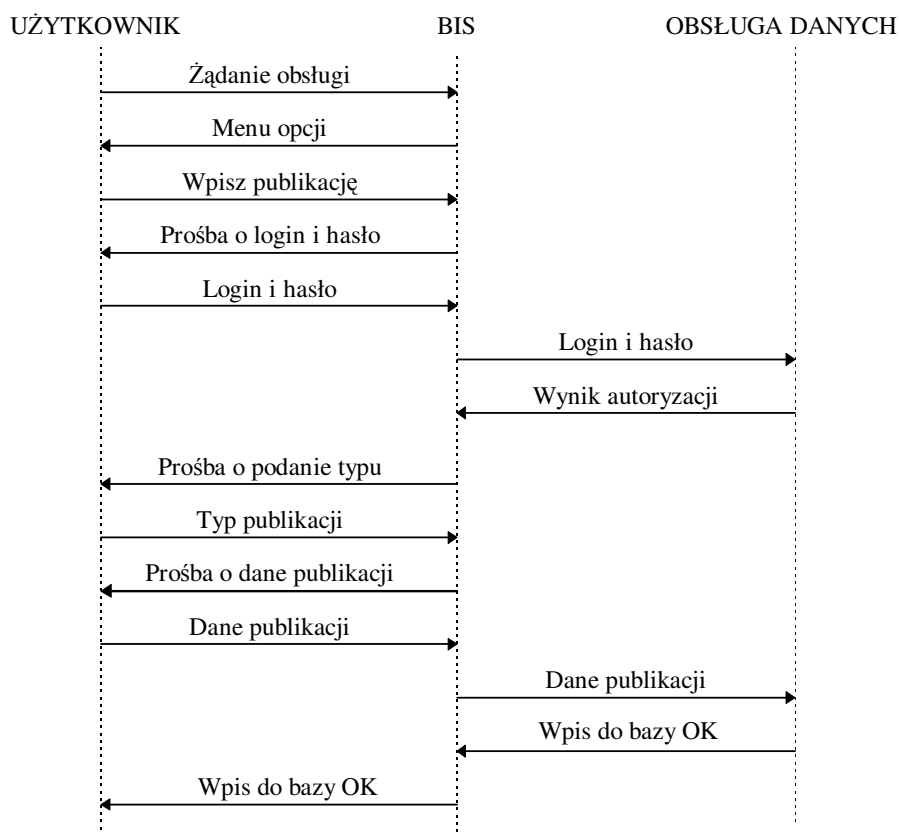
Użytkownik (pisarz) wybiera jeden z dozwolonych typów publikacji.

BIS umożliwia wprowadzenie danych o nowej publikacji.

Pisarz wpisuje dane i potwierdza zakończenie tej czynności.

BIS uaktualnia dane bazy danych.

BIS informuje użytkownika o dokonanej aktualizacji.



Rysunek 54: Ścieżka zdarzeń odpowiadająca scenariuszowi II

2.6.1.4.Scenariusz III : Pisarz wpisuje nową publikację decydując się na utworzenie nowego typu publikacji.

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję wpisywania nowej publikacji.

BIS prosi o podanie loginu i hasła uprawniającego do zapisu.

Użytkownik podaje wymagane informacje.

BIS sprawdza, czy podane login i hasło znajdują się na liście uprawnionych (pisarze).

BIS umożliwia pozytywnie zweryfikowanemu pisarzowi wybranie typu publikacji.

Pisarz wybiera tworzenie nowego typu publikacji.

BIS prosi o określenie typów danych, jakie charakteryzują nowy typ publikacji.

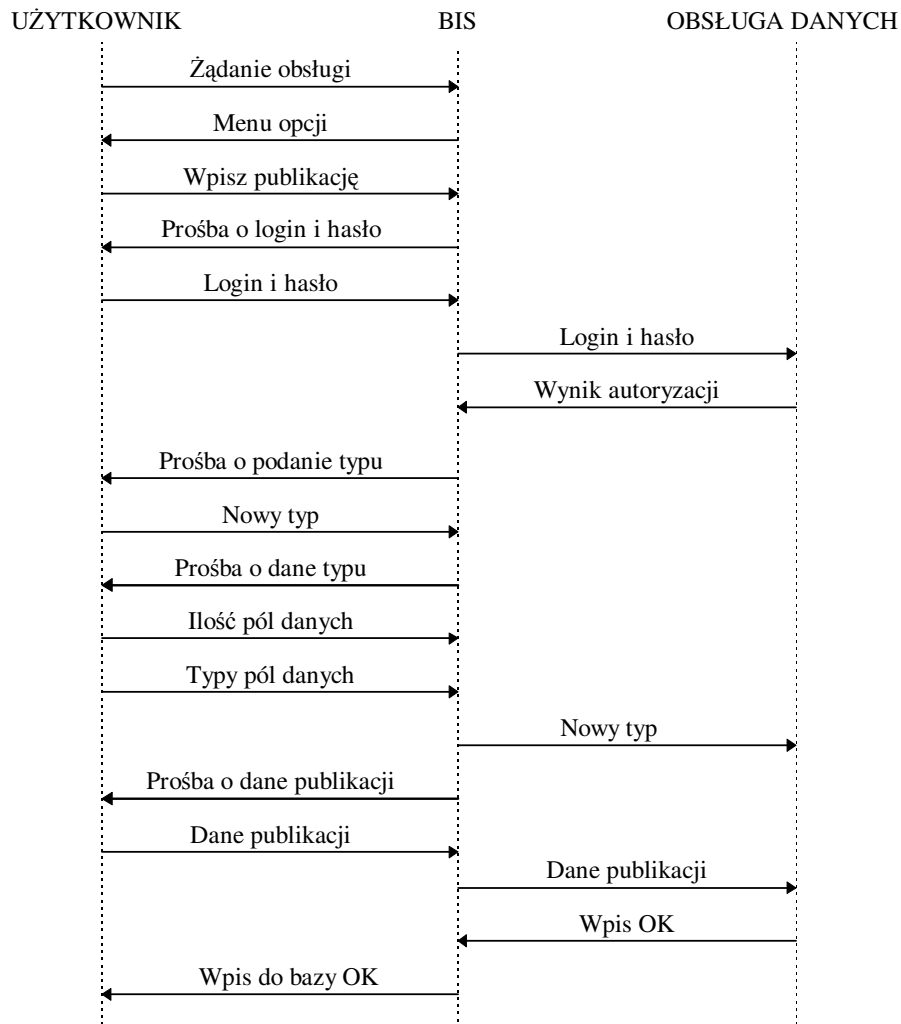
Pisarz wprowadza ilość pól danych i określa ich typy.

BIS tworzy nowy typ publikacji i informuje o tym fakcie użytkownika.

Pisarz wprowadza dane nowej publikacji.

BIS uaktualnia dane bazy danych.

BIS informuje użytkownika o dokonanej aktualizacji.



Rysunek 55: Ścieżka zdarzeń odpowiadająca scenariuszowi III

2.6.1.5.Scenariusz IV : Administrator dokonuje wprowadzenia nowego użytkownika

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję wpisywania nowego pisarza.

BIS prosi o podania loginu i hasła, opisującego użytkownika.

Administrator wprowadza login i hasło.

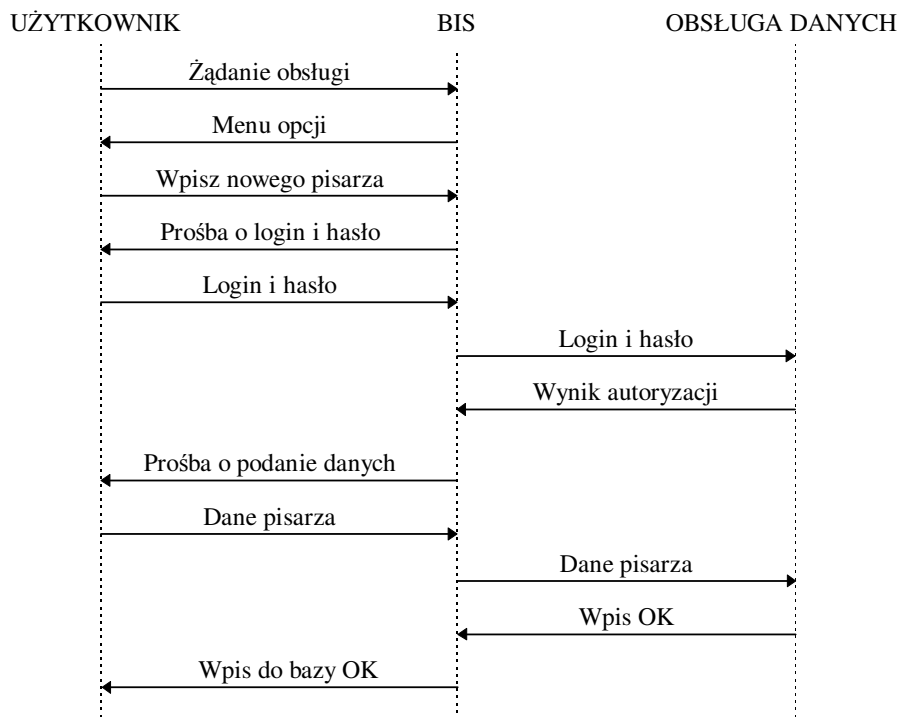
BIS sprawdza czy podana osoba ma uprawnienia Administratora.

BIS prosi o wprowadzenie danych nowego pisarza.

Administrator wprowadza dane nowego pisarza.

BIS uaktualnia dane bazy danych.

BIS informuje użytkownika o dokonanej aktualizacji.



Rysunek 56: Ścieżka zdarzeń odpowiadająca scenariuszowi IV

2.6.1.6.Scenariusz V : Administrator wykonuje dump bazy danych (back-up)

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję wykonywania back-up.

BIS prosi o podania loginu i hasła, opisującego użytkownika.

Administrator wprowadza login i hasło.

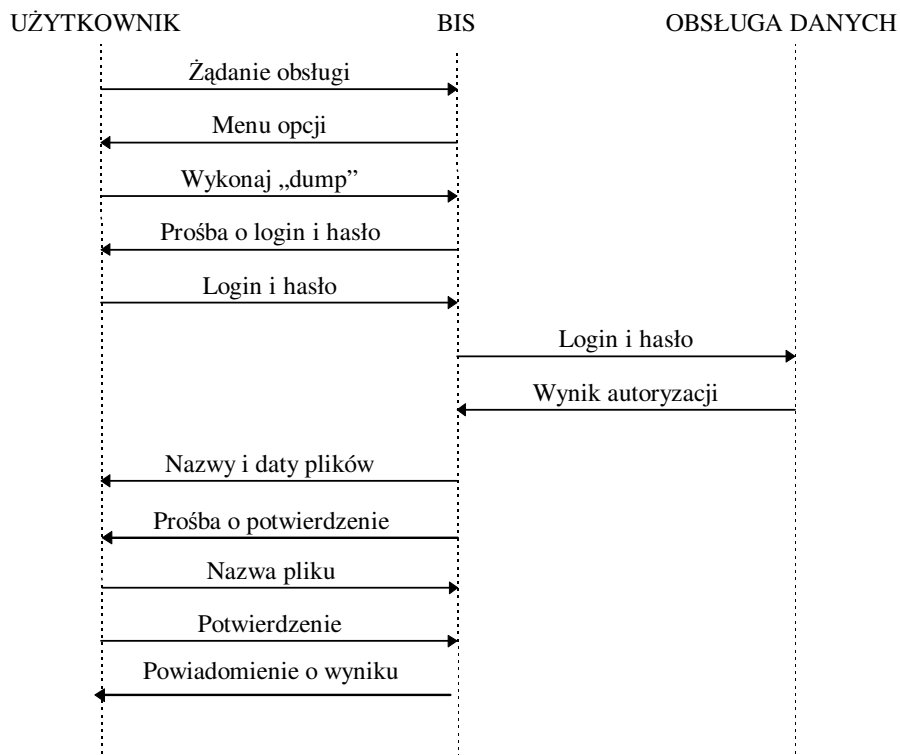
BIS sprawdza czy podana osoba ma uprawnienia Administratora.

BIS wyświetla informację o znalezionych plikach ze zarchiwizowanymi stanami bazy danych z prośbą o potwierdzenie żądania wykonania back-up i pytaniem o nazwę pliku.

Administrator potwierdza chęć wykonania back-up i podaje nazwę pliku.

BIS przeprowadza archiwizację systemu i zapamiętuje stan bazy danych w pliku.

BIS informuje Administratora o zakończeniu archiwizacji.



Rysunek 57: Ścieżka zdarzeń odpowiadająca scenariuszowi V

2.6.1.7.Scenariusz VI : Administrator dokonuje przywrócenia stanu bazy danych z pliku ze zarchiwizowanymi danymi.

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję wykonywania odtworzenia z back-up.

BIS prosi o podania loginu i hasła, opisującego użytkownika.

Administrator wprowadza login i hasło.

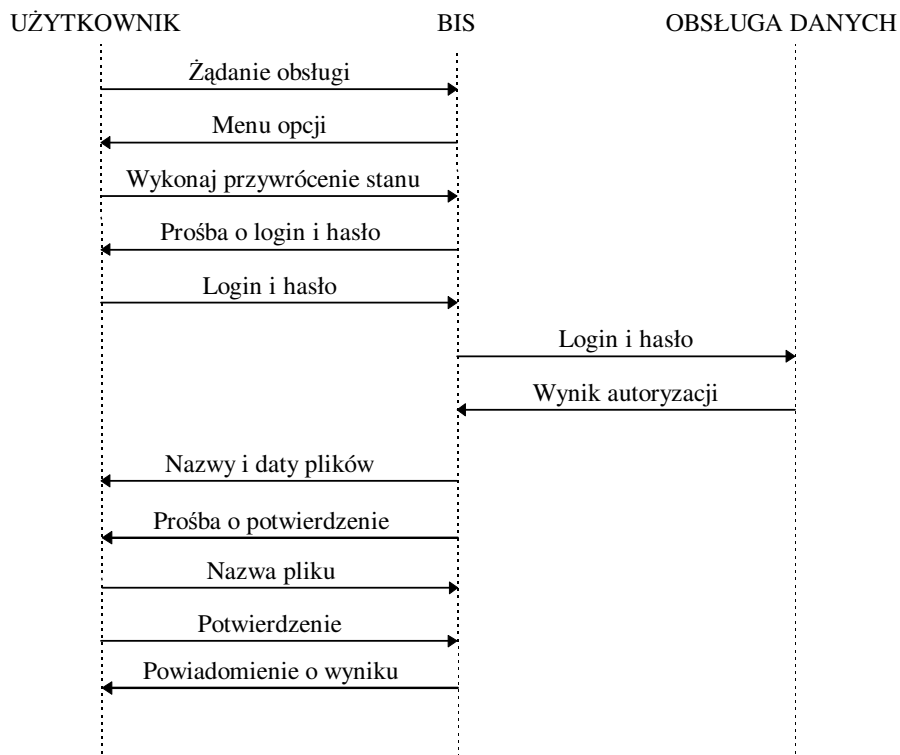
BIS sprawdza czy podana osoba ma uprawnienia Administratora.

BIS wyświetla informację o znalezionych plikach ze zarchiwizowanymi stanami bazy danych z prośbą o potwierdzenie żądania wykonania odtworzenia stanu z back-up i pytaniem o nazwę pliku.

Administrator potwierdza chęć wykonania odtworzenia z back-up i podaje nazwę pliku.

BIS przeprowadza archiwizację systemu i odtwarza stan bazy danych z pliku.

BIS informuje Administratora o zakończeniu procesu.



Rysunek 58: Ścieżka zdarzeń odpowiadająca scenariuszowi VI

2.6.1.8.Scenariusz VII : Administrator dokonuje usunięcia publikacji.

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję usuwania publikacji.

BIS prosi o podania loginu i hasła, opisującego użytkownika.

Administrator wprowadza login i hasło.

BIS sprawdza czy podana osoba ma uprawnienia Administratora.

BIS wyświetla listę publikacji, z pytaniem którą z nich należy usunąć.

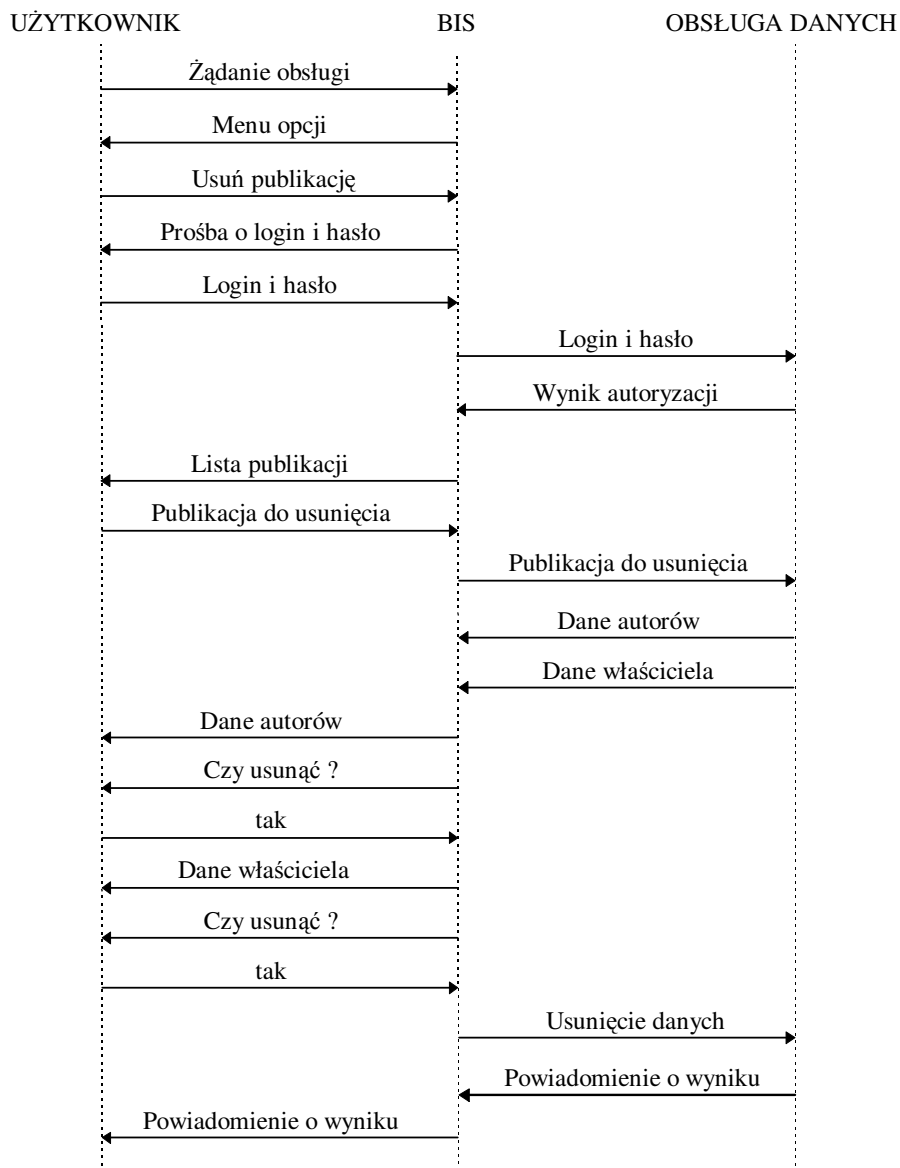
Administrator zaznacza publikację do usunięcia.

BIS sprawdza, czy dane autorów i właścicieli występują w innych publikacjach i pyta Administratora, czy usunąć te dane.

Administrator potwierdza.

BIS usuwa dane o publikacji i dane związane.

BIS powiadamia Administratora o tym fakcie.



Rysunek 59: Ścieżka zdarzeń odpowiadająca scenariuszowi VIII

2.6.1.9.Scenariusz VIII : Pisarz próbuje usunąć publikację, której sam nie wprowadził.

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję usuwania publikacji.

BIS prosi o podanie loginu i hasła uprawniającego do zapisu.

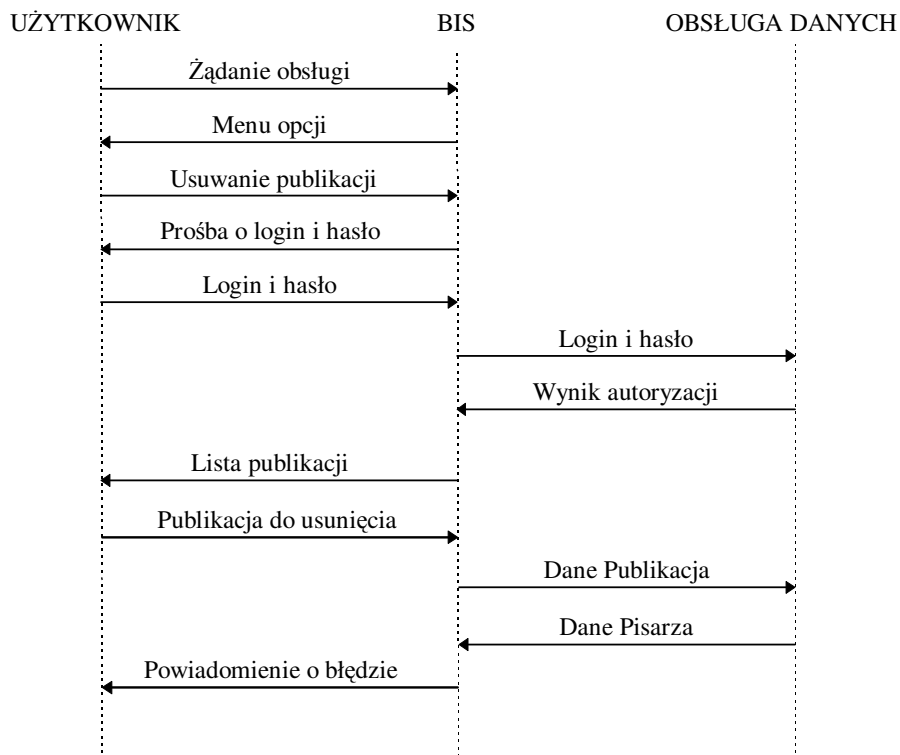
Użytkownik podaje wymagane informacje.

BIS pokazuje listę publikacji z pytaniem, którą z nich usunąć.

Pisarz wskazuje publikację do usunięcia.

BIS sprawdza, czy publikacja była wpisana przez pisarza.

BIS informuje, że usunięcie publikacji jest niemożliwe.

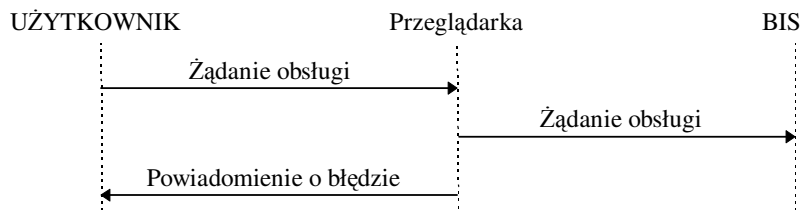


Rysunek 60: Ścieżka zdarzeń odpowiadająca scenariuszowi VIII

2.6.1.10. Scenariusz IX : Użytkownik próbuje połączyć się z systemem BIS, podczas zakłóceń łączności.

Użytkownik łączy się z serwerem BIS.

Przeglądarka Użytkownika informuje go o niemożności nawiązania połączenia z serwerem BIS.



Rysunek 61: Ścieżka zdarzeń odpowiadająca scenariuszowi IX

2.6.1.11. Scenariusz X : Administrator usuwa zbędną dziedzinę.

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję usuwania dziedzin.

BIS prosi o podania loginu i hasła, opisującego użytkownika.

Administrator wprowadza login i hasło.

BIS sprawdza czy podana osoba ma uprawnienia Administratora.

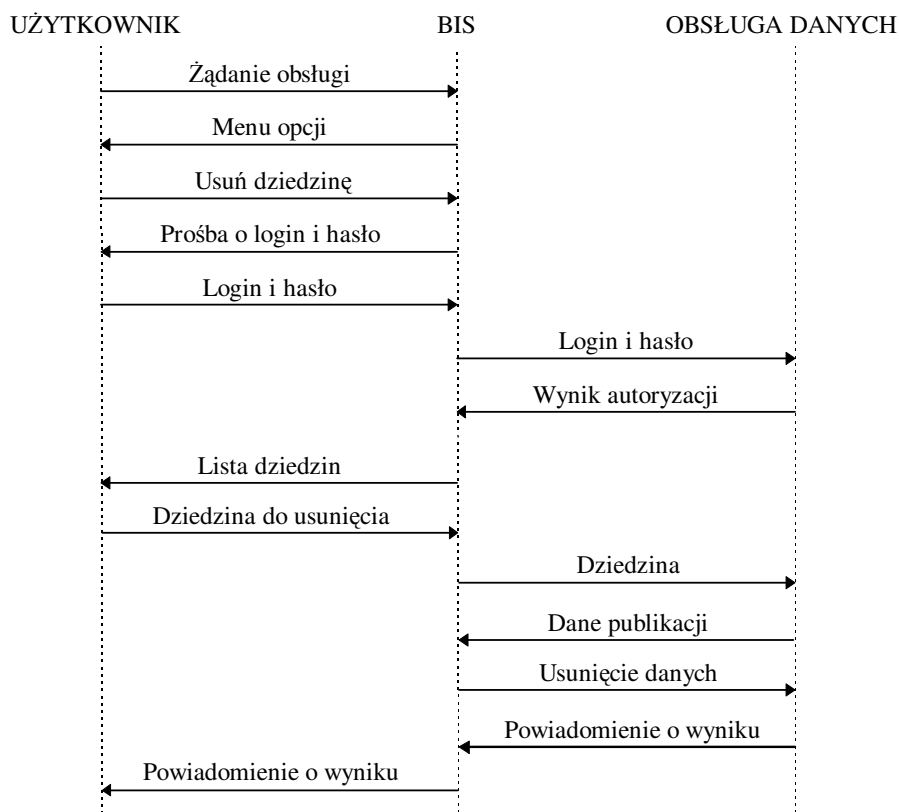
BIS wyświetla listę dziedzin, z pytaniem którą z nich należy usunąć.

Administrator zaznacza dziedzinę do usunięcia.

BIS sprawdza, czy istnieją publikacje należące do danej dziedziny.

BIS usuwa powiązania publikacji z usuwaną dziedziną.

BIS usuwa dziedzinę i powiadamia użytkownika o zakończeniu działalności.



Rysunek 62: Ścieżka zdarzeń odpowiadająca scenariuszowi X

2.6.1.12.Scenariusz XI : Administrator próbuje usunąć Administratora.

Użytkownik łączy się z serwerem BIS.

BIS zgłasza się stroną główną, podającą możliwie do wyboru opcje.

Użytkownik wybiera opcję usuwania użytkowników.

BIS prosi o podania loginu i hasła, opisującego użytkownika.

Administrator wprowadza login i hasło.

BIS sprawdza czy podana osoba ma uprawnienia Administratora.

BIS wyświetla listę użytkowników, z pytaniem którego z nich należy usunąć.

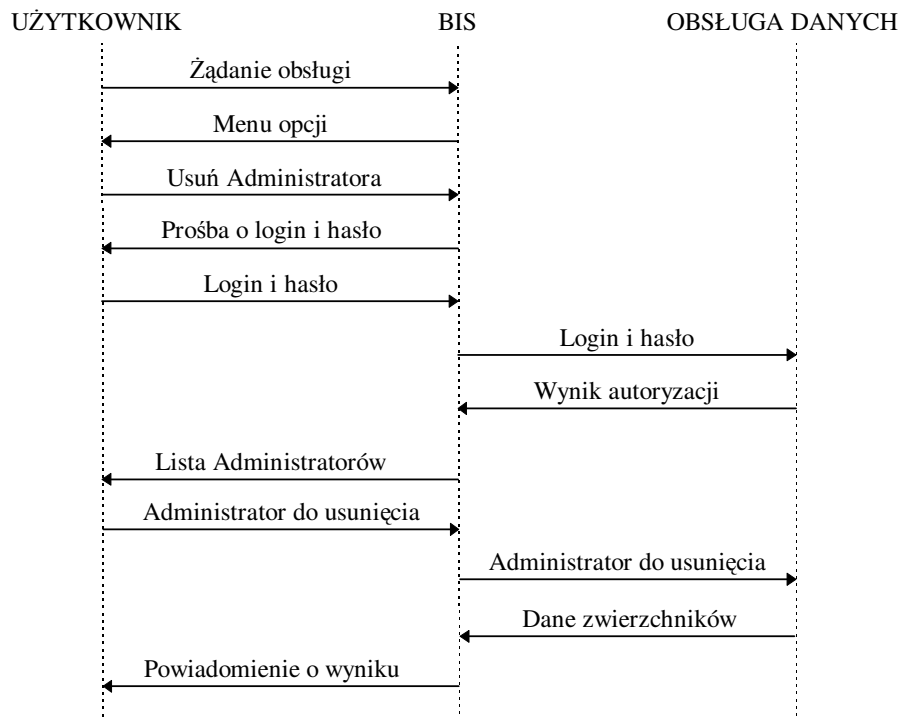
Administrator wskazuje jednego z użytkowników.

BIS sprawdza, do jakiej grupy uprawnień należy użytkownik wskazany przez administratora.

BIS otrzymuje informację, że użytkownik jest Administratorem.

BIS sprawdza, czy Administrator żądający usunięcia użytkownika jest wyżej od niego w strukturze (hierarchii Administratorów).

Ponieważ BIS nie uzyskał odpowiedzi pozytywnej, informuje Administratora, że jego żądanie nie jest możliwe do zrealizowania.



Rysunek 63: Ścieżka zdarzeń odpowiadająca scenariuszowi XI

2.6.2. Przykładowe diagramy przejść stanów

2.6.2.1. Uwagi wstępne

Zamieszczone poniżej diagramy przejść stanów są wynikiem zarówno procesu analizy, jak i projektowania. Dzięki iteracji procesu modelowania, możliwe stało się wprowadzenie pojęć wypracowanych na późniejszych etapach

modelowania: systemu bazy danych i systemu WWW. Poniższe punkty przedstawiają kolejne fazy modelowania, wraz z krytycznym komentarzem do nich.

2.6.2.2. Zarys systemu

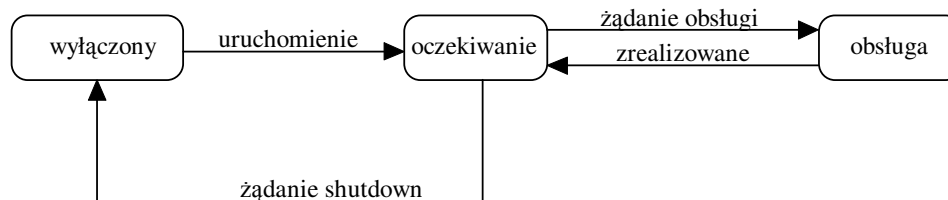
Na pierwszym, najprostszym diagramie uwzględniliśmy jedynie najprostsze działanie systemu. Jest ono opisane za pomocą trzech stanów:

wyłączony - stan, w jakim znajduje się system bezpośrednio po instalacji lub o wyłączeniu. Procesy są zatrzymane, realizacja żądań - niemożliwa.

oczekiwanie - system pracuje, oczekując na użytkowników

obsługa - system realizuje zapytania (wyszukiwanie lub aktualizację danych)

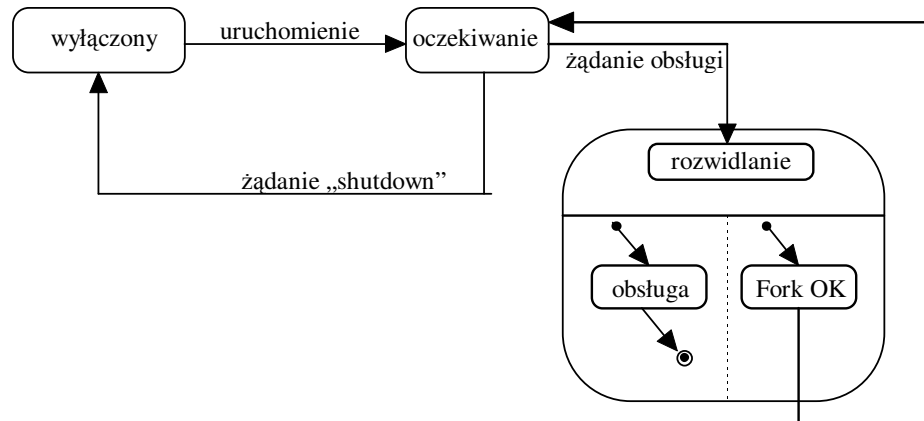
Wadą tego modelu jest jego prostota - nie uwzględnia on m.in. obsługi wołań współbieżnych. Jak jednak okazało się w dalszych etapach modelowania notacja Harela - nawet poszerzona o możliwość modelowania współbieżności nie jest wystarczająca do dobrego przedstawienia naszego systemu. Powodem jest specyfika systemu BIS - jeśli bowiem próbujemy modelować proces obsługi kolejnego żądania (jest on realizowany przez specjalnie do tego celu tworzony proces-potomka, który ginie po wykonaniu zadania), to obok „normalnej” współbieżności procesów mamy do czynienia z relacją podrzędności obiektów (proces potomka - proces rodzica).



Rysunek 64: Najprostszy diagram Harela dla systemu BIS

2.6.2.3. Implementacja współbieżności: pierwsze podejście

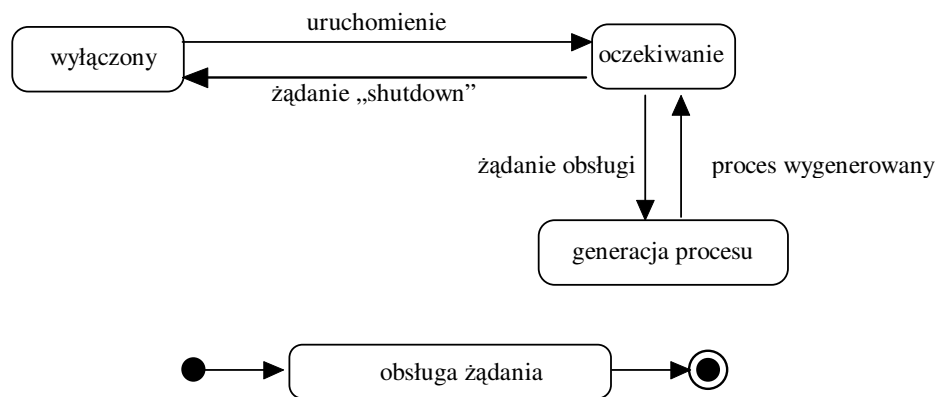
Opisane w poprzednim punkcie problemy próbowaliśmy rozwiązać modelując zachowanie systemu na niżej pokazanym diagramie. Okazuje się jednak, że model taki nie jest najlepszy - zwłaszcza modelowanie procesu obsługującego żądanie obsługi. Ostatecznie, zdecydowaliśmy się na model, w którym występują dwa niespójne (w sensie teorii grafów) diagramy Harela.



Rysunek 65: Współbieżność systemu BIS - wersja I

2.6.2.4. Implementacja współbieżności: drugie podejście

Diagram ukazuje sposób działania systemu z punktu widzenia obsługi współbieżności przy użyciu serwera WWW - po odebraniu „żądania użytkownika” na porcie 80 generowany jest proces, który przejmuje jego obsługę. Po jej zakończeniu, proces terminuje się.

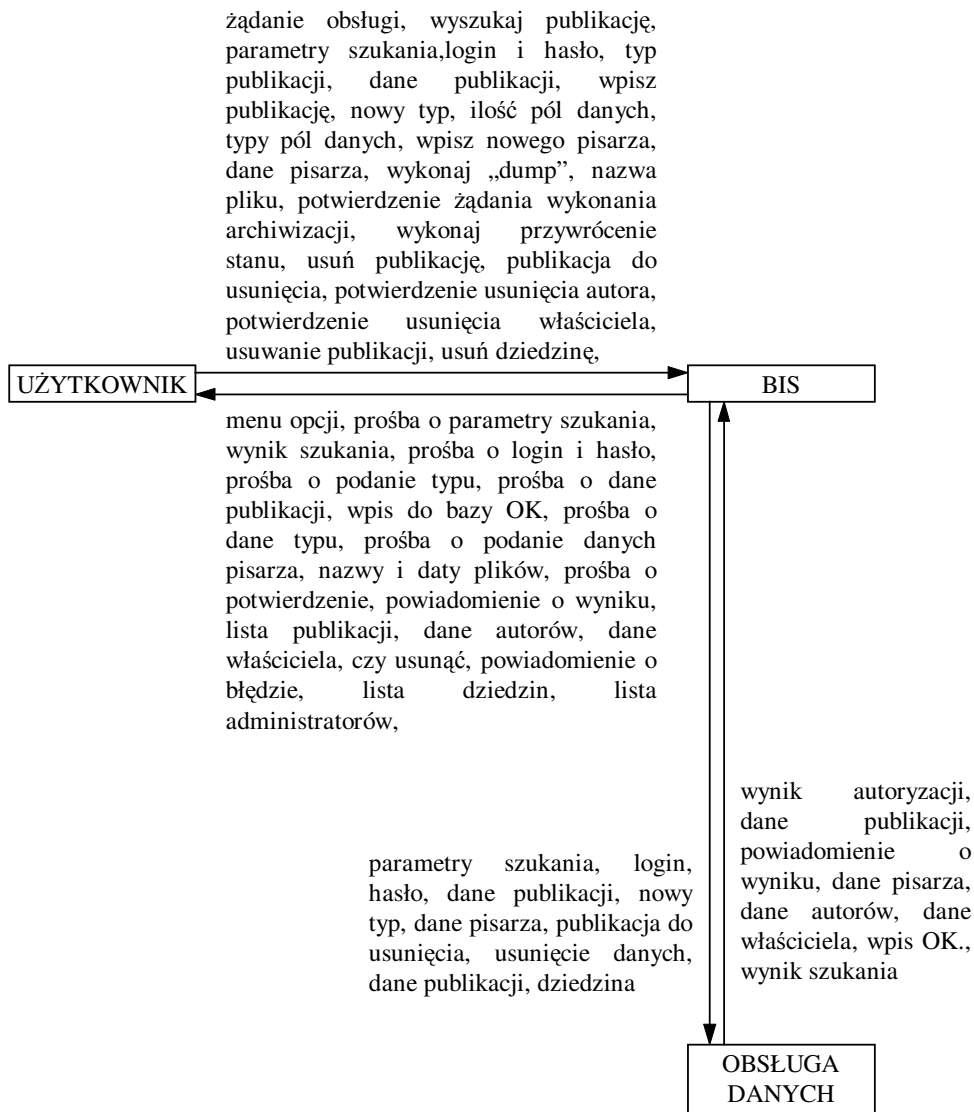


Rysunek 66: Współbieżność systemu BIS - wersja II

2.6.2.5. Diagram przepływu zdarzeń

Diagram ten powstał w oparciu o wcześniej pokazane scenariusze zdarzeń. Jest on, zgodnie z przyjętą metodologią modelowania (patrz [2], str. 175), krokiem pośrednim w celu wypracowania diagramów przejść stanów. Podobnie jak poprzednio, system podzielono na dwa komponenty. Pierwszy, nazwany dla uproszczenia BIS reprezentuje logikę aplikacji, drugi - Obsługa Danych - reprezentuje podsystem gromadzenia i przetwarzania danych (bazę danych). Ujęcie

takie ma uzasadnienie, ponieważ tylko podsystem BIS zachowuje się w sposób dynamiczny (i podlega modelowaniu dynamicznemu), podczas gdy podsystem bazy danych jest tu mniej istotny²⁸.



Rysunek 67: Diagram przepływu zdarzeń systemu BIS

Pełny model na diagramie przejść stanów nie mógłby być czytelny, dlatego rozbiliśmy go na kilka mniejszych komponentów. Poniżej zamieszczamy te części,

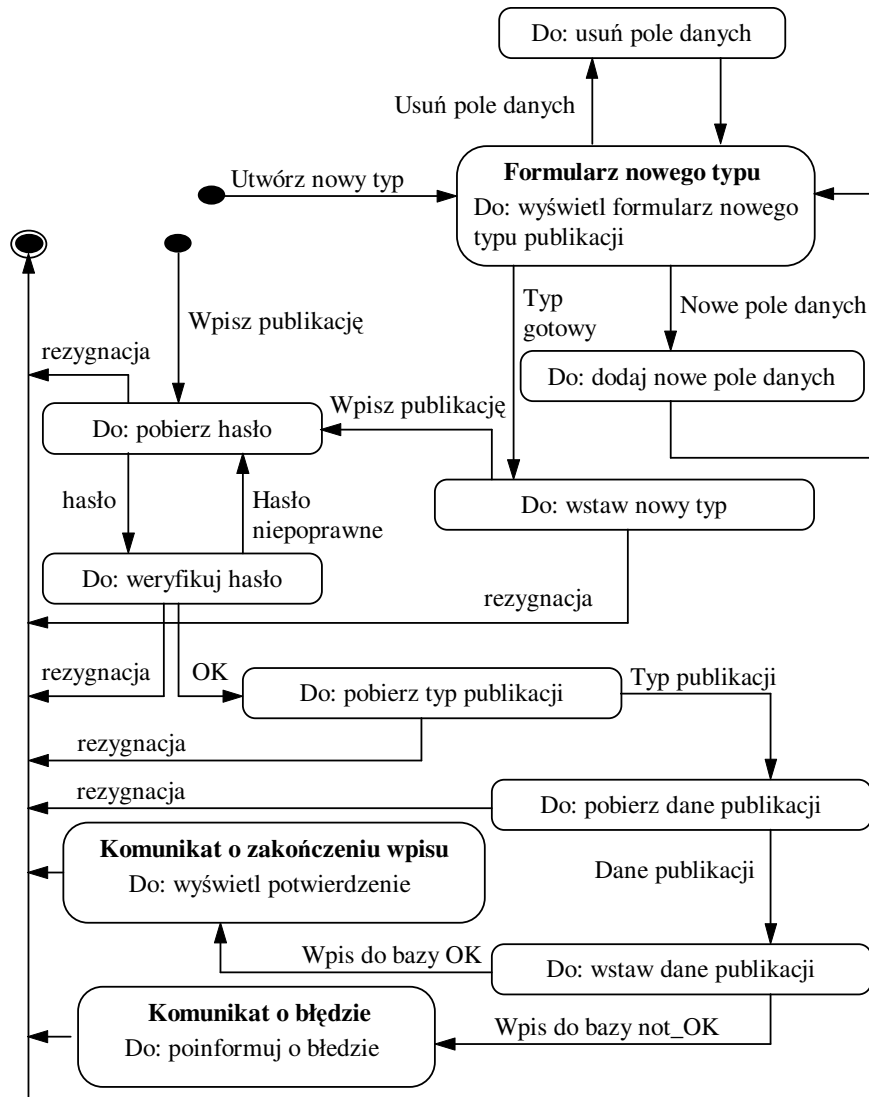
²⁸ W metodologii Rumbough bazy danych uznaje się za mało znaczące w modelu dynamicznym,

począwszy od najmniej skomplikowanych, aby na końcu pokazać ich uogólnienie na dużym diagramie.

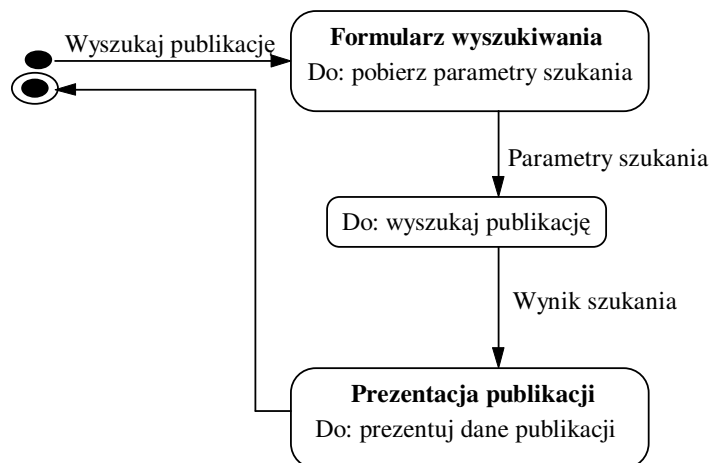
Pierwszy diagram obrazuje sposób, w jaki odbywa się wstawianie nowej publikacji do bazy danych. Przy tej operacji, pisarz może zażyczyć sobie utworzenia nowego typu publikacji. Realizowana jest wówczas procedura rozszerzona - ustalenie struktury danych tablicy, która będzie przechowywać dane nowych publikacji.

Kolejny diagram ukazuje sposób wyszukiwania przez czytelnika informacji w systemie. Jest to najprostsze, ale i najczęściej realizowane zadanie BIS (przewaga odczytów nad zapisami do bazy danych). Jest to jedyna operacja, przy której nie jest stosowana autoryzacja dostępu za pomocą sprawdzania loginu i hasła użytkownika.

Dodanie nowej publikacji polega na pobraniu od użytkownika jego danych do identyfikacji, a następnie na pobraniu (i sprawdzeniu pod względem poprawności) danych publikacji. Zweryfikowane dane są wstawiane do bazy danych.

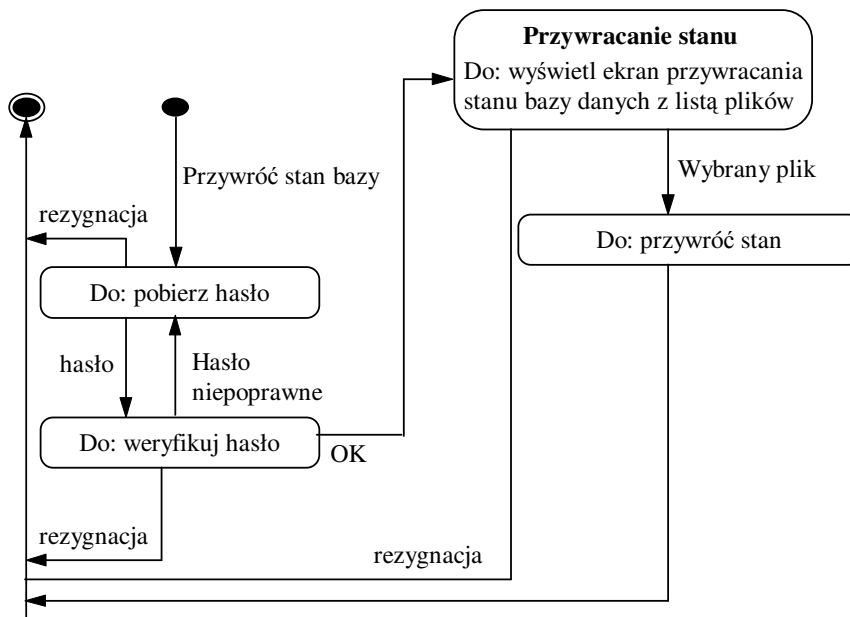


Rysunek 68: Diagram przejść stanów - wstawianie publikacji przez pisarza



Rysunek 69: Diagram przejść stanów - wyszukiwanie publikacji przez czytelnika

Kolejny diagram prezentuje sposób, w jaki odbywa się przywracanie stanu bazy danych z plików zawierających zarchiwizowany stan bazy.

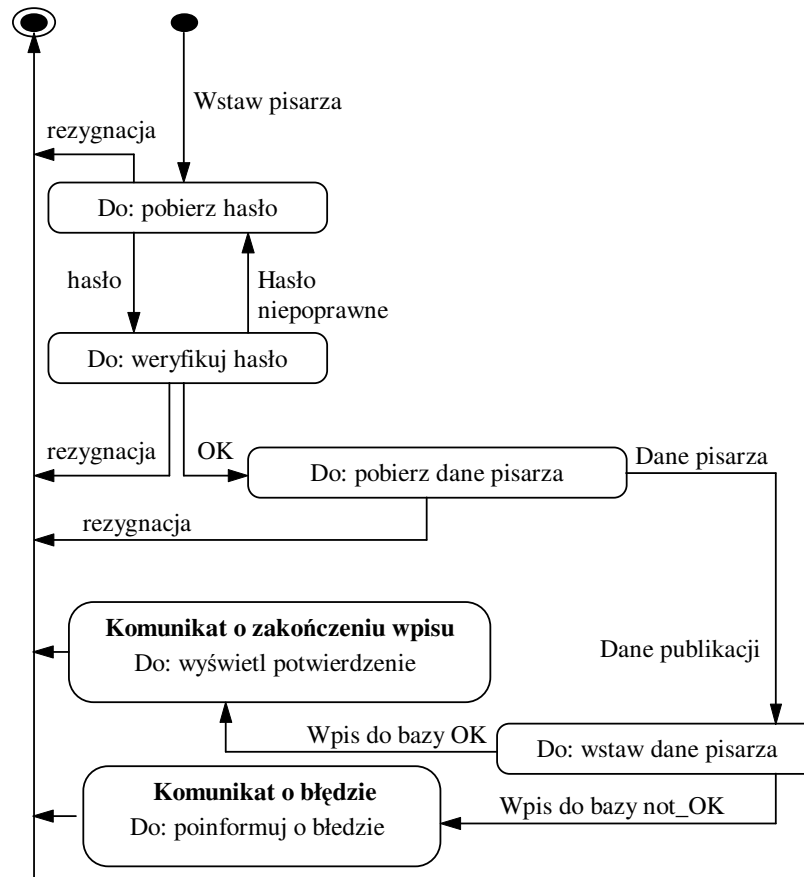


Rysunek 70: Diagram przejść stanów - przywracanie stanu bazy danych

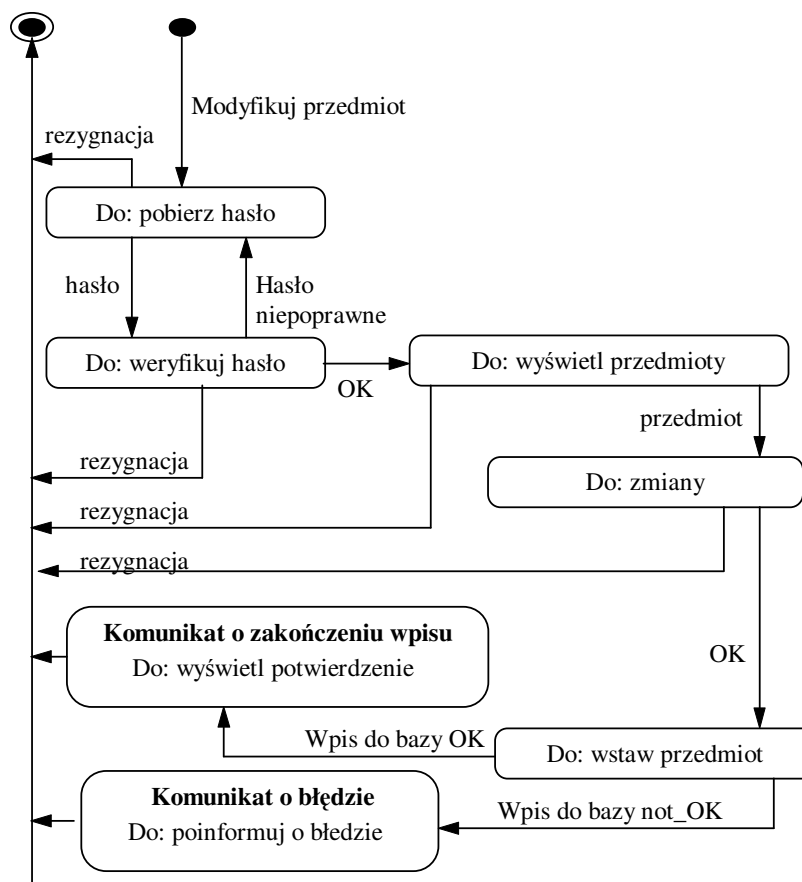
Wstawianie nowego pisarza (patrz rys. 54), to operacja, którą może wykonać jedynie administrator. Dzięki temu, że użytkownik-pisarz różni się od użytkownika-administratora jedynie prawami dostępu, możliwe jest zastosowanie tej samej procedury autoryzacji, co znacznie upraszcza kod aplikacji (implementacja autoryzacji w języku HTML nie jest zadaniem łatwym z uwagi na bardzo ograniczone możliwości języka - język HTML powstał jako język opisu strony, a nie język programowania²⁹). Kolejny diagram (rys. 55) opisuje rozwiązanie problemu archiwizacji. Na etapie analizy nie rozwiązywaliśmy jeszcze problemu dostępu współbieżnego w kontekście zachowywania i odtwarzania stanu bazy danych (nie znaliśmy jeszcze niektórych cech zastosowanego później oprogramowania). Opis zastosowanej metody ochrony spójności znajduje się w 4.2.7.

²⁹ W standardowym HTML nie ma zmiennych, instrukcji przypisania, iteracji, instrukcji warunkowych itp. Nie można także zmienić kolejności interpretowanych poleceń (tagów) np. instrukcją GOTO czy podobną.

Wstawianie nowej dziedziny nie różni się od wstawiania nowego słowa kluczowego i wstawiania przedmiotu, dlatego zastosowaliśmy tylko jeden diagram (patrz rys. 56) do opisu tych operacji. Wstawianie to polega na sprawdzeniu czy podane dane już nie istnieją w bazie danych. Jeśli nie, to wstawianie jest możliwe.



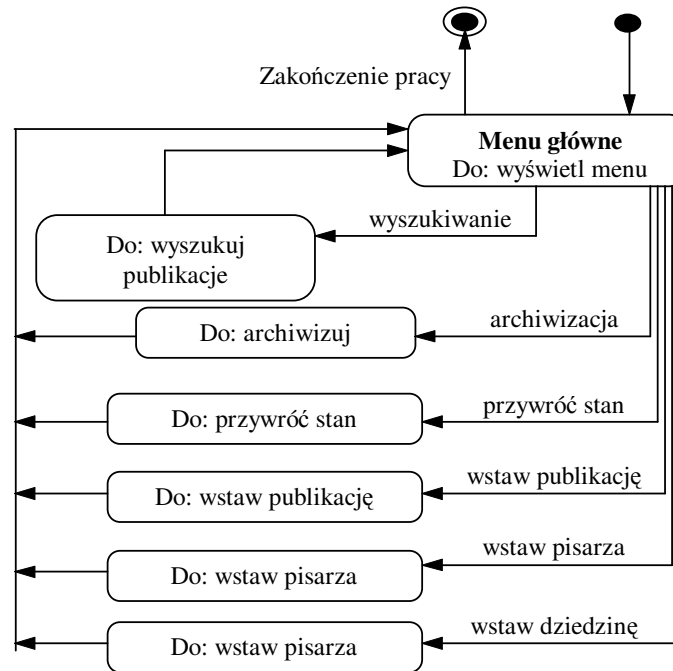
Rysunek 71: Diagram przejść stanów - wstawianie pisarza przez Administratora



Rysunek 74: Diagram przebiegu stanu - modyfikowanie przedmiotu

Ostatni diagram pokazuje sposób połączenia wszystkich operacji na jednym diagramie. Są one w jednakowy sposób wybierane z menu głównego (ekranu naszej aplikacji, widzianej w przeglądarce WWW). Diagram zawiera pewne uproszczenia (np. brak na nim operacji rozróżnienia wstawianie/kasowanie/modyfikacja - odbywają się one w zbliżony sposób³⁰, pominięto wstawianie przedmiotu i słowa kluczowego - są praktycznie identyczne z analogicznymi operacjami wykonywanymi na dziedzinach itp.), są one jednak uzasadnione, ponieważ nie rzutują one istotnie na wyniki analizy, a ich opuszczenie czyni diagram czytelniejszym.

³⁰ np. modyfikacja jest praktycznie równoznaczna ze wstawieniem - różnica polega jedynie na tym, że pole wprowadzania danych nie jest puste, a zainicjowane wartością początkową, a dane zostaną wstawione na miejsce już istniejących (zamiast nowego ID zostanie wybrany istniejący).

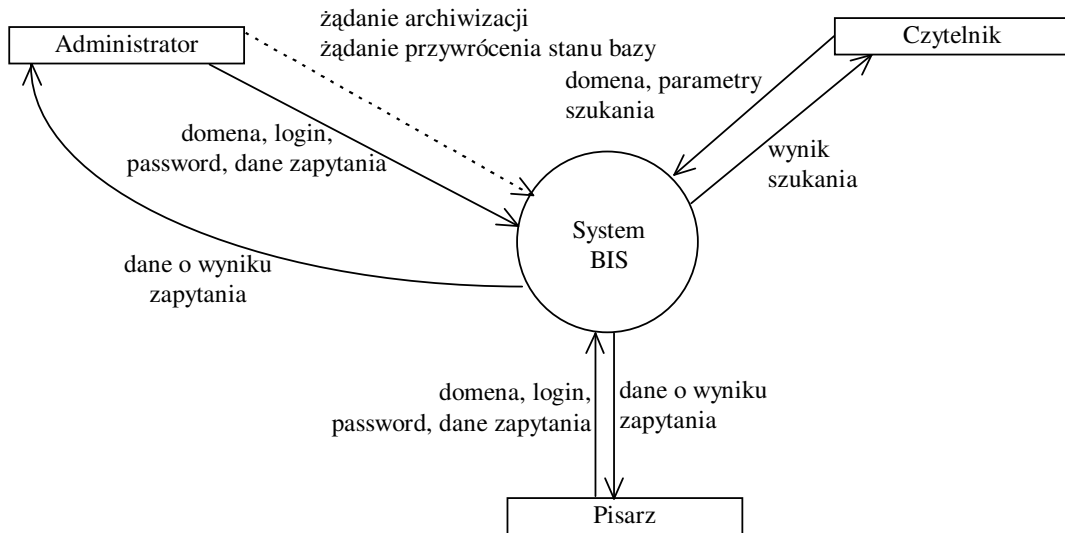


Rysunek 75: Diagram przejść stanów - poziom uogólniający

2.7. Model funkcjonalny

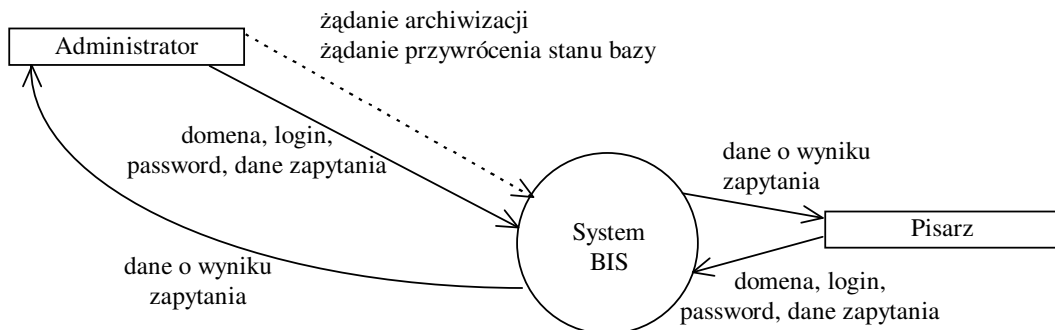
2.7.1. Identyfikacja danych wejściowych i wyjściowych

Diagram kontekstowy zawiera trzy terminatory: Administratora, Czytelnika i Pisarza. Rozróżnienie to jest trochę sztuczne, ponieważ np. Administrator może działać jako Czytelnik i Pisarz, ale przyjęcie takiego uproszczenia znacznie ułatwia modelowanie systemu. Pewnym uproszczeniem jest także przyjęcie ogólnych nazw „dane zapytania” i „dane wyniku zapytania”. Oznaczają one możliwość zadania różnych typów zapytań (przykład: „wyszukania publikacji o tytule rozpoczynającym się od słowa *modelowanie*”).



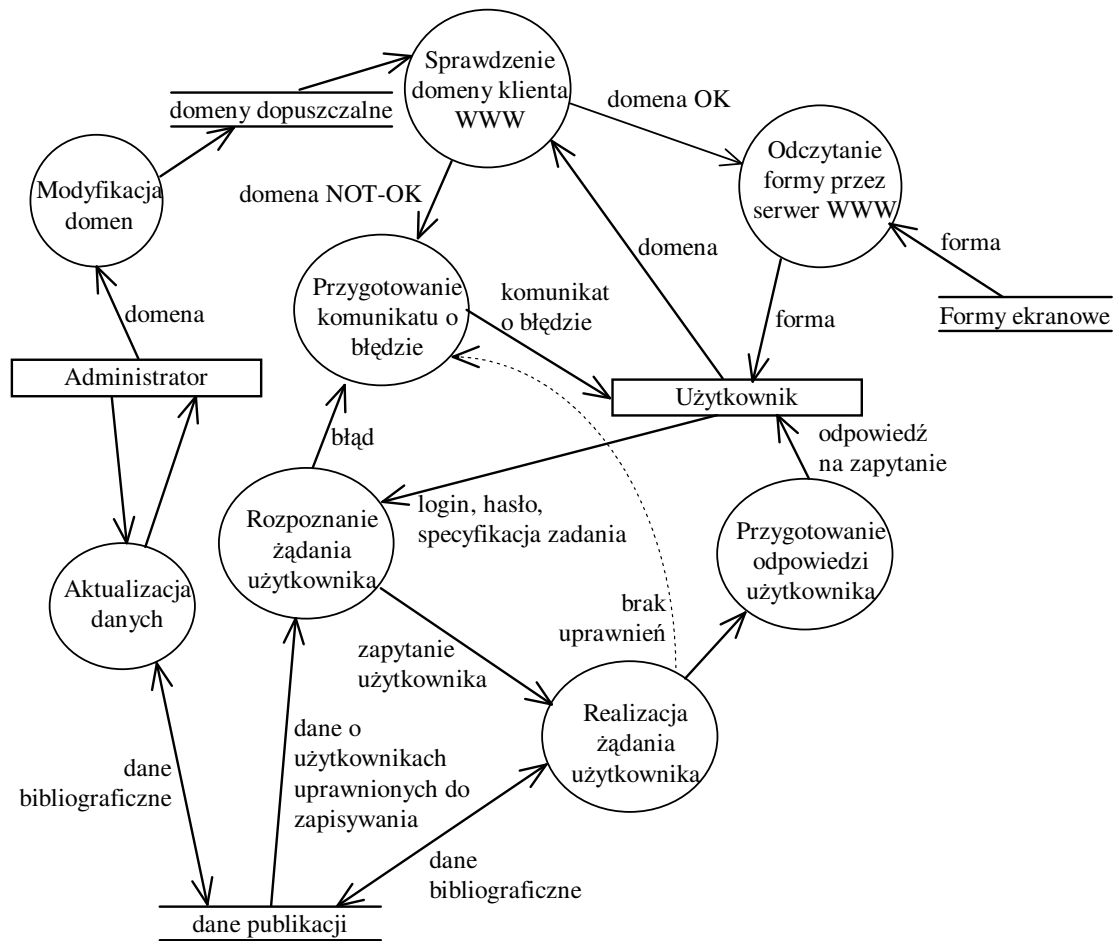
Rysunek 76: Identyfikacja danych wejściowych i wyjściowych (I)

Dla dalszych rozważań wygodnie połączyć terminatory Pisarza i Czytelnika w jeden obiekt o nazwie Użytkownik. Diagram przybierze wówczas następującą postać:



Rysunek 77: Identyfikacja danych wejściowych i wyjściowych (II)

2.8. Diagram przepływu danych



Rysunek 78: Diagram przepływu danych

2.8.1. Opis funkcji

2.8.1.1. Sprawdzenie domeny klienta WWW

Proces zajmuje się wstępną autoryzacją użytkownika (drugi etap przeprowadzany jest w procesie Rozpoznanie zadania użytkownika). Odpowiednia konfiguracja serwera WWW pozwala na automatyczne sprawdzanie dopuszczalnych domen użytkowników. Serwer ten porównuje dane określające domenę użytkownika ze spisem dopuszczalnych domen znajdującym się w pliku konfiguracyjnym serwera. Jeżeli domena użytkownika znajduje się na liście domen dopuszczalnych generuje sygnał sterujący domena_OK (w przeciwnym wypadku domena_NOTOK). Na podstawie uzyskanych informacji można podjąć decyzję o udostępnieniu (bądź nie) informacji z bazy danych.

2.8.1.2.Odczytanie formy przez serwer WWW

Proces zajmuje się udostępnieniem użytkownikowi stron systemu zawierających m.in. formy umożliwiającej konstruowanie zapytań. Dzięki tej formie możliwe jest przesłanie danych niezbędnych do pracy z systemem. Dane te w formie plików w formacie HTML będą znajdowały się w odpowiednim katalogu pod kontrolą serwera WWW.

2.8.1.3.Modyfikacja domen

Proces umożliwia zmianę (dodanie, usunięcie, modyfikacje istniejących) zawartości listy domen dopuszczalnych. Jego działanie polega na dostępie do pliku konfiguracyjnego serwera WWW i wprowadzeniu odpowiednich zmian. Ze względu na specyfikę operacji i wiarygodne sterowanie dostępem do systemu może ją wykonywać tylko administrator. W praktyce zmiana konfiguracji dopuszczalnych domen zwykle wymaga zrestartowania procesu serwera (uwaga odnosi się do oprogramowania: WWW Server Apache, wersja UNIX).

2.8.1.4.Przygotowanie komunikatu o błędzie

Proces w zależności od otrzymanych (od innych procesów - patrz diagram poziomu zerowego) komunikatów o błędach przesyła do użytkownika odpowiednią informację. Jego zawartość to: typ, opis i konsekwencje wystąpienia błędu. Opisana metoda zakłada całkowite generowanie strony z opisem błędu „w locie”, co pozwala na zrezygnowanie z konieczności tworzenia osobnych plików HTML (i zarządzania nimi). Niektóre serwery WWW pozwalają na własnoręczne definiowanie odpowiedzi (wyglądu stron pokazywanej użytkownikowi) jako reakcji na wystąpienie błędu.

2.8.1.5.Przygotowanie odpowiedzi dla użytkownika

Proces ten ma za zadanie przekształcenie odpowiedzi uzyskanej z bazy danych (wynik zapytania użytkownika) do postaci możliwej do przedstawienia użytkownikowi. Realizacją tego zadania zajmie się skrypt napisany w języku HTML z wykorzystaniem interfejsu API bazy danych, który będzie formatował uzyskiwane dane w czasie rzeczywistym i przesyłał je w odpowiedniej formie do klienta WWW po stronie użytkownika. Mechanizm tworzenia dynamicznego stron HTML został opisany w p. 1.5.5.

2.8.1.6. Aktualizacja danych

Proces umożliwia aktualizację danych znajdujących się w systemowej bazie danych. Uruchomienie tego procesu jest możliwe tylko przez administratora. Proces ten umożliwia administratorowi bezpośredni nadzór nad zawartością bazy danych poprzez funkcje dodawania, modyfikacji i usuwania danych pisarzy, zmiany danych dotyczących uprawnień do zapisu itd. Proces ten jest także odpowiedzialny za realizację archiwizacji i przewracania stanu bazy danych.

2.8.1.7. Rozpoznanie żądania użytkownika

Proces ten składa się z 3 etapów: sprawdzenie uprawnień użytkownika (w przypadku żądania dodania nowego rekordu do bazy), rozpoznania typu zadania (czy będzie to dodanie rekordu, czy użytkownik chce uzyskać informacje o publikacji, czy chce uzyskać pomoc) i przygotowania zapytania (przetworzenie danych z formy na zapytanie zrozumiałe dla bazy danych).

2.8.1.8. Realizacja żądania użytkownika

Zadaniem tego procesu jest realizacja zapytania użytkownika. Na podstawie zapytania użytkownika dokonuje się selekcji niezbędnych źródeł danych (tablic) po to, aby zminimalizować liczbęostępów do tablic. W przypadku wystąpienia błędu generowany jest odpowiedni komunikat. Żądanie użytkownika może być prośbą o wywołanie ekranu pomocy lub zapytaniem do bazy danych (odczyt lub modyfikacja)

2.9. Podsumowanie procesu analizy

Proces analizy pozwolił wyraźnie zdefiniować zakres systemu BIS. Określone zostały pożądane cechy systemu, wyspecyfikowane jego podstawowe funkcje i zachowania. Diagram modelu obiektowego stał się punktem wyjścia do implementacji bazy danych, jednak z uwagi na zastosowanie relacyjnej bazy danych konieczne stało się przejście z „obektowego” punktu widzenia świata na postać relacyjną (model ERD, normalizacja relacji itp. - zagadnienia te przybliży p. [3.6]).

W projekcie BIS już na etapie analizy zostały podjęte pewne decyzje projektowe - np. podział systemu na serwer WWW i serwer bazy danych. Było to uzasadnione, ponieważ wiadomo było, że tylko taka technologia zapewnia powodzenie naszego projektu. Wstępnie założyliśmy użycie bazy danych Informix z modułami Informix-CGI (baza danych Informix jest dostępna w Zakładzie Zastosowań Informatyki WETI PG, na której praca była tworzona, więc nie było

konieczności zakupu oprogramowania), jednak później, po zapoznaniu się z możliwościami motoru mSQL, zdecydowaliśmy się na zmianę oprogramowania (mSQL może być dostępny bezpłatnie dla każdej placówki edukacyjnej).

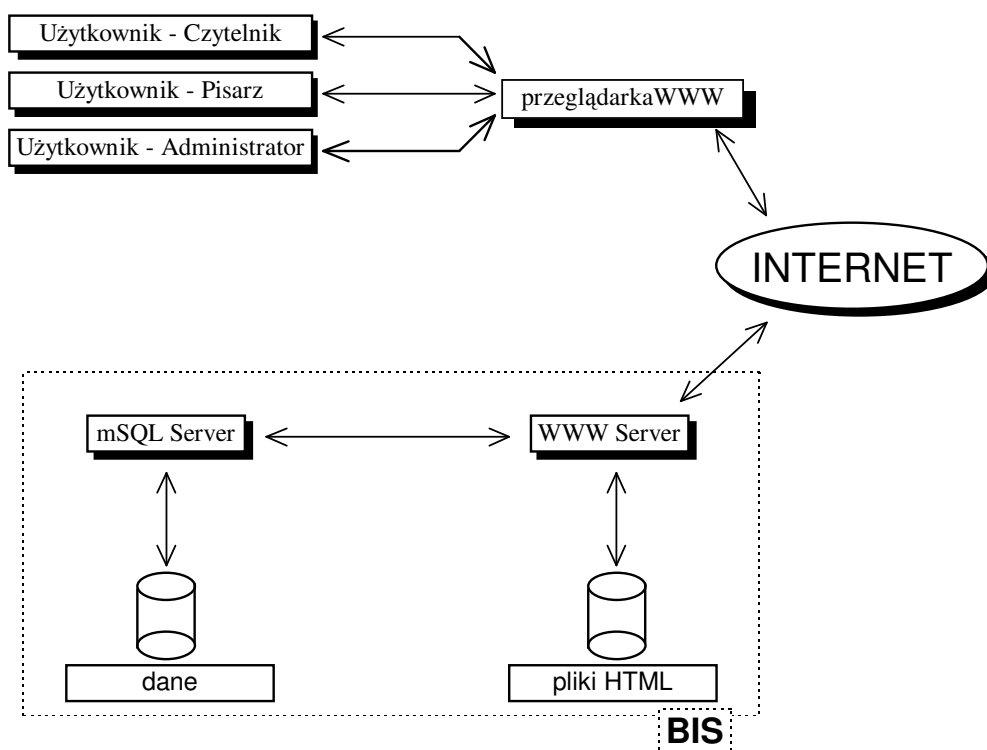
Proces analizy kilkakrotnie był iterowany, ponieważ często się okazywało, że np. pewne funkcje mogą być np. zastąpione innymi lub usunięte bez konieczności ograniczenia funkcjonalności systemu BIS. Bardzo ważnym elementem stało się zapewnianie modyfikowalności i elastyczności proponowanego przez nas rozwiązania - temu celowi służyło np. wydzielenie klasy grup użytkowników z klasy uprawnień.

W kontekście stosowanej metodologii, należy ocenić, że jej charakter wpływał w niewielkim stopniu na proces analizy. Przed podjęciem prac rozważaliśmy możliwość modelowania za pomocą metod strukturalnych, obawiając się, że relacyjny charakter bazy danych może w późniejszych etapach utrudnić proces projektowania (na bazie wniosków płynących z analizy obiektowej), jednak ostatecznie okazało się, że decyzja przyjęta przez nas - użycie obiektowej metodologii Rumbaugh - nie stwarzała problemów.

3. Projekt systemu BIS

3.1. Koncepcja systemu

Poniższy rysunek ukazuje koncepcję implementacji systemu BIS. Użytkownicy, korzystają ze swoich przeglądarek WWW (browserów). Mogą one pracować w dowolnym systemie operacyjnym (np. Unixowy Netscape Navigator lub MS Internet Explorer dla Windows). Przeglądarka komunikuje się z serwerem WWW przez sieć Internet, choć możliwa jest również komunikacja za pomocą sieci lokalnej (Intranet). Wymiana danych odbywa się za pomocą protokołu HTTP. Serwer WWW korzysta z zapisanych na dysku lokalnym plików HTML i podczas ich przetwarzania, po natrafieniu na zapytanie SQL, wykonuje je, odwołując się serwera bazy danych.



Rysunek 79: Koncepcja systemu BIS

Możliwy jest także podział systemu na dwie, fizycznie oddzielone części: serwer WWW i serwer bazy danych. Oba podsystemy mogą się wtedy komunikować za pomocą sieci lokalnej, a nawet Internetu (to ostatnie rozwiązanie niesie

oczywiście za sobą potencjalne niebezpieczeństwo - serwer bazy danych musi być otwarty na żądania przesyłane „z zewnątrz”).

3.2. Podział na podsystemy

Zakładając, że system BIS jest aplikacją działającą w przygotowanym środowisku (system operacyjny zapewniający m.in. obsługę protokołów sieciowych) można wyraźnie wyróżnić kilka modułów:

- **podsystem obsługi WWW**
 - **podsystem realizacji protokołu http** (oprogramowanie serwera WWW, realizujące nasłuch na porcie 80 i obsługujące przychodzące tam żądania zgodnie z protokołem HTTP 1.0. Dla celów systemu BIS zastosowano serwer Apache 1.2 beta 7, dostępny w sieci Internet jako oprogramowanie *freeware*)
 - **podsystem interpretacji języka opisu strony WWW**
 - **podsystem realizacji wołań HTML** (umożliwia realizację języka HTML w postaci HTML 3.0 - w wersji realizowanej przez oprogramowanie Netscape Navigator 3.0 i Microsoft Internet Explorer 3.0³¹)
 - **podsystem realizacji rozszerzonego HTML** (moduł PHP/FI³² umożliwia wzbogacenie języka HTML o dodatkowe instrukcje, m.in. pętle, zmienne, instrukcje warunkowe i iteracyjne, wywołania interfejsu API bazy danych mSQL. Z uwagi na zastosowaną technikę - rekompilacja kodu źródłowego serwera WWW prowadząca do rozbudowy wbudowanego parsera - rozwiązanie nie wymaga od użytkownika żadnych czynności dodatkowych, rozbudowujących jego przeglądarkę, jest dla klienta systemu BIS całkowicie „przezroczyste”)

³¹ Z formalnego punktu widzenia język HTML 3.0 nie istnieje w postaci standardu - dostępne jest jedynie forum dyskusyjne dotyczące proponowanych rozwiązań języka, organizowane przez twórców języka HTML 2.0 (język ten posiada zdefiniowany standard, opisany normą ISO). Jedyną „oficjalną” przeglądarką HTML 3.0 jest browser Arena. Przykładem rozbieżności różnych wersji HTML 3.0 (promowanych przez różne firmy, zwłaszcza Netscape i Microsoft) jest tag <BLINK> umożliwiający migotanie tekstu w przeglądarkach Netscape, a ignorowany przez IE w wersji 3.0.

³² Personal Home Page / Form Interface, © I.Rasmus (email rasmus@vex.net). Produkt licencjonowany jako freeware for non-commercial and educational use (ponadto uzyskaliśmy specjalną zgodę autora na użycie modułu w naszej pracy dyplomowej)

- **podsystem obsługi danych** (zrealizowany jako serwer relacyjnej bazy danych, mSQL 2.0 beta 10, posiadający interfejs w postaci oprogramowania monitorującego i służącego do wydawania zapytań typu command-line, a także wewnętrzny API, do użycia w języku C/C++)

3.3. Przydział podsystemów do procesorów

W przypadku systemu BIS punkt ten jest trudny do wykonania w sposób formalny, opisany w [2] za pomocą diagramu. Zamiast niego, można zastosować opis nieformalny: i tak, przyjmując rozszerzoną definicję procesora, jako element zasobów (w świetle takiego ujęcia procesorem może być zarówno procesor komputera, jak i człowiek - operator systemu, czy program wspomagający) można podać następujący przydział zadań. Uwagi:

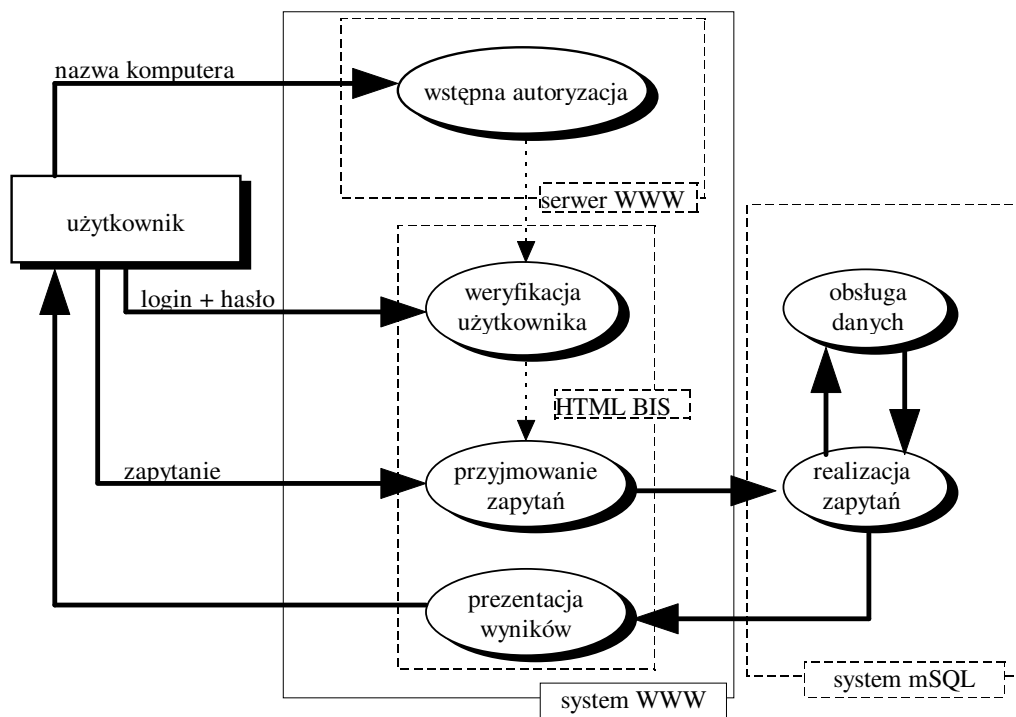
- dla uproszczenia, pod ogólną nazwą HTML-BIS rozumiemy podsystem stworzony na bazie funkcji phpfi, w postaci plików HTML, zajmujący się interfejsem i interpretacją danych wpisywanych przez użytkownika.
- System kontroli bezpieczeństwa jest dokładnie omówiony w p. 3.7

Zadanie	podsystem	opis i uzasadnienie
wstępna autoryzacja	jądro serwera WWW	kontrola dostępu oparta na analizie nazwy hosta jest możliwa za pomocą wbudowanych funkcji serwera WWW
weryfikacja użytkownika	HTML-BIS	kontrola dostępu login-password może być zaimplementowana z użyciem phpfi
przechowywanie danych	mSQL	rolę strażnika i managera danych pełni wydzielony podsystem serwera bazy danych
interfejs użytkownika	HTML-BIS	interfejs jest tworzony za pomocą języka HTML, dane przychodzące są odbierane za pomocą tego samego podsystemu, po czym przekazywane do podsystemu ich obsługi
realizacja łączności	serwer WWW	za pomocą protokołu HTTP 1.0

użytkownik-BIS		
----------------	--	--

Tabela 21: Przydział zadań w systemie BIS

W ten sposób, system BIS można przedstawić na poniższym diagramie:



Rysunek 80: System BIS

Powyższy rysunek przedstawia podział zadań pomiędzy podsystemy i ich przydział do procesorów dla przypadku realizacji zapytania przez pisarza lub Administratora. W przypadku Czytelnika nie jest przeprowadzana weryfikacja użytkownika za pomocą hasła i loginu - z diagramu można usunąć podsystem „weryfikacja użytkownika”.

3.4. Problem budowy interfejsu: Java, ActiveX, czy API-CGI ?

Rozwiązanie tego problemu ten to jedna z najważniejszych decyzji projektowych. W systemie WWW sposób konstrukcji interfejsu ma bowiem olbrzymie znaczenie - zależy od niego praktycznie cała koncepcja powiązania serwera WWW z bazą danych. System BIS został realizowany z użyciem interfejsu na „czystym HTML”. Wady i zalety tego rozwiązania przedstawia poniższe zestawienie:

- + możliwość prostego połączenia z bazą danych w oparciu o API serwera WWW i mSQL
- + prosta kompozycja dokumentu HTML - łatwość modyfikacji i rozbudowy
- + prosta generacja kodu (nie są potrzebne dodatkowe narzędzia np. w przypadku Javy - Java Developer Kit)
- + krótkie dokumenty - małe obciążenie łącz (łatwo komunikować się z odległych hostów)
- + pełna kompatybilność z przeglądarkami tekstowymi
- „mała interaktywność” menu - np. korzystając z Javy można wymusić reakcję interfejsu na ustawienie myszki nad określonym punktem ekranu (bez naciskania klawisza)
- nie można wymusić interaktywnej pomocy kontekstowej (realizowanej np. w oparciu o ActiveX - patrz www.vex.net/php)

Wykorzystanie Javy lub ActiveX (ostatnie wersje Netscape mają zapewnioną kompatybilność z komponentami ActiveX) jest na pierwszy rzut oka pożądane. Za pomocą odpowiednich narzędzi można stworzyć ciekawe w formie applety, zbudować aktywne menu itp. Okazuje się jednak że nie jest to technika często stosowana w systemach wyszukiwawczych (vide: Yahoo, Alta Vista, Wirtualna Polska). Powodem jest dodatkowy ruch w sieci, generowany przez programy Javy wysyłane do klienta. Nie należy także zapominać o istnieniu przeglądarek starszej generacji, nie posiadających możliwości odczytania appletu Javy. Zresztą, nawet „rozumiejące Javę” przeglądarki typu Netscape, czy IE bywają źle skonfigurowane, co wymaga dodatkowej pracy od użytkownika (uzyskanie dodatkowych modułów, konfiguracja środowiska itp.). Z tych powodów, w systemie BIS podjęliśmy decyzję o niestosowaniu Javy, ani techniki komponentów ActiveX³³. Bardzo istotny był tu profil potencjalnych odbiorców naszego oprogramowania - znaczący procent używa przeglądarek tekstowych lub starszych wersji browserów graficznych (np. Moscaic czy Netscape 2.0)³⁴. Na wyborze mechanizmu interfejsu zaważył także fakt, że w

³³ Próby wykorzystania techniki ActiveX podjęliśmy podczas pobytu w Niemczech, korzystając z doświadczeń tamtejszych programistów i bogatej literatury. Technologia ta wydaje się być bardzo zachęcająca, zwłaszcza w kontekście obsługi ActiveX przez najnowsze wersje Netscape.

³⁴ Powody takiego stanu rzeczy to słabe parametry techniczne komputerów (często są to maszyny 386DX, z niewielkim dyskiem i małą ilością pamięci) i brak czasu na sprowadzenie nowszych wersji programów

trakcie początkowych prac nad systemem Java nie była szeroko używana (np. pakiet programisty Javy JDK został zainstalowany dopiero niedawno w Zakładzie Zastosowań Informatyki). Nie było też szeroko dostępnych mechanizmów zapewniających współpracę z bazą danych (obecnie JDBC).

3.5. Zachowanie systemu w warunkach granicznych

3.5.1. Inicjalizacja

Proces inicjalizacji systemu powinien być przeprowadzany wg określonej kolejności: najpierw start procesu motoru bazy danych, następnie wypełnienie danymi (jeśli jest to konieczne - np. przy pierwszym uruchomieniu systemu), wreszcie - uruchomienie serwera WWW. Taka kolejność zabezpiecza przed niepożądanymi błędami wynikającymi np. z niemożności zrealizowania zapytania (przy próbie dostępu do plików HTML systemu BIS przy niedziałającym procesie serwera bazy danych pojawiają się komunikaty „mSQL error: cannot execute query”).

3.5.2. Shutdown

Wykonanie shutdown systemu obejmuje zastopowanie procesu serwera WWW, a następnie procesu bazy danych. Wykonanie tych czynności w odwrotnej kolejności może doprowadzić do błędów na stronach dokumentów HTML.

3.5.3. Sytuacje krytyczne

W trakcie prac nad systemem kilkakrotnie pojawił się problem z oprogramowaniem bazy danych - mSQL. Serwer awaryjnie kończył działanie w przypadku wysłania do niego pewnych błędnie sformułowanych zapytań. Prawdopodobnie jest to spowodowane użyciem wersji beta (w chwili gdy pisaliśmy naszą pracę pełna wersja nie była jeszcze gotowa).

W takiej sytuacji nie jest możliwe (lub jest skomplikowane implementacyjnie) automatyczne wykrycie upadku serwera msql. Prowadzi to do problemów z użytkowaniem systemu BIS - na ekranie użytkownika pojawiają się komunikaty „cannot execute query - cannot contact mSQL server”. W miejscach krytycznych (wpis do bazy danych, kasowanie rekordu) zastosowaliśmy mechanizm

przechwytywania informacji o błędnie wykonanym zapytaniu, ale umożliwia to tylko powiadomienie użytkownika o błędnej sytuacji - nie jest możliwa automatyczna naprawa z uwagi na brak systemowego logu (patrz też opis serwera mSQL). W przypadku wystąpienia takiej sytuacji krytycznej zaleca się ingerencję Administratora i ew. ponowne wypełnienie danymi (reload) bazy danych z plików zawierających archiwum.

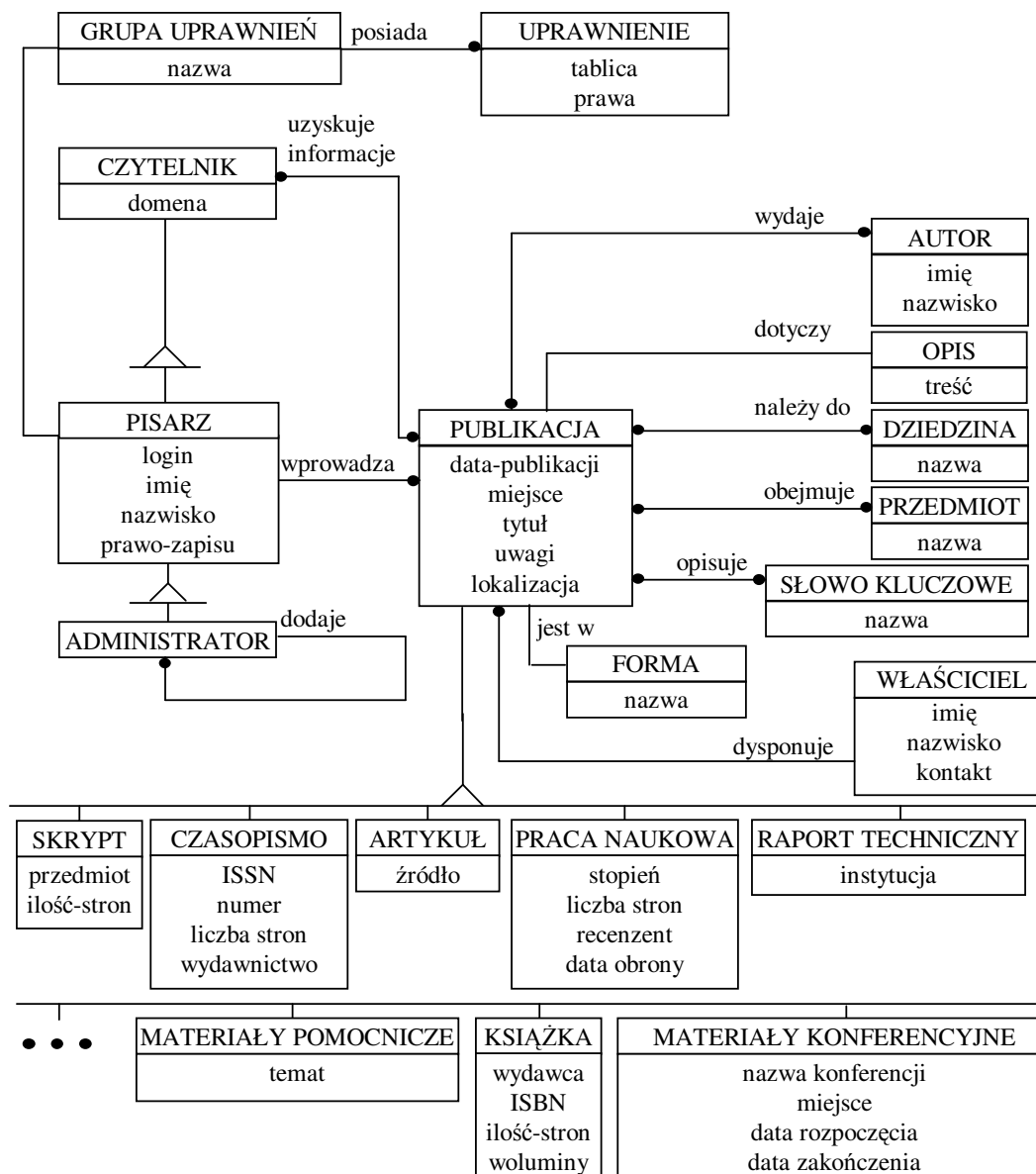
3.6. Model obiektowy a postać relacyjna

3.6.1. Uwagi wstępne

Zastosowanie modelowania obiektowego wymusiło zastosowanie określonych narzędzi modelowania. Wyniki tego procesu operują oczywiście pojęciami charakterystycznymi dla „obiektozoorientowanego” widzenia świata - takimi jak klasa, obiekt, czy związek. Specyfika naszego projektu i zastosowanie w nim relacyjnej bazy danych³⁵ (wiążącej się w pojęciami takimi jak tabela, czy encja) wymagało dopasowania obu punktów widzenia: relacyjnego i obiektowego. Punktem kluczowym okazało się przejście od diagramu obiektowego do formy schematu bazy danych. Proces ten rozłożyliśmy na kolejne etapy: stworzenia diagramu związków encji na bazie modelu obiektowego i budowy schematu bazy danych.

Punktem wyjścia do stworzenia modelu relacyjnego była ostateczna wersja modelu obiektowego (patrz rys. 58). Model relacyjny (diagram ERD) był krokiem w kierunku utworzenia schematu bazy danych - choć schemat ten w ostatecznej postaci różnił się od niego znacznie. Różnice te wynikały głównie z konieczności dostosowania schematu do planowanej możliwości rozbudowy struktury bazy danych w trakcie pracy systemu.

³⁵ Uzasadnienie tej decyzji znajduje się p. 3.2 i 3.11.3

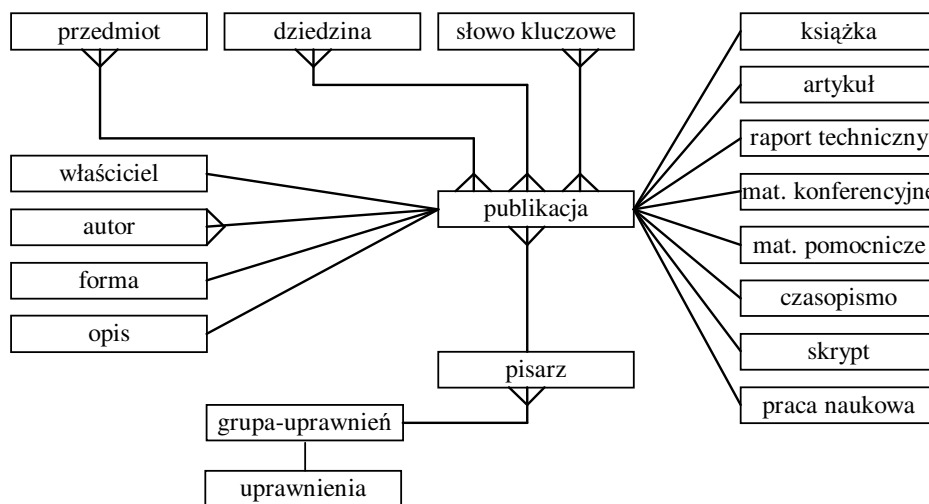


Rysunek 81: Ostateczny model obiektowy

3.6.2. Diagram ERD

Poniżej prezentujemy „pośredni” model ERD. Powstał on jako prosta „relacyjna” wersja modelu obiektowego - główne różnice to brak encji odzwierciedlających klasy *Administrator* i *Czytelnik*. Przyczyną braku klasy *Czytelnik* jest fakt, że nie jest konieczne trzymanie informacji o czytelnikach w bazie danych (tym bardziej że praktycznie nie jest możliwe uzyskanie informacji o loginie użytkownika korzystającego z systemu). Warto także zauważyć, że jako osobna encja nie występują tu *Dziedziny dopuszczalne* - ew. wpisy odbywają się do plików konfiguracyjnych serwera WWW. Klasa *Administrator* znalazła odbicie w klasie

Pisarz - diagram ERD nie modeluje metod obiektów, a jest to jedyna istotna cecha różniąca administratora od pisarza. Model jest uproszczony - nie zawiera atrybutów klas, pominięto także opis relacji z uwagi na trudności z rozmieszczeniem danych na rysunku. Model ten był zresztą tylko krokiem pośrednim w celu wypracowania schematu bazy danych (patrz niżej - p.3.6.3).



3.6.3. Problem zmiany struktury bazy danych w trakcie działania systemu

Kolejnym etapem miało być tworzenie schematu bazy danych. Po jego stworzeniu na podstawie ERD pojawił się jednak kolejny problem. Zgodnie z nowopowstałą ideą, chcieliśmy umożliwić tworzenie nie tylko nowych publikacji, ale także ich typów. Sytuacja taka wydaje się naturalna - pisząc system BIS mogliśmy przecież przeoczyć ważne w konkretnym zastosowaniu typy publikacji. Przykładem może być np. specjalistyczne zastosowanie systemu BIS do przechowywania danych o publikacjach muzycznych, które charakteryzują się określonymi specjalnymi właściwościami (np. obok klasy *Autor* powinna wystąpić klasa *Wykonawca* lub *Kompozytor*). Zadanie takie jest bardzo ambitne - nie jest nam znany do tej pory ani jeden system WWW realizujący zmianę struktury bazy danych podczas pracy systemu. Nawet największe na świecie systemy wyszukiwania danych jak Yahoo, jeśli tworzona jest nowa „jakość” w ich bazach danych (nowa kategoria tematyczna) nie tworzą nowych tablic - użytkownik musi dostosować charakter swoich danych do gotowych form. Ostatecznie, okazało się, że nawet przy zastosowaniu motoru bazy

danych o tak ograniczonych możliwościach, jak stosowany przez nas mSQL istnieje możliwość takiej implementacji, która zapewnia realizację postawionego zadania. W tym celu system musi:

- mieć możliwość utworzenia nowych tablic
- umieć określić, jakie tablice są aktualnie dostępne w bazie danych
- wczytać nazwy kolumn każdej tablicy z bazy danych

Wymaga to nowej implementacji schematu bazy danych, a ograniczone możliwości systemowe motoru bazy danych³⁶ wymagały szczególnie uważnego projektowania. W rezultacie, powstał schemat podany niżej. Nowa tablica SysTableName ma symulować tablice systemowe dużych baz danych. Są w niej przechowywane nazwy i nazwy symboliczne wszystkich tabel bazy danych. Np. encji Pisarz z modelu ERD (którą początkowo implementowała tablica Writers³⁷), teraz odpowiada tablica T2 i krotka (3,T2,"Pisarze") w tablicy SysTableName. W schemacie występują także tablice łączące (T12, T14, T16 i T18), stanowiące realizację związków wiele-do-wiele z modelu ERD.

SysTableName (ID, code, name)

T1 (ID, tytuł, rok_wyd, miejsce_wyd, lokal, ID_typu, ID_pisarza, ID_Formy, ID_Wlasc, ID_Opisu)

T2 (ID, login, passwd, ID_Upowaz, ID_wprow)

T3 (ID_U, imie, nazwisko, ulica, nr_domu, nr_m, kod, miasto, kraj, email, tel, fax)

T4 (ID, poziom)

T5 (ID, nazwa, twórca)

T6 (ID, rodzaj)

T7 (ID, ISBN, wyd, l_str, wolumin, ID_Pub)

T8 (ID, ISSN, numer, l_str, wyd, ID_Rodz, ID_Pub)

T9 (ID, imie, nazwisko, kontakt)

T10 (ID, nazwa)

³⁶ W tym miejscu warto zaznaczyć, że podczas pobytu autorów systemu BIS na praktyce w Niemczech, podobny problem bezskutecznie próbowali rozwiązać członkowie projektu DARIF (zarządzanie informacją w przedsiębiorstwach w oparciu o bazy danych i WWW), pracujący na zaawansowanym motorze MS SQL 6.5 (wersja dla Windows NT Server 4.0)

³⁷ Ze względów czysto implementacyjnych (krótsze nazwy, brak polskich liter) w bazie danych tablice nazywaliśmy słowami wziętymi z języka angielskiego

- T11 (ID, imie, nazwisko)
- T12 (ID_A, ID_P)
- T13 (ID, nazwa)
- T14 (ID_D, ID_P)
- T15 (ID, nazwa)
- T16 (ID_S, ID_P)
- T17 (ID, nazwa)
- T18 (ID_Prz, ID_P)
- T19 (ID, treść)
- T20 (ID, ID_G, ID_TB, R, W, D, U)

Podsumowując ten etap należy zwrócić uwagę, jak wielkie trudności sprawia realizacja założonej idei. Należy opracować strategię kontroli spójności danych, zaprojektować mechanizmy zarządzania tabelami o nieznanym strukturze, nadawania do nich uprawnień itp. Bardzo trudne jest również zbudowanie interfejsu HTML, który potrafiłby dynamicznie dostosować się do tablicy o niezdefiniowanej liczbie i typach kolumn - problem ten jest szerzej omówiony dalej. Sposób rozwiązania tego zagadnienia w projekcie BIS, w którym zastosowano minimalne zasoby (z uwagi na ograniczenia kosztów) wydaje się być dużą zaletą naszego produktu. Implementacja skryptu tworzącego bazę danych podana jest w dodatku.

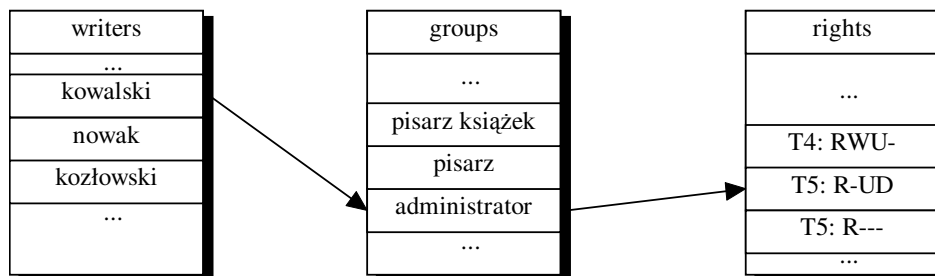
3.7. Strategia zapewniania bezpieczeństwa

W systemie BIS krytyczny punkt to kontrola dostępu pisarzy i administratorów. System musi rozpoznawać użytkowników, mieć możliwość sprawdzania uprawnień i zezwalać tylko na wykonanie tych czynności, do których prawa są udzielone. Jest to realizowane za pomocą dwustopniowej kontroli dostępu - przy użyciu funkcji wbudowanych serwera WWW i zaimplementowanej bazy danych użytkowników. Baza ta stanowi część bazy BIS. Określanie praw użytkowników odbywa się poprzez tablice Writers, Rights i Groups³⁸. Użytkownicy (ich loginy i hasła) są pamiętani w tablicy Writers, natomiast ich przynależność do grup

³⁸ Właściwie - w tablicach T2, T4 i T12, które są reprezentacją umownych tablic Writers, Rights i Groups w bazie danych.

użytkowników - w tablicy Groups. Tablica Rights określa prawa poszczególnych grup do określonych tablic.

I tak, wyznaczenie uprawnień użytkownika o loginie „kowalski” do tablicy „T5” odbywa się następująco:



Rysunek 82: Określanie praw użytkownika w systemie BIS

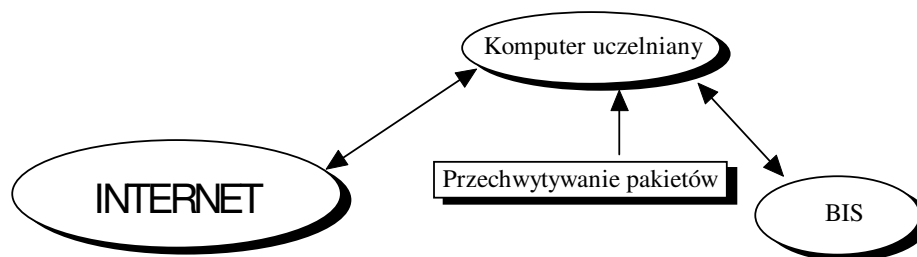
Taka implementacja daje możliwość elastycznego nadawania praw i pozwala na prostą modyfikację systemu BIS - np. w przypadku rozbudowy o nową grupę pisarzy, uprawnioną tylko do jednej z tablic. Grupa taka mogłaby np. zawierać pracowników dużej biblioteki, zajmujących się działem czasopism.

Do kontroli dostępu, wykorzystane są także mechanizmy systemowe dostarczane w środowisku Unix - nadawania praw do katalogów i programów. Można dzięki temu ograniczyć dostęp osób nieupoważnionych do plików HTML. Zabezpieczenie to uniemożliwia ingerencję w te pliki, dzięki czemu możliwe byłoby „włamanie” do bazy danych (wystarczyłaby zmiana pliku zajmującego się wczytywaniem haseł użytkowników, np. tak, aby hasła te przekazywać do określonego pliku).

Potencjalne niebezpieczeństwo kryje się jeszcze w jednym miejscu: w medium transmisyjnym. Teoretycznie jest możliwe przechwytywanie krążących w sieci Internet pakietów i analizowanie ich danych. Co prawda, taki atak jest raczej mało prawdopodobny, ponieważ wymaga posiadania praw pozwalających na zainstalowanie programu podsłuchującego jeden z węzłów sieci. Warto jednak wspomnieć o tej możliwości - mimo, iż tworzony przez nas system bibliograficzny nie jest programem przechowującym dane specjalnego znaczenia.

Z istotnym zagrożeniem mamy do czynienia w przypadku odkrycia haseł uprawniających do korzystania z systemu, w którym pracuje BIS lub haseł poszczególnych użytkowników BIS. Oprogramowanie takie może być zainstalowane

na jednym z komputerów, dzięki którym system BIS łączy się z Internetem - np. na komputerze uczelnianym:



Rysunek 83: Przykład konfiguracji umożliwiającej podsłuch pakietów (atak typu „net sniffing”)

Standardowe mechanizmy transmisji danych nie zabezpieczają danych w sieci - np. hasła używane w do autoryzacji użytkowników rozpoczynających sesję telnet przesyłane są w postaci niezakodowanej (!). Także i komunikacja serwer WWW - przeglądarka (przesyłanie danych z formy „wpisz hasło”) nie zapewnia kodowania danych.

Istnieje jednak droga do zabezpieczenia się przed „podsłuchem sieci”. Przede wszystkim standardową metodą stosowaną w takich sytuacjach jest niedopuszczenie do zdalnego logowania użytkownika root³⁹. Realizuje się to przez usunięcie nazw terminali zdalnych z pliku `/etc/securetty`. Po wtóre, instaluje się tzw. system shadow-password, co uniemożliwia odczytanie pliku `/etc/passwd`, w którym przechowywane są (w postaci zakodowanej) hasła użytkowników systemu. Warto także zablokować możliwość wykonywania „z zewnątrz” komendy `finger`, przez co utrudni się dostęp do loginów użytkowników, co utrudnia klasyczny atak typu „brute force”⁴⁰. Aby uniknąć przechwycenia haseł użytkowników systemu operacyjnego, należy także zainstalować w systemie tzw. secure shell (`ssh`⁴¹), w którym hasła przekazuje się przez sieć dopiero po zakodowaniu.

³⁹ Hasło tego użytkownika jest oczywiście najbardziej poszukiwane przez „program podsłuchujący”

⁴⁰ Polega on na uzyskaniu loginu jednego z użytkowników, a następnie systematycznym próbowaniu haseł generowanych losowo, lub na podstawie słowników najczęściej używanych haseł. Okazuje się, że taki „naiwny” mechanizm stosunkowo często prowadzi do uzyskania praw dostępu do komputera, zwłaszcza w systemach, w których użytkownicy nie dbają o odpowiednią formę haseł (litery małe i wielkie, stosowanie znaków specjalnych i liczb).

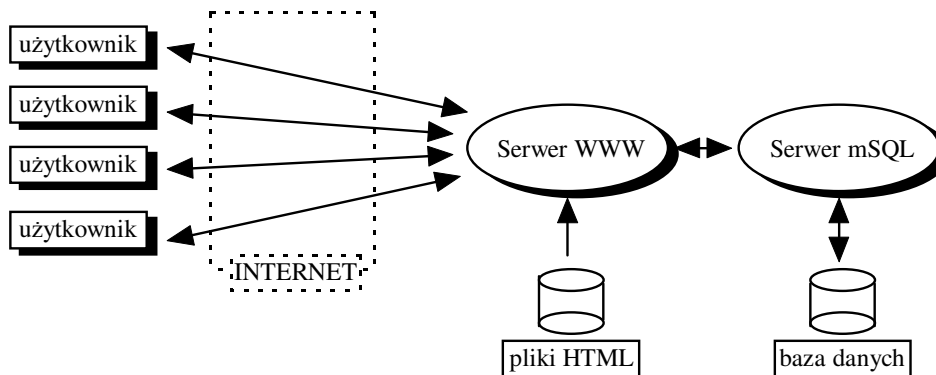
⁴¹ Secure shell umożliwia sprawdzenie użytkownika za pomocą jednej z metod: password (Knowledge of password), securid (SecurID authentication), skey (S/KEY one-time passwords), opie (NRL OPIE one-time passwords), publickeytest (Tests whether a key is acceptable), publickey (Possession of private key), hostbased (Identity of client host and user), kerberos4 (Kerberos v4 authentication),

Aby zabezpieczyć kanał transmisji przeglądarka - serwer WWW (wpisywane przez użytkownika hasło jest przekazywane przez sieć jak zwykły tekst) należy zainstalować oprogramowanie wykorzystujące tzw. SSL - Secure Socket Layer.

3.8. Współbieżny dostęp do bazy danych

3.8.1. Określenie punktów systemu, w których istnieje dostęp współbieżny

Sprawa współbieżnego dostępu do bazy danych jedną z najważniejszych kwestii projektowych, którą należało rozwiązać. Problem ten mógł być analizowany dopiero po określeniu granic podsystemów systemu BIS, zaprojektowaniu mechanizmów komunikacji między nimi i ich funkcji:



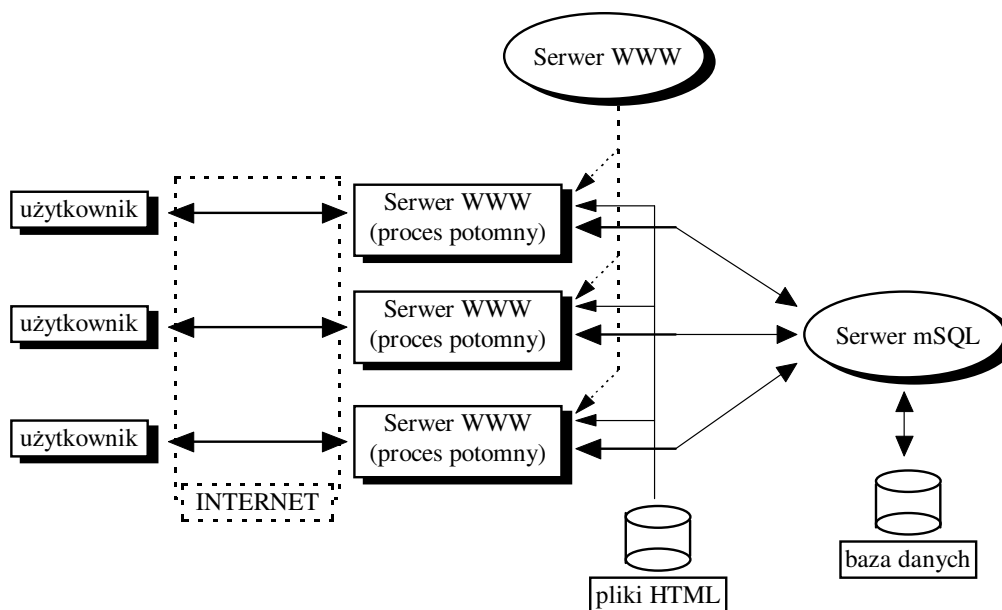
Rysunek 84: Współbieżność w systemie BIS - ujęcie ogólne

Rysunek powyżej przedstawia ogólną koncepcję systemu. W rzeczywistości, istotne jest, w jaki sposób realizowana jest obsługa współbieżnych wołań do serwera WWW. Zastosowany w implementacji serwera Apache realizuje każde wołanie użytkownika za pomocą osobnego procesu. Po odebraniu żądania strony HTML na porcie 80, proces główny serwera, sprawdza, czy istnieją wolne procesy potomne, które mogą obsłużyć wołanie⁴² i jeśli to konieczne - tworzy nowy taki proces. Każdy

kerberos5 (Kerberos v5 authentication), kerberos-afs (AFS Kerberos authentication). Dokładniejsze informacje na ten temat można znaleźć w [1].

⁴² Standardowo, po uruchomieniu serwera generowanych jest pięć procesów. Jest to wartość, którą można konfigurować za pomocą dyrektywy `START_SERVERS` - np. w przypadku systemów o niewielkiej liczbie odwołań współbieżnych liczbę aktywnych procesów można zmienić na 2-3. W sytuacji, gdy kolejny ($MAX+1$) użytkownik żąda strony WWW, kreowany jest dodatkowy proces do jego obsługi. System serwera WWW dba o to, aby niepotrzebne procesy ulegały terminacji, a liczb aktywnych była możliwie bliska ustalonej. Aby uniknąć przeładowania systemu i możliwego wyczerpania się zasobów logicznych (identyfikatory, wydajność itp.) określa się (za pomocą

z potomków ma autonomiczny dostęp do serwera mSQL - sytuację tą prezentuje kolejny rysunek.



Rysunek 85: Współbieżność w systemie BIS - działanie serwera WWW

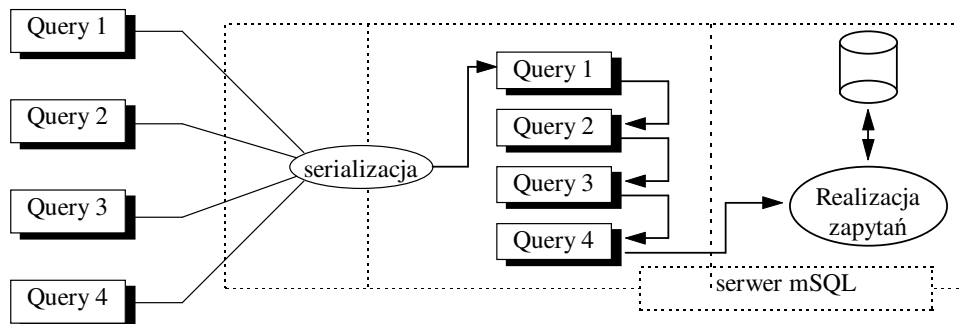
Rywalizacja o dostęp do plików HTML przechowywanych w systemie plików Unix jest rozwiązywana na poziomie systemu operacyjnego.

Każdy serwer WWW (proces potomny) może w dowolnej chwili zgłosić zapytanie do serwera mSQL. Niestety, w chwili obecnej oprogramowanie to nie jest w stanie obsłużyć zapytań w pełni współbieżnie - są one serializowane⁴³. Sytuacja taka oznacza, że jeśli dwóch użytkowników wykonuje zapytania (nawet takie, które mogą być współbieżnie wykonane bez konieczności nakładania rygli) to jeśli pierwsze zapytanie wykonuje się stosunkowo długo (przykład: skomplikowane złączenie wielu tablic), drugie będzie zatrzymane do czasu jego ukończenia. Powstaje w tym miejscu pytanie o sensowność wykorzystania motoru mSQL do tworzenia bazy danych. Za jego użyciem przemawia oprócz względów ekonomicznych, także fakt, że firma Hughes zapowiada w najbliższej przyszłości

dyrektywy MAX_SERVERS) maksymalną liczbę procesów, które mogą działać współbieżnie - po jej przekroczeniu każdy kolejny użytkownik uzyska odpowiedni komunikat („Cannot connect to server - server overloaded” -lub podobny), aż do zmniejszenia się liczby aktywnych procesów. Poza bardzo obciążonymi serwerami wyszukiwania sytuacja ta występuje bardzo rzadko.

⁴³ Autorzy serwera mSQL (Hughes Software Inc.) obiecują rozwiązanie tego problemu w najbliższych wersjach oprogramowania (patrz uwagi w p. 6.1)

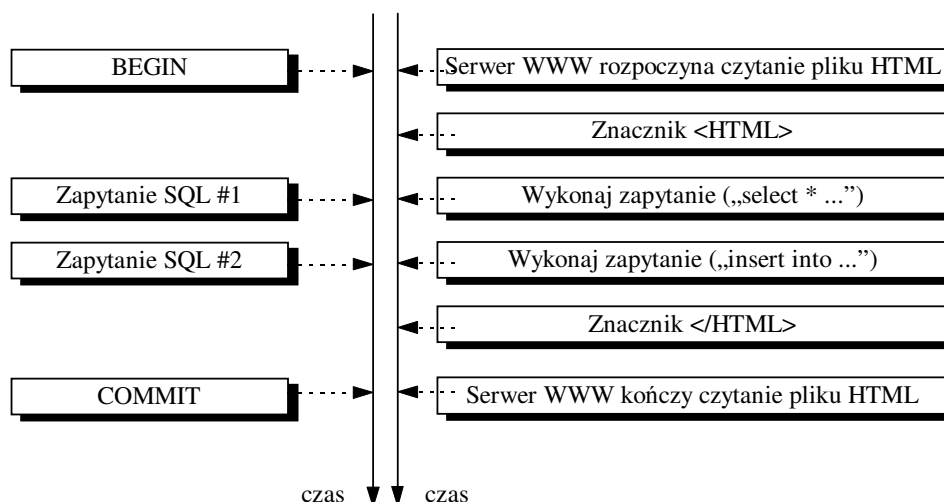
wyprodukowanie kolejnej wersji, wzbogaconej o współbieżne realizowanie zapytań i mechanizm transakcji (BEGIN, COMMIT i ROLLBACK).



Rysunek 86: Współbieżność w systemie BIS - działanie serwera bazy danych

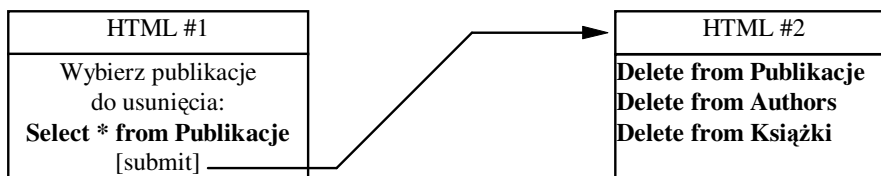
3.8.2. Charakterystyka systemu - specyfika zapytań realizowanych ze stron WWW.

Wybór serwera mSQL jako bazy danych systemu BIS niósł więc za sobą określone konsekwencje: należało zaimplementować odpowiedni sposób zabezpieczeń, chroniących spójność danych przechowywanych w bazie. Przede wszystkim, przeanalizowaliśmy z punktu widzenia naszego systemu, grożące niebezpieczeństwa: problem utraconej aktualizacji i niespójnej analizy. Problemy te rozpatruje się w kontekście transakcji, a choć de facto takie w naszym systemie nie występują (patrz uwagi p. 3.11.3 i 6.2), to stosunkowo podobne do nich są realizacje wielu zapytań znajdujących się na jednej stronie HTML. Oczywiście podobieństwo to można rozpatrywać jedynie w kategorii tych samych problemów, ponieważ realizacje stron HTML **nie są** transakcjami (w szczególności nie spełniają warunku atomowości - jedno z zapytań na stronie może się wykonać, podczas gdy kolejne - nie):



Rysunek 87: Porównanie transakcji z realizacją zapytań osadzonych w pliku HTML

Kolejną cechą naszego systemu jest specyfika czasowa zapytań. Zwykle, w systemach baz danych większość działań użytkownika wykonywanych jest w stosunkowo krótkim czasie. W systemie BIS mamy często do czynienia z sytuacją, w której odwołania do bazy danych realizujących jedną czynność występują na dwóch stronach WWW:



Rysunek 88: Odwołanie do bazy danych z użyciem dwóch stron (plików) HTML

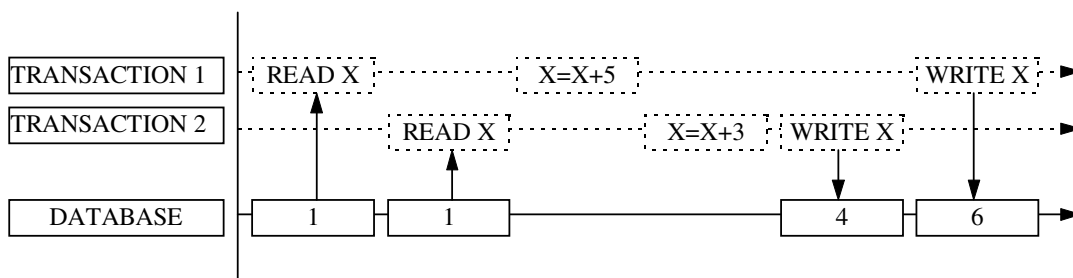
W przykładzie powyżej na jednej ze stron generujemy (za pomocą polecenia select, które odczyta dane i wstawi je do przygotowanej tabeli, przyporządkując do odpowiednio nazwanych przycisków wyboru⁴⁴) spis publikacji. Użytkownik może wybrać publikację do usunięcia i po naciśnięciu przycisku submit spowodować przejście do następnej strony. Pojawia się tu poważne niebezpieczeństwo: w czasie wypełniania formy znajdującej się na stronie #1, warunki panujące w sieci mogą ulec zmianie i załadowanie strony #2 może się bardzo opóźnić - w szczególnym

przypadku może się nawet okazać niemożliwe (utrata łączności z serwerem). Zagrożenie może spowodować również sam użytkownik, jeśli np. zrezygnuje z dalszego użytkowania systemu BIS po wypełnieniu formy na stronie #1 a przed załadowaniem strony #2. Jeśli obie formy realizują wspólnie jakąś czynność, mogą wystąpić nieprzewidziane efekty, które można porównać do przerwania w połowie transakcji. Nie ma też mechanizmów pozwalających na „cofnięcie” pierwszej części takiej „transakcji”. Wynika stąd podstawowa dyrektywa projektowa: należy do minimum ograniczyć realizację czynności (np. wstawianie, usuwanie danych) z użyciem kilku stron HTML. Jeśli to jest konieczne, wszystkie strony HTML poza ostatnią nie powinny zawierać zapytań modyfikujących dane (insert, delete, update).

Problem „wydłużonych w czasie zapytań” może też być przeciwwskazaniem do prób implementacji rygli za pomocą dodatkowych kolumn o znaczeniu „record under modification” - nałożenie takiego rygla może doprowadzić do blokowania „na stałe” rekordów, jeśli użytkownik np. stracił połączenie z serwerem, lub zrezygnował z wypełnienia formy.

3.8.3. Problem utraconej aktualizacji

Z problemem utraconej aktualizacji mamy do czynienia w przypadku, gdy dwie transakcje wysyłające zapytania do bazy danych próbują zmienić zawartość krotki w sytuacji pokazanej na poniższym rysunku. Każda z transakcji ma wyróżnione oddzielne etapy READ (odczyt wartości danych) i WRITE (zapis danych do bazy):



Rysunek 89: Problem utraconej aktualizacji

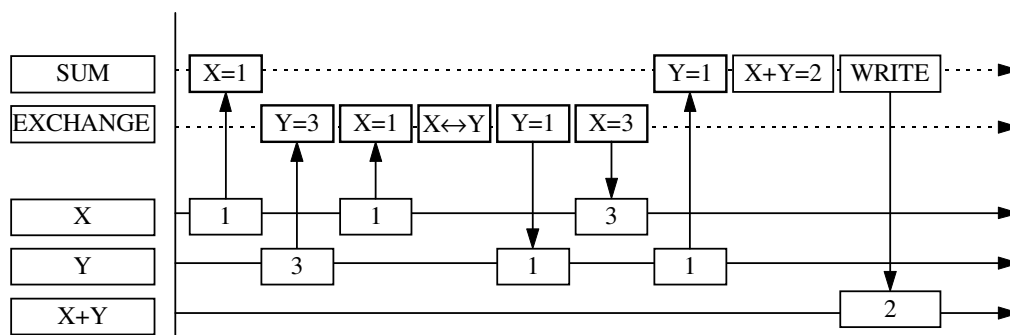
⁴⁴ patrz 3.9 - opis elementów interfejsu

W takim przypadku, w zależności od kolejności wykonania się poszczególnych faz, możliwe są cztery wyniki końcowe: $X=1+3=4$, $X=1+5=6$, $X=1+3+5=9$ lub $X=1+5+3=9$. Istota problemu tkwi w tym, że o ile w przypadku zserializowania transakcji (kolejność AB lub BA) „obydwa” wyniki 9 mają logiczne uzasadnienie, to wyniki 4 i 6 nie daje się uzyskać w ogóle. Warto jednak wskazać, że jeżeli obydwie transakcje nie stosują obliczeń bazujących na odczytanych danych, a jedynie je zmieniają (tzn. $X=X+5$ jest zabronione, natomiast $X=5$ - dozwolone), problem utraconej aktualizacji można przy pewnym ograniczeniu wymagań pominąć, ponieważ zbiór możliwych wyników transakcji jest taki sam zarówno w przypadku wykonania transakcji współbieżnie, jak i w kolejności (nie jest możliwe wygenerowanie błędnego wyniku). Pozostający problem polega na odpowiedzi na pytanie, czy można dopuścić do sytuacji, w której użytkownik modyfikuje dane, które zostały zmodyfikowane przez kogoś innego (modyfikacja w tym momencie oznacza wpisanie zupełnie nowych danych, a nie np. inkrementację czy dekrementację istniejących).

W systemie BIS, mimo iż nie występują modyfikacje typu „inkrementuj”, zdecydowaliśmy się wykorzystanie mechanizmu znaczników czasowych (timestamp). Rozwiązanie to ma zresztą tylko charakter czasowy - patrz uwagi w p. 3.11.3 i 6.2

3.8.4. Problem niespójnej analizy

Problem ten najprościej pokazać na przykładzie dwóch transakcji, z których jedna sumuje dwie dane, a druga zamienia je miejscami. W przypadku zserializowania tych dwóch transakcji, otrzymamy zawsze wynik 4 (dodawanie jest przemienne, więc nie gra roli, czy zamiana będzie miała miejsce przed czy po sumowaniu), wykonanie współbieżne może wygenerować błędny wynik - 2.



Rysunek 90: Problem niespójnej analizy

Istota tego problemu to zmiana danych, podczas gdy dane te są używane przez inną transakcję. Problem ten jest szczególnie niebezpieczny, jeśli zauważymy, że dane te mogą być używane jako klucze zewnętrzne - istnieje bowiem ryzyko utraty spójności danych.

3.8.5. Problem archiwizacji w trakcie działania systemu.

W większości systemów komputerowych archiwizacja danych odbywa się w trybie „off-line” - normalne działanie systemu jest zawieszane do czasu ukończenia aktualizacji. W systemie BIS postanowiliśmy umożliwić tworzenie kopii zapasowych bez konieczności zatrzymywania systemu. Dzięki temu wykonywanie backup'u jest możliwe również „z zewnątrz” - przez administratora korzystającego z usługi WWW. Problem, który trzeba rozwiązać, to ochrona przed realizacją dostępu typu „insert” w momencie przeprowadzania archiwizacji - może to bowiem doprowadzić do utraty spójności bazy danych. Załóżmy, że w momencie backup, użytkownik wpisuje nową publikację (artykuł). Może się zdarzyć sytuacja, w której proces już zarchiwizował zawartość tablicy Artykuł, natomiast nie zdążył zarchiwizować tablicy Publikacja. W tym momencie zrealizowany zostaje wpis nowej publikacji - jej dane umieszczane są w tablicach Publikacja i Artykuł. Po zakończeniu tego procesu backup archiwizuje tablicę Publikacja. W efekcie, po wykonaniu przywrócenia stanu bazy danych, spójność bazy danych zostanie utracona (w bazie będzie tylko część danych nowej publikacji - ta w tablicy Publikacja).

Ponieważ instrukcja mSQL informuje, że baza danych kolejkuje zapytania i uniemożliwia ich współbieżne wykonanie, istniała możliwość, że zachowanie stanu bazy (ang. dump) bazy jest również realizowany jako pojedyncze zapytanie. W takiej sytuacji przed zaistnieniem niebezpiecznej sytuacji chroniłby mechanizmy

systemowe mSQL. Niestety - przeprowadzone testy⁴⁵ wykazały, że dump bazy danych nie jest operacją atomową i podczas jego trwania możliwe jest zarówno wydanie zapytań bezpiecznych (select), jak i grożących utratą spójności (insert, update, delete).

3.8.6. Błędne usuwanie na skutek współbieżnej modyfikacji

Problem ten ma aspekt głównie implementacyjny, dlatego omówiono go dokładniej w 4.2.2.

3.8.7. Podsumowanie - przyjęta strategia zapewniania współbieżności

Opisane wyżej zagrożenia są bardzo istotne z punktu widzenia bezpieczeństwa systemu. W pewnym momencie autorzy stanęli nawet przed koniecznością ponownego rozpatrzenia celowości wykorzystania mSQL, w kontekście problemów, jakie stwarza on w dziedzinie współbieżności. Okazało się jednak, że producent oprogramowania - firma Hughes Inc. - zapowiada na najbliższy okres kolejną edycję serwera mSQL, tym razem poszerzonego o implementację transakcji. Mechanizm taki zabezpieczył bazę danych przed wyżej opisanymi niebezpieczeństwami, a jego kluczową zaletą byłaby możliwość wykonania komendy ROLLBACK i zapewnienie atomowości transakcji. Firma zapowiada, że motor ten, będzie nadal dostępny dla uczelni bez konieczności ponoszenia opłat licencyjnych. Niestety, wąskie ramy czasowe projektu, wykorzystanie transakcji nie jest możliwe w obecnej wersji systemu BIS. Na podstawie konsultacji przeprowadzonych z Promotorem pracy - dr inż. K. Goczyłą, przyjęliśmy następującą strategię:

- projekt BIS nie będzie czekał na wyprodukowanie nowej wersji mSQL
- implementacja zaawansowanych mechanizmów obsługi współbieżności mija się z celem, ponieważ za kilka miesięcy pojawi się możliwość zastosowania transakcji i ew. następcy, rozwijający projekt będą mogli je wykorzystać
- współbieżność w bazie BIS powinna być zapewniona na warunkach wystarczających do zabezpieczenia przed możliwością utraty spójności bazy danych (propozycja Promotora: ryglowanie dostępu „write” do bazy danych, ew.

⁴⁵ Test polegał na stworzeniu nowej bazy danych, wypełnieniu jej elementami i przeprowadzeniu archiwizacji, podczas której z innej konsoli wykonywaliśmy zapytania (baza liczyła 30.000 krotek zawierającej pola char[255], aby zapewnić wystarczająco długi czas jej trwania).

rygłowanie dostępu do tablic - uwzględnia to charakter bazy danych: częsty dostęp typu „read” i sporadyczny dostęp typu „write”⁴⁶)

W ostatecznej wersji rozszerzyliśmy tą strategię o zastosowanie znaczników czasowych, co zapewnia praktycznie najlepszą technikę obsługi współbieżności możliwą do realizacji przy użyciu dostępnych środków.

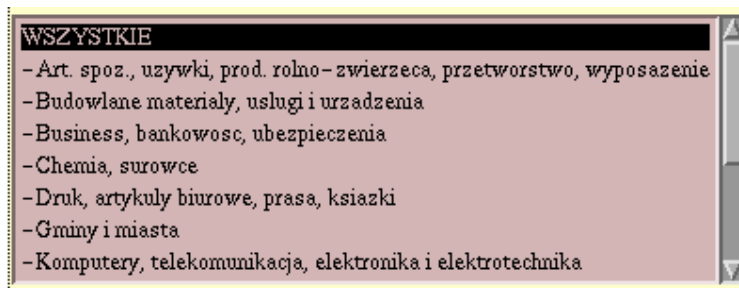
3.9. Konstrukcja interfejsu

3.9.1. Możliwości oferowane przez HTML

3.9.1.1. Uwagi wstępne

Możliwości oferowane przez HTML są stosunkowo ubogie. Wynika to z konieczności dopasowania konstrukcji samego języka do różnych środowisk, w których będzie on implementowany: począwszy od przeglądarki tekstowej Lynx, aż do zaawansowanych wersji Microsoft Internet Explorer 4.0 i 5.0, będących integralnymi składnikami środowiska (i interfejsu) Windows 95/NT. Należy tu także wspomnieć o różnicach pomiędzy implementacjami HTML 3.0.

Konstruując interfejs w języku HTML należy zwrócić uwagę na kłopoty, jakie pojawiają się na skutek innej interpretacji tagów (elementów określających własności tekstu w języku HTML) przez różne przeglądarki. Przykładem może tu być odmienny wygląd list w Netscape Navigator (Unix) i MS IE (Windows):



Rysunek 91: Przykład listy zinterpretowanej przez program Netscape Navigator 3.0 Gold (Unix)

⁴⁶ Dodawanie nowych publikacji odbywa się stosunkowo rzadko. W praktyce dostęp współbieżny typu „write” jest stosunkowo mało prawdopodobny (w normalnym działaniu bazy, tj. po zakończeniu wprowadzania pierwszych danych)



Rysunek 92: Przykład listy zinterpretowanej przez program MS Internet Explorer 4.0 (Windows 95)

Dodatkowym problemem - dotyczącym już tylko przeglądarek graficznych jest występowanie innych palet barw w Netscape Navigator i MS IE. Nieprofesjonalne tworzenie rysunków (zwłaszcza kodowanych z utratą danych - format jpg) prowadzi do tego, że choć w jednej przeglądarce kolor jest jednolity, w drugiej pojawiają się widoczne, nieestetyczne zaburzenia.

Interfejs interaktywny jest w praktyce bardzo trudny do zrealizowania. Właściwie drogi są dwie - użycie drogi zalecanej przez określonego wytwórcę (np. ActiveX w systemie Windows) lub zastosowanie multiplatformowej Javy. Mimo niewątpliwych zalet tego ostatniego rozwiązania spotyka się je stosunkowo rzadko - jedynie na najlepszych stronach WWW (interfejs oparty o Javę ma ok. 5% stron). Przykład strony z zastosowanym komponentem ActiveX można znaleźć pod adresem www.vcv.net/php

3.9.2. Przekazywanie informacji do użytkownika

Poniższa charakterystyka obejmuje niezbędny zasób wiedzy do skonstruowania interfejsu systemu BIS i do prześledzenia problemów implementacyjnych podanych dalej.

3.9.2.1. Parametryzowanie tekstu

Język HTML nie oferuje wielkich możliwości ingerencji w postać wyprowadzanych danych. Jeśli chodzi o tekst, zastosować można jedynie pogrubienie (bold), pochylenie (italic) i podkreślenie (underline)⁴⁷. W wersji HTML 3.0 można również zdefiniować kolor tekstu i font użyty do jego przedstawienia.

HTML nie oferuje możliwości justowania tekstu do obu marginesów - jedynie do lewej lub prawej strony. Jedynym wyjściem jest układanie tekstu w tablice - jest

⁴⁷ Są to tzw. metody określania atrybutów fizycznych. Zalecany, ale stosunkowo rzadko używany sposób to określanie logiczne np. strong zamiast bold. Różnica jest następująca: jeśli przeglądarka nie może ze względów technicznych (np. wyświetlacz nie może zrealizować italic - np. stasze typy monitorów) to tag fizyczny zostanie zignorowany, tag logiczny - zamieniony na inny (np. tekst o stylu italic zostanie podkreślony).

to zwłaszcza przydatne w przypadku prezentacji wyników zapytań (krotek) bądź budowania form umożliwiających wprowadzanie danych.

3.9.2.2. Wyświetlanie grafiki

HTML w wersji 3.0 oferuje możliwość zamieszczania rysunków w formatach jpg i gif. Za wykorzystaniem jpg przemawia wyjątkowo ekonomiczny sposób w jaki te rysunki są przechowywane (kompresja danych z utratą informacji za pomocą rozwinięcia Fouriera). Format gif jako jedyny umożliwia wykorzystanie rysunków transparentnych (technika bardzo popularna, kolor przezroczysty jest zastępowany tłem dokumentu HTML co daje możliwość uzyskania ładnych efektów małymi środkami). Format ten pozwala również na wprowadzenie gifów animowanych (tzw. animated gif - gif taki zawiera kolejne fazy ruchu), ale rysunki takie zajmują zwykle bardzo dużo miejsca.

3.9.2.3. Filmy i animacje inne niż gify animowane

Są dostępne głównie w formacie *.avi oraz innych, do których trzeba przygotować przeglądarkę poprzez dołączenie dodatkowych modułów. W naszej pracy nie stosujemy takich technik, więc kwestię tę pomijamy w niniejszej dokumentacji. Warto jedynie wspomnieć, że stosowanie sekwencji animacji na stronach WWW łatwo może doprowadzić do „zalania” łącza nadmiarem danych - doświadczenia pokazują (próba zastosowania widoku z kamery na ulicę Długą w firmie Itnet), że już dwóch - trzech użytkowników potrafi doprowadzić do nadmiernego obciążenia nawet pojemnych łącz (technologia Frame Relay).

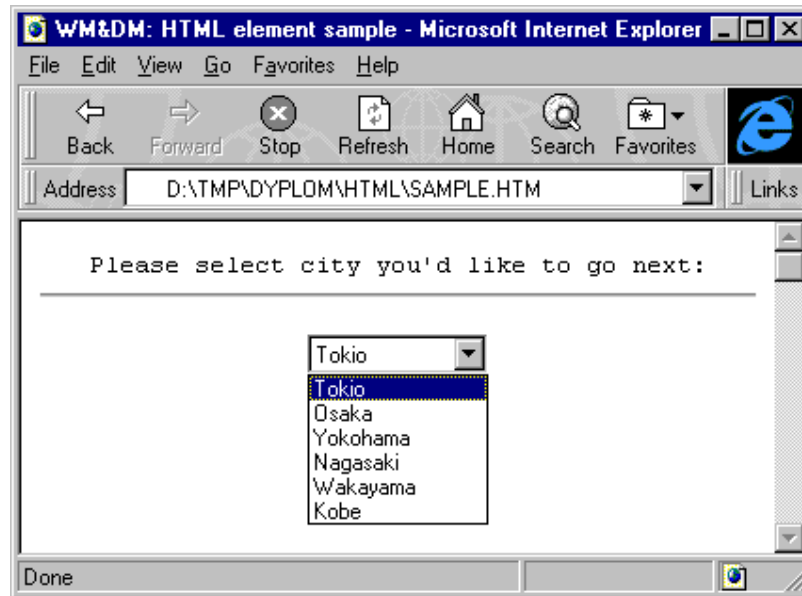
3.9.3. Wprowadzanie danych

3.9.3.1. Formy (formularze)

Jedyną możliwością wprowadzenia danych w HTML jest wykorzystanie tzw. formularzy (pomijamy rozbudowę HTML o aplety Javy lub komponenty ActiveX). Formularz jest definiowany przez podanie ilości i rodzaju pól wprowadzania danych, a także ich niektórych parametrów (np. długość pola tekstowego). Należy także podać adres dokumentu, do którego zostaną przesłane wprowadzone dane. Niestety, nie jest możliwe podanie różnych adresów - po dokumencie z formą, w której użytkownik zakończył wprowadzanie danych (kończy się to zawsze kliknięciem klawisza „SUBMIT”) zawsze wywoływany jest dokument o podanej lokalizacji.

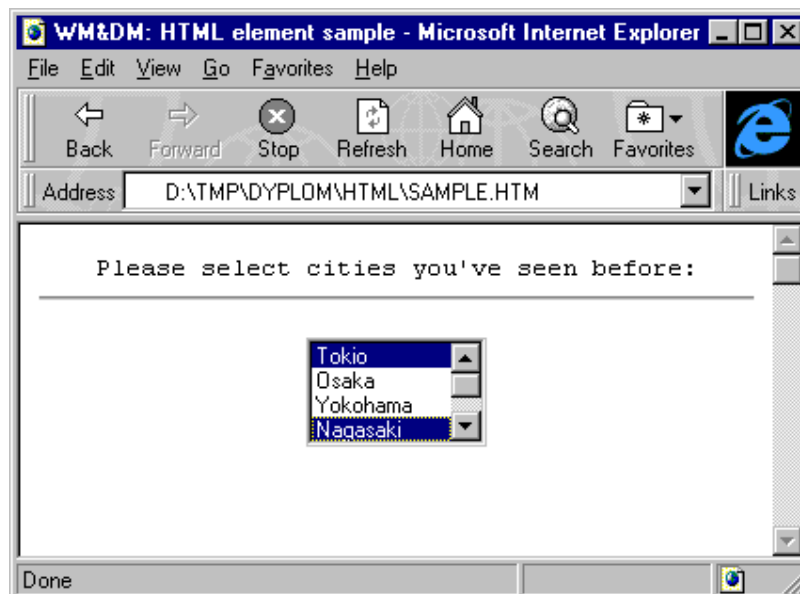
3.9.3.2. Lista rozwijana (pull-down)

Umożliwia wybór jednej (i tylko jednej) z opcji. Zaletą tego elementu jest fakt, że gdy nie jest on używany (nie był „kliknięty”), pokazywany jest tylko element wybrany, co ułatwia gospodarzkę wolnym miejscem na ekranie przeglądarki.



Rysunek 93: Elementy interfejsu HTML - lista rozwijana (typu „pull-down”)

3.9.3.3. Lista przewijana (wybór opcji)

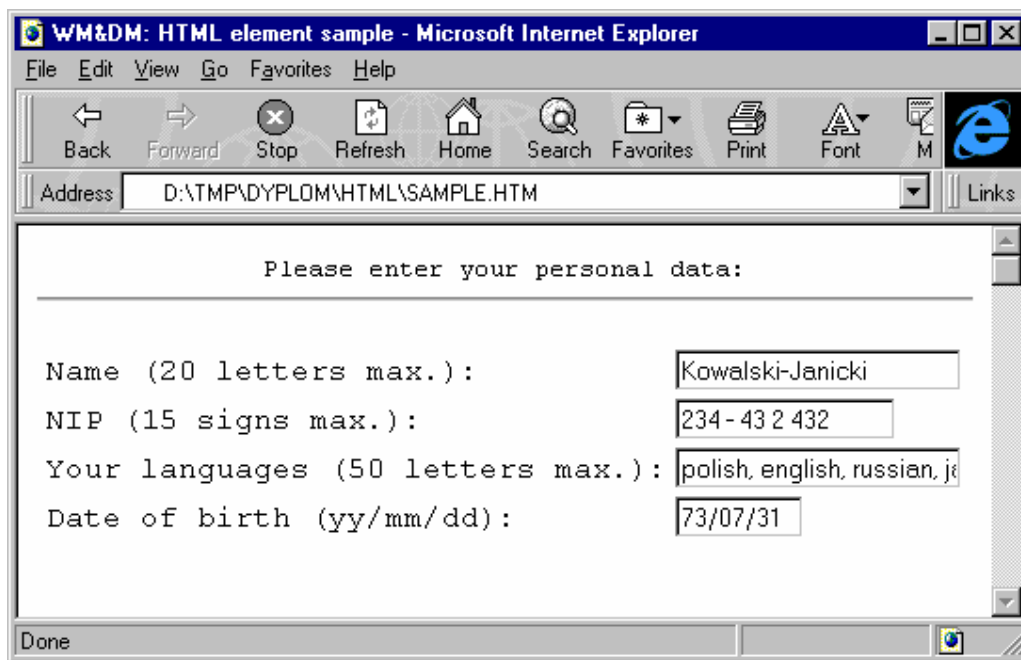


Rysunek 94: Elementy interfejsu HTML - lista przewijana

Umożliwia wybór jednej lub wielu spośród podanych opcji. Zastosowanie tego elementu jest zbliżone do listy rozwijanej (pull-down). Możliwe jest określenie ile elementów ma być jednorazowo wyświetlanych na liście („długość listy”) - elementy niewidoczne są dostępne po użyciu strzałek przesuwania umieszczonych z boku listy.

3.9.3.4. Pola wprowadzania danych

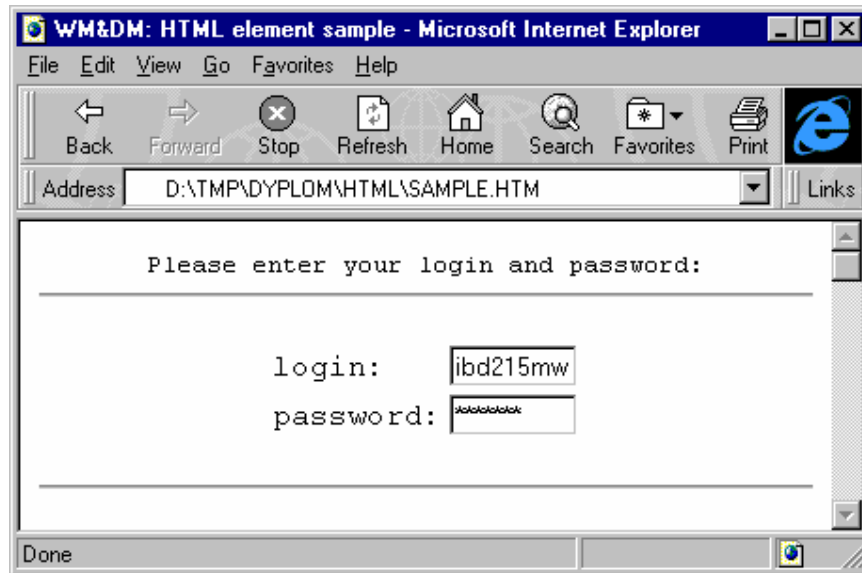
Tą nazwą określa się wszystkie pola, w które można wprowadzać dane - liczby lub tekst. Pola te charakteryzują się maksymalną długością i wielkością na ekranie (np. pole do którego można wpisać 50 znaków może na ekranie zajmować wielkość 10 znaków). Można także określić ich wartość początkową. Niestety, nie jest możliwa automatyczna kontrola poprawności wpisywanych danych: ani logiczna (np. odrzucanie daty 31.02.97), ani syntaktyczna (można wpisać litery w polu, którego używamy do wpisywania liczb).



Rysunek 95: Elementy interfejsu HTML - pola wpisu danych

3.9.3.5. Specjalne pole wprowadzania danych - typ password

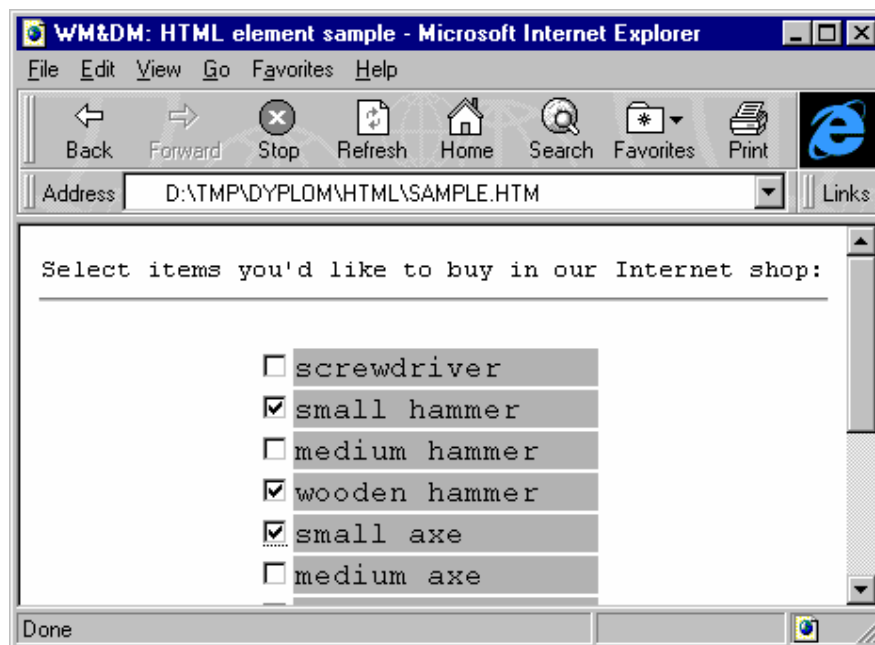
Pole to jest realizowane w identyczny sposób jak inne pola tekstowe, ale znaki nie są wyświetlane na ekranie (zastępuje je zwykle znak gwiazdki - *).



Rysunek 96: Elementy interfejsu HTML - pole wpisu hasła

3.9.3.6. Pole wyboru (ang. checkbox)

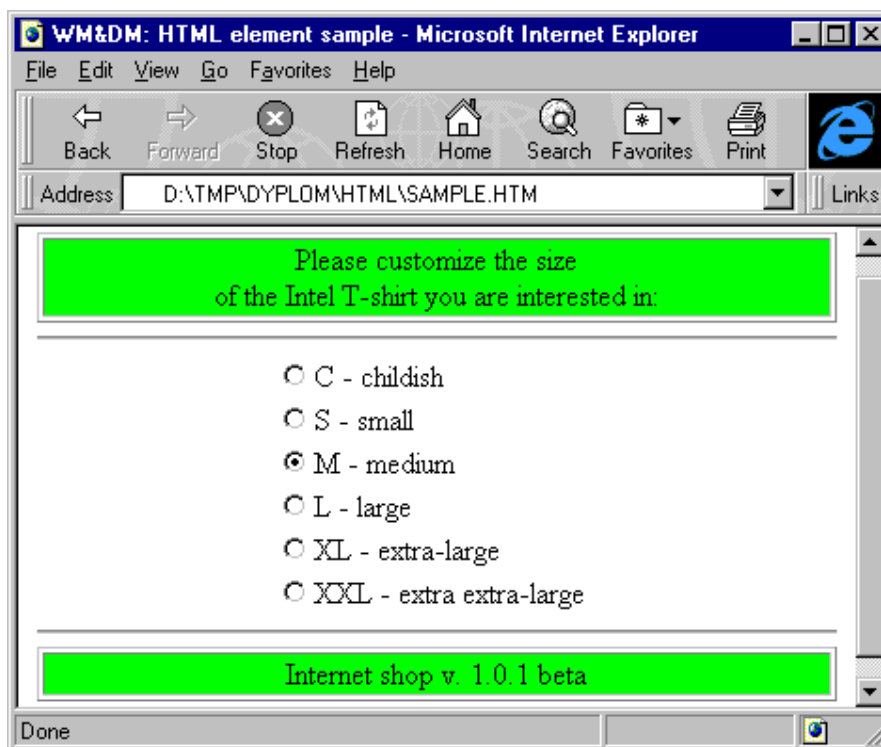
Pole wyboru jest elementem interfejsu, który można wykorzystać do zaznaczania wielu pozycji w sformatowanym tekście. Jest on szczególnie użyteczny przy oznaczaniu rekordów do modyfikacji lub usunięcia, a także przy tworzeniu długich list wyboru.



Rysunek 97: Elementy interfejsu HTML - checkbox

3.9.3.7. Przycisk wyboru opcji (ang. radiobutton)

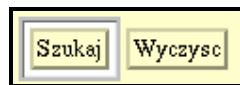
Przycisk wyboru opcji jest elementem interfejsu, który można wykorzystać do zaznaczania dokładnie jednej spośród pewnej liczby pozycji w sformatowanym tekście. Jest on szczególnie użyteczny np. przy oznaczaniu wariantów cechy (gdzie zależy nam na tym, aby użytkownik wybrał dokładnie jedną z opcji).



Rysunek 98: Elementy interfejsu HTML - radiobutton

3.9.3.8. Przyciski SUBMIT i RESET

Przyciski SUBMIT i RESET są dwoma typami klawiszy dopuszczanymi w formach. Pierwszy z nich jest niezbędny do prawidłowego działania formy - po jego naciśnięciu dane wyprowadzone przez użytkownika są przesyłane do serwera (do pliku którego lokalizacja określona jest poleceniem ACTION). Drugi z nich jest używany dość rzadko - jego naciśnięcie powoduje usunięcie danych wpisanych przez użytkownika i przygotowanie „czystej” formy (dzieje się to po stronie klienta - bez odnawiania kontaktu z serwerem). Czasami spotyka się go w obsłudze bardzo rozbudowanych form, lub w formach obsługujących sklepy internetowe, jako polecenie „opróżnienia” koszyka z zakupami.



Rysunek 99: Elementy interfejsu HTML - przyciski SUBMIT i RESET
(Netscape Navigator)

3.9.4. Prototypowanie form dla systemu BIS

Interfejs programu BIS był tworzony w oparciu o interfejsy klasycznych programów wyszukiwawczych: Yahoo, Alta Vista itp. Zapewniało to zgodność z powszechnie przyjętymi standardami. Podstawowe problemy, jakie należało rozwiązać to:

- zapewnienie zgodności z przeglądarkami tekstowymi (Lynx)
- zbliżony kształt interfejsu w oknach Netscape Navigator i MS IE
- funkcjonalność (np. unikanie formularzy, których długość przekracza długość okna przeglądarki)
- oszczędne gospodarowanie miejscem (trzy listy umieszczone na ekranie praktycznie uniemożliwiają wstawienie innych elementów interfejsu bez wydłużenia strony HTML poza „okienko” przeglądarki)
- polskie litery (który ze standardów kodowania wybrać ?)

Problem zgodności z przeglądarkami tekstowymi rozwiązaliśmy dokonując systematycznego porównania, które znaczniki (tagi) są interpretowane, a które - nie. Okazało się, że Lynx ignoruje np. znacznik <TH> stosowany do wskazywania pola tabeli będącego nagłówkiem (pole takie jest automatycznie centrowane i wytłuszczone). Problem grafik został rozwiązany przez zastosowanie parametru ALT, oznaczającego tekst zastępczy który wyświetlany jest w oknie przeglądarki nie potrafiącej wyświetlać grafiki. W ten sposób, sekwencja:

To jest przykład grafiki:
na stronie HTML.

zostanie zinterpretowana w oknie Lynxa, jako:

To jest przykład **grafiki**: ikonka-myszki na stronie HTML

Różnice w interpretacji kodu HTML przez Netscape Navigatora i MS Internet Explorera zostały zminimalizowane poprzez wprowadzenie dodatkowych

parametrów, określających wygląd elementów „różniących” - jak np. listy (zastosowano parametry WIDTH, SIZE, SELECTED itp.). Zastosowano także formatowanie tekstu przez określenie rodzaju czcionek, który ma być użyty na ekranie. Znacznik został początkowo wprowadzony przez Microsoft (wykorzystano fakt, że czcionki w środowisku Windows są łatwe do wskazania, ponieważ system operacyjny gromadzi o nich informację), ale obecnie Netscape Navigator, również interpretuje takie określenie wyglądu liter.

Dużym problemem stało się zapewnienie odpowiedniej funkcjonalności. Dotyczyło to zwłaszcza dynamicznie generowanych plików z formularzami. Sytuacja taka miała miejsce, gdy budowaliśmy strony, dzięki którym możliwe było wprowadzanie danych do tablic nie istniejących jeszcze w bazie danych (zmiana struktury bazy danych podczas działania systemu - por. 4.5.3). Oznaczało to konieczność wypracowania mechanizmu, który pozwalałby na **automatyczną konstrukcję interfejsu**, na podstawie podanej struktury danych, co przy bardzo ograniczonych możliwościach HTML było bardzo trudne do zrealizowania. Problem został rozwiązany przez wielokrotną metodę prób i błędów - nie spotkaliśmy się nigdzie z istniejącym rozwiązaniem, z którego doświadczenia możnaby przenieść lub zmodyfikować do naszych potrzeb.

Zagadnienie polskich liter, który w początkach pracy dyplomowej stanowiło poważny problem (w użyciu było kilka metod kodowania⁴⁸, z upływem czasu straciło na ważności - obecnie stosuje się właściwie tylko standard ISO lub Windows. Najpopularniejszą metodą staje się zapis plików HTML w jednym z tych standardów i zamieszczenie odpowiedniej stopki informującej użytkownika, jak powinien skonfigurować przeglądarkę. Rozwiązanie alternatywne - generowanie stron w obu standardach jest stosunkowo kłopotliwe z uwagi na podwojenie się zajmowanego obszaru dysku i dodatkowy nakład pracy związany z aktualizacją danych w dwóch miejscach⁴⁹.

⁴⁸ Jeszcze pod koniec 1996 na polskich stronach domowych znaleźć można było 9 różnych metod kodowania polskich znaków narodowych (najpopularniejszymi były standard ISO, Windows, Mazovia, oraz „bez polskich liter”).

⁴⁹ Zaawansowaną metodą jest stworzenie skryptu CGI, tłumaczącego „w locie” tekst na wybrany wcześniej przez użytkownika. Rozwiązanie takie jest jednak stosunkowo skomplikowane i jest wykorzystywane bardzo rzadko.

Ostateczna wersja interfejsu bazuje za wykorzystaniu formularzy i tabel jako elementów formatujących tekst i rozmieszczających go na ekranie. Wykorzystano także mechanizm ustalania koloru wypełnienia pól tablic do zaznaczania opcji w menu.

3.9.5. Ocena jakości interfejsu

Uzyskany interfejs cechuje się dość dobrym poziomem funkcjonalności. Wydaje się, że bez ograniczenia funkcjonalności trudno byłoby wprowadzić jakiegokolwiek ulepszenia. Interfejs systemu BIS jest także dość podobny do popularnych systemów wyszukiwawczych co powinno ułatwić obsługę nowym użytkownikom. Ewentualne modyfikacje powinny dotyczyć wprowadzenia nowych wersji językowych, oraz rozwiązania problemu polskich liter w oparciu o skrypt CGI (pozostaje otwarta kwestia połączenia mechanizmu CGI z API).

3.10. Wymagania nakładane na zasoby

3.10.1. System operacyjny

Konieczny Unix, w wersji na PC, Sun lub HP, jądro systemu operacyjnego powinno zawierać wkompiłowaną obsługę kart sieciowych (dla celów demonstracyjnych i testowych nie jest to konieczne - odwołania do bazy mogą następować przez tzw. localhost). Konieczne są biblioteki systemowe lgm i lgb. W przypadku stosowania Linuxa zalecane wersje Debian, Red Hat, Slackware lub inne, zapewniające standard ELF.

3.10.2. Procesor

W przypadku wykorzystania komputera klasy PC, ze względów wydajnościowych co najmniej 486SX, zalecany co najmniej 486DX33. W przypadku komputerów klasy Sun lub HP praktycznie dowolna maszyna typu serwer, a nawet stacja robocza.

3.10.3. Pamięć operacyjna

Wymagania dotyczą komputera klasy PC: minimalnie 4 MB (konieczne odpowiednie ustawienie partycji swap w systemie Linux), zalecane co najmniej 8MB.

3.10.4.Karta sieciowa

Dowolna, np. NE2000. Nie jest niezbędna do celów demonstracyjnych i testowych (użycie typu localhost).

3.10.5.Internet lub inna sieć komputerowa

Podłączenie stałe (bezpośrednio do Internetu np. za pomocą serwera uczelnianego przez sieć lokalną). Opcjonalna wersja połączenie „sztywne” (dzierżawiona linia telefoniczna), raczej odradzana ze względu na wyższe koszty (zwłaszcza w odniesieniu do placówek uczelnianych).

3.10.6.Pamięć masowa

W odniesieniu do komputerów klasy PC zalecany dysk twardy min. 250 MB (uwzględniamy odpowiednio dużą partycję swap i miejsce potrzebne na zainstalowanie systemu operacyjnego). W praktyce dysk może być mniejszy (należy wówczas poważnie ograniczyć składowe systemu operacyjnego takie jak system X-Window, oprogramowanie wspomagające itp.), ale nie jest to rozwiązanie wygodne. W przypadku stosowania innych komputerów, zwłaszcza z już istniejącym systemem (przykład: stacja SunSparc Classic loqi16.eti.pg.gda.pl. w Zakładzie Zastosowań Informatyki PG), wystarcza ok. 20 MB na serwer bazy danych, WWW i pliki HTML.

3.10.7.Inne

Wymagane prawa „root” do uruchomienia oprogramowania serwera WWW (o ile pracować ma on na standardowym dla usługi WWW porcie 80, możliwe jest także ustawienie do pracy z wykorzystaniem portu 8080 lub innego). W praktyce może zostać to zrealizowane przez wpisanie zlecenia uruchamiania oprogramowania do plików startowych w katalogu /etc lub ustawienie bitu s (set uid) w prawach dostępu do pliku httpd.

3.10.8.Monitor

Po zainstalowaniu systemu nie jest niezbędny, dostęp może być realizowany poprzez sieć.

3.11. Wybór komponentów systemu

3.11.1.Komponenty sprzętowe

System przez nas zaproponowany nie jest ściśle związany z określoną konfiguracją sprzętową - może on pracować zarówno w standardowej konfiguracji

komputera PC, jak i na maszynach Sun Sparc, czy HP-UX. Podczas prac nad systemem używaliśmy trzech konfiguracji⁵⁰:

- a. komputer PC 486DX30, 20 MB RAM, 2 x HDD 200MB, NE 2000, S3 PCI (monitor był odłączony - praca odbywała się zdalnie)
- b. komputer PC 486DX2 66, 8MB 250MB, S3805 LB, Deawoo 14"
- c. komputer PC Pentium 75, 16MB 1.2GB (300MB dla Linux), S3764V+ PCI 1 MB, GoldStar 14"

3.11.2. System operacyjny

Jako platformę operacyjną dla systemu BIS wybraliśmy system UNIX w wersji dla komputerów PC. Implementacja zastosowana w naszym systemie to Linux Debian. Oprogramowanie to jest dostępne w sieci Internet, a większa jego część posiada licencję GNU. Ponieważ instalacja systemu operacyjnego była wykonywana kilkakrotnie, na różnych komputerach, zastosowaliśmy zarówno metodę instalacji „przez sieć” (z wykonywaniem podmontowywania (za pomocą polecenia mount) katalogu exportowanego za pośrednictwem PC-NFS), jak i klasyczną instalację z dyskietek, CD czy też dołączonego dysku twardego.

Używany przez nas Linux był oznaczony wersją 2.0.27. Z uwagi na to, że domyślnie jądro systemu nie posiada wbudowanych procedur umożliwiających obsługę sieci, zaistniała konieczność przeprowadzenia rekompilacji (z uwzględnieniem karty sieciowej NE2000).

Podczas jednej z instalacji wykonaliśmy próbę instalacji opcjonalnego oprogramowania (uzyskanego z sieci Internet) podsystemu obsługi haseł tzw. Shadow-password. Chroni ona przed niepowołanym dostępem plik /etc/passwd, co ma duże znaczenie przy ocenie bezpieczeństwa systemu. Zabezpieczyliśmy się także przed realizacją żądań zewnętrznych typu finger, co również jest zalecane w celu podwyższenia wiarygodności systemu (utrudnia to potencjalnemu włamywaczowi identyfikację loginów użytkowników, co w pewnym stopniu zabezpiecza przed atakiem typu „brute-force”).

⁵⁰ Przez krótki czas system był tworzony na komputerze Sun Sparc 5 w laboratorium ZZI WETI PG (p. 623)

3.11.3. Motor bazy danych

Dla potrzeb systemu BIS wybraliśmy relacyjny motor bazy danych firmy Hughes Software Inc - mSQL⁵¹. Jest to oprogramowanie komercyjne, ale o specjalnej licencji dla odbiorców pracujących w placówkach edukacyjnych (freeware for educational staff). Użytkownicy, których dotyczy ta licencja mają prawo do bezpłatnej dystrybucji oprogramowania, wraz z całością kodu źródłowego.

O wyborze tego oprogramowania zdecydowały zarówno czynniki ekonomiczne, jak i sprawdzona wartość tego oprogramowania - jest ono używane w kilku profesjonalnych systemach WWW - np. TOW, baza danych Panoramy Firm w Internecie itp. Innymi możliwościami przez nas rozpatrywanymi były:

- MS SQL Server 6.5 - duży serwer bazy danych firmy Microsoft, pracujący w systemie Windows NT (Win NT Server 4.0). Autorzy systemu BIS zetknęli się z tym oprogramowaniem podczas praktyki w Niemczech, realizowanej w ostatnim roku studiów. W projekcie DARIF używaliśmy serwera MS SQL 6.5 w połączeniu z MS Internet Information Server (serwer WWW firmy Microsoft) i bazami danych Access 7.0 i Oracle. Zalety tego systemu to bogate możliwości oferowane w zakresie zarządzania bazami danych (platforma ODBC umożliwia połączenie z innymi bazami, istnieje możliwość zastosowania rozproszonej bazy danych itp.), szybkość realizowania zapytań, narzędzia systemowe pozwalające skrócić proces implementacji i ułatwić maintenance itp. Zasadniczą wadą tego systemu jest jego koszt, wyrażany zarówno jego bardzo wysoką ceną, jak i ilością zasobów potrzebnych do instalacji⁵² (środowisko Windows NT, duży i szybki dysk twardy, szybki procesor itp.).

- MS Access 7.0 - baza danych Microsoft, pracująca pod systemami 32-bitowych Windows (Win 95, Win NT). Podobnie jak MS SQL Server wymaga środowiska Windows, a co za tym idzie serwera WWW pracującego w tym środowisku (łączenie się z serwera pracującego w innym środowisku - np. Unix jest możliwe, ale stosunkowo kłopotliwe ze względów implementacyjnych). Rozwiązanie

⁵¹ mini SQL - nazwa pochodzi od faktu, że serwer mSQL nie realizuje niektórych konstrukcji ANSI SQL (jego polecenia są podzbiorem specyfikacji ANSI SQL)

⁵² We wspomnianym projekcie DARIF system, na którym pracowaliśmy składał się z komputera COMPAQ Pentium Pro 200 MHz, 128 MB RAM, 4 BG HDD

to zarzuciliśmy ze względów finansowych (konieczność zakupienia licencji Windows, Accessa i IIS).

- Solid SQL - sharewareowa baza danych pracująca w systemie Unix. Rozwiązanie to jest bardzo podobne do mSQL, zdecydowaliśmy się na ten ostatni z uwagi na pewne doświadczenia w jego użytkowaniu.

- mySQL - relacyjna baza danych SQL, dostępna w sieci Internet jako public domain. Funkcjonalnie zgodna z mSQL, zdecydowaliśmy się na ten ostatni z uwagi na pewne doświadczenia w jego użytkowaniu.

Projekt mSQL jest rozwijany na bieżąco - średnio co miesiąc pojawiają się nowe wersje tego oprogramowania, zwykle dostępne dla użytkowników jako wersje beta. W porównaniu z poprzednią pełną wersją systemu - mSQL 1.0, nowa wersja 2.0 (dostępna od lutego 1997) wprowadziła m.in. indeksowanie tabel, obsługę znaczników czasowych, optymalizację zapytań, przyspieszony algorytm obsługi dysku itp. Szczegółowe informacje na ten temat można znaleźć pod adresem <http://Hughes.com.au>, w pliku history.html. Na najbliższe miesiące zapowiadana jest kolejna edycja oprogramowania - mSQL 3.0, wzbogacona m.in. o obsługę transakcji. Zapowiedź ta, była jednym z czynników, które zadecydowały o zastosowaniu tego oprogramowania - ułatwi to ew. rozwój systemu BIS o pełną obsługę współbieżności.

3.11.4. Serwer WWW

W projekcie BIS stosujemy serwer WWW Apache v. 1.2.10 beta, który jest dostępny w sieci Internet pod adresem www.apache.org w postaci skompilowanych modułów programowych (binaria) i kodu źródłowego. Serwer ten powstaje jako efekt projektu rozwojowego, realizowanego stale od kilku lat. Część kodu powstała w centrum obliczeniowym uniwersytetu Illinois (National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign), algorytmy szyfrowania i ochrony danych pochodzą m.in. z firmy RSAData Security Inc. (MD5 Message-Digest Algorithm) poszerzonych o wyniki badań laboratoriów koncernu Spyglass Inc., Caregie Mellon University, and Bell Communications Research, Inc. (Bellcore). Na pewną część kodu prawa autorskie (patrz dokumentacja on-line serwera) posiada Henry Spencer.

Serwer WWW Apache jest produktem freeware zarówno dla rozwiązań komercyjnych, jak i edukacyjnych. Jest to oprogramowanie u ustalonej marce (1 miejsce pod względem liczby wykorzystujących je hostów), szeroko stosowane na świecie i uważane za jeden z najbardziej stabilnych i wiarygodnych systemów WWW. Wiele w pełni profesjonalnych systemów używa tego oprogramowania - np. giełda papierów wartościowych TOW, czy system informacji kolejowej HAFAS.

3.11.5.PHPFI - Interfejs API serwera WWW (rozszerzenie poleceń dostępnych w HTML)

PHP/FI (Personal Home Page / Form Interpreter) jest modułem napisanym w języku C, rozbudowującym standardowy parser wchodzący w skład serwera WWW Apache 1.2. Jego rola sprowadza się do połączenia interfejsów API serwera Apache i mSQL⁵³. Dzięki temu możliwe staje się rozszerzenie standardowego języka HTML o bogaty zestaw dodatkowych instrukcji, m.in. iteracje, instrukcje warunkowe i decyzyjne.

Moduł ten powstał początkowo jako napisany w języku perl skrypt cgi. Jego zadaniem miało być śledzenie liczby odwołań do stron HTML (nie miał to być zresztą produkt dostępny dla szerokiej rzeszy użytkowników, a jedynie pomoc dla jego autora). Z uwagi na to, że serwer, na którym skrypt miał pracować był bardzo obciążony, pojawiały się problemy z tworzeniem procesów potomnych (instrukcja fork). W związku z tym skrypt został przepisany w języku C (aby uniknąć potrzeby forkowania procesu perla za każdym dostępem do strony HTML). Ponieważ pojawiło się zapotrzebowanie na podobne narzędzie, autor - P. Rasmus - zdecydował się na jego udostępnienie. Pakiet, w skład którego wchodziła również dokumentacja i FAQ (a także treść rozważań z listy dyskusyjnej poświęconej pakietowi) otrzymał nazwę Personal Home Page Tools, zamienioną następnie na Personal Home Page Construction Kit.

W tym samym czasie autor zajmował się również łączeniem baz danych ze stronami HTML. Napisał w tym celu prosty skrypt CGI, służący do zamieszczania zapytań SQL w stronach HTML i tworzenia na ich podstawie tabel i form. Narzędzie to otrzymało nazwę FI (Form Interpreter).

⁵³ Lub jednego z innych motorów: Postgress 95, MySQL, Solid, Sybase, Oracle. Serwery WWW, dla których PHP/FI został przetestowany to NCSA Server i Netscape.

Z tych dwóch modułów, połączonych i przepisanych na nowo (w celu optymalizacji kodu) powstał produkt o nazwie PHP/FI 2.0. Pozwala on na uniknięcie konieczności stosowanie wielu krótkich programików cgi (można dzięki temu uzyskać znaczne polepszenie wydajności związane z uniknięciem wielokrotnego forkowania procesów). Ułatwia on także budowanie bazy danych z interfejsem WWW.

3.12. Komentarz końcowy procesu projektowania

Proces projektowania nie był dokładnie zgodny z metodologią Rumbaugh. Wynikało to ze głównie specyfiki systemu - zastosowanie relacyjnej bazy danych wymusiło niejako zmianę punktu widzenia z „obiekтового” na „relacyjny”. Zmiany te nie były jednak zbyt poważne i przyjęta wcześniej metodologia po wprowadzeniu pewnych poprawek posłużyła za bazę do całości procesu projektowego.

4. Implementacja

4.1. Uwagi wstępne

Proces implementacji odbywał się w kilka miesięcy po ukończeniu etapu analizy. W tym okresie technologia WWW posunęła się zdecydowanie naprzód i rozwiązania, które do tej pory były nowatorskie, upowszechniły się, zaistniało także wiele nowych faktów, których nie było można przewidzieć wcześniej. Należy tu zwłaszcza wskazać na szybki rozwój rynku usług WWW, rzutujący na prędkość z jakim rozwijał się rynek produktów związanych z łączeniem baz danych z serwerami WWW. Przykładem zmian, jakie powstały już w trakcie implementacji (i wymagały powtórzenia wielu punktów analizy i projektu) może być np. zamiana technologii skryptów CGI na technologie interfejsu API w budowie interfejsu użytkownika.

4.2. Wybrane problemy implementacyjne

Rozdział ten pełni dwojaką funkcję - po pierwsze, jest zapisem problemów, na jakie autorzy natknęli się implementując system BIS, po drugie - stanowi dokumentację techniczną programu, mającą ułatwić zrozumienie pewnych rozwiązań ew. następcom.

4.2.1. Dynamiczne przydzielanie nazw elementom interfejsu

Problem ten pojawiał się wielokrotnie w trakcie implementacji systemu. W języku HTML, do umieszczenia na stronie elementu typu checkbox (lub radiobutton) służy polecenie

```
<INPUT TYPE=CHECKBOX VALUE=ON NAME="check">
```

Dzięki temu, po naciśnięciu klawisza SUBMIT i wywołaniu pliku HTML określonego w poleceniu ACTION, możemy określić, czy checkbox o nazwie „check” został przez użytkownika wciśnięty, czy nie (dyrektywa VALUE oznacza, że checkbox powinien być w stanie „on” po załadowaniu strony). Zbudowanie strony z większą ilością przycisków nie jest problemem, o ile nadamy im różne nazwy, np.

Please choose items you would like to order:

```
<TABLE BORDER=1>
```

```
<TR><TD><INPUT TYPE=CHECKBOX  
NAME="check01"></TD><TD>Shoes</TD><TR>  
<TR><TD><INPUT TYPE=CHECKBOX  
NAME="check02"></TD><TD>Hat</TD><TR>  
<TR><TD><INPUT TYPE=CHECKBOX  
NAME="check03"></TD><TD>Socks</TD><TR>  
<TR><TD><INPUT TYPE=CHECKBOX  
NAME="check04"></TD><TD>Suit</TD><TR>  
<TR><TD><INPUT TYPE=CHECKBOX  
NAME="check05"></TD><TD>Shirt</TD><TR>  
</TABLE>
```

Niestety, przy budowie interfejsu do bazy danych, często nie jesteśmy w stanie stwierdzić, ile krotek będzie liczyła tabela w momencie zapytania. Trzeba było opracować mechanizm budowy nazw i automatycznego ich wstawiania do strony HTML w zależności od ilości elementów. Standardowy HTML nie udostępnia takich możliwości, użyliśmy więc specyficznych własności PHP/FI. Początkowo próbowaliśmy wykorzystać zmienne tablicowe, ale okazało się, że zmienne te nie są przekazywane do pliku określonego parametrem ACTION⁵⁴. Ostatecznie, jedyną drogą okazało się wykorzystanie dynamicznej generacji HTML, jak pokazuje poniższy listing (podkreślamy, że **obydwa** listingi, mimo różnej formy to pliki HTML).

```
$result=mysql($database,"select * from items");  
$num=mysql_numrows($result);  
echo „Please choose items you would like to order:”;  
$i=0;  
echo "<TABLE BORDER=1>”;  
while ($i <= $num);  
echo "<TR>”;  
$name="check”;  
if ($i <= 10); $name+="0”; endif;
```

⁵⁴ Na błąd ten zwróciliśmy uwagę autorowi PHP/FI, który przyznał, że istotnie jego moduł nie realizuje jeszcze tej możliwości. Usterka ta zostanie prawdopodobnie usunięta w kolejnej wersji modułu.

```
$name+=$i;  
echo "<TD><INPUT TYPE=CHECKBOX NAME=/'$name/'></TD><TD>";  
echo mysql_result($result,$i,"name")+</TD>";  
echo "</TR>";  
$i++;  
endwhile;  
echo "</TABLE>";
```

Dodatkowym czynnikiem utrudniającym budowę dynamiczną interfejsu jest konieczność odpowiedniego „przechwycenia” danych w pliku wskazywanym dyrektywą ACTION. W przypadku statycznym nie sprawia to problemu, ponieważ z góry wiadomo ile zmiennych i o jakich nazwach trzeba będzie obsłużyć. Przypadek dynamiczny takich możliwości nie daje - konieczne staje się więc przekazanie dodatkowej informacji z pliku zawierającego formę - informacji o liczbie elementów na stronie. Najprościej można to uzyskać doklejając dodatkową zmienną do adresu URL (jest to metoda wykorzystywana klasycznie do transferu danych): np.

```
<FORM ACTION="/shopping/items_for_sale.html?liczba=$num">
```

Jest to jednak rozwiązanie nieeleganckie, ponieważ powoduje ukazywanie się w oknie przeglądarki „sztucznych” adresów URL (choć np. serwer Yahoo wykorzystuje tę metodę). W systemie BIS zastosowaliśmy mechanizm przekazywania przez tzw. pola utajone (które mimo występowania w nich tagu INPUT nie powodują wyświetlenia w przeglądarce pole do wprowadzenia danych):

```
<INPUT TYPE=HIDDEN NAME=liczba VALUE=$num>
```

W obu przypadkach, w pliku wskazanym w poleceniu ACTION można zastosować poniższy mechanizm. Warto zwrócić uwagę na linię `if ($$name!="on"); $comm+=" not ";endif;` - podwójny znak \$ nie jest błędem, lecz przykładem stosowania metazmiennych (nazwa autora modułu PHP/FI).

```
$i=0;  
while ($i <= $liczba);
```

```
$name="check";
if ($i<10); $i+="0"; endif;
$name+=$i;
$comm="You have";
if ($$name!="on"); $comm+=" not ";endif;
$comm+="ordered item #"+$i;
echo $comm;
$i++;
endwhile;
```

4.2.2. Ochrona przed błędnym kasowaniem wynikającym ze współbieżnego dostępu

Problem ten wiąże się z kwestią omawianą w 4.2.1 - generowaniem dynamicznych nazw elementów interfejsu. Jego istota łączy się ze specyfiką systemu. Rozpatrzmy sytuację, w której chcemy usunąć z bazy pewne dane - np. jedną (lub więcej) dziedzin. W tym celu na jednej ze stron HTML prezentujemy (za pomocą mechanizmu opisanego w 4.2.1) listę dziedzin wraz z odpowiadającymi im checkboxami. Po zaznaczeniu żądanych dziedzin i naciśnięciu klawisza SUBMIT, ładowana jest następna strona HTML - zawierająca zapytania do bazy danych (typu delete). Pomiędzy wypełnieniem pierwszej strony a wykonaniem drugiej może jednak nastąpić modyfikacja bazy danych - np. usunięcie jednej z krotek. Zapytanie delete usunie wówczas nie tę krotkę, którą chcemy, lecz następną ! Aby zabezpieczyć się przed tym, musimy zmodyfikować plik nr 1, umieszczając w nim informację o ID krotek do usunięcia:

```
$database="BIS";
$result=mysql($database,"select * from dziedziny);
$num=mysql_numrows($result);
echo "<INPUT TYPE=HIDDEN NAME=liczba VALUE=$num>";
echo "Please choose tuples to delete:";
$i=0;
echo "<TABLE BORDER=1>";
while ($i <= $num);
echo "<TR>";
$name="check";
```

```
if ($i <= 10); $name+="0"; endif;
$name+=$i;
echo "<TD><INPUT TYPE=CHECKBOX NAME=/"$name/"></TD><TD>";
$id="i";
if ($i <= 10); $id+="0"; endif;
$id+=$i;
echo "<INPUT TYPE=HIDDEN NAME=/"$id/" VALUE=$val>";
echo mysql_result($result,$i,"name")+</TD>";
echo "</TR>";
$i++;
endwhile;
echo "</TABLE>";
```

Dzięki temu, w pliku wskazanym przez ACTION, dysponować będziemy (w zmiennych o nazwie iXX, gdzie XX- numer krotki) identyfikatorem krotki do usunięcia. Jeśli zapewnimy, aby identyfikator był unikalny i niepowtarzalny również „w sensie czasu” (raz użyty ID nie zostaje ponownie przydzielony krotce, nawet po skasowaniu tej, której był przydzielony uprzednio), skasowanie nieodpowiedniej krotki nie będzie możliwe. Nadawanie identyfikatorów omówiono w p. 4.2.6.

4.2.3. Zmiany struktury bazy danych podczas pracy systemu

Zmiana struktury bazy danych jest jedną z najciekawszych cech systemu BIS. Aby było możliwe jej zaimplementowanie, musieliśmy stworzyć mechanizm, pozwalający na trzymanie informacji o nazwach tablic aktualnie używanych w systemie. Osiągnęliśmy to przez „emulację” tablic systemowych dużych baz danych - rolę tę pełni tabela SysTbName. Wpisuje się do niej nawy tablic systemu BIS. W ten sposób, system może w każdej chwili określić, jakie tablice wchodzi w jego skład, a także (z pośrednictwem interfejsu API mSQL) określić strukturę każdej z tablic (ilość kolumn, ich nazwy, typ, wielkość i rodzaj - np. kolumna nr 3 w tablicy T25: „wydawca”, „znakowy”, „char[40]”, „not null”).

4.2.4. Bezpieczeństwo formularzy - metody POST i GET

Metody POST i GET to dwie metody realizacji formularzy. Określają one sposób, w jaki przekazywane będą dane ze strony zawierającej formę, do strony zajmującej się obsługą wprowadzonych danych (odbywa się to przez specjalna

zmienną systemowa lub przez adres URL). Okazuje się, że standardowe metody mogą być potencjalnie niebezpieczne dla systemu - istnieje możliwość włamania przez odwołanie się do adresu URL, np. jeśli strona, zawierająca formę dodającą nowego użytkownika ma adres `www.bis.com/authorform.html` (natomiast w formie wywołujemy plik `add.html`), włamywacz może próbować samodzielnie ustawić zmienne, w których przechowuje się dane, podając w swojej przeglądarce adres:

```
www.bis.com/add.html?name="John"&right="root"
```

Aby uniknąć tego typu włamań nie należy stosować metody GET, ponieważ nie jest ona zabezpieczona przed „doklejaniem” zmiennych (rodzaj metody specyfikuje się w znaczniku FORM np. `<FORM ACTION=add.html METHOD=POST>`). W przypadku stosowania PHPFI warto także stosować mechanizm tzw. zmiennych bezpiecznych: wywołanie pliku html ze zmiennymi zainicjowanymi poza formą (przez „ręczne” doklejenie wartości do adresu URL) powoduje wystąpienie błędu, którego obsługę można przechwycić (i np. skierować przeglądarkę do innego pliku HTML).

4.2.5. Ukrywanie parametrów przekazywanych przez zmienne

Stosunkowo często wymagane jest przekazanie wartości zmiennych występujących w jednym z plików HTML do innego. Najpopularniejszą metoda jest przekazywanie za pomocą zmiennych doklejanych do adresu URL. Postać ogólna takiego adresu to:

```
http://www.adres.com/katalog/plik.html?zm1=wartość1  
&zm2=wartość2& ...
```

Wywołania takiego nie uważa się za rozwiązanie eleganckie - z uwagi na „sztuczność” adresu i na potencjalne niebezpieczeństwo włamania opisane w 4.2.4. Dodatkowo, wywołanie pliku bez podania wartości zmiennych może doprowadzić do nieprzewidzianych efektów. Zamiast tego, można zastosować mechanizm pól utajonych. Należy wówczas w pliku HTML wpisać linię:

```
<INPUT TYPE=HIDDEN NAME="nazwa" VALUE="wartość">
```

Adres pliku HTML pozostaje niezmienny, wartość jest przekazana w sposób „niezauważalny”. Niestety, sposób ten jest do zastosowania tylko w tych

plikach HTML, w których użyty jest tag <FORM>, ponieważ przekazywanie przez pola utajone jest częścią obsługi form.

4.2.6. Nadawanie identyfikatorów

Początkowo nadawaliśmy identyfikatory za każdym razem wyszukując wolne ID w tablicach. Metodę tę automatyzowaliśmy tworząc w języku HTML funkcję (dzięki zastosowaniu modułu PHPFI). Dzięki temu osiągnęliśmy pewne skrócenie kodu. Następnie dopracowaliśmy funkcję tak, że po usunięciu krotki o danym ID, możliwe było ponowne wykorzystanie identyfikatora. Niestety, nie znaleźliśmy metody pozwalającej na ominięcie problemów opisanych w 4.2.2. Stało się to możliwe dopiero po wykorzystaniu bazy danych mSQL 2.0 (w pierwszych wersjach systemu BIS używaliśmy jeszcze wersji mSQL 1.0) i systemowego mechanizmu sekwencji (sequence). Umożliwia on pobranie unikalnego identyfikatora dla danej tablicy i jednocześnie zapewnia jego niepowtarzalność „w czasie”. Zapewnia to ochronę spójności danych przy dostępie współbieżnym.

4.2.7. Ochrona procesu archiwizacji danych

Tak jak to pokazano w p. 4.2.7., dostęp do bazy danych (typu „write”) podczas archiwizacji mógłby zagrozić spójności bazy danych. Aby uniknąć takiego niebezpieczeństwa, stosujemy mechanizm śledzenia „transakcji” - każdy dostęp użytkownika umieszcza informacje o sobie w specjalnej bazie danych (BIS-T). Nie mogła być to wydzielona tablica systemu BIS, ponieważ w takim przypadku byłaby ona również archiwizowana. Informacją tą jest numer procesu i jedno typ (czy dostęp jest typu read, czy zawiera instrukcje modyfikujące - upadate, insert, bądź delete). W ten sposób, przed wykonaniem archiwizacji, wystarczy sprawdzić, czy trwające procesy mogą być „niebezpieczne” (np. zapis stanu bazy danych jest możliwy, gdy realizowane są jedynie zapytania typu select). Jeśli taki proces trwa, proces archiwizujący czeka na zakończenie się ich (po wykonaniu zapytań proces usuwa „swoją” krotkę z tablicy w bazie BIS-T). Oczywiście, proces archiwizujący musi zabezpieczyć się przed nowymi żądaniem użytkowników (w terminologii współbieżności sytuację taką określa się jako zagłodzenie procesu (patrz [20]). W tym celu, wpisuje on do bazy BIS-T swoją krotkę, która oznacza zdarzenie „proces archiwizacji czeka na możliwość rozpoczęcia działania”. Użytkownik, próbujący skorzystać z bazy danych (w sensie zapisu do niej) otrzyma komunikat „czynność

niemożliwa do zrealizowania z uwagi na trwającą archiwizację danych - proszę spróbować później”.

Zastosowany mechanizm jest bardzo wygodny dla użytkownika, ponieważ pozwala na realizację zapytań najczęściej stosowanych, a więc typu „read”. Jednocześnie warto w tym miejscu podkreślić, że stosowana metoda ochrony jest rozwiązaniem prawdopodobnie przekraczającym potrzeby systemu BIS (można przyjąć, że zapisy do bazy są tak rzadkie, a archiwizacja tak krótka, że uzasadniona byłaby nawet strategia ochrony przez wyłączanie systemu na czas backup’u - „off-line backup”) i można je traktować raczej jako przykład rozwiązania problemu ogólniejszego.

4.2.8. Ochrona dostępu - implementacja haseł.

Implementacja haseł możliwa jest w dwojaki sposób⁵⁵: poprzez interfejs Apache lub z wykorzystaniem PHPFI. Obie te metody bazują na wywoływaniu w przeglądarce okienka „Authenticate”, pozwalającego na wpisanie hasła i loginu użytkownika. Autoryzacja dotyczy połączenia z serwerem WWW, co oznacza, że po użytkownik raz rozpoznany, nie musi powtórnie wpisywać hasła, dopóki korzysta ze stron serwera jednego WWW (cacheowanie informacji nie zaburza działania tego mechanizmu). Warto tu wskazać, że strategia taka jest wygodna dla użytkownika (wpisanie kilku publikacji w czasie jednej sesji nie wymaga kilkukrotnego logowania), ale nakłada na niego rygor zakończenia połączenia po skończonej pracy (przez wyjście z aplikacji przeglądarki, lub odwołanie się do innego serwera WWW, ponieważ pozostawienie „otwartego” połączenia z serwerem BIS np. w ogólnodostępnym laboratorium pozwoliłoby osobom nieuprawnionym na korzystanie z praw pisarza).

4.2.9. Śledzenie zmian w bazie danych

Ponieważ baza danych ma teoretycznie dosyć szeroki dostęp (potencjalnie duża grupa pisarzy - np. pracownicy PG), wydaje się wskazane stworzenie systemu nadzoru wprowadzenia danych, aby administrator mógł określić, kto wprowadził określoną informację. W systemie BIS jest to realizowane przez dodawanie do

⁵⁵ Teoretycznie istnieje również możliwość wykorzystania form, ale jest to bardzo skomplikowane implementacyjnie (np. trzeba zapewnić bezpieczny przekazywanie uprawnień pomiędzy stronami HTML, co w przypadku wykorzystania możliwości Apache lub PHPFI jest realizowane systemowo) i nieefektywne.

wprowadzonych danych identyfikatora wprowadzającego (niestety, baza danych nie implementuje tego systemowo). Dzięki temu każdy z pisarzy może dowolnie aktualizować (oraz kasować) dane przez siebie wprowadzone, a Administrator może ustalić kto wprowadził błędne lub nieprawidłowe dane (np. stworzył niepotrzebną dziedzinę lub typ publikacji) i poinformować o tym ich autora, albo wyciągnąć z tego odpowiednie konsekwencje.

5. Testowanie systemu

Testowanie systemu miało na celu sprawdzenie jego wydajności, funkcjonalności i integralności. Opis przeprowadzonych czynności, wraz z ich wynikami i komentarze znajduje się w osobnym dodatku do niniejszej pracy.

5.1. Zbieżność z założonym projektem (stopień rozwiązania problemu)

Zaimplementowany system realizuje założenia projektowe w prawie 100 %. Jedyne różnice wiążą się ze zrezygowaniem z hierarchizacji administratorów, w ostatecznym rozwiązaniu tworzą oni grupę równouprawnioną, z wyróżnionym administratorem „root”, który jest ich zwierzchnikiem. Powodem tej zmiany jest konkluzja, że tworzenie hierarchii jest dość pracochłonne i czasochłonne, natomiast nie rozszerza ono w znacznym stopniu funkcjonalności systemu (hierarchia administratorów miała być zresztą zaimplementowana opcjonalnie i nie była jednym z wymagań nakładanych na system). Rozbudowa zarządzania administratorami jest możliwa w dalszych wersjach systemu BIS.

5.2. Koszt systemu

5.2.1. Uwagi wstępne

Zgodnie ze wcześniejszymi założeniami, przygotowaliśmy system zminimalizowany pod względem kosztów. System ten może być używany w placówkach niekomercyjnych i edukacyjnych - w innych sytuacjach konieczne jest uzyskanie opłat licencyjnych na serwer bazy danych mSQL. Poniżej zamieszczamy przykładowe kalkulacje cenowe, oparte na cennikach firm komputerowych z czerwca 1997.

Wszystkie oferty zakładają, że system BIS nie będzie pracował na dedykowanym komputerze - przy takim założeniu po instalacji można usunąć z zestawu monitor, klawiaturę, mysz bez pogorszenia się własności użytkowych systemu (administracja zdalna).

Zwraca uwagę, że system BIS zachowuje się wydajnie nawet w minimalnych konfiguracjach sprzętowych dostępnych na rynku.

5.2.2. Opcja minimalna (oszczędnościowa) - sprzęt używany

Lp.	komponent	koszt w PLN	uwagi
1.	system operacyjny Linux Debian w wersji CD	0	freeware,
2.	moduł PHP/FI	0	freeware,
3.	serwer bazy danych mSQL	0	shareware / eduware
4.	serwer Apache 1.2.7	0	freware
5.	procesor	50	486 DX 50MHz
6.	płyta główna	200	486
7.	obudowa	75	MiniTower
8.	monitor	303	mono 14"
9.	pamięć	87	4MB
10.	dysk twardy	400	540 MB
11.	karta grafiki	50	Trident 521KB
12.	klawiatura	15	standard
13.	karta sieciowa	95	NE2000 COMBO 2 lata gw.
14.	Mysz	10	?
	RAZEM	1195	

Tabela 22: Kosztorys systemu BIS - sprzęt używany

5.2.3. Opcja minimalna (oszczędnościowa) - sprzęt nowy

Lp.	komponent	koszt w PLN	uwagi
1.	system operacyjny Linux Debian w wersji CD	0	freeware,
2.	moduł PHP/FI	0	freeware,
3.	serwer bazy danych mSQL	0	shareware /

			eduware
4.	serwer Apache 1.2.7	0	freeware
5.	procesor	103	486 DX2 66MHz PCI
6.	płyta główna	287	486 PCI
7.	obudowa	96	MiniTower
8.	monitor	303	mono 14"
9.	pamięć	87	4MB
10.	dysk twardy	600	brak oferty poniżej 1 GB
11.	karta grafiki	89	Trident 521KB
12.	klawiatura	15	standard
13.	karta sieciowa	95	NE2000 COMBO 2 lata gw.
14.	Mysz	19	ACT
	RAZEM	1684	

Tabela 23: Kosztorys systemu BIS - sprzęt nowy

5.3. Ocena zastosowanej metodologii

Podczas prac nad systemem stosowana była metodologia obiektowa Rumbaugh. Jej użycie nie nastęrczało wielkich problemów, mimo, że system miał „charakter” bardziej relacyjny niż obiektowy. Etap analizy praktycznie nie wymagał odejść od klasycznego toku OMT, pewne rozbieżności pojawiły się dopiero w trakcie projektu. Warto także wspomnieć, że implementacja nie mogła być przeprowadzona w języku obiektowym, ponieważ HTML nie ma praktycznie żadnych cech obiektowych (HTML jest właściwie językiem opisu strony, więc przypisywanie mu nawet cech języków strukturalnych jest pewną nieścistością, dopóki nie stosuje się modułu PHPFI, rozbudowującego go o instrukcje iteracyjne, warunkowe, zmienne itp.).

5.4. Bezpieczeństwo

System BIS cechuje się dość wysokim, jak na standardy niekomercyjnych systemów WWW poziomem bezpieczeństwa. Należy podkreślić, że zastosowanie

bardziej zaawansowanych mechanizmów kontroli - np. protokołu SSL wiązałyby się z dodatkowymi kosztami finansowymi lub wydajnościowymi (np. zainstalowanie oprogramowania firewall zmusiłoby nas do stosowania lepszego procesora i prawdopodobnie poszerzenia przestrzeni dyskowej zajmowanej przez system). Mając na uwadze, że system BIS ma gromadzić jedynie dane bibliograficzne (a więc o niewielkim ryzyku w przypadku włamania⁵⁶), należy ocenić jego bezpieczeństwo jako dobre lub nawet bardzo dobre⁵⁷.

5.5. Wydajność

Przeprowadzone testy (patrz dodatki) wykazują, że jedno z podstawowych kryteriów oceny wydajności naszego systemu - szybkość realizacji zapytań jest „mocnym punktem” naszego systemu. Podczas przeprowadzonych prób, okazało się, że zapytania realizowane są stosunkowo szybko (trudno jest zmierzyć ten czas przy pracy zdalnej z uwagi na duże zakłócenia pomiaru wprowadzane przez zmienny czynnik opóźnienia sieci⁵⁸). Warto tu podkreślić, że testy te były przeprowadzane na komputerze o przeciętnej szybkości - 486DX. Jedynym mankamentem, jaki stwierdzony został podczas tych testów, jest dość szybki przyrost wielkości plików bazy danych przy wstawianiu nowych elementów). Prawdopodobnie wiąże się to z istnieniem plików indeksowych (wstawianie do tablic nieindeksowanych wykazuje mniejsze przyrosty plików).

5.6. Współbieżność

Współbieżność w systemie BIS została zaimplementowana z użyciem mechanizmów ochrony integralności danych. Mechanizmy te zabezpieczają całkowicie przed naruszeniem spójności danych przez dostęp współbieżny, nie ograniczają jednak funkcjonalności programu (np. podczas operacji backup możliwy jest dostęp czytelników do bazy). Należy podkreślić, że w następnej wersji mSQL zostanie umożliwione wykorzystywanie transakcji, co pozwoli na dalsze ulepszanie

⁵⁶ Z oczywistych względów systemy operujące na danych „finansowych” np. sprzedaż internetowa są ochraniać znacznie silniej, ale też i koszty, które użytkownik ponosi za stworzenie mechanizmów zabezpieczenia systemu stanowią dużo większy procent całości kosztów systemu.

⁵⁷ Współczynnik: zysków związanych z włamaniem do systemu do kosztów jego złamania jest u nas bardzo niski (koszty złamania systemu mogą być szacowane np. przez określenie nakładów finansowych, jakie należy ponieść do złamania zabezpieczenia np. dla szyfru DES, używanego w kodowaniu haseł w systemie Unix - niedawno koszt taki był szacowany na kilka tysięcy dolarów dla jednego klucza - patrz [4]).

systemu BIS (ograniczenie kodu, przeniesienie ciężaru ochrony na mechanizmy systemowe). Szczególną uwagą powinno się przyłożyć do ochrony procesu archiwizacji - obecna forma (wydaje się być ona najlepszą możliwą z wykorzystaniem dostępnych mechanizmów) jest wrażliwa na awarię systemu podczas trwania procesu - pozostawienie krotki „procesu backup” powoduje zablokowanie dostępu typu „write”. Administrator powinien wtedy ręcznie usunąć tę krotkę (sytuacja taka jest stosunkowo mało prawdopodobna - konieczne byłby wypadek awarii podczas trwania backup’u).

5.7. Odporność na błędy i awarie. Odtwarzanie danych.

W chwili obecnej system BIS nie ma mechanizmów pozwalających na automatyczne naprawianie błędów. W sytuacji nieprawidłowego działania potrafi on jedynie stwierdzić, gdzie występuje nieprawidłowość i poinformować o tym użytkownika (np. „Cannot read from table Writers”). BIS pozwala także na wykrycie błędów polegających na powtórnym wprowadzeniu słowa kluczowego, dziedziny itp.

System BIS nie posiada zaawansowanych mechanizmów odtwarzania danych (np. automatycznego przywracania spójności danych). Mechanizmy takie towarzyszą zwykle transakcyjnym bazom danych i są często oparte o różnorakie formy zapisywania podejmowanych czynności (logi transakcji) - co stanie się możliwe dopiero po zastosowaniu nowej wersji motoru bazy danych. W przypadku prostych systemów, wysublimowane systemy zabezpieczeń nie są zresztą zwykle potrzebne.

5.8. Modyfikowalność

Dziedzina ta jest silną stroną programu BIS. Zarówno projekt, jak i sam kod były tworzone z uwzględnieniem przyszłej rozbudowy systemu. Możliwe jest także wykorzystanie BIS jako podstawy do stworzenia bazy danych o zupełnie innej tematyce. Służy temu bogaty zestaw napisanych przez nas funkcji⁵⁹ (/bis/private/functions) do zarządzania informacją wyświetlana na ekranie (listy, dynamiczny przydział nazw zmiennym, automatyczne tworzenie elementów typu checkbox i radiobutton itp.) i zarządzaniem danymi (tworzenie nowych tablic, zarządzanie użytkownikami i grupami użytkowników). Warto podkreślić, że niektóre

⁵⁸ Duże opóźnienia pojawiały się np. w czasie pracy z bazą danych O2 w laboratorium Jakości Oprogramowania.

⁵⁹ Ich stworzenie możliwe jest dzięki modułowi PHP/FI - w klasycznym języku HTML nie jest to wykonalne

rozwiązania zastosowane w projekcie, znacznie przekraczały swoją funkcjonalnością potrzeby systemu BIS - pisane były właśnie z myślą o wykorzystaniu w późniejszych rozbudowach i modyfikacjach,

5.9. Ocena końcowa systemu

System BIS sprawdza się jako wytworzony produkt, odpowiadający na konkretne potrzeby. Wydaje się nam, że w hipotetycznym przypadku zaistnienia przetargu na rozwiązanie systemu informacji bibliograficznej dla Zakładu Zastosowań Informatyki (z możliwością wykorzystania w innych placówkach edukacyjnych), nasz system miałby duże szanse zwycięstwa. Oczywiście, stosunkowo nietrudno byłoby podać przykład systemu o większych możliwościach (zawansowana baza danych - jak Oracle, MS SQL Server, czy Informix, serwer WWW o większych możliwościach - np. MS Internet Information server, interfejs Java lub Active X itp.), ale system taki nie miałby większych szans w konkurencji z naszym - w aspekcie finansowym. Podstawową bowiem zaletą BIS, jest jego niska cena i dobre dopasowanie do warunków środowiska, w jakim będzie pracował. Nie sztuką jest bowiem zrobić dobry produkt przy zaangażowaniu wielkich nakładów, sztuką jest zrobić produkt dobry niskimi kosztami.

6. Przyszłość systemu BIS

6.1. Transakcje

Transakcje są pierwszą modyfikacją, której powinien zostać poddany BIS. Niestety, nie mogły one zostać od początku zaimplementowane w systemie z uwagi na ich brak w serwerze mSQL. Firma Hughes Inc. zapowiada jednak na najbliższy czas nową wersję motoru, będącego już pełną transakcyjną bazą danych (implementacja znaczników⁶⁰ czasowych miała być pierwszym etapem na drodze do budowy takiego serwera). Po implementacji transakcji kod systemu BIS będzie mniejszy i bardziej odporny na błędy. Zapewne będzie również dostępny log transakcji, co poprawi możliwości systemu w zakresie przywracania stanu bazy po awarii.

6.2. System doradzania oparty na analizie stanu bazy danych

System ten to idea wypracowana w końcowej fazie projektu BIS. Jego implementacja nie wchodziła jednak w skład wymagań, więc nie został on włączony w skład systemu BIS. W ogólnych zarysach ma on służyć pomocą przy wprowadzaniu danych, sugerując dobór odpowiednich słów kluczowych, dziedzin i przedmiotów. Jeśli bowiem do bazy danych wprowadzono kilkanaście publikacji na temat np. grafiki komputerowej i formatów plików graficznych, podając w każdej z nich zestaw słów kluczowych „bmp” „tif” „jpg”, „gif”, kolejny pisarz wprowadzając publikację (na ten temat) i podając słowa kluczowe „bmp” i „jpg” powinien rozważyć dodanie „tif” i „tif”. Osiągnąć to można przechowując informację o związkach między wprowadzonymi słowami kluczowymi. System ten może doradzać także dobór odpowiednich dziedzin na podstawie słów kluczowych (wprowadzając słowo „skaner” warto jako dziedzinę podać obok „informatyka” również i „elektronika”, o czym użytkownik mógł zapomnieć).

6.3. Rozszerzenie badania wstawianych elementów

W chwili obecnej system analizuje tylko dokładne wystąpienia słów kluczowych. Oznacza to, że np. po wprowadzeniu słowa „wektor” można (na skutek przypadku) wprowadzić słowo „wektory” (powtórne wprowadzenie słowa „wektor” spowoduje wygenerowanie komunikatu ostrzegawczego). Wydaje się celowe

⁶⁰ Dostępne od wersji mSQL 2.0

wprowadzenie dodatkowego rozszerzenia tego systemu, opartego o „niedokładne” porównywanie wartości (np. „Jack” jest „niedokładnie” równe „Jacek”). Metoda taka podwyższa poziom zabezpieczeń przed wprowadzeniem słowa w obcej pisowni (np. interfejs - interface), w innej formie gramatycznej („postaci” - „postacie”⁶¹) lub błędnie wprowadzonego („wektor” - wejtor”). Warto przy tym wskazać, że celowe wydaje się rozróżnianie miejsca na którym występują różnice pomiędzy wyrazami - wystąpienie na końcu może sugerować inną formę gramatyczną, na początku - błąd pisowni. Dalsze rozważania pozostawiamy osobom rozwijającym system BIS.

⁶¹ Obie formy są poprawne (wg. Słownika jez. polskiego prof. Dobrowolskiego)

7. Literatura

- [1] „SSH Communications Security” Tatu Ylonen <ylo@ssh.fi> March 22, 1997
Network Working Group
- [2] „Object Oriented Analysis and Design” - Rumbaugh, 1995
- [3] „Internet” - [red], README Warszawa 1993
- [4] Dorothy E. Robling-Denning "Kryptografia i ochrona danych", WNT 1993
- [5] „Budowa systemu operacyjnego Unix” - M. Bach - WNT Warszawa 1995
- [6] Net World - roczniki 1995, 1996
- [7] Stan Schatt „Zrozumieć sieci lokalne” Intersoftland 1994
- [8] Craig Hunt, TCP/IP Administracja Sieci, README, Warszawa 1996
- [9] A. Marciniak „Słownik Informatyczny angielsko-polski” PWN, Warszawa-Poznań 1991
- [10] R. Sobczak, P. Szpryngier, W. Molisz, W. Pawlak, S. Połomski, D. Markowski, W. Mościbrodzki „Strategia wdrażania Internetu dla TP S.A.” - analiza teoretyczna, Gdańsk 1996/97
- [11] Douglas Maughan, Mark Schertler, Mark Schneider, Jeff Turner : „Internet Security Association and Key Management Protocol (ISAKMP)”, IPSEC Working Group 1997
- [12] „Object Oriented Analysis and Design”, C. Booch - Benjamin and Cummings Publishing Co. 1994
- [13] „SSH Transport Layer Protocol” , Network Working Group, Tatu Ylonen <ylo@ssh.fi>
- [14] „SSH Authentication Protocol”, Network Working Group, Tatu Ylonen <ylo@ssh.fi>
- [15] Transport Layer Security Working Group „The SSL Protocol, Version 3.0” Alan O. Freier, Philip Karlton (Netscape Communications) Paul C. Kocher (Independent Consultant) Nov. 1996
- [16] „Security Extensions For HTML” E. Rescorla, A. Schiffman Terisa Systems, Inc
- [17] „The Secure HyperText Transfer Protocol” E. Rescorla, A. Schiffman, Terisa Systems, Inc.
- [18] „Considerations for Web Transaction Security” Network Working Group, G.

Bossert, S. Cooper (Silicon Graphics Inc.), W. Drummond (IEEE, Inc.)

January 1997

- [19] „Nowa koncepcja skryptów CGI’ A. Montefusco, Software 8/96 AVT -
Korporacja
- [20] „Programowanie współbieżne” - Zbigniew Weiss, WNT 92
- [21] „Internet” - roczniki 1996, 1997
- [22] „Software” - roczniki 1996,1997

ponadto w pracy wykorzystano dane udostępniane na wielu serwerach WWW (konkretne adresy podano w treści) oraz notatki z następujących wykładów:

- Relacyjne bazy danych
- Obiektowe bazy danych
- Systemy operacyjne
- Metody wytwarzania oprogramowania
- Metody prezentacji informacji
- Metody modelowania systemów
- Analiza i projektowanie systemów informatycznych
- Techniki uruchamiania oprogramowania