ISBN 978-83-921270-5-5

Warszawska Wyższa
Szkoła Informatyki
ul. Lewartowskiego 17
00-169 Warszawa

www.wwsi.edu.pl

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

informatyka+ tom 2 Multimedia, technologie internetowe, bazy danych i sieci komputerowe

ZBIÓR WYKŁADÓW WSZECHNICY POPOŁUDNIOWEJ

tom 2

Multimedia, technologie internetowe, bazy danych i sieci komputerowe

informatyka+

Człowiek – najlepsza inwestycja

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Multimedia, technologie internetowe, bazy danych i sieci komputerowe

Publikacja współfinansowana ze środków Unii Europejskiej
w ramach Europejskiego Funduszu Społecznego

Człowiek – najlepsza inwestycja

Tom 2.
Multimedia, technologie internetowe, bazy danych i sieci komputerowe

Redaktor merytoryczny: prof. dr hab. Maciej M. Sysło
Redaktor: Magdalena Kopacz

Publikacja opracowana w ramach projektu edukacyjnego Informatyka+
– ponadregionalny program rozwijania kompetencji uczniów szkół ponadgimnazjalnych
w zakresie technologii informacyjno-komunikacyjnych (ICT)
www.informatykaplus.edu.pl
kontakt@informatykaplus.edu.pl

Wydawca:
Warszawska Wyższa Szkoła Informatyki
ul. Lewartowskiego 17, 00-169 Warszawa
www.wwsi.edu.pl
rektorat@wwsi.edu.pl

Wydanie pierwsze

Copyright©Warszawska Wyższa Szkoła Informatyki, Warszawa 2011
Publikacja nie jest przeznaczona do sprzedaży.

ISBN 978-83-921270-5-5

Projekt graficzny: FRYCZ I WICHA

Wszystkie tabele, rysunki i zdjęcia, jeśli nie zaznaczono inaczej, pochodzą z archiwów autorów lub zostały przez nich opracowane.

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

ZBIÓR WYKŁADÓW WSZECHNICY POPOŁUDNIOWEJ

tom 2

Multimedia, technologie internetowe, bazy danych i sieci komputerowe

informatyka+

MULTIMEDIA, TECHNOLOGIE INTERNETOWE, BAZY DANYCH I SIECI KOMPUTEROWE
ZBIÓR WYKŁADÓW WSZECHNICY POPOŁUDNIOWEJ

WSTĘP

Zgodnie z założeniami, projekt Informatyka+ – ponadregionalny program rozwijania kompetencji uczniów szkół ponadgimnazjalnych w zakresie technologii informacyjno-komunikacyjnych (ICT) ma na celu podniesienie poziomu kluczowych kompetencji uczniów szkół ponadgimnazjalnych w zakresie informatyki i jej zastosowań, niezbędnych do dalszego kształcenia się na kierunkach informatycznych i technicznych lub podjęcia zatrudnienia, oraz stworzenie uczniom zdolnym innowacyjnych możliwości rozwijania zainteresowań naukowych w tym zakresie. Program ten jest alternatywną formą kształcenia pozalekcyjnego.

Realizacja projektu zbiegła się w czasie ze światowym kryzysem kształcenia informatycznego. Od upadku dot.comów na początku tego wieku, aż o 50% spadło zainteresowanie studiami informatycznymi w Stanach Zjednoczonych i podobne tendencje zaobserwowano w Wielkiej Brytanii oraz w innych krajach. Powszechny i łatwy dostęp do najnowszej technologii komputerowej i prostota w opanowaniu jej podstawowych funkcji doprowadzają młodych użytkowników tej technologii do przekonania, że posiedli jej najważniejsze tajniki i szkoda czasu na głębsze studia w tym kierunku. Rynek pracy jednak jest w stanie wchłonąć każdą liczbę wysoko i średnio wykwalifikowanych informatyków i specjalistów z dziedzin pokrewnych.

Projekt Informatyka+ jest formą działań określanych mianem *outreach*, które są adresowane przez uczelnie do uczniów i mają na celu głębsze zaprezentowanie, czym jest informatyka, przybliżenie jej zastosowań oraz wskazanie możliwości dalszego kształcenia się w kierunkach związanych z profesjonalnym wykorzystaniem technologii komputerowej i technologii informacyjno-komunikacyjnej. Inicjatywę tę należy uznać za niezmiernie aktualną i potrzebną, wpisującą się zarówno w myślenie o przyszłości dziedziny informatyka i o przyszłych karierach młodych Polaków w zawodach informatycznych, jak i rozwoju nowoczesnego państwa. Szczegółowe informacje o projekcie i jego efektach są zamieszczane na stronie http://www.informatykaplus.edu.pl/.

Niniejszy zbiór wykładów prowadzonych w ramach Wszechnicy Popołudniowej, stanowiącej jedną z form realizacji projektu, oddajemy przede wszystkim do rąk uczniów. Tom 2 zawiera wykłady z zakresu Multimediów, Baz danych i Sieci komputerowych.

Pierwsza grupa tematów dotyczy korzystania z sieci Web 2.0, czyli poszukiwania treści multimedialnych oraz funkcjonowania i tworzenia stron internetowych, zwłaszcza w wersji dynamicznej, oraz podstaw budowy obrazów i dźwięków i ich kodowania w wersji elektronicznej.

Druga część zagadnień zawartych w tomie odnosi się do baz danych, mających bardzo ważne zastosowania praktyczne. Obecnie większość dobrze zorganizowanych zasobów danych i informacji w sieci występuje w postaci baz danych, co ułatwia ich przeszukiwanie, utrzymywanie i rozbudowę.

Wreszcie ostatnia grupa tematów jest związana z techniczną stroną budowy sieci komputerowych. Przedstawione zagadnienia dotyczą działania sieci, również w wersji bezprzewodowej, wybranych usług sieciowych oraz bezpieczeństwa w sieci w odniesieniu do informacji i korzystających z nich użytkowników.

Do lektury zamieszczonych tekstów wystarczy znajomość matematyki i informatyki na poziomie szkoły ponadgimnazjalnej. Nowe pojęcia są wprowadzane na przykładach i w sposób intuicyjny, umożliwiający nadążanie za tokiem wykładu.

Prof. dr hab. Maciej M. Sysło
Merytoryczny Koordynator
Projektu Informatyka+

Warszawa, jesienią 2011 roku

Multimedia, grafika i technologie internetowe

W poszukiwaniu treści multimedialnych

Witryna w Internecie – zasady tworzenia i funkcjonowania

Obraz jako środek przekazu informacji

Metody kodowania i przechowywania sygnałów dźwiękowych

W poszukiwaniu treści multimedialnych

Piotr Kopciał
Politechnika Warszawska
piotrkopcial@gmail.com

Streszczenie

Pojęcie „multimedia” jest dzisiaj w powszechnym użytku. Ale czy tak naprawdę wiemy, co się pod nim kryje? Czy potrafimy wskazać elementy, składające się na przekaz multimedialny? Pierwsza część wykładu jest poświęcona przybliżeniu pojęcia multimediów i podkreśleniu ich znaczenia w pracy, nauce i rozrywce. Następnie przedstawiono charakterystykę treści i form multimedialnych dostępnych w Internecie. Zasadnicza część wykładu dotyczy strategii i narzędzi służących do efektywnego wyszukiwania informacji w Internecie w postaci graficznej, dźwiękowej i filmowej. Dzięki strumieniowemu przesyłaniu muzyki i filmów w sieci możliwe jest obecnie słuchanie przekazu dźwiękowego i oglądanie filmów bezpośrednio na stronie WWW; użytkownik może także pobierać te zasoby na swój komputer. Coraz większą rolę odgrywają otwarte zasoby edukacyjne w sieci z różnych dziedzin, wśród których te najatrakcyjniejsze przyjmują postać przekazu multimedialnego. Prezentacja jest bogato ilustrowana ciekawymi stronami internetowymi, zawierającymi m.in. demonstracje, symulacje zjawisk, nagrania dźwiękowe i filmy. Wykład służy uporządkowaniu i rozszerzeniu wiedzy na temat multimedialnych zasobów Internetu i umiejętności korzystania z nich.

Spis treści

1 WPROWADZENIE

Pojęcia multimedia używa się dość często. Już kilkanaście lat temu komputery stacjonarne nazywano komputerami multimedialnymi. W otaczającym nas świecie multimedia są wszechobecne. Ale, czy tak naprawdę zdajemy sobie sprawę, co się kryje za tym określeniem. Czym są multimedia? Czy potrafimy wskazać poszczególne elementy składające się na przekaz multimedialny?

W części wstępnej oprócz przybliżenia pojęcia multimediów zwrócimy uwagę na ich znaczenie w naszym codziennym życiu.

Multimedia to techniki komputerowe, umożliwiające łączenie wielu sposobów przekazywania informacji: dźwięku, obrazu, animacji, tekstu oraz słowa mówionego w jeden przekaz. Multimedia można zatem rozumieć jako połączenie wielu mediów (sposobów przekazywania informacji). Cechą charakterystyczną przekazu multimedialnego jest zaangażowanie użytkownika (tzw. interakcja z użytkownikiem). Przykładem może być komputer multimedialny, za pomocą którego możemy nie tylko oglądać filmy i słuchać muzyki, ale także grać w gry komputerowe, czy też rozmawiać przez Skype (internetowy komunikator) z osobą znajdującą się w dowolnym miejscu na Ziemi, widząc ją na ekranie monitora i słysząc jej głos w głośnikach, a także czytając jej wypowiedzi wyświetlane na ekranie.

Jak multimedia ułatwiają nam życie

Być może nie zdajemy sobie z tego sprawy, ale multimedia dostępne za pośrednictwem Internetu znakomicie ułatwiają nam życie:

- Zakupy przez Internet są udogodnieniem dla osób, dla których wyjście z domu jest problemem: osób opiekujących się dziećmi, chorymi, osób niepełnosprawnych.
- Wideokonferencje umożliwiają jednoczesną rozmowę osób znajdujących w różnych częściach kuli ziemskiej. Spotkanie się tych osób wiązałoby się z dużymi kosztami i poświęceniem czasu.
- Podczas dłuższych podróży multimedia umożliwiają obserwację przebiegu podróży, relaks przy oglądaniu filmów lub słuchaniu audycji muzycznych, oglądanie wiadomości; słowem: umożliwiają „kontakt ze światem”.
- Internetowe telefony i komunikatory (np. Skype) pozwalają zaoszczędzić na kosztach rozmów telefonicznych.
- Elektroniczne biblioteki udostępniają swe zasoby dla większego grona czytelników. Znika problem braku książki na półce, „bo ktoś ją wypożyczył przed nami”. Z książki w wersji elektronicznej może korzystać wiele osób jednocześnie.
- Wirtualne muzea umożliwiają poznanie dziedzictwa kulturowego osobom, które być może nigdy nie odwiedziłyby miejsca znajdującego się na innym kontynencie.
- Gry komputerowe, gry sieciowe stanowią znakomitą rozrywkę.

Zastosowania multimediów

Zastosowania multimediów dotyczą wielu dziedzin współczesnego życia:

- Zarówno w szkole, jak i w pracy spotykamy się z prezentacjami multimedialnymi, przygotowanymi najczęściej w programie Power Point.
- W domu korzystamy z komunikatorów internetowych, takich jak np. Gadu Gadu, Skype.
- Coraz większą rolę odgrywają multimedia w edukacji – na płytach CD są dostarczane encyklopedie multimedialne (zawierające oprócz definicji słownych również zdjęcia i animacje), słowniki (umożliwiające odsłuchanie brzmienia danego słowa wypowiadanego w obcym języku) oraz kursy języków obcych (wzbogacone o elementy zabawy: quizy i gry).
- Powszechne staje się korzystanie z nawigacji GPS w czasie podróży, np. samochodem.

Wirtualna rzeczywistość, uzyskiwana przy użyciu multimediów, znajduje zastosowanie nie tylko w grach komputerowych:

- Młodzi piloci, przed objęciem sterów prawdziwego samolotu, szkolą swoje umiejętności na symulatorach lotów.

- Architekci projektujący budynki i mosty najpierw tworzą ich konstrukcje w komputerze. Specjalne programy (tzw. programy CAD, ang. *Computer Aided Design*) służą do badania konstrukcji budynków lub mostów, wyliczania działających obciążeń i naprężeń itd.
- Osoby szykujące się do egzaminu z prawa jazdy mogą doskonalić swoje umiejętności przy użyciu komputerowego symulatora.

2 MULTIMEDIA

Z uwagi na ogromne znaczenie multimediów w naszym życiu, warto przyjrzeć im się bliżej.

2.1 ELEMENTY PRZEKAZU MULTIMEDIALNEGO

Elementami przekazu multimedialnego są:

- tekst;
- obraz;
- animacja;
- film;
- dźwięk;

jak również:

- trójwymiarowa grafika (tzw. grafika 3D);
- dźwięk dookólny (przestrzenny);
- napisy, wyświetlane podczas projekcji filmów obcojęzycznych;
- hipertekst, czyli wzajemne powiązanie pomiędzy dokumentami (stronami internetowymi) za pośrednictwem tzw. hiperłączy.

Grafika

Grafika jest jednym z najważniejszych elementów przekazu multimedialnego. To dzięki niej przekaz multimedialny zyskuje na atrakcyjności. Mówi się, że *jeden obraz jest wart więcej niż tysiąc słów*. Słowa te oddają, jak wiele możemy przekazać przy pomocy elementów graficznych. Najprostszym tego przykładem są zdjęcia.

Przy konstruowaniu obrazów bierze się pod uwagę właściwości ludzkiego wzroku. Można na przykład przedstawić informację trójwymiarową na płaskim (dwuwymiarowym) ekranie. Można stworzyć optyczne wrażenie głębi w obrazie, czyli efekt trójwymiarowości.

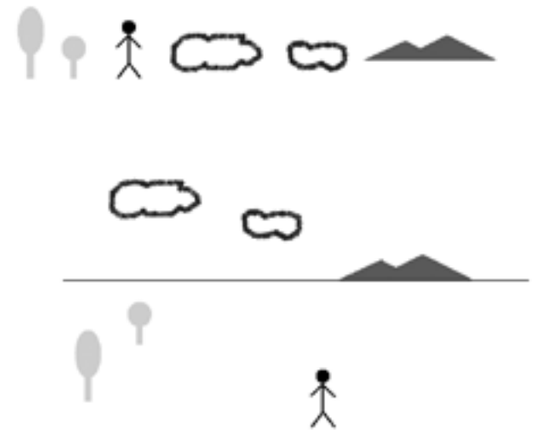
Aby to zrobić, wykorzystuje się techniki stosowane od dawna w malarstwie. Zademonstrujemy dwie techniki. Pierwszą z nich jest **częściowe przesłanianie** obiektów, co służy zilustrowaniu wzajemnych relacji przestrzennych pomiędzy obiektami (rys.1). Na rysunku po lewej stronie dwa obiekty znajdują się obok siebie. Wydają się leżeć w jednej płaszczyźnie. Natomiast po prawej stronie obiekt przesłaniany wydaje się być głębiej w obrazie, niż obiekt przesłaniający. Obraz uzyskuje cechy trójwymiarowej głębi.

Druga metoda uzyskiwania głębi polega na podzieleniu obrazu za pomocą **linii widnokręgu** (istniejącej w wyobraźni obserwatora). Ta pozioma linia rozdziela obraz na część górną (która odpowiada niebu) i dolną (odpowiadającą ziemi). Na rysunku 2 w górnej części widzimy obiekty umieszczone obok siebie. Wydają się one leżeć w jednej płaszczyźnie. W dolnej części rysunku obiekty znajdujące się bliżej linii widnokręgu wydają się być położone głębiej w obrazie (od tych, które są bardziej oddalone od linii widnokręgu, a tym samym znajdujące się bliżej obserwatora).

Animacja

Animacja komputerowa powstaje przez wyświetlenie serii obrazów następujących po sobie. Stwarza to wrażenie ruchu na ekranie. Współczesne bajki animowane tworzone są przy użyciu specjalistycznych programów komputerowych.

Rysunek 1.
Technika przesłaniania obrazów, służąca uzyskaniu efektu głębi w obrazie



Rysunek 2.
Dodanie linii widnokręgu, służące uzyskaniu efektu głębi w obrazie

Czy wiecie, skąd wzięło się potoczne określenie bajek – kreskówki? Otóż dawniej należało bajkę rysować klatka po klatce. Szybkie nałożenie klatek na siebie stwarzało wrażenie ruchu. Dziś robi się to w inny sposób. Twórca animacji, posługując się programem komputerowym, definiuje jedynie obraz pomiędzy tzw. klatkami kluczowymi, a obiekt porusza się zgodnie z parametrami ruchu zdefiniowanymi w programie komputerowym.

Film

Film składa się z **klatek**, tzw. kadrów. Pomiędzy poszczególnymi kadrami zachodzą zmiany, a szybkie przełączanie kadrów stwarza wrażenie ruchu.

Zazwyczaj film trwa stosunkowo długo (około dwóch godzin), co wymaga ogromnej liczby kadrów. Z tego powodu przy zapisie filmu stosuje się kompresję. Kompresja polega na zakodowaniu serii klatek na podstawie podobieństwa pomiędzy nimi. Jest to kompresja stratna – pewne informacje o obrazie (np. szczegóły) są tracone. Jednakże oko ludzkie nie jest w stanie wychwycić tak znikomych strat w obrazie.

Dźwięk

Zastosowanie dźwięku podnosi atrakcyjność przekazu multimedialnego. Dotyczy to szczególnie przekazu wielokanałowego, a co najmniej stereofonicznego.

Czy zastanawialiście się kiedyś, dlaczego film oglądany w kinie robi większe wrażenie niż oglądany w telewizji? To nie wielkość ekranu ma aż takie znaczenie, a właśnie system nagłośnienia. Widownię otacza kilkanaście głośników, rozmieszczonych w odpowiedni sposób. Dzięki temu uzyskuje się efekt określany jako tzw. **brzmienie przestrzenne**. Takie efekty uzyskuje się w przypadku dźwięku wielokanałowego. Każda ścieżka dźwiękowa nagrywana jest osobno w innym kanale. Znajomość właściwości ludzkiego słuchu pozwala odtwarzać ścieżki dźwiękowe w taki sposób, abyśmy mogli rozróżnić głosy dobiegające z prawej strony ekranu, słyszeć strzały w oddali, a także dźwięk zamykanych za nami drzwi.

2.2 CO JEST POTRZEBNE DO KORZYSTANIA Z MULTIMEDIÓW

Multimedia w sieci Internet to dwa nierozłączne tematy. Pierwszym z nich są źródła multimediów, takie jak:

- strony z plikami graficznymi, audio i wideo do pobrania;
- strony oferujące strumieniową transmisję audio i wideo;
- radio internetowe.

Drugim tematem są narzędzia służące do odtwarzania multimediów z Internetu, pobierania i słuchania/oglądania. Odtwarzanie multimediów wiąże się z pewnymi wymaganiami dotyczącymi sprzętu i oprogramowania. Sprzętem jest komputer. Niezbędne oprogramowanie to program komputerowy, nazywany **odtwarzaczem**. Możemy mieć do czynienia z odtwarzaczami: obrazu, animacji, dźwięku i filmów.

Sprzęt

Każdy współczesny komputer jest wyposażony w procesor oraz kartę dźwiękową na tyle wydajne, aby odtwarzać dźwięki. Najważniejszym elementem komputera, odpowiedzialnym za odtwarzanie multimediów, jest karta graficzna, od której zależy, czy będziemy mogli oglądać obraz wideo w trybie pełnoekranowym (odpowiednio wysoka rozdzielczość), czy obraz będzie płynny (czy nie będzie się „zacinać"). Prawdą jest, że jakość odtwarzanego obrazu zależy również od odtwarzanego pliku oraz szybkości połączenia internetowego, jednakże odpowiednio szybka karta graficzna odgrywa największą rolę. Współczesne komputery są wyposażone w zintegrowaną kartę graficzną, która w zupełności wystarcza, aby czerpać przyjemność z multimedialnych dobrodziejstw Internetu.

Mamy już komputer wyposażony w procesor, kartę dźwiękową oraz graficzną. Czy to wystarczy? Nie. Potrzebne są jeszcze głośniki. W przypadku komputerów przenośnych (tzw. laptopów) głośniki umieszczone są wewnątrz komputera. Do komputera stacjonarnego należy dołączyć je oddzielnie. Do rozmowy np. przez Skype, będziemy potrzebować także mikrofon, a jeśli chcemy widzieć osobę, z którą rozmawiamy – to również kamerę internetową.

Oprogramowanie

Mamy do czynienia z różnego rodzaju odtwarzaczami audio i wideo, przystosowanymi do określonego rodzaju (formatu) plików. Często używamy innego odtwarzacza do muzyki, a innego do filmów.
Najczęściej odtwarzacze multimediów można pobrać za darmo z Internetu. Często są one instalowane razem z systemem operacyjnym. Przykładem takiego odtwarzacza jest program Windows Media Player, który odtwarza zarówno dźwięk, jak i obraz.

Do korzystania z multimedialnych zasobów Internetu niezbędna jest również przeglądarka internetowa. Współczesne techniki sieciowe umożliwiają odsłuchiwanie muzyki i oglądanie filmów bezpośrednio ze strony internetowej. Jest to możliwe, ponieważ współczesne przeglądarki są wyposażone w funkcje obsługi elementów multimedialnych umieszczonych na stronach internetowych.

2.3 STRUMIENIOWANIE

Strumieniowanie to technika rozsyłania informacji (multimedialnych danych), w sposobie działania podobna do tradycyjnej telewizji czy radia. Informacje przesyłane są ze strony źródłowej w postaci strumienia – ciągu danych. Są one odbierane (i odtwarzane) przez komputer użytkownika na bieżąco, w momencie ich przybycia. Techniką strumieniowania są przesyłane obrazy, dźwięki, teksty oraz inne dane multimedialne. Najważniejszą cechą strumieniowania jest to, że informacje są rozsyłane nie w postaci pliku, lecz strumienia danych. Strumieniowanie ma wiele zastosowań:

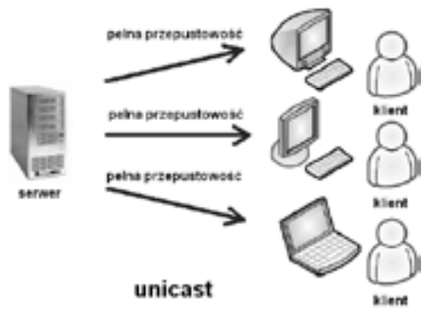
- rozrywka – np. zastąpienie tradycyjnego radia przez audycje internetowe;
- monitoring – ochrona budynków;
- edukacja na odległość – transmisja prezentacji i wykładów przez Internet;
- medycyna – transmisja zabiegów chirurgicznych, konsultacje na odległość z lekarzami.

W metodzie strumieniowania sygnał trafia do odbiorcy natychmiast po nawiązaniu połączenia ze źródłem (dostawcą mediów). W Internecie przyjęło się nazywać źródło danych **serwerem**, natomiast odbiorcę – **klientem**.

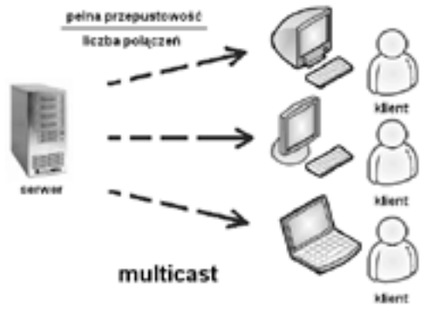
Strumieniowanie może odbywać się na dwa sposoby: unicast oraz multicast.

1. W metodzie **unicast** (rysunek 3) przepływność łączy pomiędzy serwerem a klientami jest jednakowa. Każdy odbiorca (klient) otrzymuje dobrej jakości strumień danych. Metoda unicast wymaga łączy o sporych przepustowościach (tzw. łącza szerokopasmowe) i z tego powodu stosowana jest najczęściej w sieciach lokalnych (komputery w obrębie jednego budynku lub firmy).

2. W metodzie **multicast** (rysunek 4) przepustowość łącza, przez które płynie strumień danych, dzielona jest pomiędzy wszystkich odbiorców (klientów). Zaletą tej metody jest możliwość obsłużenia kilku odbiorców nawet przy niewielkiej przepustowości łącza. Wadą natomiast jest to, że wraz ze wzrostem liczby odbiorców maleje jakość sygnału. Multicast stosuje się najczęściej w Internecie. Można zaobserwować, że filmy na YouTube ładują się szybciej w godzinach porannych, kiedy mniej osób korzysta z komputerów, natomiast wolniej działają w godzinach wieczornych.



Rysunek 3.
Ilustracja strumieniowania typu unicast



Rysunek 4.
Ilustracja strumieniowania typu multicast

3 WYSZUKIWANIE INFORMACJI W INTERNECIE

Korzystanie z zasobów Internetu wydaje się łatwe, dopóki polega na przeglądaniu stron o znanych nam adresach (takich jak np. portale informacyjne: onet.pl, gazeta.pl itp.). W tym przypadku wystarczy wpisać w przeglądarce adres strony, nacisnąć [Enter], a po chwili na ekranie zostanie wyświetlony zbiór informacji.

Niestety, takie możliwości nie wystarczają na długo. Prędzej czy później przyjdzie moment, w którym nie będziemy znać adresów właściwych stron, na których znajdują się poszukiwane przez nas informacje, np. zwyczaje dinozaurów żyjących kiedyś na naszej planecie. Szansa na to, że przypadkiem natkniemy się na stronę z takimi informacjami, odwiedzając wielotematyczne serwisy informacyjne, jest niewielka. W Internecie istnieją przecież miliardy stron. Na szczęście posiadamy systemy ratujące nas z takiej sytuacji – ułatwiające wyszukiwanie informacji w Internecie. Są to serwisy wyszukiwawcze, zwane potocznie wyszukiwarkami.

Aby w sposób świadomy korzystać z dobrodziejstw Internetu, konieczne jest poznanie sposobu działania wyszukiwarek. Tylko wtedy będziemy w stanie szybko i efektywnie odnaleźć w gąszczu światowych zasobów sieciowych informacje, których szukamy.

3.1 WYSZUKIWARKA I ZASADA JEJ DZIAŁANIA

Wyszukiwarka to strona internetowa dająca dostęp do bazy danych, zawierającej katalog słów kluczowych i adresów stron, na których te słowa występują. Kiedy wpisujemy słowo w polu wyszukiwania i naciskamy [Szukaj], polecamy wyszukiwarce przeglądanie bazy, odszukanie adresów stron i wyświetlenie ich w postaci listy uporządkowanej według stopnia prawdopodobieństwa napotkania słowa (które wpisaliśmy w polu wyszukiwania) na stronie.

Uwaga! Nie należy mylić terminów *wyszukiwarka i przeglądarka*. Wyszukiwarka to strona internetowa, której zadaniem jest wyszukiwanie innych stron. Natomiast **przeglądarka** to program komputerowy, służący do oglądania stron internetowych. Przykładem przeglądarki jest Internet Explorer i Mozilla Firefox, a wyszukiwarki – Google.

Wyszukiwarka jako narzędzie służące do przeszukiwania Internetu, składa się z:

1. *Robotów* – są to programy, które wędrując po sieci zbierają informacje ze stron. Roboty przeglądają opisy stron (w trybie tekstowym) znajdujące się na serwerach;
2. *Indekserów* – programów, które na podstawie informacji zebranych przez roboty budują bazę danych – *Indeks* napotkanych stron;
3. *Indeksu* – jest to baza danych o odpowiedniej strukturze. W bazie tej wyszukiwarka przeprowadza wyszukiwanie.

Ponadto, niektóre roboty podążają za linkami znalezionymi na stronie, indeksując w ten sposób także inne strony, powiązane z daną witryną. Baza danych zawiera posortowane informacje o stronach odwiedzonych przez robota. Baza ta jest nieustannie aktualizowana o najnowsze informacje (np. zmiany i aktualizacje na stronach).

Wyszukiwarka tworzy ranking stron na podstawie słów kluczowych, nagłówka strony oraz złożonego algorytmu, specyficznego dla każdej wyszukiwarki. Przykładowo, może przypisywać punkty za to, ile razy słowo występuje na stronie lub w których miejscach strony występuje. Każdej zaindeksowanej stronie przypisywane są punkty określające jej miejsce w rankingu.

Popularne wyszukiwarki

W naszym kraju prym wśród wyszukiwarek wiedzie Google. Korzysta z niej 97% polskich internautów. Drugie i trzecie miejsce zajmują MSN i NetSprint [http://ranking.pl/pl/rankings/search-engines.html].

W światowym rankingu wyszukiwarek pierwszą pozycję zajmuje również Google, osiągając w grudniu 2009 roku liczbę 87 mld zapytań. Kolejne miejsca przypadają wyszukiwarkom Yahoo (9,4 mld), Baidu.com (8,5 mld), Microsoft SN (prawie 4 mld) [za: http://www.comscore.com/Press_Events/Press_Releases/2010/1/Global_Search_Market_Grows_46_Percent_in_2009].

3.2 STRATEGIA WYSZUKIWANIA W INTERNECIE

Aby szukanie przyniosło efekty, musimy się w pierwszej kolejności zastanowić, czego szukamy. Oto przykładowe pytania, na które powinniśmy sobie odpowiedzieć:

- Czego dotyczy mój problem?
- Jakie zagadnienia są z nim związane?
- Jakimi słowami mogę je wyrazić?

W odpowiedzi na powyższe pytania zgromadzimy listę słów kluczowych, od których zaczniemy wyszukiwanie.

Proste wyszukiwanie

Istnieje wiele wyszukiwarek internetowych: Yahoo, Alta Vista itp., ale, jak wspomniano wyżej, wśród nich jedna zasługuje na wyróżnienie. Oczywiście: Google. Ponieważ większość polskich internautów używa właśnie tej wyszukiwarki, w omawianych przykładach będziemy najczęściej odwoływać się do Google. Proste wyszukiwanie polega na wpisaniu słów kluczowych w polu wyszukiwania i naciśnięciu [Szukaj].

Zawężanie wyników wyszukiwania

Proste wyszukiwanie zazwyczaj sprawdza się znakomicie. Jednakże gdy musimy odszukać więcej informacji na dany temat, zaczynają się problemy. Zwróćmy uwagę, że wyświetlenie odnośników do kilku tysięcy stron jest tylko na pozór dobrą informacją. Po pierwsze, przeszukanie choćby kilkuset z nich zajęłoby nam kilka dni. Po drugie większość z tych stron będzie raczej dotyczyć innej tematyki. W takiej sytuacji pojawia się konieczność zawężenia wyników wyszukiwania. Jak to zrobić?

Efekt ten uzyskujemy poprzez podanie bardziej jednoznacznych słów kluczowych oraz przez wyłączenie słów, które nie powinny występować.

Łączenie słów znakiem cudzysłowu

Najprostszym zabiegiem jest złączenie słów kluczowych, które powinny występować razem. Przykładowo, wpisując słowa kluczowe *prędkość światła*, odszukamy również strony zawierające tekst „prędkość samochodu, który ma włączone światła". Podobnych stron będzie ponad 400 000 [20.08.2009].

Natomiast kiedy słowa kluczowe zamkniemy znakami cudzysłowu: *„prędkość światła"*, to zostaną odnalezione tylko te strony, na których słowa te występują obok siebie. Wyniki wyszukiwania z ponad 400 000

zmniejszą się do około 26 000 stron [20.08.2009]. Widzimy zatem, że ogromna liczba stron odnalezionych w pierwszym przypadku była zupełnie nie na temat.

Przeszukiwanie tylko polskich stron

Zawężenie wyników wyszukiwania możemy uzyskać również poprzez zaznaczenie pola [Szukaj na stronach kategorii: język polski]. Pole to należy zaznaczyć przed rozpoczęciem wyszukiwania. W przypadku słów polskojęzycznych, różnica w liczbie odnalezionych stron nie będzie duża. Jednakże przy szukaniu obcojęzycznych terminów, lista odnalezionych stron może skrócić się wielokrotnie (np. ze 119 mln do 1,13 mln jak na rysunku 5).

Rysunek 5.
Zawężenie wyników wyszukiwania poprzez przeszukiwanie wyłącznie polskich stron

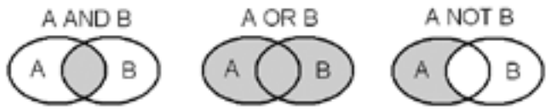
Użycie łączników

Współczesne wyszukiwarki są bardziej „inteligentne” niż nam się wydaje. „Rozumieją”, kiedy chcemy odszukać stronę zawierającą *słowo_pierwsze* **i** *słowo_drugie*, *słowo_pierwsze* **lub** *słowo_drugie*, a także, kiedy chcemy odszukać stronę zawierającą *słowo_pierwsze* **bez** *słowa_drugiego*. Słowa **i**, **lub**, **bez** to tak zwane **łączniki**, a ich angielskie odpowiedniki to **AND**, **OR**, **NOT**.

Znaczenie tych łączników dla wyszukiwarki jest następujące:

- łącznik AND oznacza, że powinny zostać wybrane strony, na których słowa kluczowe (terminy) występują jednocześnie;
- OR oznacza, że na zwróconych stronach powinno występować jedno lub drugie słowo kluczowe;
- natomiast NOT wskazuje, jakie terminy nie powinny występować na zwróconych stronach.

Działanie łączników zilustrowano symbolicznie na rysunku 6.



Rysunek 6.
Działanie łączników AND, OR i NOT – rezultat jest zaciemniony

Przykładowo, jeśli szukamy informacji o *samolotach* i *lotnikach*, to powinniśmy użyć łącznika AND, wpisując w wyszukiwarce: *samolot AND lotnik*. Jeśli natomiast szukamy informacji o samolotach lub lotnikach, to

użyjemy łącznika OR i w wyszukiwarce wpiszemy *samolot* OR *lotnik*. Jeśli natomiast chcemy ograniczyć wyszukiwanie do stron na temat lotników, ale nie odnoszących się do samolotów, użyjemy łącznika NOT, wpisując w wyszukiwarce *lotnik* NOT *samolot*.

Uwaga! Łączniki AND, OR, NOT wpisujemy WIELKIMI LITERAMI. Łączniki możemy zapisywać skrótowo przy użyciu znaków:

- + (plus) oznacza AND;
- – (minus) oznacza NOT;
- natomiast brak znaku odbierany jest jako OR.

Zaprezentujemy teraz przykład użycia łącznika NOT. Przykładowo, jeśli poszukujemy informacji na temat procesorów i nie chcemy sięgać do zasobów Wikipedii, powinniśmy wpisać w polu wyszukiwania wyrażenie procesory – *wikipedia* (rysunek 7). Minus stojący przed słowem wikipedia oznacza, że strony zawierające to słowo będą zignorowane (i nie zostaną umieszczone na liście wyników wyszukiwania).

Rysunek 7.
Przykład użycia łącznika NOT

Wyszukiwanie zaawansowane

Przykłady przedstawione wcześniej umożliwiają szybkie odnalezienie wartościowych informacji poprzez skuteczne zawężenie wyników wyszukiwania. Jednakże w przypadku niektórych słów, często używanych lub szeroko opisywanych w Internecie, konieczne jest jeszcze bardziej drastyczne zawężenie wyników wyszukiwania. Należy wtedy skorzystać z mechanizmu **wyszukiwania zaawansowanego**.

W celu uruchomienia wyszukiwania zaawansowanego, należy kliknąć odnośnik [Szukanie zaawansowane], który znajduje się na stronie głównej Google. Po kliknięciu odnośnika ujrzymy formularz, który umożliwi sprecyzowanie zasad wyszukiwania. Wystarczy wypełnić formularz i kliknąć przycisk [Szukaj w Google]. Strony będą wyszukiwane zgodnie z wytycznymi w formularzu.

Uwaga! Formularz wyszukiwania zaawansowanego zawiera wiele pól. Jeśli nie jesteśmy pewni, co zaznaczyć w danym polu (lub nie potrzebujemy którego z kryteriów wyszukiwania zaawansowanego), najlepiej pozostawmy pole puste.

3.3 CO ROBIĆ, JEŚLI NIE ZNAJDUJEMY ODPOWIEDZI NA NASZE PYTANIE

Pomimo że w odpowiedzi na nasze pytanie wyszukiwarka zwraca tysiące stron, nadal często zdarza się, że nie znajdujemy rozwiązania naszego problemu. Co wtedy?

Możemy rozszerzyć listę słów kluczowych, za pomocą których szukamy. Nawet jeśli odnalezione strony nie przynoszą pełnej odpowiedzi na postawione przez nas pytanie, to możemy na nich znaleźć terminy i pojęcia pomocne w określeniu tego, czego szukamy. Zanotujmy je i użyjmy jako słowa kluczowe w wyszukiwarce.

Jeśli to nie pomoże, możemy użyć innej wyszukiwarki. Różne wyszukiwarki mają dostęp do różnych stron. Lista wyników wyświetlona w odpowiedzi na nasze zapytanie będzie się różnić. Dlatego warto korzystać z wielu wyszukiwarek.

Jeśli potrafimy posługiwać się językiem obcym, warto skorzystać z wyszukiwarki w obcym języku. Największą popularnością cieszą się wyszukiwarki anglojęzyczne. Należy zdawać sobie sprawę, że informacje

na temat nowości pojawiających się na świecie dochodzą do Polski z pewnym opóźnieniem. Obszerne informacje na najnowsze tematy niekiedy są dostępne tylko w języku angielskim.
Możemy użyć tzw. multiwyszukiwarki. **Multiwyszukiwarka** to system korzystający z kilku wyszukiwarek na raz. Wyniki zwrócone przez każdą z wyszukiwarek są zbierane, powtarzające się odnośniki są usuwane, co w rezultacie prowadzi do wyświetlenia ostatecznej listy wyników. Multiwyszukiwarka oszczędza czas, który poświęcilibyśmy na skorzystanie z kilku wyszukiwarek po kolei. Przykładowe multiwyszukiwarki to: Search.com (http://www.search.com/), MetaCrawler (http://www.metacrawler.com/) oraz Dogpile (http://www.dogpile.com/).

Przeszukiwanie grup dyskusyjnych
Jeszcze innym sposobem znalezienia informacji na temat, którego szukamy jest korzystanie z tematycznych grup dyskusyjnych. **Grupę dyskusyjną** można sobie wyobrazić jako miejsce w Internecie, gdzie użytkownicy wymieniają się poglądami i wzajemnie sobie doradzają. Grupa dyskusyjna zajmuje się konkretną tematyką, gromadząc specjalistów i entuzjastów w danej dziedzinie.

Zasada działania grupy dyskusyjnej polega na tym, że użytkownik porusza jakiś temat, umieszczając wiadomość na stronie, a inni mogą ją komentować, odpowiadać. W ten sposób rodzi się dyskusja, w której udział może wziąć każdy. Z Internetu korzystają codziennie miliony użytkowników. Jest zatem bardzo prawdopodobne, że ktoś wcześniej spotkał się z problemem, który mamy. Zadając pytanie grupie dyskusyjnej, znajdziemy tam również odpowiedzi udzielone przez innych użytkowników.

Należy przy tym pamiętać, że jeśli mamy jakiś problem, to przed zadaniem pytania grupie dyskusyjnej, najpierw przeczytajmy znajdujące się na niej wiadomości. Jest bardzo prawdopodobne, że ktoś zadał podobne pytanie już wcześniej, inna osoba odpowiedziała – a my mamy gotowe rozwiązanie!

Grupy dyskusyjne okazują się bardzo przydatne w sytuacjach podobnych do poniższych:
- utknąłem w grze komputerowej, w którą gra już niewiele osób;
- mam stary aparat fotograficzny i chcę dowiedzieć się o nim więcej;
- potrzebuję pomocy w obsłudze maszyny do szycia kupionej w 1995 roku.

Aby skorzystać z grup dyskusyjnych, należy wpisać w wyszukiwarce całą frazę lub zdanie. Sprawdzonym narzędziem przeszukiwania grup dyskusyjnych jest wydzielona część wyszukiwarki Google, do której dostęp uzyskamy po kliknięciu w odnośnik [Grupy dyskusyjne] na stronie głównej http://www.google.pl. Pojawi się strona umożliwiająca przeszukiwanie, wyszukiwanie oraz zakładanie grup dyskusyjnych. W celu przeszukania archiwów grup dyskusyjnych należy wpisać w polu wyszukiwania słowa kluczowe, np. „zmiana domyślnej przeglądarki Firefox Internet Explorer Windows" i nacisnąć [Przeszukuj grupy]. W rezultacie otrzymamy listę tematów poruszanych na różnych grupach dyskusyjnych, związanych z naszym zapytaniem.

Polska wyszukiwarka grup dyskusyjnych to np. Niusy na portalu Onet.pl (http://niusy.onet.pl). Menu w lewej części pomaga we wstępnej selekcji grup. W odróżnieniu od Google, która jest wyszukiwarką o zasięgu międzynarodowym, Niusy Onet dotyczy tylko polskich grup dyskusyjnych.

Wyszukiwanie w katalogu
Można odnieść wrażenie, że wiedza na temat wyszukiwania informacji w Internecie, którą do tej pory nabyliśmy, jest w zupełności wystarczająca na co dzień. Jest to jednak błędne przekonanie. Wyszukiwarki sprawdzają się dobrze, kiedy szukamy konkretnych informacji, związanych z kilkoma słowami kluczowymi. Jednak do wyszukiwania serwisów internetowych związanych z pewnym obszarem tematycznym, najlepszym narzędziem są **serwisy katalogowe**, potocznie zwane internetowymi **katalogami**.

Czym jest serwis katalogowy? Jest to strona internetowa, na której są zgromadzone informacje o tematyce innych stron. Przykładowo, możemy zgłosić własną stronę internetową do katalogu, podając słowa

kluczowe, które określają tematykę naszej strony. Kiedy szukamy strony poświęconej żeglowaniu, wystarczy że odwiedzimy katalog, wybierzemy kategorię Sport, Rekreacja, Jeziora lub podobną, a otrzymamy listę adresów stron poświęconych tej tematyce. Przykładowe katalogi to Onet.pl (http://katalog.onet.pl/), Gazeta.pl (http://szukaj.gazeta.pl), Hoga.pl (http://www.hoga.pl/) oraz Open Directory Project (http://www.dmoz.org/) i Yahoo (http://dir.yahoo.com).

Należy zwrócić uwagę, że katalog działa na innej zasadzie niż wyszukiwarka. Wyszukiwarka sama odwiedza strony i przypisuje im słowa kluczowe. Natomiast w przypadku katalogu, to autor strony decyduje, do jakiej kategorii ma być przypisana jego strona (poprzez podanie kilku charakterystycznych słów kluczowych).

W celu przejrzenia zasobów katalogu Onet.pl należy wpisać w przeglądarce adres http://katalog.onet.pl/ i nacisnąć [Enter]. Pojawi się strona, na której oprócz pola wyszukiwania (działającego podobnie jak w wyszukiwarce) znajduje się lista kategorii, w które pogrupowane są strony internetowe. Aby dotrzeć do stron o interesującej nas tematyce, należy wybierać kolejne, coraz bardziej szczegółowe nazwy kategorii. Przykładowo, możemy przejść od kategorii ogólnej *Sport i Turystyka* przez kategorie pośrednie, np. *Ośrodki sportowe i rekreacyjne*, do kategorii szczegółowej np. *Stanice wodne*.
Uwaga! Zawartość danej kategorii możemy dalej poddawać przeszukiwaniu. Jeśli na przykład w wybranej przez nas kategorii *Stanice wodne* chcemy odszukać ośrodki znajdujące się w Mikołajkach, w polu [Gdzie szukać] zaznaczamy opcję [Wybrana kategoria], następnie w polu tekstowym wpisujemy *Mikołajki*, i naciskamy [Szukaj]. Po chwili lista stron zostanie zaktualizowana.

Google również oferuje wyszukiwanie w katalogu. Katalog Google (http://www.google.pl/dirhp/) jest zarządzany i aktualizowany przez grupę ochotników liczącą 20 000 osób. Każda z tych osób jest odpowiedzialna za konkretną kategorię. Dzięki temu strony są zamieszczane we właściwych kategoriach, a przypisy do stron są zrozumiałe.

Cenne wskazówki: jak właściwie zadawać pytania, jakich błędów unikać
1. **Jako słowa kluczowe stosuj przede wszystkim rzeczowniki.**
   Czasowniki i przymiotniki mogą dotyczyć wielu zagadnień, przez co wyniki wyszukiwania nie będą trafne. Przykładowo, *atrakcyjna* może być osoba, praca, lokata, oferta, czy też atrakcyjne miejsce do wypoczynku. Zatem wpisanie tylko tego przymiotnika dostarczy wiele stron o różnorodnej tematyce.

2. **Uwzględniaj liczbę mnogą.**
   Wpisując przykładowo *rower, rowery, rowerem,* zawęzisz wyszukiwanie do tych stron, na których jest najwięcej informacji na temat rowerów, np. poświęconych rowerzystom w miastach, rowerowej turystyce, kolarstwu, portalom dla rowerzystów. Pominięte zostaną strony, na których słowo *rower* pojawia się przypadkowo, np. w jakimś artykule na portalu informacyjnym.

3. **Unikaj używania słów bardzo popularnych.**
   Setki tysięcy odnalezionych stron to nie jest dla Ciebie dobra wiadomość. Większość z nich w ogóle nie będzie na temat, który Cię interesuje.

4. **Używaj kilku wyrazów, fraz w jednym zapytaniu.**
   Na przykład: *„nowa planeta" „system słoneczny" odkrycie OR znalezienie*. Rezultaty wyszukiwania zostaną zredukowane do tych najbardziej miarodajnych.

5. **W pierwszej kolejności wpisuj słowa najważniejsze.**
   Aby wyszukiwarka „wiedziała" co jest szukane. Strony są wybierane na podstawie tzw. metaznaczników, dzięki którym można wyróżnić informacje na stronie pod względem ważności.

6. **Używaj wyszukiwarek obsługujących łączniki.**
   Dzięki użyciu łączników AND, OR, NOT możesz sprecyzować wyniki wyszukiwania, przez co trafienia będą dokładniejsze.

Fenomen Google i smutna prawda

Kiedy wejdziemy na stronę Google, uderza nas ona swą prostotą. W odróżnieniu od portali informacyjnych, zawierających „najświeższe informacje", na stronie głównej znajdujemy jedynie logo Google oraz pole do wpisania szukanych informacji. Jednakże pod tą prostotą kryje się potężny mechanizm, który co i raz zachwyca nas umiejętnością odnajdywania poszukiwanych przez nas informacji. Nieraz byliśmy zdziwieni, jak szybko i instynktownie Google potrafi odgadywać, czego dokładnie szukamy.

Smutna prawda

1. **Jednakże żadna wyszukiwarka nie wie wszystkiego, nawet Google.**
   Wyszukiwarka ta indeksuje ponad 8 miliardów stron. Należy zdawać sobie sprawę, że informacje w sieci Internet nieustannie się zmieniają – tak szybko, że nie sposób za tymi zmianami nadążyć. Istnieją zatem strony, o których Google po prostu nie wie. Poza tym informacje umieszczane są w Internecie w przeróżnych formatach (nie tylko dokumenty HTML, pliki Word, PDF czy PPT). Niektórych formatów danych wyszukiwarka nie potrafi odczytać i „zrozumieć". Jest też w Internecie wiele ukrytych zasobów, o które trzeba wiedzieć, gdzie i jak zapytać, np. zasoby bibliotek. Często szukanie informacji wiąże się z wypełnieniem pól formularza na stronie.

2. **W Internecie nie ma wszystkiego.**
   Pomimo tego, że w Internecie istnieją miliardy stron, jest prawdopodobne, że na żadnej z nich nie ma informacji, których szukamy.
   Poza tym w Internecie nie można umieścić wszystkiego. Obserwujemy tendencję do umieszczania coraz większej ilości treści multimedialnych na stronach internetowych. Treści te są plikami graficznymi, muzycznymi, video i innymi, i jako takie nie mogą mieć zbyt dużego rozmiaru, ponieważ taka strona ładowałaby się bardzo wolno. Zmniejszenie rozmiaru plików wpływa na obniżenie ich jakości. Dlatego np. filmy oglądane w Internecie nigdy nie będą takiej jakości jak te, oglądane w domu na DVD.

3. **Wyniki wyszukiwania nie są stałe.**
   Roboty odwiedzają strony w Internecie i zbierają informacje o zmianach i aktualizacjach ich treści. Zatem wyniki wyszukiwania dla tego samego zapytania mogą się zmieniać z dnia na dzień.

4. **Wyniki wyszukiwania w Google nie są aktualnym obrazem stanu sieci Internet.**
   Zazwyczaj mija kilka dni, zanim robot Google odwiedzi nowo utworzoną witrynę i dostarczy wyszukiwarce informacji o niej. Zatem należy oczekiwać, że Google dowie się o naszej stronie z pewnym opóźnieniem.

Przestroga

- Patrzmy krytycznie na to, co znajdziemy w Internecie. Pamiętajmy, że każdy może założyć własną stronę internetową i umieszczać na niej co chce. Nie wszystkie znalezione przez nas informacje będą wiarygodne.
- W odróżnieniu od książki lub czasopisma, często nie wiemy, kto jest autorem treści umieszczonej na stronie. Poza tym nikt nie sprawdza błędów, nie recenzuje, nie ma redaktora nadzorującego, jak w przypadku książki drukowanej.
- Zwróćmy uwagę, kto jest autorem strony. Jeśli stronę prowadzi Polska Partia Przyjaciół Piwa, podejdźmy do takiej strony z ostrożnością.
- Nauczmy się odróżniać artykuły promocyjne i reklamowe od właściwych treści. Jeśli np. dany model aparatu fotograficznego opisywany jest jako bezkonkurencyjny, może warto poczytać na jego temat również na stronie innej niż witryna producenta.

- Nie dajmy się nabrać. Wiele stron tworzonych jest dla żartu. Niektóre strony mogą być parodią tych prawdziwych. Przykład: http://georgewbush.com oraz http://gwbush.com.

4 WYSZUKIWANIE MULTIMEDIÓW W INTERNECIE

W tej części powiemy, w jaki sposób możemy korzystać z multimedialnych zasobów Internetu.

4.1 WYSZUKIWANIE OBRAZÓW I ANIMACJI

W podobny sposób jak z wyszukiwarki stron internetowych, możemy skorzystać z mechanizmu wyszukiwania obrazów. Wystarczy kliknąć odnośnik [Grafika] znajdujący się na stronie głównej serwisu Google. Wyszukiwarka grafiki jest bliźniaczo podobna do wyszukiwarki stron.

W celu odnalezienia obrazów i grafik należy wpisać w polu wyszukiwania dowolne słowo, np. *mazury* i nacisnąć [Szukaj obrazów]. Po chwili na ekranie pojawi się galeria miniatur obrazów.

Kliknięcie konkretnej miniatury prowadzi nas do strony zawierającej dany obraz. Jeśli naszym celem jest wyświetlenie jedynie obrazu, a nie strony, która zawiera dany obraz, należy kliknąć na miniaturze znajdującej się w górnej części wyświetlonej strony. Obraz zostanie wyświetlony w pełnych rozmiarach, a po kliknięciu prawym klawiszem myszy na obrazie, możemy go zapisać na dysku.

Zaawansowane wyszukiwanie obrazów

Podobnie jak w przypadku wyszukiwania stron, wyświetlone wyniki mogą nie spełniać naszych potrzeb. Pomijając ogromną liczę wyników, jednym z częstych problemów jest zbyt mała rozdzielczość znalezionych obrazów. Z tego powodu, również w przypadku wyszukiwania grafiki, pomocny jest mechanizm wyszukiwania zaawansowanego.

Pierwszym sposobem na przefiltrowanie wyników jest wybranie rozmiaru obrazów: małe, średnie lub duże obrazy. Narzędzie do filtrowania według rozmiaru dostępne jest na stronie wyników wyszukiwania, a po wybraniu szukanej wielkości obrazów, następuje natychmiastowe przeładowanie strony z wynikami.

Drugim sposobem na zawężenie wyników wyszukiwania jest wpisanie słów kluczowych w odpowiedni sposób w polu wyszukiwania. Podobnie jak w przypadku wyszukiwania stron możemy łączyć słowa kluczowe zamykając je znakami cudzysłowu, jak również wyłączać z wyszukiwania słowa kluczowe poprzedzając je znakiem minus.

Trzecim sposobem jest skorzystanie z formularza zaawansowanego wyszukiwania grafiki. Formularz ten dostępny jest z poziomu strony głównej wyszukiwarki grafiki Google. Przy użyciu formularza możemy definiować słowa kluczowe, z którymi mają być powiązane wyszukiwane obrazy, albo słowa, które nie powinny się znajdować na stronie zawierającej dany obraz. Możemy określać rozmiar (rozdzielczość) wyszukiwanego obrazu, konkretny typ plików (JPEG, GIF lub PNG), zabarwienie (obrazy czarno-białe lub kolorowe). Możemy również wskazać fragment adresu strony, który powinien pojawić się w adresie wyszukiwanych obrazów. Po wypełnieniu formularza naciskamy przycisk [Szukaj w Google].

Nie tylko Google oferuje wyszukiwanie grafiki w sieci. Możliwość taką dają także wyszukiwarki MSN, Yahoo oraz NetSprint. Zwrócimy jednak uwagę na wyszukiwarkę firmy Microsoft o nazwie Bing, która charakteryzuje się nowoczesnym interfejsem graficznym. Po wpisaniu zapytania otrzymamy w wyniku inne obrazy, niż przy użyciu Google.

Zwróćmy uwagę, że mamy również inne możliwości sortowania wyników (opcje umieszczone w lewej części strony). Po najechaniu kursorem myszy na wybrany obrazek, zostaje on powiększony i pojawiają się dodatko-

we informacje na jego temat. Wszystkie znalezione obrazki są wyświetlane na jednej stronie. Możemy przesuwać je za pomocą suwaka umieszczonego po prawej stronie okna. Wszystko to sprawia, że korzystanie z tej wyszukiwarki jest bardzo wygodne.

Warto zwrócić uwagę na obrazy ruchome – animacje (najczęściej występujące w formacie .GIF). Animacje można wyszukać wpisując w polu wyszukiwania np. fish.GIF, jeśli chcemy znaleźć animację ryby.

4.2 SŁUCHANIE, POBIERANIE I ODTWARZANIE MUZYKI

Słuchanie transmisji muzycznych

W Internecie znaleźć można wiele źródeł muzyki. Dobrym przykładem strony udostępniającej dźwięki jest Wrzuta.pl (http://wrzuta.pl/). Po wejściu na stronę główną klikamy na przycisku [audio], znajdującym się na prawo od pola wyszukiwania. Zostaniemy przekierowani na stronę z plikami muzycznymi. Wystarczy kliknąć na wybranym tytule, aby rozpocząć strumieniowe pobieranie dźwięku na nasz komputer. Za pomocą przycisków możemy zatrzymywać odtwarzanie utworu, przewijać go, regulować głośność (rysunek 8).

Rysunek 8.
Słuchanie muzyki przez Internet

Inną, wartą polecenia witryną muzyczną jest Polskastacja.pl (http://www.polskastacja.pl). Na tej stronie możemy wybierać rodzaj muzyki, która nas interesuje. Znajdziemy utwory pogrupowane w całe *play listy*, dzięki czemu nie będziemy musieli wyszukiwać utworów pojedynczo.

Słuchanie transmisji radiowych

W Internecie możemy również słuchać transmisji radiowych. Jedną z wielu stron, które to umożliwiają jest Radio.biz.pl (http://www.radio.biz.pl). Po wejściu na stronę główną możemy wybrać stację radiową. Po wybraniu stacji, np. RMF MAXXX, usłyszymy transmisję radiową, wyświetli się okno z nazwiskiem wykonawcy i tytułem utworu oraz *playlistą* (rysunek 9). Zwróćmy uwagę, że nie potrzebujemy żadnego dodatkowego programu! Jest to bardzo wygodne rozwiązanie.

Rysunek 9.
Słuchanie radia przez Internet

Pobieranie i odtwarzanie plików muzycznych

W Internecie można znaleźć wiele stron z muzycznymi plikami w formacie mp3 do pobrania. Jedną z najpopularniejszych jest strona topmp3.pl (http://topmp3.pl). Po wybraniu kategorii z menu w lewej części strony, otrzymamy listę najsłynniejszych utworów w danej kategorii. Po kliknięciu na wybrany utwór następuje odświeżenie strony, a po kliknięciu [Pobierz] wyświetli się okno pobierania piku na nasz komputer. Klikamy [Zapisz] i po chwili utwór zostanie zapisany na naszym komputerze. Aby go posłuchać, możemy użyć odtwarzacza Windows Media Player.

4.3 OGLĄDANIE, POBIERANIE I ODTWARZANIE FILMÓW

Wyszukiwanie plików wideo

Istnieją wyszukiwarki, które pomogą nam w znalezieniu filmów wideo. Na następnej stronie zaprezentowano przykład wyszukiwarki MetaCrawler (rysunek 10). Po wybraniu zakładki [Video] i wpisaniu *winnie the pooh* otrzymujemy listę filmów o Kubusiu Puchatku, które możemy obejrzeć lub pobrać.

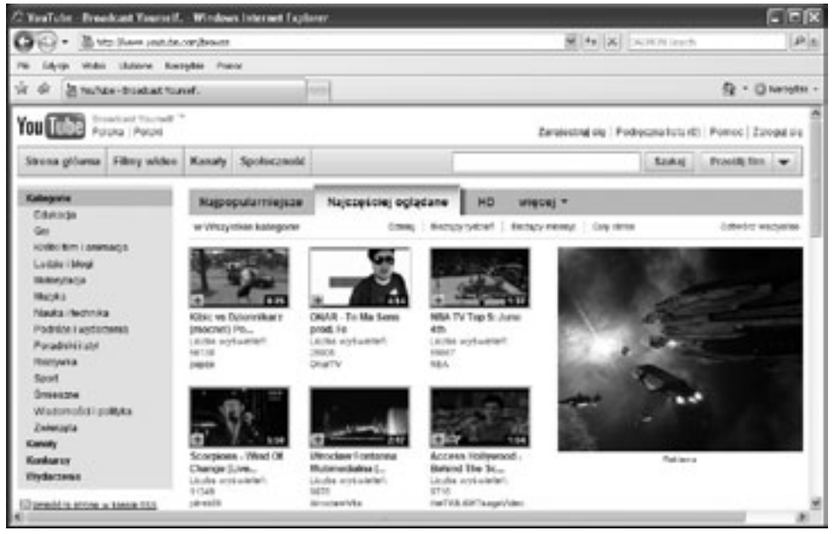
Google również oferuje wyszukiwanie plików video. Wystarczy kliknąć łącze [Wideo] znajdujące się na stronie głównej, aby po chwili znaleźć się na stronie wyszukiwarki Wideo.

Oglądanie transmisji wideo

Portal Wrzuta.pl umożliwia również oglądanie filmów. Wybierając przycisk [filmy] znajdujący się na prawo od pola wyszukiwania, już po chwili znajdziemy się na stronie z filmami wideo, z których najpopularniejsze mamy przed oczami od razu. W celu wyszukania filmu, który nas interesuje, wpisujemy jego nazwę w polu wyszukiwania i klikamy [OK]. Wpisujemy *terminator* i po chwili otrzymamy filmy o terminatorze. Aby obejrzeć wybrany z nich, wystarczy na niego kliknąć. Za pomocą przycisków możemy przewijać i zatrzymywać film oraz regulować głośność.

Największą bazą filmów dysponuje portal YouTube (http://youtube.com). Oprócz wyszukiwania filmu poprzez wpisanie jego nazwy, możemy przeglądać filmy według kategorii (analogicznie jak w katalogu internetowym). Kategorie znajdują się w lewej części strony (rysunek 11).

Rysunek 10.
Wyszukiwarka wideo MetaCrawler



Rysunek 11.
Portal YouTube.com

W trakcie oglądania filmu mamy dostęp do kilku przycisków (rysunek 12). Przy ich użyciu możemy wyświetlać film w trybie pełnoekranowym, możemy również obejrzeć go w lepszej jakości (przycisk HQ, ang. *High Quality*).

Rysunek 12.
Oglądanie filmu na portalu YouTube

Pobieranie i odtwarzanie plików wideo

W Internecie można znaleźć również serwisy udostępniające filmy do pobrania. Zwróćmy uwagę na serwis, za pośrednictwem którego można pobierać pliki wideo udostępniane w serwisie YouTube. Po wejściu na stronę (http://keephd.com/), w polu o nazwie *Enter YouTube URL* wklejamy adres filmu z serwisu YouTube, a następnie klikamy *Download*. Pojawi się okno dialogowe, w którym możemy wybrać format pobieranego pliku (np. .flv lub .MP4). W oknie [Pobieranie pliku] klikamy [Zapisz] i już po kilku minutach film znajdzie się na naszym komputerze. Możemy go odtworzyć na przykład za pomocą odtwarzacza Media Player Classic (odtwarzacz ten możemy za darmo pobrać z Internetu).

4.4 OTWARTE ZASOBY EDUKACYJNE

Otwarte zasoby edukacyjne to materiały dydaktyczne i naukowe przedstawione w formie cyfrowej, z otwartym i wolnym dostępem dla studentów, wykładowców i samouków, którzy mogą z nich korzystać w celach edukacyjnych i badawczych. Są to często bardzo wartościowe materiały dydaktyczne. Otwarte zasoby edukacyjne są rodzajem elektronicznej biblioteki publicznej, ułatwiającej wszystkim naukę, studia i zdobywanie wiedzy. Najczęściej umieszczane w Internecie materiały dydaktyczne to:

- nagrania wykładów: audio i wideo;
- wykłady w formie tekstowej;
- podręczniki multimedialne;
- archiwa publikacji, zdjęć;
- zestawienia danych;
- programy komputerowe.

Jednym ze źródeł otwartych zasobów jest Internetowe Centrum Zasobów Edukacyjnych MEN o nazwie Scholaris przeznaczony dla uczniów i nauczycieli. W tym portalu można znaleźć interaktywne materiały edukacyjne,

takie jak kursy, ćwiczenia oraz e-Lekcje (rysunek 13). Materiały podzielone są według przedmiotów (menu w lewej części strony) oraz według typu: symulacje, prezentacje, testy, filmy, zdjęcia itp.

Warto zwrócić uwagę na portal ZamKor (http://www.zamkor.pl/zamkor.pl). Jest to portal wydawnictwa książek szkolnych, jak również źródło zasobów edukacyjnych, głównie z fizyki. Polecamy zwłaszcza animacje zjawisk fizycznych na stronie http://fizyka.zamkor.pl/alpety/programy_fizyka_liceum/start.htm. Na stronie http://fizyka.zamkor.pl/ można również znaleźć filmy dydaktyczne, foliogramy, zestawy doświadczalne. W portalu ZamKor można również słuchać i oglądać wykłady *on-line*.

Rysunek 13.
Portal edukacyjny Scholaris [źródło: http://www.scholaris.pl/]

Na stronie http://www.jakubas.pl/ natomiast zamieszczonych jest wiele linków do materiałów dydaktycznych i animacji przydatnych w nauce matematyki.

Polska Wszechnica Informatyczna (http://www.pwi.edu.pl/) jest z kolei portalem internetowym przeznaczonym dla studentów, wykładowców i absolwentów kierunku informatyka polskich uczelni wyższych. Materiały zgromadzone na tym portalu mogą okazać się pomocne również uczniom szkół średnich np. przy wyborze kierunku studiów. W portalu można przeglądać listę wykładów oraz zapoznać się z ich tematyką, a także je obejrzeć lub pobrać prezentację. W tym portalu będą zamieszczane również nagrania wykładów z Projektu Informatyka +, przeznaczone dla uczniów ze szkół.

Innym serwisem informatycznym, gromadzącym materiały edukacyjne do studiowania informatyki jest „ważniak", dostępny pod adresem: http://wazniak.mimuw.edu.pl/.

PODSUMOWANIE

Atrakcyjność multimediów sprawia, że niekiedy trudno się od nich oderwać. Szczególnie dotyczy to Internetu oraz gier komputerowych. Jednak w trosce o własne zdrowie należy przestrzegać higieny pracy z komputerem, starać się robić regularne przerwy i wyjść czasem na świeże powietrze.

Mamy nadzieję, że udało nam się zainteresować słuchaczy multimedialnymi treściami, do których mamy dostęp przez Internet. Zapewne z wielu z nich korzystaliście do tej pory. Mamy nadzieję, że niektóre dopiero odkryjecie. W Internecie dostępnych jest wiele treści multimedialnych. Z uwagi na ograniczony czas naszego wykładu, omówiliśmy tylko wybrane z nich. Z pewnością będą pojawiać się nowe treści multimedialne. Każdy z Was będzie korzystać z Internetu, przez co będziecie mieć okazję do ich odkrycia.

LITERATURA

1. Battelle J., *Szukaj. Jak Google i konkurencja wywołali biznesową i kulturową rewolucję*, WN PWN, Warszawa 2006
2. Calishain T., Dornfest R., Adams D.J., *Google. Leksykon kieszonkowy*, Helion, Gliwice 2003
3. Sokół R., *Internet. Ilustrowany przewodnik*, Helion, Gliwice 2007

Witryna w Internecie – zasady tworzenia i funkcjonowania

Piotr Kopciał
Politechnika Warszawska
piotrkopcial@gmail.com

Streszczenie

Internet wkracza w coraz to nowe obszary naszego życia: e-nauczanie, elektroniczne biblioteki, wirtualne laboratoria, medycyna, usługi (bankowość, turystyka). Podstawowym elementem tych i podobnych serwisów są strony internetowe, które składają się na bardziej złożone witryny, portale i platformy internetowe. Wykład jest poświęcony funkcjonowaniu stron internetowych. W pierwszej części opisano mechanizmy działania stron internetowych, w tym m.in. komunikację w standardzie klient-serwer i strony dynamiczne. Następnie są omawiane zalety i wady stron statycznych i dynamicznych oraz mechanizmy interakcji serwisów internetowych z użytkownikiem, stosowane na współczesnych stronach internetowych. Jednym z celów wykładu jest uwrażliwienie słuchaczy na dobre praktyki projektowania i tworzenia stron internetowych. Wykład kończy przedstawienie kilku przykładowych interaktywnych serwisów WWW w działaniu. Prezentacja jest bogato ilustrowana różnymi aspektami stron internetowych.

Spis treści

1 WPROWADZENIE

Jeszcze 50 lat temu trudno było uwierzyć, że komputery z całego świata mogą zostać ze sobą połączone. W tamtych czasach jedynie niewielka grupa zapaleńców marzyła o współpracy użytkowników komputerów z całego świata, o błyskawicznym wymienianiu się informacjami i różnego rodzaju danymi, takimi jak dokumenty, pliki itp.

Internet a intranet

Internet to sieć komputerowa o ogólnoświatowym zasięgu, to sieć sieci, do której dostęp może mieć każdy użytkownik komputera.

Uwaga! Nazwę Internet pisze się wielką literą, ponieważ jest to nazwa własna.

Czasem można spotkać pojęcie **intranet**. Nie jest to przejęzyczenie. Internet to gigantyczna sieć składająca się z komputerów rozsianych po całym świecie. Natomiast intranet to sieć o mniejszym zasięgu – np. obejmująca komputery w firmie lub organizacji. W skład sieci intranet wchodzi znacznie mniej komputerów, mogą być nawet dwa.

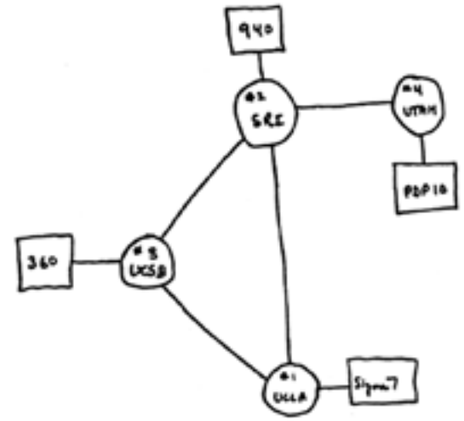
Historia i przyszłość

Historia Internetu sięga roku 1969, gdy została uruchomiona pierwsza sieć komputerowa ARPANet (sieć o przeznaczeniu militarnym). Prekursorem rozwiązań internetowych był Paul Baran, Polak z pochodzenia. Na rysunku 1 przedstawiono schemat sieci ARPANet. Sieć ta rozpoczęła działanie od 4 węzłów ulokowanych na amerykańskich uczelniach: Uniwersytecie Kalifornijskim w Los Angeles, Uniwersytecie Stanforda, Uniwersytecie Kalifornijskim w Santa Barbara i Uniwersytecie Utah.

W latach 1971/72 Ray. Tomlinson opracował protokół poczty elektronicznej, co znacznie przyspieszyło wymianę wiadomości. W roku 1983 powstał protokół TCP/IP – podstawowy protokół służący do wymiany danych pomiędzy komputerami w sieci. Jego twórcami byli Vinton Cerf i Robert Kahn.

W roku 1991 Tim Berners-Lee utworzył pierwszą stronę internetową oraz oprogramowanie do wyświetlania takich stron (przeglądarkę) – uważa się ten moment za początek **serwisu WWW** (ang. *World Wide Web*), czyli **globalnej pajęczyny**.

W Polsce po raz pierwszy połączono się z Internetem latem 1991 roku, chociaż wcześniej możliwy był dostęp do innych sieci o światowym zasięgu, takich jak BITNET.



Rysunek 1.
Schemat sieci ARPANet [źródło: www.computerhistory.org/internet_history/]

Internet w dzisiejszej postaci określa się mianem sieci Web 2.0, odnoszącym się do serwisów internetowych, w których podstawową rolę odgrywa użytkownik wraz z generowanymi przez siebie treściami, zasobami, a także serwisami.

A co czeka nas w przyszłości? Web 3.0 to określenie dalszej ewolucji Internetu w kierunku systemu przekazu wiedzy i modelu sieci semantycznej, czyli sieci „rozumiejącej" swoją zawartość oraz użytkowników sieci.

Co można znaleźć w Internecie

W Internecie istnieje wiele informacji i danych, głównie dostępnych poprzez **strony internetowe** w serwisach WWW, począwszy od bardzo wartościowych materiałów i aktualnych treści, przez rozrywkę pod różnymi postaciami (np. gry sieciowe), a skończywszy na treściach zbędnych i szkodliwych.

Przykładem wartościowych informacji, do których mamy dostęp w Internecie są:

- multimedialne encyklopedie i materiały edukacyjne (przykładowe adresy: http://portalwiedzy.onet.pl/encyklopedia.html, http://pl.wikipedia.org/wiki, http://mediawiki.ilab.pl/index.php, http://www.pwi.edu.pl/, http://wazniak.mimuw.edu.pl);
- wirtualne muzea (www.1944.wp.pl, http://www.zamek-lancut.pl);
- obserwacje z życia np. zwierząt (transmisja na żywo obrazu z kamery) (http://www.bociany.ec.pl, http://www.teleskopy.net);
- elektroniczne biblioteki (http://www.cm.umk.pl/~biblio/ambgb/b_eksiazki.htm, Google Book Search).

2 STRONA W INTERNECIE

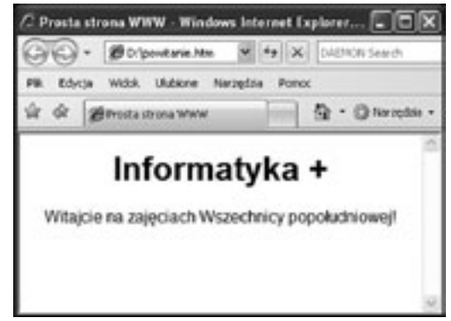
Strona internetowa jest wynikiem interpretacji **dokumentu HTML**, czyli dokumentu napisanego w języku HTML. Taki dokument może być pobrany z lokalnego dysku komputera lub z serwera internetowego i jest interpretowany po stronie użytkownika przez przeglądarkę. Na stronie internetowej można umieszczać tekst, obrazy, tabele, wstawki dźwiękowe, animacje, sekwencje wideo.

Często słyszymy określenie „witryna internetowa". **Witryna internetowa** jest określeniem rozbudowanej strony internetowej, która może składać się w wielu stron, do których dostęp uzyskujemy poprzez wybranie odpowiedniej opcji w menu witryny. W dalszej części będziemy na ogół pisać o stronie, bo witryna to zbiór stron.

HTML (ang. *Hypertext Markup Language*) jest językiem, który służy do tworzenia opisów stron internetowych. Język HTML to zestaw znaczników, pomiędzy którymi umieszcza się tekst lub inne elementy mające pojawić się na stronie. Przykładowo dla fragmentu kodu HTML `raz<b> dwa</b> trzy`, wyraz dwa zostanie wyświetlony czcionką pogrubioną, ponieważ jest ograniczony znacznikiem <b>.

```
<HTML>
 <HEAD>
  <TITLE> Prosta strona WWW </TITLE>
 </HEAD>
 <BODY>
  <FONT FACE="Arial">
  <CENTER>
   <H1> Informatyka + </H1>
   Witajcie na zajęciach Wszechnicy popołudniowej!
  </CENTER>
 </BODY>
</HTML>
```


Strona o tym opisie jest przedstawiona na rysunku 2.



Rysunek 2.
Prosta strona internetowa

Serwer to komputer, na którym znajduje się plik zawierający opis strony internetowej utworzonej w języku HTML wraz z plikami zawierającymi elementy składowe strony (np. obrazy). Serwer udostępnia stronę innym komputerom za pośrednictwem sieci Internet. W sieci istnieje wiele serwerów.

Przeglądarka to program służący do pobierania opisu stron internetowych z serwera i wyświetlania ich zawartości na ekranie monitora użytkownika. Przeglądarka „tłumaczy" kod HTML strony na postać oglądaną na monitorze.

Adres URL (ang. *Uniform Access Locator*) to adres, pod którym jest dostępna konkretna strona internetowa. Przykładowy adres URL to http://www.google.pl/. Adres URL jest adresem serwera, z którym przeglądarka kontaktuje się w celu pobrania opisu strony.

Znaczenie poszczególnych części adresu URL zestawiono w tabeli 1.

Tabela 1.
Znaczenie poszczególnych części adresu URL

http:// (https://)	nazwa_serwera.pl/	katalog/	plik.html
nazwa protokołu sieciowego (sposobu przesyłania danych z serwera do przeglądarki)	nazwa domenowa serwera, z którego zostanie pobrany dokument HTML (wyświetlona jako strona)	nazwa folderu (katalogu) na serwerze	nazwa pobieranego pliku (dokumentu HTML) znajdującego się w tym folderze (katalogu)

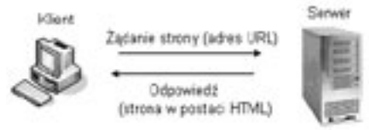
Zasada działania strony internetowej

Po utworzeniu, strona internetowa jest umieszczana na serwerze. W tym momencie staje się dostępna dla wszystkich użytkowników Internetu. Tak jak budynki na ulicy, każdy serwer ma swój adres (tzw. adres domenowy); a tak jak mieszkania w budynku, każda strona ma swój unikatowy adres.

Gdy użytkownik wpisze adres URL strony w przeglądarce, ta stara się odnaleźć w pierwszej kolejności serwer, a następnie daną stronę. Jeśli znajdzie, serwer odsyła do przeglądarki żądaną stronę w postaci pliku HTML, ewentualnie wraz z uzupełniającymi go plikami graficznymi. Przeglądarka wyświetla stronę na ekranie komputera użytkownika w postaci zdefiniowanej w pliku HTML.

Po to, aby komputer użytkownika (a dokładniej jego przeglądarka) mógł się porozumieć z serwerem, obydwa komputery komunikują się za pomocą protokołu HTTP (ang. *Hypertext Transfer Protocol*).

Powyższą komunikację nazywamy komunikacją **klient-serwer** (rysunek 3). **Klientem** w tym określeniu jest komputer użytkownika, który przy użyciu przeglądarki żąda wyświetlenia wskazanej strony, której opis znajduje się na **serwerze**.



Rysunek 3.
Komunikacja klient-serwer

3 TWORZENIE STRONY INTERNETOWEJ

Istnieje wiele powodów, dla których warto umieć tworzyć strony internetowe:

- dla przyjemności – budowanie i prowadzenie własnej strony internetowej może przynieść wiele satysfakcji, możemy np. zaprezentować na niej swoją twórczość milionom internautów;
- w dzisiejszych czasach korzystanie z Internetu stało się tak powszechne, jak korzystanie z edytora tekstu Word do pisania;
- nie musimy płacić za zrobienie czegoś, co można zrobić samemu;
- tworzenie stron internetowych może sprawiać frajdę – być dobrą zabawą, niewymagającą szczególnych umiejętności; na początek wystarczy znajomość języka HTML, który jest łatwy do opanowania.

3.1 CO MOŻNA UMIEŚCIĆ NA STRONIE INTERNETOWEJ

1. Tekst
   Niektóre strony zawierają wyłącznie tekst. Zaletą takich stron jest zwykle duża wartość informacyjna oraz szybkość wyświetlania w przeglądarce. Wadą jest brak elementów atrakcyjnych dla użytkownika.
2. Obrazy
   Obrazy i elementy graficzne przyciągają uwagę. Mogą to być np. własne zdjęcia, rysunki oferowanych przez firmę produktów lub mapka dojazdu na miejsce. Pobranie strony zawierającej elementy graficzne z serwera do przeglądarki trwa jednak dłużej.
3. Formularze
   Formularze stosuje się do zbierania informacji od użytkowników odwiedzających daną stronę (rejestracja, ankieta itp.) lub przekazywania danych przez użytkowników, chcących np. uzyskać informacje od właściciela strony. Formularze stanowią także formę zamówień w transakcjach internetowych.
4. Obramowania
   Ramki stosuje się do podziału strony na kilka części, w których można grupować podobne informacje. Przykładowo na stronie księgarni internetowej informacje ogólne i pole wyszukiwania oddzielone są od części strony zawierającej opisy poszczególnych działów tematycznych.
5. Multimedia: sekwencje audio i wideo
   Multimedia umieszczone na stronie są najbardziej atrakcyjnymi elementami dla odwiedzających Internet.

3.2 PROJEKTOWANIE WITRYNY

Nie można kopać dołu na fundamenty, nie mając gotowego projektu domu [1]. Słowa te oddają, jak ważne jest przygotowanie dobrego planu (projektu), przed przystąpieniem do realizacji praktycznej. Dotyczy to również tworzenia stron internetowych.

Zaprojektowanie strony, którą chcemy utworzyć, to podstawa. Od tego, co i w jaki sposób chcemy umieścić na stronie, zależeć będą dalsze czynności. Należy odpowiedzieć sobie na pytania:

- co skłoniło mnie od tworzenia tej właśnie strony?
- do kogo strona jest adresowana?
- co chcę na niej umieścić?
- w jaki sposób chcę zaprezentować siebie (lub np. swoją firmę) innym?

Najczęściej na stronie umieszcza się:

- informacje o swoich zainteresowaniach (hobby) lub własnej firmie;
- zdjęcia prywatne lub zdjęcia oferowanych produktów wraz z ich opisem;
- formularze, dzięki którym osoby odwiedzające stronę mogą przekazywać i wymieniać informacje z właścicielem strony;
- elementy graficzne, które czynią stronę bardziej atrakcyjną wizualnie.

Zawartość strony zależy od przeznaczenia strony i jej odbiorców.

Kiedy już zdecydujemy, co ma znajdować się na stronie, należy następnie rozrysować jej układ na kartce papieru. Typowy układ strony internetowej przedstawiono na rysunku 4.

Rysunek 4.
Typowy układ strony internetowej

Nie jesteśmy odbiorcami swojej strony

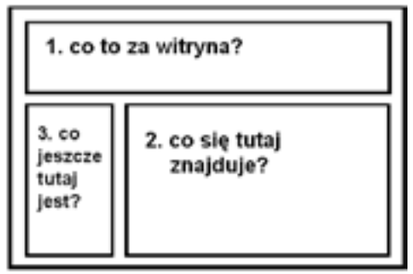
Gdy tworzymy stronę na temat, który nas interesuje, możemy dojść do wniosku, że wszyscy odbiorcy strony są podobni do nas. To błędne przekonanie. Należy zdawać sobie sprawę, że wiemy na temat naszej witryny znacznie więcej niż osoby, które widzą ją po raz pierwszy. Oznacza to jednocześnie, że na temat użytkownika – odbiorcy naszej witryny wiemy mniej, niż nam się wydaje. Jest to jedna z największych trudności piętrząca się przed twórcami stron internetowych.

Pamiętajmy! Nie projektujemy strony dla siebie. Tworzymy ją dla innych użytkowników Internetu, którzy będą naszą stronę odwiedzać.

Najczęściej popełniane błędy:
- złe zaplanowanie struktury (układu) strony;
- brak przemyślanego grupowania informacji na wybrany temat;
- używanie żargonu i słów niezrozumiałych dla innych;
- zbytnie przeładowanie strony elementami, które rozpraszają, a nie przyciągają uwagę.

Jak ludzie widzą witryny internetowe

Kiedy użytkownik trafi na stworzoną przez nas stronę, powinien mieć możliwość szybkiego zorientowania się, co może na niej znaleźć. Należy wyraźnie zasygnalizować, jakie informacje użytkownik może uzyskać i w jaki sposób. Internauci są niecierpliwi. Jeżeli w ciągu 15 sekund użytkownik stwierdzi, że nie może znaleźć tego, czego szuka (np. informacji na dany temat, gry, łącza do innych stron), to jest bardzo prawdopodobne, że opuści stronę bezpowrotnie.



Rysunek 5.
Kolejność przeglądania strony internetowej przez internautę

Badania zachowań internautów wykazują, że intuicyjnie przeglądają oni strony internetowe według pewnego powtarzającego się schematu. Zwykle na początku spoglądają na górną jej część, aby zorientować się, co to za strona (rys. 5). Następnie kierują wzrok ku środkowi. Jeśli nie znajdą tam tego, czego szukają, zmierzają wzrokiem w kierunku lewej części strony, gdzie spodziewają się odszukać elementy nawigacji (menu).

Jak ludzie nawigują w Internecie

Czasem warto zastanowić się, jak ludzie zachowują się, kiedy surfują po Internecie.
Z różnych mediów korzystamy w różny sposób:
- czasopisma – czytamy;
- radia – słuchamy;
- telewizję – oglądamy;
- w Internecie – nawigujemy tak, jakby to była przestrzeń.

W dzisiejszych czasach obserwujemy konwergencję mediów. Zarówno czasopisma, jak i radio oraz telewizja dostępne są przez Internet.

Globalna sieć jest zupełnie innym środkiem przekazu niż druk, radiofonia czy telewizja. Ludzie przemieszczają się pomiędzy stronami w wirtualnej przestrzeni. Na każdej stronie poszukują sygnałów nawigacyjnych, skupiają się na dotarciu do miejsc docelowych i zastanawiają się, gdzie przejść, którą witrynę odwiedzić.

Kursor myszy stanowi niejako przedłużenie ręki użytkownika. Z tego powodu nawigacja w Internecie jest podobna do sterowania w przestrzeni fizycznej. Dlatego poruszanie się w obrębie projektowanej witryny jest tak ważne.

5 skutecznych sposobów na odstraszenie użytkowników Internetu

Marzeniem każdego projektanta strony jest to, aby jego strona zyskała popularność i uznanie użytkowników. Może jednak zdarzyć się sytuacja, w której użytkownik odwiedzający naszą stronę po raz pierwszy, już nigdy nie zechce na nią wrócić. Aby do tego nie doszło, należy unikać następujących sytuacji:
1. Wyłączenie serwera, na którym umieszczona jest nasza strona (nikt nie będzie mógł się do niej dostać) – jeśli nie dysponujemy komputerem, który mógłby pełnić rolę serwera i pracować bez przerwy, skorzystajmy z usług firm świadczących usługi hostingowe.
2. Umieszczanie zbyt wielu elementów multimedialnych (grafika, dźwięk, film), spowalniających wyświetlanie strony (przeglądarka użytkownika będzie pobierać stronę bardzo długo).
3. Zmiana rozmieszczenia elementów na stronie, co powoduje, że użytkownik powracający na stronę nie może poruszać się znajomymi drogami i znaleźć tego, czego szuka.
4. Wstawienie odnośników do stron, których nie można wyświetlić (użytkownik spotka się z niezrozumiałym komunikatem serwera).
5. Brak aktualizowania treści (artykuły, zdjęcia, odnośniki do innych stron) witryny – jeśli co jakiś czas nie będą pojawiać się świeże informacje, to użytkownik nie będzie miał powodu do ponownych odwiedzin.

5 sposobów poprawy witryny

1. Skoncentruj się przede wszystkim na tym, żeby strona dobrze funkcjonowała.
   Wygląd ma znaczenie drugorzędne. *Strony internetowe muszą się szybko ładować, jeśli ludzie mają ich używać. Być może konieczny będzie kompromis pomiędzy efektami, jakie chcemy uzyskać, a szybkością, która jest ograniczana przez te efekty* [1].
2. Myśl o użytkowniku.
   Projektant strony powinien wcielić się w użytkownika i wyobrazić sobie, jak to jest, gdy korzysta się z powolnego łącza internetowego.
3. Projektuj stronę zgodnie z przyjętymi konwencjami.
   W ciągu ostatnich lat wypracowano sprawdzony schemat układu strony, do którego użytkownicy są przyzwyczajeni. W obrębie dobrze zaprojektowanej witryny użytkownik porusza się intuicyjnie.
4. Zwróć uwagę na szczegóły.
   Potocznie błahe błędy, takie jak brak wyrównania czy odpowiednich oznaczeń, mogą sprawić kłopot użytkownikowi.
5. Testuj.
   Najlepszym sposobem na sprawdzenie działania witryny jest jej przetestowanie przez użytkowników, a następnie poprawienie według poczynionych spostrzeżeń i sugestii.

3.3 JĘZYK HTML I STRUKTURA DOKUMENTU HTML

Opis stron internetowych jest tworzony w języku HTML. Nauka tego języka jest dość łatwa. HTML jest zestawem znaczników. Każdy znacznik umieszczony jest w nawiasach ostrych < >.

Przykładowo – znacznikiem rozpoczęcia opisu strony jest `<HTML>`. Większość znaczników występuje jako część otwierająca i zamykająca. Część zamykająca zawiera dodatkowy znak – ukośnik /. Znacznikiem zamykającym stronę jest zatem `</HTML>`.

Strukturę dokumentu HTML opisującego stronę określają 3 znaczniki: `<HTML>`, `<HEAD>` i `<BODY>`.

`<HTML>` – użycie tego znacznika jest obowiązkowe, gdyż wskazuje on na początek i koniec dokumentu. Znacznik `<HTML>` musi znaleźć się w pierwszym wierszu kodu strony.

`<HEAD>` – znacznik definiujący nagłówek dokumentu. Można w nim określić takie elementy, jak nazwa i styl dokumentu, tytuł strony. Nagłówek umieszczamy na początku dokumentu, a kończymy go znacznikiem `</HEAD>`.

<BODY> – pomiędzy znacznikami <BODY> oraz </BODY> zawarta jest zasadnicza treść dokumentu. W tej części można definiować: rodzaj czcionki, kolor tekstu, tło strony itd.

Przykład prostego dokumentu HTML został podany wcześniej, przy okazji prezentowania na rysunku 2 efektu jego interpretacji przez przeglądarkę.

Hiperłącza

Fragmenty na stronie internetowej, a także inne obiekty mogą odgrywać rolę **łącza** z innymi stronami i witrynami w Internecie – łącza takie nazywamy **hiperłączami**.

Tekst na stronie internetowej określa się mianem **hipertekstu**, gdyż może zawierać hiperłącza (krócej łącza) i elementy multimedialne, nie będące tekstem.

Hiperłączy można używać na dwa sposoby:
- jako odsyłaczy do innych stron naszej witryny,
- jako odsyłaczy do innych stron w Internecie.

Poniżej zilustrowano ten drugi przypadek. Umieszczenie hiperłącza na stronie wymaga użycia odpowiedniego znacznika HTML:

```
<HTML>
 <HEAD>
  <TITLE> Prosta strona WWW </TITLE>
 </HEAD>
 <BODY>
  <FONT FACE="Arial">
  <CENTER>
   <H1> Informatyka + </H1>
   Witajcie na zajęciach Wszechnicy popołudniowej!</br>
   Więcej na temat programu Informatyka+ znajdziecie na
   <a href="http://informatykaplus.edu.pl/">
   stronie projektu</a>
  </CENTER>
 </BODY>
</HTML>
```

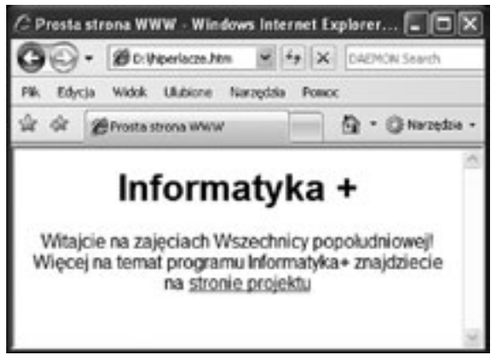
Strona o tym kodzie ma postać jak na rysunku 6, a efektem kliknięcia w hiperłącze na tej stronie jest przejście do witryny pokazanej na rysunku 7.

Narzędzia do tworzenia stron

Kod HMTL można napisać w prostym edytorze tekstu, np. w Notatniku Windows. Wystarczy znać znaczniki HTML i zasady ich stosowania. Jednakże dużym ułatwieniem jest posłużenie się specjalnym programem do tworzenia stron internetowych, tzw. edytorem języka HTML.

Edytory są pomocne przy tworzeniu bardziej złożonych elementów stron, takich jak np. tabele. Zamiast tworzyć w Notatniku każdą komórkę tabeli osobno, wystarczy skorzystać z narzędzia tworzenia tabel w takim edytorze.

Przykładem edytora HTML jest program FrontPage firmy Microsoft. Istnieje również wiele darmowych edytorów (lub dostępnych za darmo przez określoną liczbę dni, np. 60), które można pobrać z Internetu. Darmowe edytory HTML można ściągnąć z następujących stron: http://agerwebedytor.com/ (Ager Web Edytor), http://www.darmoweprogramy.org/programy/edytory_html.php (Alleycode HTML Editor, Web Design Toy, EasyHTML, HotHTML).



Rysunek 6.
Prosta strona powitalna zawierająca hiperłącze

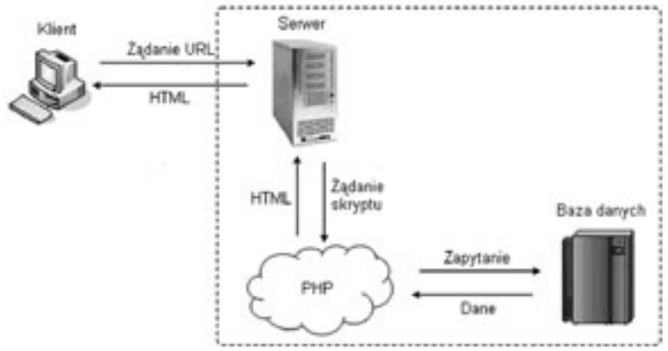
Rysunek 7.
Strona wyświetlona jako efekt kliknięcia w hiperłącze na stronie przedstawionej na rysunku 6

4 DYNAMICZNA STRONA INTERNETOWA

Dzięki dynamicznym stronom internetowym można np. witać użytkownika odwiedzającego stronę kolejny raz w następujący sposób: „Witaj ponownie, Krzysiu!".

Dynamiczna strona internetowa jest tworzona przez serwer w momencie, kiedy użytkownik żąda jej wyświetlenia. Strony dynamiczne są generowane na bieżąco i mogą zawierać różne treści, co jest uwarunkowane tym, kto je pobiera i w jakich okolicznościach. Na przykład tło strony może być jasne lub ciemne, w zależności od tego, czy akurat jest dzień czy noc.

4.1 ZASADA DZIAŁANIA STRONY DYNAMICZNEJ



Rysunek 8.
Działanie dynamicznej strony WWW

Na rysunku 8 przedstawiono działanie dynamicznej strony WWW. Interakcja pomiędzy klientem a serwerem zaczyna się w momencie wpisania w przeglądarce adresu strony lub kliknięcia łącza do strony dynamicznej. Za pomocą protokołu HTTP przeglądarka nawiązuje połączenie z serwerem. Serwer przesyła żądanie do interpretera języka skryptowego (np. PHP), który wykonuje kod skryptu (**skryptem** nazywamy kod napisany w języku przeznaczonym do tworzenia stron dynamicznych). Jeśli w skrypcie PHP są zapisane zapytania do bazy danych (np. w celu pobrania informacji o użytkowniku), interpreter języka skryptowego odpowiada za komunikację serwera z bazą danych. Po pobraniu zawartości strony, przeglądarka analizuje kod HTML, po czym wyświetla gotową stronę na ekranie monitora użytkownika.

Należy zwrócić uwagę, że dynamiczne fragmenty strony internetowej nie istnieją, dopóki ktoś nie zażąda wyświetlenia strony. Dopiero wtedy serwer buduje taką stronę według instrukcji zawartych w kodzie HTML oraz w kodzie skryptu, a gdy użytkownik zamyka stronę dynamiczną w przeglądarce, to dynamiczne fragmenty strony przestają istnieć. W przypadku kolejnego wyświetlenia takiej strony, jej dynamiczne fragmenty są tworzone na nowo. Dzięki temu na stronach mogą ulegać zmianie: godzina, data, prognoza pogody, notowania giełdy itp.

W odróżnieniu od strony dynamicznej, treść strony statycznej nie zmienia się od momentu jej utworzenia do chwili zmiany opisu strony lub usunięcia go z serwera.

4.2 STRONA STATYCZNA A STRONA DYNAMICZNA

Statyczne strony WWW opisane w języku HTML, są przechowywane na serwerze i przesyłane w takiej samej postaci do wszystkich użytkowników. Oznacza to, że każdy użytkownik widzi taką samą stronę pod względem treści i układu.

Natomiast strony dynamiczne są generowane przez serwer na bieżąco, w zależności od tego kim jest użytkownik (np. użytkownik zalogowany do serwisu ma dostęp do treści niedostępnych dla użytkowników niezalogowanych). Mechanizm ten wymaga od serwera większej pracy niż w przypadku stron statycznych, kiedy to rola serwera sprowadza się do przechowywania plików, oczekiwania na żądanie i przesłania strony wskazanej przed użytkownika do jego przeglądarki.

Ponadto potrzebna jest baza danych zawierająca treści, które mają pojawić się na stronie. **Baza danych** jest elektronicznym magazynem informacji (danych) i narzędziem do zarządzania tymi informacjami.

Zarówno strony statyczne, jak i strony dynamiczne mają swoje wady i zalety, co zilustrowano w tabeli 2.

Statyczne strony WWW, nawet te najbardziej atrakcyjne pod względem treści i grafiki, mają wadę, która polega na tym, że aktualizacja ich treści zajmuje sporo czasu, ponieważ wymaga modyfikowania każdej strony niezależnie. Wady tej są pozbawione witryny z elementami dynamicznymi, których treść przechowywana jest w bazie danych i pobierana przy każdym otwarciu strony przez odwiedzającego. Ponadto zmiana treści dynamicznego fragmentu strony wymaga modyfikacji w jednym tylko miejscu – w bazie danych.

Tabela 2.
Wady i zalety stron statycznych i dynamicznych

	Wady	Zalety
Strony statyczne	– nie można szybko zmienić treści kilku stron – interakcja z użytkownikiem bardzo ograniczona	– łatwo je utworzyć (kod HTML)
Strony dynamiczne	– sposób tworzenia jest bardziej skomplikowany (języki skryptowe są trudniejsze do opanowania niż HTML) – wymagają bazy danych na serwerze	– łatwo i szybko można zmienić treść kilku stron – umożliwiają interakcję z użytkownikiem

4.3 TWORZENIE STRONY DYNAMICZNEJ – JĘZYK SKRYPTOWY

Dynamiczne strony internetowe tworzy się za pomocą tzw. **języków skryptowych**. Fragmenty kodu napisane w języku skryptowym są umieszczane pomiędzy znacznikami kodu HTML strony. W języku skryptowym można zdefiniować polecenia dla serwera, w jaki sposób ma budować (generować) stronę. Można np. wyświetlić aktualną datę i godzinę lub pobrać aktualne informacje (np. na temat pogody) z bazy danych.

Najczęściej stosowanym i najprostszym do nauki językiem skryptowym jest wspomniany już PHP. Poniżej przedstawiono kod skryptu generującego aktualną datę.

```
<HTML>
 <HEAD>
  <TITLE> Prosta strona WWW </TITLE>
 </HEAD>
 <BODY>
  <FONT FACE=”Arial”>
  <CENTER>
   <H1> Informatyka + </H1>
   Witaj na zajęciach Wszechnicy Popołudniowej w dniu:
   <?php
    echo date(„Y-m-d”);
   ?>
  </CENTER>
 </BODY>
</HTML>
```

Efekt działania tego skryptu przedstawi kolejny rysunek.

Rysunek 9.
Strona wyświetlająca aktualną datę

Za każdym razem, gdy ta strona jest wyświetlana, pobierana jest aktualna data.

4.4 INTERAKCJA Z UŻYTKOWNIKIEM WIZYTÓWKĄ NOWOCZESNYCH STRON INTERNETOWYCH

Strony WWW uważa się za **interaktywne**, jeśli przechowują sesję użytkownika. **Sesja** jest sposobem przekazywania informacji o użytkowniku (np. towary, jakie gromadzi w koszyku w internetowym sklepie) pomiędzy stronami (czyli następującymi po sobie żądaniem klienta i odpowiedzią serwera).

Interaktywność polega na tym, że treść strony może się dynamicznie zmieniać w zależności od:

- Profilu użytkownika – osoby, które korzystały wcześniej z serwisu mogą automatycznie otrzymywać informacje na interesujące je tematy (np. wyniki sportowe, lokalna prognoza pogody). Inny przykład to różny zakres opcji menu dostępnych na stronie w zależności od tego, czy dana osoba jest administratorem, moderatorem czy zwykłym użytkownikiem;
- Wprowadzonych danych – biorąc pod uwagę dane wprowadzone wcześniej przez użytkownika, treść strony może być wyświetlana w różny sposób. Przykładem jest personalizacja serwisu, często stosowana w serwisach społecznościowych – użytkownicy mogą zmieniać wygląd strony (tło, format tekstu itp.) według własnych upodobań;
- Przeglądarki oraz systemu operacyjnego użytkownika – języki skryptowe (np. PHP) umożliwiają tworzenie aplikacji pobierających informacje o systemie operacyjnym i przeglądarce użytkownika po to, aby jak najlepiej dostosować do nich sposób wyświetlania strony;
- Czasu – np. wyświetlanie ciemniejszego tła serwisu w godzinach nocnych lub różnych motywów na stronie w zależności od pory roku;
- Położenia geograficznego.

Do ciekawych efektów stosowanych na stronach internetowych należą m.in.:

- podświetlanie przycisków po najechaniu na nie kursorem myszy;
- zmiana kształtu kursora myszy;
- pojawianie się okien dialogowych;
- mechanizm przeciągnij-i-upuść;
- manipulowanie grafiką (np. przełączanie obrazków);
- uruchamianie wyskakujących okienek (np. pojawianie się okienka informacyjnego, gdy użytkownik umieści wskaźnik myszy na obrazku);
- przesuwanie mapy;
- zwijanie i rozwijanie menu.

Efekty te tworzy się przy użyciu techniki AJAX (ang. *Asynchronous Javascript And XML*), wykorzystującej m.in. język skryptowy JavaScript.

4.5 PRZYKŁADY SERWISÓW INTERAKTYWNYCH

Poniżej przykłady interaktywnych serwisów internetowych: platformy aukcyjnej Allegro.pl, utworzonej przy użyciu języka PHP, oraz nawigatora Google Suggest i mapę Google Maps, utworzonych przy użyciu techniki AJAX.

Serwis aukcyjny

W sieci istnieje wiele serwisów aukcyjnych. Można na nich kupić niemal wszystko: począwszy od drobiazgów i używanych rzeczy, które nie są już potrzebne ich właścicielom, poprzez nowe, firmowe towary wystawiane przez sklepy internetowe, skończywszy na samochodach. Takie zróżnicowanie oferty wpływa na popularność serwisu. W roku 2008 na największym polskim serwisie aukcyjnym sprzedano 97 mln przedmiotów o wartości ponad 5,2 mld PLN, a liczba użytkowników przekroczyła 8 mln. Do końca roku 2010 w tym serwisie aukcyjnym sprzedano 162 mln przedmiotów, a liczba użytkowników osiągnęła 12,5 mln [za: http://allegro.pl/country_pages/1/0/marketing/about.php].

Serwis aukcyjny jest wyposażony w narzędzia ułatwiające pracę. Początkującego użytkownika prowadzi przyjazny system pomocy. Aby sprzedać przedmiot, należy się zarejestrować. Dla zapewnienia bezpieczeństwa transakcji i płatności, serwis oferuje bezpieczną rejestrację. Każdy użytkownik posiada swój profil oraz możliwość monitorowania prowadzonych przez siebie aukcji i zarządzania nimi. Za pośrednictwem forum można nawiązać kontakt z innymi użytkownikami. Procedura wystawiania przedmiotu na aukcji jest intuicyjnie prosta. Użytkownik prowadzony jest krok po kroku: wybiera typ aukcji, kategorię przedmiotu, wprowadza informacje o sprzedawanym przedmiocie, dołącza pliki graficzne (np. zdjęcia przedmiotu), podaje cenę i czas trwania aukcji. Po kilku minutach przedmiot jest dostępny dla milionów internautów. W momencie zakupienia produktu, wystawca dostaje stosowną informację w wiadomości e-mail, a kupujący kontaktuje się z nim w celu sfinalizowania transakcji. Kupujący i sprzedający mogą wystawiać sobie nawzajem komentarze, do których inni użytkownicy mają również dostęp. Pozytywne komentarze przyczyniają się do tworzenia reputacji wiarygodnego sprzedawcy, jak również uczciwego kupującego.

Google Suggest

Google Suggest oraz Google Maps są częścią największej na świecie wyszukiwarki internetowej Google, która zawiera skatalogowane informacje o ponad 8 miliardach stron.
Aplikacja Google Suggest jest mechanizmem podpowiedzi generowanych podczas wpisywania pytania do wyszukiwarki. Google Suggest działa bardzo szybko, na bieżąco aktualizując listę podpowiedzi każdorazowo po naciśnięciu klawisza przez użytkownika.

Każda wygenerowana podpowiedź wiąże się z odpytaniem serwera Google (a dokładniej wielu serwerów), zawierającego informacje o zawartości stron WWW. Algorytm aplikacji Google Suggest na podstawie wcześniejszych wyszukiwań dokonanych przez innych użytkowników określa, jakie podpowiedzi powinny być wyświetlone. W trakcie wpisywania hasła w polu wyszukiwania liczba prezentowanych podpowiedzi jest na ogół ograniczona. Jednakże znając możliwości wyszukiwarki Google, można stwierdzić, że liczba wszystkich możliwych podpowiedzi liczona jest w milionach.

Google Maps

Google Maps to połączenie wyszukiwarki z przeglądarką map. Po wejściu na stronę http://mapy.google.pl mapa przedstawia całą Polskę. Poszukiwanie polega na wpisaniu nazwy miejsca i naciśnięciu przycisku [Przeszukaj mapy]. Dzięki dostępnym opcjom wyszukiwanie można zawęzić do konkretnych ulic lub obiektów, takich jak szpitale, restauracje, hotele. Przesuwanie mapy odbywa się przy użyciu kursora myszy bez konieczności przeładowywania strony. Kontrolki znajdujące się w lewej części mapy umożliwiają jej powiększenie bez odrywania od niej wzroku.

Rysunek 10.
Przykład użycia mapy internetowej Google Maps

Na rysunku 10 przedstawiono przykład użycia Google Maps. W odpowiedzi na wpisane zapytanie w lewej części ekranu została wyświetlona lista wyników, a w prawej części ekranu – mapa z zaznaczonymi na niej obiektami. Lista wyników zawiera zdjęcie obiektu, adres, opinie użytkowników na temat danego obiektu. Obiekty na mapie są zaznaczone balonikami. Litera na baloniku odpowiada pozycji obiektu na liście. Wybranie (przy użyciu kursora myszy) obiektu z listy powoduje wyświetlenie dodatkowego okna (tzw. okna pop-up), a czasem nawet niewielkie przesunięcie się mapy, aby obiekt został jak najlepiej pokazany. Okno pop-up zawiera dodatkowe informacje i opcje (np. wyświetlenie trasy dojazdu).

Wymienione powyżej przykłady serwisów wyróżnia duża popularność i innowacyjność. Rola i zastosowanie Internetu nieustannie się zmienia. Wkracza on w nowe obszary naszego życia: e-nauczanie, elektroniczne biblioteki, wirtualne muzea i laboratoria, medycyna, usługi (bankowość, turystyka), handel, rozrywka itp. Omówione przykłady interaktywnych narzędzi są tego dowodem.

PODSUMOWANIE

Żywimy nadzieję, że pozyskana wiedza okaże się słuchaczom pomocna na co dzień i skłoni do wzięcia udziału w innych zajęciach tego projektu, np. dotyczących praktycznych umiejętności tworzenia stron internetowych.

LITERATURA

1. Cohen J., *Serwisy WWW. Projektowanie, tworzenie, zarządzanie*, Helion, Gliwice 2004
2. Price J., Price L., *Profesjonalny serwis WWW*, Helion, Gliwice 2002
3. Sokół R., *Internet. Ilustrowany przewodnik*, Helion, Gliwice 2007

Obraz jako środek przekazu informacji

Andrzej Majkowski
Politechnika Warszawska
amajk@ee.pw.edu.pl

Streszczenie

Obecnie trudno jest znaleźć dziedzinę nauki, a także i rozrywki, w której nie byłoby jakiegoś elementu związanego z cyfrowym przetwarzaniem obrazów. Na wykładzie poznamy, jak powstaje obraz i jak jest postrzegany przez człowieka. Poznamy, po co są tworzone i do czego używane modele barw. Opisane będą takie pojęcia, jak: kontrast, korekcja gamma, temperatura barwowa, balans bieli oraz podane zostanie, jak są one powiązane z jakością fotografii czy filmów. W dalszej części wykładu poznamy podstawy telewizji analogowej i cyfrowej. Omówione zostaną popularne systemy telewizji analogowej NTSC (ang. *National Television System Commitee*) i PAL (ang. *Phase Alternating Line*), cyfrowa telewizja systemu DVB (ang. *Digital Video Broadcast*), standard telewizji HDTV (ang. *High Definition TV*). Opisane zostaną algorytmy poprawy jakości obrazu stosowane w telewizji cyfrowej, umożliwiające eliminację migotania – technika 100 Hz – redukcję artefaktów wynikających z kompresji, lepsze wyeksponowanie konturów obrazu. W części końcowej omówiona zostanie zasada działania wyświetlaczy LCD i ekranów plazmowych. Te dwie technologie zostaną również porównane ze sobą.

Spis treści

1 CYFROWE PRZETWARZANIE OBRAZÓW

Zmysł wzroku odgrywa w życiu człowieka niezwykle istotną rolę, związaną nie tylko z czysto fizycznym rozpoznawaniem i rozróżnianiem otaczających nas przedmiotów i zjawisk, ale wrażenia wzrokowe wpływają także na naszą psychikę czy nastrój. Warto również podkreślić, że tą drogą mózg człowieka przyswaja największą ilość informacji z otaczającego nas świata. Z fizycznego punktu widzenia rejestracja promieniowania świetlnego jest realizowana na siatkówce oka. Siatkówkę oka można przyrównać do pewnego rodzaju światłoczułej matrycy, na której znajdują się receptory widzenia. Takimi receptorami są **pręciki**, które rejestrują jedynie natężenie światła, bez możliwości jego analizy barwnej, oraz **czopki**, które reagują na światło o określonej barwie (rysunek 1). Widzenie barwne jest wynikiem reakcji fotochemicznej, w której substancje białkowe zawarte w czopkach, zwane **opsynami**, reagują na światło absorbując poszczególne składowe promieniowania barwnego. Istnieją trzy rodzaje opsyn: absorbujące światło niebieskie, zielone i czerwone. Umożliwiają one barwne widzenie dzienne. Brak opsyny jednego rodzaju (np. absorbującej światło czerwone) powoduje niezdolność rozróżniania pewnych barw. W wyniku reakcji fotochemicznych energia świetlna zostaje przekształcona na impulsy nerwowe, które są dalej przesyłane przez nerw wzrokowy. Sygnały świetlne docierające do mózgu są zamieniane na cechy, takie jak: kształt, kolor, czy wzajemne relacje przestrzenne obiektów.

Rysunek 1.
Budowa oka [źródło: http://wikipedia.org/wiki/oko]

Obrazy cyfrowe reprezentują te same sceny, które możemy obserwować, ale przedstawione w postaci dwuwymiarowych tablic pikseli. Technika cyfrowa umożliwia przeprowadzenie wielu operacji obróbki obrazu, w tym także działań niewykonalnych tradycyjnymi metodami przy pomocy szklanych filtrów optycznych lub analogowej elektroniki. Jedną z pierwszych prób wykorzystania techniki cyfrowej w praktyce było przesyłanie obrazów na odległość z wykorzystaniem kabla.

Rysunek 2.
Obraz przetransmitowany i odtworzony przez Bartlane System [źródło: http://www.hffax.de/history/html/bartlane.html]

Do przesłania obrazów użyto opracowanego w 1920 roku tzw. Bartlane System, który umożliwiał skanowanie obrazu element po elemencie. Negatyw fotografii był poddawany naświetleniom w pięciu różnych czasach ekspozycji. Obraz był rejestrowany na płytkach cynkowych. Każdy punkt zdjęcia był tym samym charakteryzowany kombinacją pięciu bitów opisującą wzrastającą jasność obrazu. W wyniku skanowania powstawała taśma papierowa rejestrująca poziomy szarości obrazu (5-bitowe kodowanie). Wykorzystanie kabla transatlantyckiego umożliwiło przesłanie obrazów przez Ocean Atlantycki (Londyn – Nowy Jork). Nietrudno sobie wyobrazić jak bardzo uprościło to wymianę informacji. Jeden z pierwszych przesłanych tą drogą obrazów jest przedstawiony na rysunku 2.

Dalszy bardzo szybki rozwój technik cyfrowych nastąpił w latach 1939-45. W czasie II wojny światowej bardzo potrzebne były efektywne systemy rozpoznawania wojskowego, prowadzono więc szeroko zakrojone badania w tym kierunku. Techniki cyfrowe wykorzystano głównie do podwyższania jakości obrazu fotograficznego (dystorsja, nieostrość, kontrast). Początek lat 60. XX wieku to jednocześnie początek misji kosmicznych NASA (misje Rangera). Rysunek 3 przedstawia obraz Księżyca sfotografowany przez statek Ranger 7.

Rysunek 3.
Pierwszy obraz Księżyca sfotografowany przez statek Ranger 7 [źródło: http://pl.wikipedia.org/wiki/Ranger_7]

Fotografię wykonano w 1964 roku przy użyciu kamery telewizyjnej i następnie przesłano na Ziemię. Zdjęcia z tej misji uzmysłowiły konieczność intensyfikacji w rozwoju metod przetwarzania i analizy obrazu. Obecnie cyfrowe przetwarzanie obrazów jest praktycznie wszechobecne. Trudno jest znaleźć dziedzinę nauki, a także i rozrywki, w której nie byłoby jakiegoś elementu związanego z cyfrowym przetwarzaniem obrazów.

1.1 MODELE BARW

Barwa jest wrażeniem psychicznym wywołanym w mózgu człowieka, w czasie gdy oko rejestruje promieniowanie elektromagnetyczne z widzialnej części fal świetlnych. Główny wpływ na to wrażenie ma skład widmowy promieniowania świetlnego, ilość energii świetlnej, obecność innych barw w polu widzenia obserwatora, ale także cechy osobnicze obserwatora: zdrowie, samopoczucie, nastrój, a nawet doświadczenie i wiedza w posługiwaniu się własnym organem wzroku. Barwa z samej swojej natury jest trudna do zdefiniowania, stąd tworzy się mnóstwo wzorców, tabel i modeli próbujących uporządkować barwy. Modele barw są próbą ich opisu przy użyciu pojęć matematycznych. Przy opisie sprzętu najczęściej wykorzystywanymi modelami barw są modele RGB i CMY/CMYK.

Model barw RGB – jest ukierunkowany na sprzęt, w którym barwa powstaje w wyniku emisji światła: monitory, skanery, cyfrowe aparaty fotograficzne. Jest to model addytywny, w którym wszystkie barwy powstają przez zmieszanie trzech barw podstawowych: czerwonej, zielonej i niebieskiej. **Mieszanie addytywne** (rys. 4a) to mieszanie barw poprzez sumowanie wiązek światła widzialnego różnych długości. Synteza addytywna zachodzi np. podczas projekcji na ekran: w miejscu oświetlonym jednocześnie światłem o różnej barwie oko ludzkie widzi odbity strumień światła będący sumą wszystkich padających w to miejsce barw (w widzianym przez nas strumieniu odbitym występują na raz wszystkie długości fal odpowiadające poszczególnym strumieniom światła padającego).

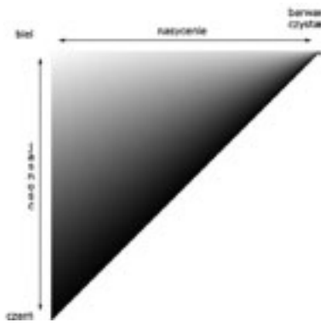
Model barw CMY – jest ukierunkowany na sprzęt drukujący: drukarki, maszyny drukarskie. Wrażenie barwy uzyskuje się dzięki światłu odbitemu od zadrukowanego podłoża. Pigment farb/atramentów pochłania określone długości fali, a odbija pozostałe. Dlatego model ten jest nazywany modelem substraktywnym. **Mieszanie substraktywne** to mieszanie barw poprzez odejmowanie wiązek światła odpowiadającego różnym długościom fal (najczęściej realizowane jest poprzez pochłanianie niektórych długości fal przez powierzchnię, od której odbija się światło białe). Synteza substraktywna zachodzi np. przy mieszaniu farb o różnych barwach: w miejscu pokrytym farbą (powstałą ze zmieszania farb o różnych barwach) oko ludzkie widzi odbity strumień światła będący tą częścią światła białego, która zostanie po pochłonięciu wszystkich składowych barwnych przez poszczególne farby wchodzące w skład mieszanki (rys. 4b). Wszystkie barwy w modelu CMY powstają przez zmieszanie trzech barw podstawowych: cyan (zielono-niebieska), magenta (purpurowa), yellow (żółta). Zmieszanie C, M i Y powoduje odfiltrowanie całego światła i powstaje kolor czarny. W praktyce trudno jest uzyskać w ten sposób idealny kolor czarny. Dlatego powstał model **CMYK**, w którym zdecydowano się na dodanie jeszcze jednego koloru – czarnego (blacK).

a) b)

Rysunek 4.
Addytywne a) i substraktywne b) mieszanie barw [źródło: http://pl.wikipedia.org/wiki/Barwy_podstawowe]

Barwy można opisać używając atrybutów barw. Atrybuty barwy to odcień, nasycenie i jasność. **Odcień** jest cechą jakościową barwy związaną z długością fali dominującej w strumieniu światła. Przy widzeniu barwnym obserwując poszczególne pasma widma o różnych długościach fali stwierdzimy, że istnieje charakterystyczna różnica między każdym z tych wrażeń. Doznawane wrażenia określamy nazywając je kolejno: fioletowy, niebieski, zielony, żółty, pomarańczowy, czerwony. Tę cechę wrażenia wzrokowego nazywamy właśnie odcieniem barwy.

Nasycenie jest cechą jakościową barwy i podaje stosunek ilości światła monochromatycznego do ilości światła białego – im większe nasycenie, tym mniejszy jest udział w widmie promieniowania fal o innych długościach niż fali dominującej. **Jasność, jaskrawość** jest cechą ilościową barwy. Jasność dotyczy obiektów odbijających światło, jaskrawość – świecących i odpowiada wrażeniu słabszego lub mocniejszego strumienia światła.



Rysunek 5.
Atrybuty barwy

Odcień barwy, jasność i nasycenie (trzy atrybuty barwy) są ze sobą ściśle związane. Zmiana jednego atrybutu pociąga za sobą zmianę pozostałych (rys. 5). W zakresie widzenia barwnego wraz ze zmianą jasności zachodzą zmiany barwy postrzeganej. Wrażenie zmiany barwy obserwujemy również, gdy bez zmiany odcienia i jasności zmniejszymy nasycenie barwy.

1.2 KONTRAST, KOREKCJA GAMMA, TEMPERATURA BARWOWA, BALANS BIELI

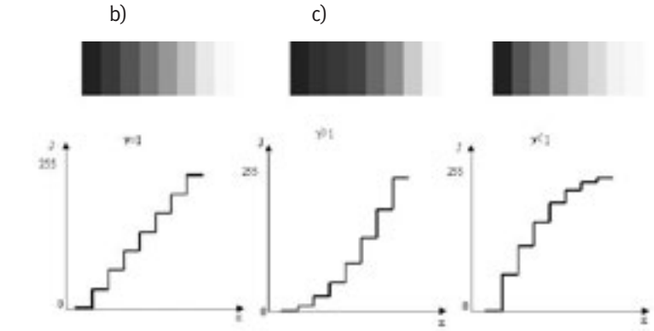
Przy przetwarzaniu obrazów rejestrowanych aparatami cyfrowymi czy kamerami cyfrowymi często używa się pewnych podstawowych pojęć. Poznanie ich znaczenia umożliwia lepsze zrozumienie bardziej złożonych procesów zachodzących podczas przetwarzania obrazów cyfrowych. Pojęcia te to kontrast, korekcja gamma, temperatura barwowa i balans bieli.

Kontrast określa zróżnicowanie jasności poszczególnych punktów ekranu. Z punktu widzenia optymalnej reprodukcji obrazu nie jest tylko istotny maksymalny stosunek pomiędzy najjaśniejszym i najciemniejszym fragmentem ekranu (kontrast maksymalny), lecz także rozkład różnic w jasności poszczególnych części obrazu (gradacja kontrastu). Dla osiągnięcia wiernej reprodukcji rzeczywistości charakterystyka jasności układu przetwarzania i wyświetlania obrazu powinna być liniowa (rys. 7a). Z subiektywnego punktu widzenia niekiedy wskazane jest specjalne kształtowanie gradacji kontrastu. Często stosuje się nieliniowe przetwarzanie np. w celu pełnego wykorzystania dynamiki obrazu (czyli poprawnego zróżnicowanie skali szarości zarówno w jasnych, jak i ciemnych partiach obrazu – rys. 6). Technika cyfrowa daje tutaj możliwości nieosiągalne dla techniki analogowej.

Często wprowadza się celowo pewną nieliniowość przetwarzania, aby w efekcie otrzymać liniową charakterystykę końcową. W przypadku liniowej charakterystyki przetwarzania (rys. 7a) jasność obrazu J jest proporcjonalna do czynnika ją wywołującego (np. napięcia x na przetworniku). Nieliniowa charakterystyka świetlna $J - x$ (rys. 7b, 7c) może być opisana w następujący sposób $J \sim x^{\gamma}$, czyli jasność obrazu jest proporcjonalna do wywołującego ją napięcia x podniesionego do potęgi γ. Wykładnik γ oznacza stopień nieliniowości przetwornika. Od greckiej litery γ, określającej ten współczynnik, korekcja charakterystyki przeprowadzana w ten sposób nosi nazwę **korekcji gamma**.

W systemie przetwarzania i wyświetlania obrazów istotną rzeczą jest wierna reprodukcja barw. Barwa obiektów zarejestrowana przez kamerę czy aparat fotograficzny zależy od koloru oświetlenia. W tym przypadku barwa np. koloru skóry czy bieli śniegu na zdjęciu może być różna od tej, jakiej oczekujemy. Zadaniem korekcji barw jest właśnie sprowadzenie postaci barw do formy akceptowalnej przez widza. Prawidłowe odwzorowanie koloru śniegu jest przykładem ustawienia **balansu bieli**. Często na odbitkach fotograficznych wykonanych z tego samego negatywu, w różnych zakładach fotograficznych widoczne są różnice w jego zabarwieniu: śnieg przybiera zabarwienie niebieskie, żółte, zielone, a niekiedy różowe. Celem ustawienia balansu bieli jest usunięcie tego zabarwienia.

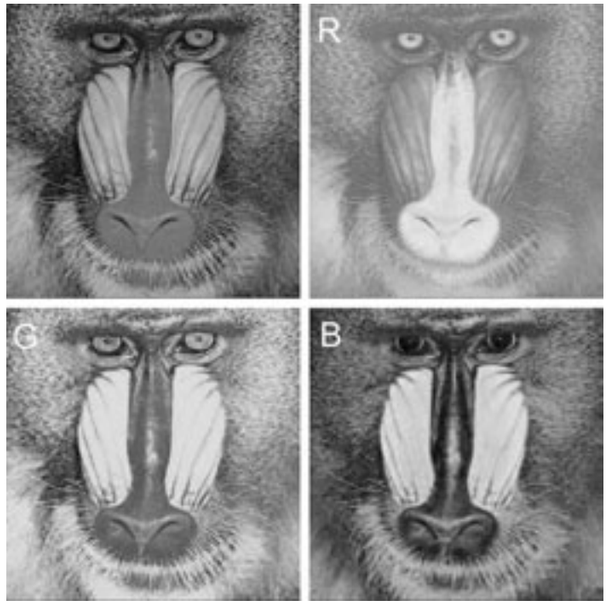
Rysunek 6.
Efekt zastosowania korekcji gamma, lewy górny róg – obraz oryginalny, pozostałe obrazy są wynikiem zastosowania korekcji gamma z różnym współczynnikiem γ



Rysunek 7.
Ilustracja korekcji gamma [źródło: 7]

Niekiedy potrzebne jest połączenie procesu korekcji barw z korekcją gamma dla obrazu czy sygnału wizyjnego rozłożonego na składowe RGB (rys. 8). Korekcje stosuje się oddzielnie dla każdego obrazu: czerwonego, zielonego i niebieskiego. Inaczej mogą się pojawić zafałszowania barw w zależności od jasności poszczególnych fragmentów obrazu. Jest to wynik tzw. braku równowagi dynamicznej bieli.

Temperatura barwowa, jako cecha określająca wrażenie percepcyjne oglądanego obrazu, zależy głównie od rodzaju oświetlenia oraz od właściwości barwnych elementów występujących w scenie obrazowej. W praktyce temperaturę barwową definiuje się na podstawie relacji, jakie zaobserwowano pomiędzy temperaturą a właściwościami emisyjnymi ciała czarnego. Temperaturę barwową oblicza się na podstawie średniej wartości kolorów całego obrazu, z pominięciem pikseli, które nie mają wielkiego wpływu na temperaturę barwową, a mianowicie pikseli koloru czarnego i tzw. pikseli samoświecących, czyli o jasności większej od wartości średniej o pewną wartość progową. Obraz kwalifikowany jest do kategorii barwowej według przedziału temperatur, do którego należy obliczona wartość. Przedziały te zostały wyznaczone doświadczalnie za pomocą badań subiektywnych (patrz tab. 1).



Rysunek 8.
Obraz rozłożony na składowe RGB (czerwony, zielony, niebieski)

Tabela 1.
Zakresy temperatur barwowych

Kategoria subiektywna	Zakres temperatur
Gorąca	1667 K ~ 2250 K
Ciepła	2251 K ~ 4170 K
Neutralna	4171 K ~ 8060 K
Zimna	8061 K ~ 25 000 K

Poniżej przedstawiono trzy zdjęcia, których temperatura barwowa jest różna. Zdjęcie z lewej na neutralną temperaturę barwową, w środku – temperatura barwowa jest przesunięta ku czerwieni, zdjęcie z prawej ma temperaturę barwową przesuniętą w stronę barwy niebieskiej.

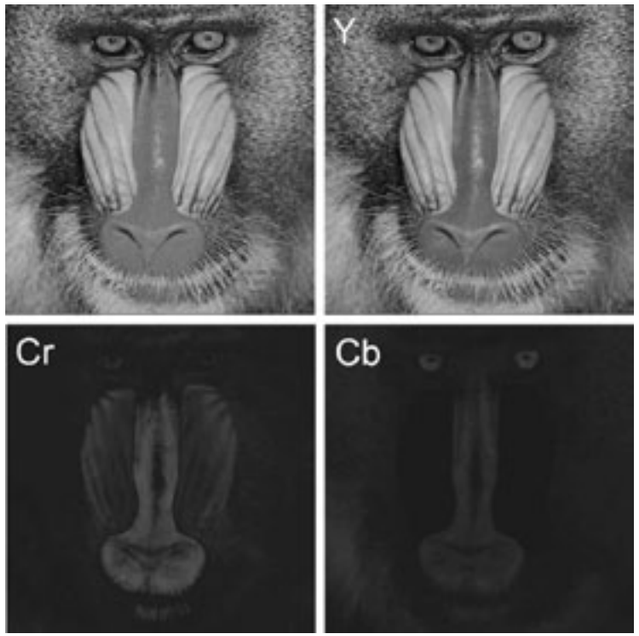
Rysunek 9.
Przykład zdjęć o różnej temperaturze barwowej

2 TELEWIZJA ANALOGOWA I CYFROWA

Prace nad systemem telewizji kolorowej, rozpoczęły się w połowie lat 50. XX wieku w Stanach Zjednoczonych. Nowy system musiał spełniać następujące założenia:

- nie komplikować budowy odbiorników telewizji kolorowej, co mogłoby wpływać na koszt produkcji odbiornika telewizyjnego i zmniejszyć jego dostępność dla widza ze względu na cenę;
- posiadać możliwości odbioru programu telewizji nadawanego w kolorze na odbiornikach czarno-białych i odwrotnie;
- wykorzystywać dotychczasowe kanały częstotliwości do przesyłania sygnałów telewizji kolorowej, nie powodując zakłóceń w kanałach sąsiednich;
- jakość przesyłanego sygnału miała być wysoka i zaspakajać wymagania widza.

Najpopularniejsze systemy telewizji analogowej to **NTSC** i **PAL**. Mają one ze sobą wiele wspólnego. W zasadzie system PAL jest udoskonaloną modyfikacją systemu NTSC. Całkowity sygnał wizyjny jest na wielu etapach przetwarzania obrazu reprezentowany przez składowe: luminancję Y i dwa sygnały różnicowe koloru C_r oraz C_b, czyli tzw. sygnały chrominancji (rys. 10). Nie stosuje się przy transmisji i kompresji sygnałów RGB, gdyż każdy z nich jest sygnałem pełnej szerokości pasma. Sygnały różnicowe koloru mogą być natomiast ograniczone częstotliwościowo w stosunku do sygnału luminancji bez wpływu na jakość zrekonstruowanego obrazu barwnego. Próbki chrominancji występują w strukturze linii (czyli w kierunku poziomym) dwukrotnie rzadziej niż elementy luminancji. Podobną zasadę można zastosować w kierunku pionowym, czyli umieszczać próbki chrominancji na co drugiej linii. Takie ograniczenie rozdzielczości w pionie nie wpływa zasadniczo na jakość obrazu kolorowego, natomiast istotnie redukuje strumień informacji o obrazie.



Rysunek 10.
Obraz rozłożony na składowe: luminancja Y (obraz w skali szarości) i składowe chrominancji C_r i C_b (przenoszące informację o kolorze)

W systemie NTSC obraz jest składany z 525 linii na ramkę, przy częstotliwości odświeżania 59,94 Hz (jest to skutkiem stosowania w USA częstotliwości prądu przemiennego wynoszącej 60 Hz) i 29,97 ramkach na sekundę. Stosowany w Polsce standard telewizji kolorowej PAL bazuje na strukturze ramki obrazu zawierającej 625 linii i składającej się z dwóch pól półobrazów powtarzanych z częstotliwością 50 Hz. W standardzie PAL stosuje się strukturę wybierania linii określaną jako wybieranie kolejnoliniowe nieparzyste. Linie należące do kolejnego półobrazu są wyświetlane na ekranie pomiędzy liniami poprzedniego. Każda pełna ramka obrazu pojawia się wobec tego 25 razy na sekundę.

2.1 STANDARD TELEWIZJI KOLOROWEJ HDTV

HDTV (ang. *High Definition TV*) to telewizja wysokiej rozdzielczości. W potocznym znaczeniu jest określeniem sygnału telewizyjnego o rozdzielczości większej niż standardowa (PAL lub NTSC). Pierwsze publiczne instalacje analogowej telewizji w wysokiej rozdzielczości zostały uruchomione w Japonii, gdzie wciąż cieszą się dużą popularnością, mimo równoległej transmisji w systemie cyfrowym. Podczas gdy w USA telewizja wysokiej rozdzielczości stawała się coraz popularniejsza, w Europie przez dłuższy czas nie była stosowana w publicznych przekazach. W końcu jednak w 2004 roku pojawiła się pierwsza stacja nadająca z europejskiego satelity Astra – euro1080. W Polsce pierwszym operatorem kablowym, który wprowadził usługę HDTV (w 2007 roku), były Multimedia Polska SA HDTV oferuje rozdzielczości:

- 720p – 1280×720 pikseli,
- 1080i/1080p – 1920×1080 pikseli,

gdzie „i" (ang. *interlaced*) oznacza obraz z przeplotem (na zmianę wyświetlane są linie parzyste i nieparzyste), po symbolu „i" czasem podawana jest liczba pól (półobrazów) na sekundę, np. 1080i60, natomiast „p" (ang. *progressive scan*) oznacza obraz bez przeplotu. Po symbolu „p" podawana jest czasem liczba klatek (pełnych obrazów) na sekundę, np. 720p60.

Przed przesłaniem do użytkownika końcowego sygnał HDTV może być zakodowany na kilka sposobów, wśród których najczęściej stosuje się: MPEG-2, H.264/MPEG-4 AVC .

2.2 CYFROWA TELEWIZJA SYSTEMU DVB

DVB (ang. *Digital Video Broadcast*) to standard cyfrowej telewizji. Charakteryzuje się jakością obrazu i dźwięku porównywalną do zapisu DVD. Telewizja DVB umożliwia często interaktywny odbiór, np. włączenie napisów w różnych językach oraz przełączenia języka ścieżki audio. W standardzie DVB obraz i dźwięk są przesyłane w systemie MPEG-2, tak jak w DVD. Niedawno wprowadzono również kodowanie w MPEG-4 (H.264). Sygnał DVB może być przekazywany z nadajników naziemnych (DVB-T), satelity (DVB-S) i stacji telewizji kablowych (DVB-C). Podstawą tego systemu jest strumień transportowy (TS) zdefiniowany i opisany w normie międzynarodowej ISO/IEC 13818-1. TS składa się ze skompresowanych składowych wizji, fonii i danych oraz tablic (PSI) umożliwiających urządzeniu odbiorczemu odbiór wybranego programu telewizyjnego lub radiowego oraz danych. Standard DVB definiuje dodatkowe tablice (SI) umieszczone w strumieniu oraz parametry transmisji w zależności od typu kanału transmisyjnego.

3 POPRAWA JAKOŚCI OBRAZU

Technika cyfrowa umożliwia zastosowanie wielu metod poprawy jakości obrazu przekazów telewizyjnych. Najczęściej spotykane zniekształcenia wynikają z pojawienia się artefaktów (zniekształceń) procesu kompresji. Do innych zakłóceń zaliczamy między innymi: szumy, interferencje (przenikanie sygnałów luminancji i chrominancji), migotanie powierzchni i linii, zaburzenia synchronizacji. Eliminacja wymienionych zjawisk jest możliwa przy wykorzystaniu dwu- i trójwymiarowych filtrów cyfrowych, filtrów grzebieniowych, układów korekcji podstawy czasu i stosowaniu odpowiednich technik (100 Hz, obraz bez przeplotu). Poprawie jakości sprzyja też sztuczne podnoszenie rozdzielczości w oparciu o technikę nadpróbkowywania i interpolacji wartości pikseli.

3.1 ELIMINACJA MIGOTANIA

Eliminacja migotania – technika 100 Hz – polega na podwajaniu częstotliwości powtarzania półobrazów. Wprowadzenie w standardzie PAL wybierania międzyliniowego z częstotliwością powtarzania półobrazów (pola) 50 Hz miało w założeniu doprowadzić do zmniejszenia efektu migotania jasnych płaszczyzn na ekranie telewizora. Zjawisko migotania obrazu staje się szczególnie dokuczliwe przy przekątnych większych niż 29 cali. W odbiornikach stosuje się więc podwajanie częstotliwości powtarzania półobrazów, czyli technikę 100 Hz. Może być ona realizowana w różnych wariantach. Załóżmy, że mamy sekwencję wizyjną złożoną z półobrazów (pól) A i B wyświetlanych co 20 ms, składających się na całkowity obraz o rozdzielczości pionowej 576 linii. W wariancie AABB półobraz A zostaje wyświetlony dwa razy pod rząd co 10 ms, a następnie tak samo reprodukowany jest półobraz B. Takie rozwiązanie eliminuje migotanie dużych jasnych powierzchni ekranu, ale wprowadza często bardziej dokuczliwe zjawisko migotania linii i konturów w obrazie. Tej wady nie ma sposób odtwarzania ABAB, wymagający jednak większej pamięci, zdolnej przechować dwa półobrazy. Wariant ten powoduje jednak zniekształcenia w odtwarzaniu szybko poruszających się obiektów (efekt „rozdwajania"). Obecnie stosuje się interpolację treści półobrazów, polegającą na utworzeniu na podstawie przesyłanej informacji nowych półobrazów A' i B'. Algorytmy interpolacyjne tak wyliczają wartości nowych pikseli, aby w rezultacie doprowadzić do poprawnego odtwarzania ruchu przy niezauważalnym migotaniu. Treść wizyjna wyświetlana jest z częstotliwością 100 Hz w kolejności AA'BB'.

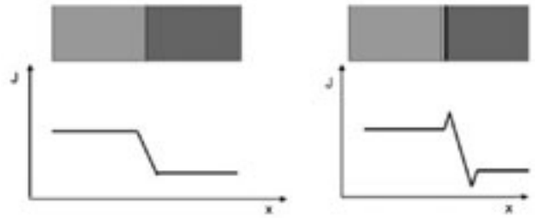
3.2 REDUKCJA ARTEFAKTÓW WYNIKAJĄCYCH Z KOMPRESJI

Stosowanie stratnej kompresji wprowadza do rekonstruowanego obrazu wiele zniekształceń zwanych **artefaktami**. Mogą one powodować wrażenie istotnego pogorszenia jakości. Za powstanie artefaktów odpowiada zwykle koder MPEG-2 stosowany po stronie nadawczej. Do typowych zjawisk należy tutaj efekt blokowy. Jest on charakterystyczny dla metod kompresji bazujących na przetwarzaniu bloków pikseli. W procesie kwantyzacji składowe stałe reprezentujące sąsiednie bloki mogą być zakodowane z różną precyzją, co powoduje później widoczne różnice w luminancji tła fragmentów obrazu. Na ekranie pojawia się wtedy charakterystyczna struktura siatki.

Innym artefaktem jest *mosquito noise*. Nazwa bierze się z faktu, że przypomina on chmarę komarów unoszących się nad obiektem w rytmie jego ruchów. Zjawisko wynika z tego, że w procesie kodowania MPEG-2 fragmenty obrazów odpowiadające wyższym częstotliwościom przestrzennym, a więc opisujące występujące w obrazie krawędzie, są kodowane z małą precyzją. Wspomniane artefakty są usuwane w układach filtrów cyfrowych. Zastosowanie prostych filtrów cyfrowych może jednak prowadzić do zmniejszenia wyrazistości obrazu lub innych efektów pogarszających jego subiektywną ocenę.

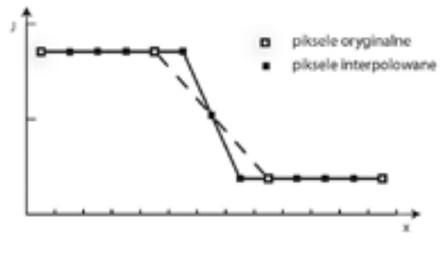
3.3 EKSPONOWANIE KONTURÓW OBRAZU

Poprawa ostrości konturów odbierana jest przez większość osób jako zwiększenie rozdzielczości. Już samo zwiększenie kontrastu, które powoduje większe różnice w jasności sąsiadujących fragmentów obrazu, potęguje wrażenie lepszej ostrości konturów. Jednak zwiększanie kontrastu w skali całego obrazu prowadzi do zatarcia się poziomów jasności w ciemnych i jasnych partiach obrazu. Stosuje się więc zabieg polegający na lokalnym powiększeniu kontrastu w bezpośrednim otoczeniu krawędzi (rys. 11). Efekt ten można osiągnąć stosując dwuwymiarowy cyfrowy filtr górnoprzepustowy. Uwypuklenie wysokich częstotliwości przestrzennych powoduje wzrost dostrzegalności drobnych szczegółów obrazu poprzez silniejsze zróżnicowanie jasności w otoczeniu konturów.



Rysunek 11.
Lokalne uwypuklenie konturów obrazu [źródło: 7]

Innym sposobem poprawy ostrości jest zwiększenie stromości zboczy sygnału wizyjnego (rys.12). Stosując technikę nadpróbkowywania można utworzyć zbiór nowych pikseli w taki sposób, aby zrekonstruowany sygnał charakteryzował się pasmem charakterystycznym dla telewizji HDTV.



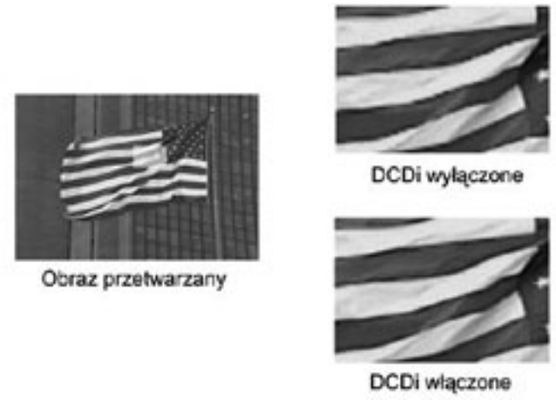
Rysunek 12.
Technika nadpróbkowywania

Do obliczenia wartości nowych pikseli są stosowane metody interpolacji. Proces **interpolacji** ma na celu utworzenie nowego, wcześniej nieistniejącego piksela na podstawie pikseli sąsiadujących z pikselem tworzonym tak, aby był on jak najlepiej dopasowany optycznie do przetwarzanego obrazu. Dobierając właściwy algorytm interpolacji można osiągnąć efekt poprawy stromości zboczy bez zwiększania lokalnego kontrastu. Przejścia pomiędzy fragmentami odpowiadającymi różnej jasności będą wtedy „bardziej strome". W stosunku do wcześniej przedstawionej metody zwiększania ostrości, ta technika nie wprowadza zniekształceń grzbietu sygnału przed i po zboczu. Wspomniany sposób poprawy ostrości wykorzystano w technologii D.I.S.T. stosowanej w niektórych odbiornikach HDTV, do zwiększenia rozdzielczości obrazu wizyjnego.

3.4 ALGORYTMY POPRAWY JAKOŚCI OBRAZU

Technologia D.I.S.T. (ang. *Digital Image Scaling Technology*) opracowana została przez firmę JVC. Umożliwia ona redukcję migotania przy jednoczesnej poprawie rozdzielczości obrazu. Obraz przekazywany w konwencjonalnym 625-liniowym standardzie PAL z przeplotem zostaje na wstępie przetworzony do trybu progresywnego (czyli obraz jest wyświetlany bez przeplotu). Odbywa się to na drodze trójwymiarowej interpolacji wartości pikseli z linii półobrazów parzystego i nieparzystego, z wykorzystaniem relacji czasowych i przestrzennych między nimi. Specjalny algorytm interpolacji umożliwia uzyskanie wysokiej rozdzielczości w kierunku pionowym i podwojenie liczby linii w ramce do 1250. Sygnał wizyjny jest następnie formowany poprzez ekstrakcję 3 pól o częstotliwości 75 Hz z dwóch ramek 50 Hz i podawany na wyjście układu DIST w trybie wybierania międzyliniowego 1250/75 Hz. Zwiększenie częstotliwości wyświetlania półobrazów przyczynia się w tym przypadku do ograniczenia efektu migotania.

Redukcję zniekształceń krawędzi i linii umożliwia **technologia DCDi** (ang. *Directional Correlation Deinterlacing*) firmy Faroudja. W konwencjonalnej telewizji nieraz dostrzegalne są zniekształcenia polegające na poszarpaniu ukośnych linii lub konturów. Wrażenie to jest spotęgowane przy sekwencjach odtwarzanych w zwolnionym tempie (np. powtórka finiszu biegu – linie bieżni). Jednym z układów redukujących tego typu zniekształcenia jest DCDi. Ta technologia jest również wykorzystywana przez nadawców w USA do konwersji standardu NTSC do telewizji wysokiej rozdzielczości HDTV. Algorytm zaimplementowany w DCDi polega na „inteligentnej" interpolacji pikseli w zależności od charakteru ruchu obiektu w analizowanej scenie i kąta nachylenia konturów. Mechanizm interpolacji przebiega dzięki temu wzdłuż krawędzi nie dopuszczając do efektu ich poszarpania lub schodkowania, przy jednoczesnym zachowaniu ostrości i wierności oddania barw w miejscu przejść między kolorami.



Rysunek 13.
Ilustracja działania DCDi [źródło: http://www.gnss.com/tch_dcdi_overview.phtml]

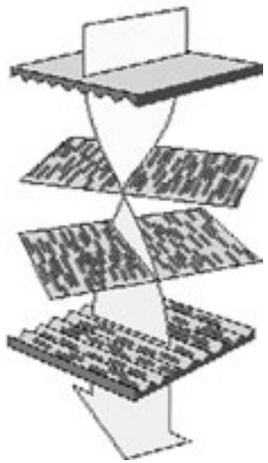
Na rysunku 13 przedstawiono zastosowanie technologii DCDi do poprawy jakości obrazu. Z lewej strony znajduje się obraz flagi łopoczącej na wietrze nadawany przez zwykłą telewizję. Jest to trudny obraz to wiernego odtworzenia w konwencjonalnej telewizji. Po prawej stronie przedstawiono powiększenia tego obrazu. W przypadku górnego obrazu filtry DCDi są wyłączone. Widać wyraźne poszarpanie krawędzi linii. W przypadku dolnego zaś DCDi jest włączona. Postrzępienie linii zniknęło, także połączenie obszarów czerwonych i białych jest bardziej naturalne.

4 WYŚWIETLACZE LCD

Ciekłe kryształy zostały wynalezione w XIX wieku przez austriackiego botanika Friedricha Reinitzera. Termin „ciekły kryształ" rozpropagował niemiecki fizyk Otto Lehmann. Ciekłe kryształy to substancje prawie przezroczyste, mające właściwości charakteryzujące zarówno ciała stałe, jak i ciecze. Światło przechodzące przez ciekłe kryształy podąża za ułożeniem tworzących je molekuł. W 1960 roku odkryto, że pobudzenie napięciem elektrycznym ciekłych kryształów zmienia położenie tych molekuł, a co za tym idzie – sposób przenikania przez nie światła. W roku 1969 James Fergason odkrył efekt skręconego nematyka (ang. *twisted nematic* – TN). Było to odkrycie o fundamentalnym znaczeniu, ponieważ wyświetlacze LCD wykorzystują właśnie to zjawisko.

Panele LCD świecą dzięki zastosowaniu specjalnych lamp z tzw. zimną katodą. Charakteryzują się one bardzo dużą wydajnością przy jednoczesnym niewielkim zużyciu energii. Użycie filtru polaryzującego światło powoduje

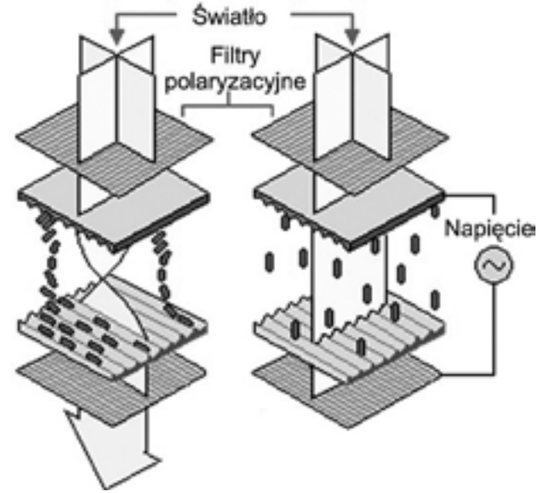
polaryzację przechodzącej przez niego wiązki światła. Polaryzacja światła zależy od orientacji wektora jego pola elektrycznego. W uproszczeniu światło to fala elektromagnetyczna. Wektory pola elektrycznego i magnetycznego są prostopadłe do kierunku fali ruchu. Lampa emituje niespolaryzowane światło, więc pole elektryczne może poruszać się w dowolnym kierunku prostopadłym do osi propagacji światła. Gdy światło przechodzi przez polaryzator, to wychodząc po drugiej stronie ma wektor pola elektrycznego skierowany w znanym kierunku (np. pionowym). Światło nie może jednak przejść przez drugi polaryzator, prostopadły do pierwszego (w tym wypadku poziomy). Ale jeżeli między dwoma polaryzatorami umieści się ciekły kryształ, to zmienia on polaryzację światła na pasującą do drugiego polaryzatora i wtedy jest ono przepuszczane przez cały układ.



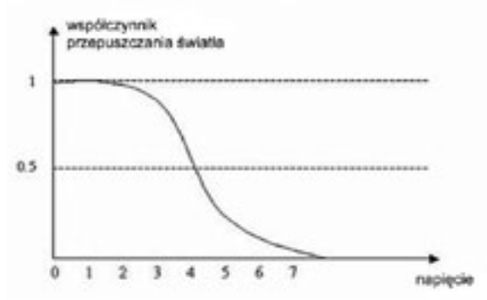
Rysunek 14.
Zasada działania wyświetlacza LCD w technologii TN [źródło: http://www.pctechguide.com/flat_panel-displays/liquid-crystal-light-polarisation-in-lcd-monitors]

W technologii TN ekran LCD składa się z dwóch warstw ciekłych kryształów umieszczonych pomiędzy dwiema odpowiednio wyprofilowanymi powierzchniami (rys. 14), z których jedna jest ustawiona najczęściej pod kątem 90° wobec drugiej (stąd w nazwie skręcenie – *twisted*). Molekuły znajdujące się między nimi muszą się przemieścić o 90°, podobnie jak światło podążające za ich położeniem. Wystarczy jednak przyłożyć do ciekłych kryształów napięcie elektryczne, a molekuły zaczną się przemieszczać pionowo, powodując przejście światła bez zmiany położenia o 90° (rys. 15). Technologia TN jest najczęściej stosowana w niedrogich modelach monitorów komputerowych o niewielkich przekątnych obrazu – 15, 17 cali. Matryce tego typu nie najlepiej reprodukują barwy, mają jednak również mocne strony. Gwarantują na przykład krótki czas reakcji – w najnowszych modelach 4 ms. Największą wadą matryc TN jest stosunkowo wąski kąt widzenia, 120 – 140 stopni w obu płaszczyznach. Panele tego typu nie nadają się do zastosowań profesjonalnych (pracy z grafiką), jednak z racji niskiej ceny i wspomnianego krótkiego czasu reakcji dobrze sprawdzą się podczas oglądania dynamicznych materiałów wideo.

Jedną z wad paneli ciekłokrystalicznych jest gorsze odwzorowanie barw w stosunku do monitorów CRT. Ma to szczególnie znaczenie przy profesjonalnej obróbce zdjęć, zwłaszcza gdy wyniki naszej pracy mają trafić do drukarni. Normalny obserwator odróżnia od 300 tysięcy do 1 miliona barw natomiast przy pomocy techniki cyfrowej można na monitorze komputera przedstawić obraz w modelu RGB, mający dokładność 24 bitów (3 kanały x 8 bitów). Każdy kanał barwy R, G, B jest opisany liczbą 8 bitową, dzięki czemu można uzyskać w każdym kanale 256 poziomów jasności. Kombinacja 256 poziomów koloru czerwonego, zielonego i niebieskiego definiuje przeszło 16 milionów kolorów. Różnicując napięcie na końcówkach ciekłego kryształu można modulować stopień zamknięcia przełącznika, aby uzyskać stany pośrednie (rys. 16). Niestety w technologii TN można w ten sposób uzyskać jedynie rozdzielczość ok. 6 bitów (8 bitów w technologii CRT). Pozostałe, brakujące barwy są zazwyczaj uzyskiwane na zasadzie interpolacji z lepszym lub gorszym skutkiem.



Rysunek 15.
Napięcie powoduje zmianę położenia molekuł ciekłego kryształu [źródło: http://www.pctechguide.com/flat-panel-displays/liquid-crystal-light-polarisation-in-lcd-monitors]

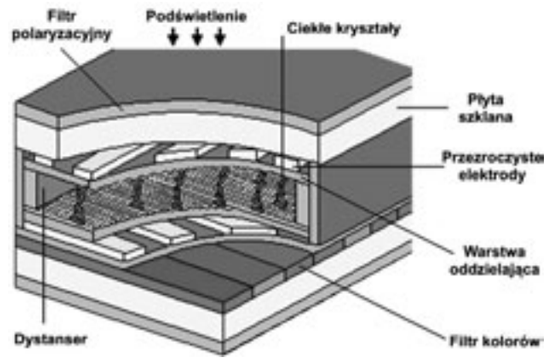


Rysunek 16.
Różnicując napięcie na końcówkach ciekłego kryształu można modulować stopień zamknięcia przełącznika, aby uzyskać stany pośrednie

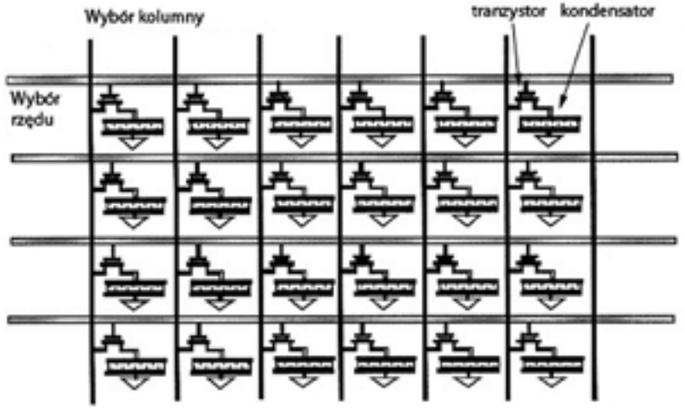
W **matrycach pasywnych** (rys. 17) ciekłe kryształy są adresowane poprzez ładunki lokalne, przy czym nic nie powstrzymuje ładunków elektrycznych przed rozpływaniem się na boki i wpływaniem na położenie molekuł kryształów sąsiednich. Adresowanie, czyli określenie piksela, który ma być w danej chwili wysterowany, realizowane jest przy użyciu dwóch krzyżujących się elektrod. Elektroda przednia jest wspólna dla całej kolumny i przewodzi prąd, natomiast tylna elektroda, wspólna dla całego rzędu, służy jako uziemienie. Czas reakcji matrycy musi być bardzo długi, nawet kilkaset milisekund, gdyż ciekły kryształ musi zachować orientację molekuł

do następnego zaadresowania. Nic nie podtrzymuje orientacji molekuł, stąd powoli wracają one do położenia pierwotnego. Matryce pasywne charakteryzują się rozmytym obrazem oraz smugami i cieniami ciągnącymi się za obiektami w ruchu.

Matryce aktywne mają podobną budowę do matryc pasywnych. Podstawową różnicę stanowi warstwa tranzystorów cienkowarstwowych (ang. *thin film transistor* – TFT). Poprzez te tranzystory sterowane są kondensatory, które gromadzą i utrzymują ładunki elektryczne, zapobiegając ich rozpływaniu się na inne piksele (rys. 18). Tranzystor powiązany jest tylko z jedną komórką ciekłego kryształu, dzięki czemu nie ma smużenia ani rozmycia obrazu. Jeśli do elektrod przyłożymy napięcie, to spowodujemy, że cząsteczki ciekłych kryształów zmienią położenie i zostaną skręcone. Zapisany ładunek pozostaje na kondensatorze i dzięki temu na końcówkach kryształów nadal jest napięcie, nawet gdy linia adresuje inny piksel. Nie powróci on więc do stanu początkowego, co miało miejsce w przypadku matryc pasywnych. Czas zapisu ładunku do kondensatora jest o wiele krótszy niż czas obrotu kryształu, co oznacza, że dane mogą być zapisane, a kolejny piksel zaadresowany jest natychmiast, bez opóźnień. Obecnie stosuje się praktycznie wyłącznie matryce aktywne.

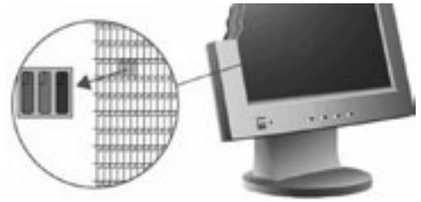


Rysunek 17.
Budowa matrycy pasywnej [źródło: http://www.pctechguide.com/flat-panel-displays]



Rysunek 18.
Budowa matrycy aktywnej [źródło: http://www.wtec.org/loyola/dsply_jp/c5_s2.htm]

Specjalne filtry nadają kolor poszczególnym subpikselom ulokowanym na przedniej warstwie szkła (rys. 19). Trzy subpiksele, każdy w kolorze czerwonym, zielonym oraz niebieskim, formują piksel. Różne kombinacje kolorystyczne subpikseli dają obraz oraz kolor na ekranie.

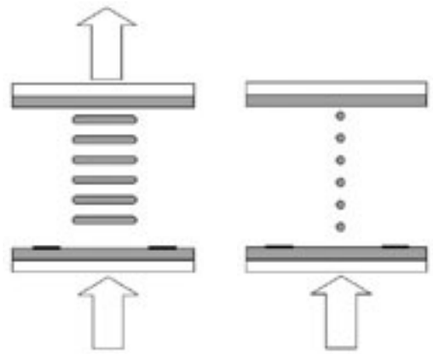


Rysunek 19.
Sposób nakładania filtrów barwnych [źródło: 3]

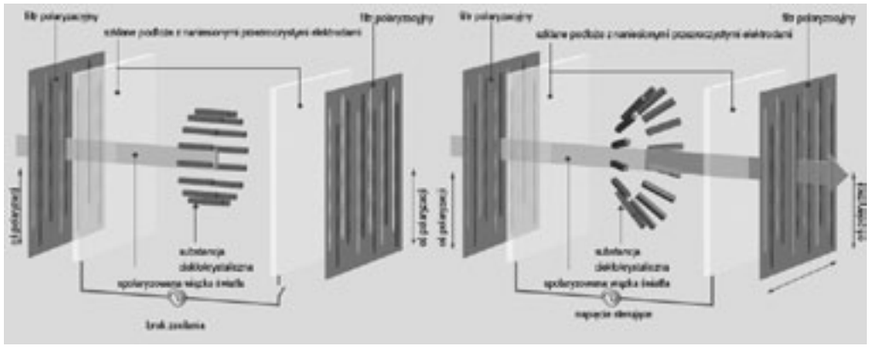
Oprócz omawianej wcześniej technologii TN wytwarzania wyświetlaczy LCD istnieją jeszcze dwie alternatywne: IPS, MVA.

Technologia **IPS, S-IPS** (ang. *In-Plane Switching*) została zaprojektowana w 1995 roku przez firmę Hitachi. Tym, co różni wyświetlacze IPS od wykonanych w technologii TN jest równoległe do powierzchni ułożenie molekuł ciekłych kryształów. Przy użyciu technologii IPS osiągany jest doskonały kąt widzenia, aż do 170°, jaki znamy z normalnych monitorów (CRT). Jednakże jest też minus: z powodu równoległego ułożenia ciekłych kryształów, elektrody nie mogą być umieszczone na obydwu szklanych powierzchniach jak w wyświetlaczu TN. Zamiast tego, muszą być zainstalowane w kształcie grzebienia na dolnej, niższej powierzchni szklanej. Prowadzi to ostatecznie do redukcji kontrastu i dlatego wymagane jest silniejsze tylne podświetlenie dla podniesienia jasności obrazu. Obecnie S-IPS ma zalety matryc VA (ładne kolory, szerokie kąty widzenia) oraz TN (szybkość działania).

Na rysunku 20 z lewej strony przedstawiono stan, w którym piksel wyświetlacza jest jasny (stan włączenia). Po przyłożeniu napięcia molekuły kryształu zaczynają się obracać o 90° pozostając w pozycji równoległej do siebie i do powierzchni wyświetlacza. Takie ułożenie molekuł ciekłego kryształu powoduje, że światło jest blokowane.



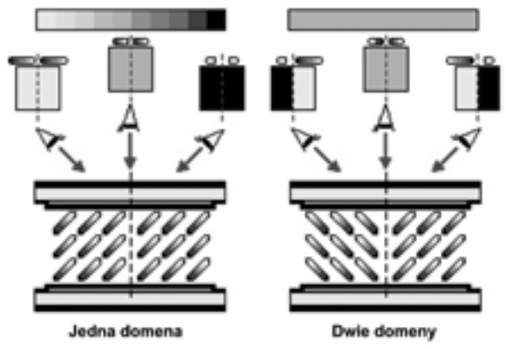
Rysunek 20.
Sposób ułożenia molekuł ciekłego kryształu w technologii IPS



Rysunek 21.
Budowa wyświetlaczy LCD wykonanych w technologii MVA [źródło: 3]

MVA, PVA, WVA to matryce uznawane za najlepiej odwzorowujące barwy. W popularnych matrycach TN ułożenie ciekłych kryształów będących w stanie pobudzenia jest nieuporządkowane, natomiast w przypadku tych technologii – wielokierunkowe, dzięki czemu obraz jest spójny, o mniej widocznej strukturze pikseli (rys. 21). Technologię PVA opracowała firma Samsung, natomiast MVA i WVA opatentowali inni znaczący producenci matryc LCD – Fujitsu i CMO. Zasada działania „trzech technologii VA" jest praktycznie taka sama, a różnice w nazewnictwie są spowodowane kwestiami praw patentowych. Panele tego typu mają bardzo szerokie kąty widzenia, minimum 170 stopni w obu płaszczyznach. Niestety mają też jedną poważną wadę – długi czas reakcji. Najszybsze modele MVA mają czas reakcji rzędu 25 ms, co sprawia, że podczas wyświetlania szybko przemieszczających się po ekranie obiektów można zauważyć smużenie.

Na rysunku 22 przedstawiono zasadę działania matrycy MVA. W tym przypadku zastosowane są tylko dwie domeny, w praktyce muszą być co najmniej cztery. Światło dochodzi do obserwatora w momencie kiedy molekuły ciekłego kryształu są ułożone prostopadle do kierunku obserwacji, a zatrzymywane jest w przypadku równoległego ułożenia molekuł. W położeniach pośrednich przepuszczana jest tylko część światła. W przypadku jednej domeny (rys. 22 po lewej) obserwator, przechodząc z lewej strony na prawą, widzi stopniowy zanik świecenia piksela. W przypadku zastosowania dwóch domen (rys. 22 po prawej) z lewej strony jedna domena przepuszcza światło, druga natomiast je blokuje, z prawej strony jest odwrotnie. Gdy obserwator patrzy na wprost, obie domeny przepuszczają tylko część światła. Ponieważ piksele są bardzo małe, oko i mózg ludzki uśredniają czarno-białe pola z lewej i prawej strony do koloru szarego. Obserwator w tym przypadku widzi jednolity szary kolor w bardzo szerokim zakresie kątów widzenia.



Rysunek 22.
Zasada działania matryc MVA [źródło: http://www.pctechguide.com/flat-panel-displays/mva-multi-domain-vertical-alignment-in-lcd-monitors]

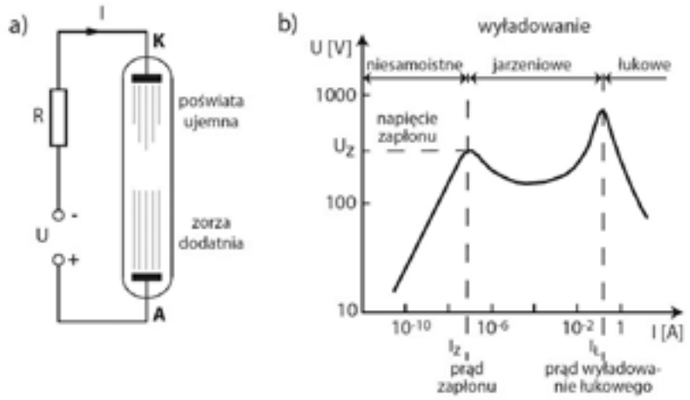
Podstawowe parametry charakteryzujące monitory ciekłokrystaliczne:

- **Rozdzielczość** – monitor LCD w przeciwieństwie do modeli CRT pracuje z maksymalną jakością tylko w rozdzielczości rzeczywistej (natywnej). Oczywiście prezentacja obrazu z inną rozdzielczością jest możliwa, jednak wtedy mamy do wyboru dwa sposoby oglądania obrazu – wyświetlany na fragmencie matrycy odpowiadającej danej rozdzielczości (np. 640x480 na panelu o rzeczywistej rozdzielczości 1024x768) lub prezentowany na całej powierzchni ekranu przy użyciu algorytmów interpolowania.
- **Częstotliwość odświeżania obrazu** w monitorach LCD – bezwładność monitorów ciekłokrystalicznych jest znacznie większa niż monitorów CRT. Każdy piksel matrycy LCD jest aktywowany oddzielnie i znajduje się w stanie włączonym albo wyłączonym. Obraz na monitorze LCD nie migocze, nie ma więc potrzeby niwelowania efektu migotania przez zwiększanie częstotliwości odświeżania. Częstotliwość odświeżania monitorów LCD dobiera się tak, aby zapewnić płynne zmiany obrazu przy animacji. Do tego celu w zupełności wystarczy odświeżanie z częstotliwością 60 Hz i z taką częstotliwością pracuje większość monitorów LCD.
- **Czas reakcji** – producenci prezentują wyniki czasu reakcji piksela przy przejściu trzech subpikseli (zielony, czerwony, niebieski) od koloru czarnego do białego i odwrotnie. Suma czasów zapalania i gaszenia piksela składa się na czas końcowy, podawany w milisekundach (ms). Warto jednak zauważyć, że takie przedstawienie sprawy nie w pełni informuje, jak monitor będzie funkcjonował w praktyce. Rzeczywisty czas reakcji będzie taki jak czas przejścia najwolniejszego z subpikseli, których kombinacje tworzą poszczególne kolory.
- **Kąty widzenia** – producenci paneli mierzą wielkość kątów widzenia poprzez utratę jakości obrazu. Moment, w którym następuje zbyt duża utrata jasności i kontrastu obrazu w porównaniu do wyjściowej, staje się kątem granicznym. Do mierzenia jakości kontrastu w monitorach LCD używa się współczynnika CR (ang. *contrast ratio*). W przeszłości kąty widzenia w monitorach LCD były bardzo małe, uniemożliwiając pracę więcej niż jednej osobie na monitorze. Po wprowadzeniu do masowej produkcji paneli w technologiach IPS oraz MVA, problem ten częściowo zniknął.

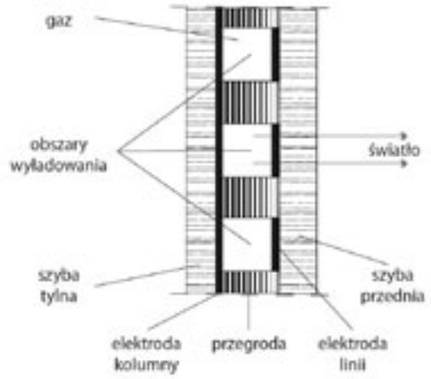
5 EKRANY PLAZMOWE

Ekrany plazmowe należą do grupy przetworników z wyświetlaniem aktywnym, wykorzystujących do wyświetlania zjawisko wyładowania jarzeniowego w plazmie. Istota tego zjawiska polega na emisji światła przez zjonizowany gaz o małym ciśnieniu (rzędu 1 hPa) wskutek przepływu przez gaz prądu elektrycznego. Świecenie gazu jest wywołane zderzeniami jonów, początkowo samoistnie występujących, przyspieszanych w polu elektrycznym występującym pomiędzy dwiema elektrodami wyładowczymi spolaryzowanymi napięciem U. Dla małych napięć U (rys. 23) świecenie jest słabe, prawie niedostrzegalne. W miarę wzrostu napięcia U liczba jonów rośnie, co powoduje, że świecenie jest coraz intensywniejsze. Po przekroczeniu pewnego napięcia progowego Uz, zwanego napięciem zapłonu, cały gaz w obszarze pomiędzy elektrodami wyładowczymi jest zjonizowany – czyli tworzy tzw. plazmę (stąd nazwa przetwornika) – i świeci równomiernym światłem. Barwa wyładowania jarzeniowego zależy od rodzaju zastosowanego gazu.

Na rysunku 24 przedstawiono zasadę konstrukcji najprostszego plazmowego wyświetlacza obrazów wykorzystującego opisane wyżej zjawisko fizyczne. Dwa zestawy elektrod ułożone prostopadle względem siebie są naniesione na wewnętrzne powierzchnie szklanych płyt, tworzących obudowę przetwornika, pomiędzy którymi znajduje się rozrzedzony gaz.



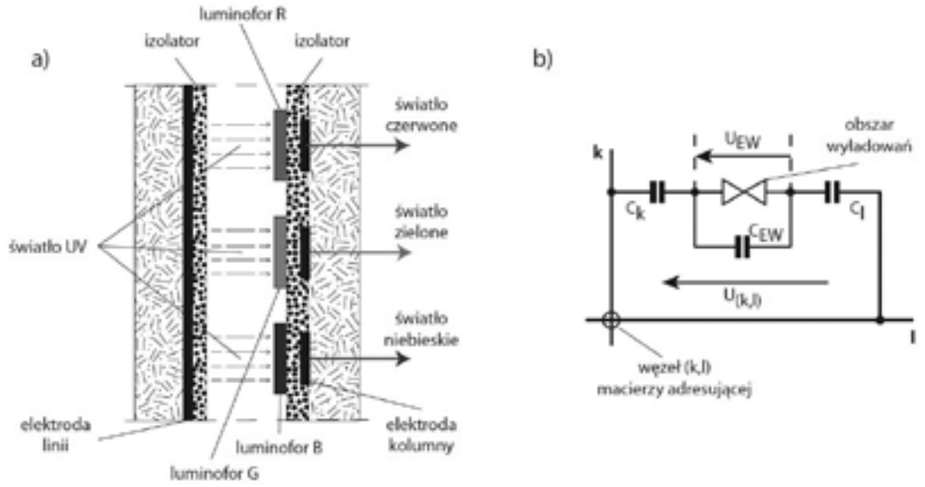
Rysunek 23.
Zjawisko wyładowania jarzeniowego w plazmie



Rysunek 24.
Konstrukcja stałoprądowego wyświetlacza plazmowego

Adresowanie ekranu polega na dołączeniu do adresujących elektrod, poprzez rezystor ograniczający prąd wyładowania, napięcia stałego większego od napięcia zapłonu. Wówczas w obszarze przestrzennego skrzyżowania adresowanej linii i adresowanej kolumny powstają warunki dla „zapłonu" i z węzła zostanie wyemitowane światło. Dla zapobieżenia rozprzestrzeniania się wyładowania do sąsiednich węzłów są one od siebie odseparowane przegrodą. Opisane wyżej rozwiązanie od sposobu sterowania jest nazywane **stałoprądowym ekranem** (wyświetlaczem) plazmowym (DC PDP). Nie jest to rozwiązanie dogodne. Istotnymi jego wadami są: bezpośredni styk elektrod sterujących ze świecącym gazem, co powoduje ich stopniowe zniszczenie, konieczność stosowania rezystorów ograniczających prąd wyładowania, a także poważne trudności z wykonaniem przegrody.

Wad stałoprądowych wyświetlaczy plazmowych nie ma **przemiennoprądowy wyświetlacz** plazmowy. Istota modyfikacji wobec ekranów DC PDP polega na odizolowaniu elektrod adresujących od gazu. Zapobiega to z jednej strony ich niszczeniu przez jony, z drugiej powoduje wtrącenie do obwodu wyładowania dwóch kondensatorów: C_K i C_L, tworzonych przez elektrody, izolator i obszar wyładowania (rys. 25).



Rysunek 25.
Konstrukcja przemiennoprądowego wyświetlacza plazmowego

Jeżeli przyjąć, że w chwili początkowej oba te kondensatory nie są naładowane, to po zaadresowaniu węzła napięciem U(k, l) › UZ (UZ – napięcie zapłonu) cały potencjał węzła odłoży się na obszarze wyładowania U_{EW}, co spowoduje jego zaświecenie i przepływ w obwodzie wyładowania krótkiego impulsu prądowego, wykładniczo malejącego, ładującego kondensatory C_K i C_L do napięcia UZ/2. Po naładowaniu kondensatorów C_K i C_L prąd w obwodzie wyładowania osiągnie wartość zerową, a obszar wyładowania – po wygenerowaniu krótkiego „błysku" (impulsu świetlnego) – przestanie świecić. Czas trwania błysku jest na tyle krótki, że wyładowanie jarzeniowe nie zdąży rozszerzyć się poza obszar adresowanego węzła, co eliminuje konieczność stosowania trudnej do wykonania przegrody izolacyjnej. Ponadto małe pojemności kondensatorów C_K i C_L ograniczają maksymalny ładunek, jaki może przepłynąć w obwodzie wyładowania, a w konsekwencji także maksymalną wartość prądu wyładowania, co czyni zbytecznym rezystor ograniczający ten prąd. Uzyskiwany impuls świetlny jest zbyt krótki i za słaby, ze względu na małe pojemności kondensatorów C_K i C_L. Problem ten można rozwiązać, zmieniając bezpośrednio po wygenerowaniu błysku polaryzację napięcia U(k,l) na przeciwną. Napięcie to doda się do napięć stałych na kondensatorach C_K i C_L, dzięki czemu łączny spadek napięcia na obszarze wyładowania U_{EW} znów przekroczy wartość napięcia zapłonu UZ i element EW ponownie zacznie świecić do czasu przeładowania kondensatorów C_K i C_L, generując kolejny błysk.

Zmieniając periodycznie polaryzację U(k,l) z dostatecznie dużą częstotliwością, przez podanie do węzła napięcia przemiennego o częstotliwości rzędu kilkuset kHz i wartości międzyszczytowej równej 2UZ, uzyskuje się ciągłą generację impulsów świetlnych z elementu wyświetlającego, którą oko – ze względu na częstotliwość powtarzania błysków, rzędu kilkudziesięciu kHz – odbiera jako ciągłe świecenie. Zmieniając czas dołączenia napięcia przemiennego do węzła można generować „pakiety błysków" o zmiennej długości, sterując w ten sposób jasnością świecenia piksela. Od charakterystycznego sposobu pobudzania obszaru wyładowczego do świecenia wyświetlacze stosujące opisaną wyżej zasadę wyświetlania noszą nazwę przemiennoprądowych ekranów plazmowych AC PDP. Do tej grupy rozwiązań zaliczają się wszystkie ekrany plazmowe, dostępne obecnie na rynku.

Rysunek 26.
Konstrukcja pojedynczego piksela wyświetlacza plazmowego

Rysunek 26 ilustruje sposób uzyskiwania obrazów wielobarwnych przez ekrany plazmowe AC PDP. Wnętrze wyświetlacza wypełnione jest gazem lub mieszaniną gazów, świecących podczas wyładowania jarzeniowego światłem nadfioletowym (UV), które pobudza do świecenia paski luminoforów naniesione od wnętrza bańki. Umiejscowienie luminoforów na ścieżce wyładowania powodowałoby ich niszczenie (wypalanie) przez jony świecącego gazu. Zjawisko to eliminuje stosowana w obecnie oferowanych rozwiązaniach wielobarwnych wyświetlaczy plazmowych typu AC PDP konstrukcja piksela ich ekranu. Elektrody wyładowcze kolumn i linii są tu umiejscowione obok siebie na przedniej szybie piksela, a luminofor rozmieszczony na jego przeciwnej ściance. Przepływ prądu wyładowania (jonów mieszaniny gazów) odbywa się pomiędzy elektrodami wyładowczymi w dużej odległości od luminoforu, do którego dociera jedynie promieniowanie UV emitowane przez świecącą plazmę.

LITERATURA

1. *Adobe Premiere Elements. Domowe studio wideo*, praca zbiorowa, Helion, Gliwice 2007
2. Beach A., *Kompresja dźwięku i obrazu wideo*, Helion, Gliwice 2009
3. Bieńkowski M., *Ciekłokrystaliczne obrazy*, „CHIP" czerwiec 2001
4. Danowski B., *Komputerowy montaż wideo. Ćwiczenia praktyczne*, Helion, Gliwice 2006
5. Nasiłowski D., *Jakościowe aspekty kompresji obrazu i dźwięku. Poglądowo o DivX*, Mikom, Warszawa 2004
6. Ozer J., *Tworzenie filmów w Windows XP. Projekty*, Helion, Gliwice 2005
7. Rak R., Skarbek W. (red.), *Wstęp do inżynierii multimediów*, Politechnika Warszawska, Warszawa 2004
8. Zieliński T.P., *Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań*, WKiŁ, Warszawa 2005

Metody kodowania i przechowywania sygnałów dźwiękowych

Andrzej Majkowski
Politechnika Warszawska
amajk@ee.pw.edu.pl

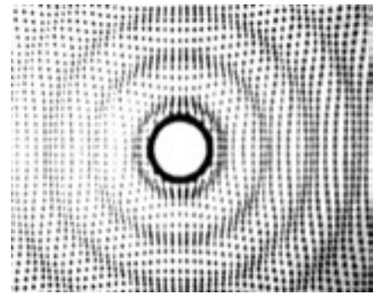
Streszczenie

Wykład jest poświęcony metodom przechowywania i kodowania sygnałów dźwiękowych. Na wstępie przedstawiamy, w jaki sposób człowiek odbiera dźwięki i podajemy podstawowe informacje dotyczące sygnałów dźwiękowych, jak również w jaki sposób oceniamy jakość sygnału dźwiękowego. Następnie przedstawiamy różne formaty zapisu dźwięku. W dalszej części wykładu opisujemy pewne właściwości sygnałów dźwiękowych wykorzystywane podczas kodowania, a szczególnie w kompresji sygnałów dźwiękowych. Poznamy, co to jest psychoakustyka i efekty maskowania. Przedstawiamy również etapy kompresji dźwięku w standardzie MP3. Szczególną uwagę zwracamy na elementy, które znacząco wpływają na jakość kodowania MP3 oraz ułatwiają znalezienie właściwego kompromisu pomiędzy stopniem kompresji a jakością nagrania.

Spis treści

1 DŹWIĘK

Fala dźwiękowa rozchodzi się jako podłużna fala akustyczna w danym ośrodku sprężystym: gazie, płynie (rys. 1). W ciałach stałych, takich jak metale, występuje również fala poprzeczna. Najczęściej mówimy o rozchodzeniu się dźwięku w powietrzu. Dźwięk, jako drgania cząsteczek, charakteryzuje się tym, że cząsteczka pobudzona przekazuje energię cząstce sąsiedniej, a sama drga wokół własnej osi. Skutkiem tego są lokalne zmiany ciśnienia ośrodka rozchodzące się falowo. Co ciekawe, w wodzie dźwięk rozchodzi się znacznie szybciej niż w powietrzu, a w próżni oczywiście nie rozchodzi się w ogóle. W potocznym znaczeniu **dźwięk** to każde rozpoznawalne przez człowieka pojedyncze wrażenie słuchowe.



Rysunek 1.
Fala dźwiękowa [źródło: http://sound.eti.pg.gda.pl/student/elearning/fd.htm]

Zakres częstotliwości od 20 Hz do 20 kHz jest zakresem częstotliwości słyszalnych (fonicznych, audio). Dźwięki o częstotliwości mniejszej od 20 Hz są nazywane **infradźwiękami**, zaś o częstotliwości większej od 20 kHz – **ultradźwiękami.**

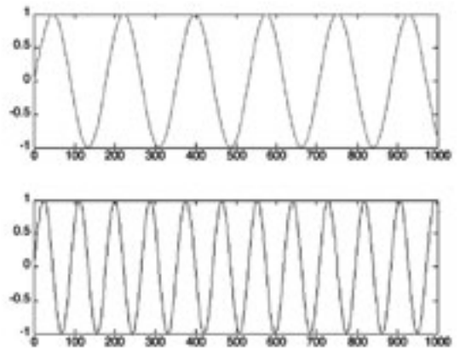
1.1 JAK ODBIERAMY DŹWIĘKI

Elementarnym rodzajem dźwięku, dla którego fala dźwiękowa ma postać sinusoidy (rys. 2) jest **ton**. **Wysokość tonu** to atrybut wrażenia słuchowego, umożliwiający uszeregowanie dźwięków na skali niskie-wysokie. Przez **wysokość dźwięku** rozumie się częstotliwość drgań fali akustycznej – im wyższa częstotliwość drgań tym wyższy dźwięk. Na rysunku 2 częstotliwość drugiego sygnału jest dwa razy większa niż pierwszego, zatem dźwięk o takim przebiegu będzie odbierany jako wyższy. Dźwięki są najczęściej sygnałami złożonymi (występuje w nich wiele składowych sinusoidalnych o różnych amplitudach i częstotliwościach). Wysokość dźwięku, często utożsamiana z częstotliwością, w dużym stopniu zależy od niej, ale nie wyłącznie. Innymi czynnikami wpływającymi na wrażenia wysokości są m.in. natężenie dźwięku czy współobecność innych tonów. Występują też różnice w postrzeganiu wysokości dźwięku między lewym i prawym uchem.

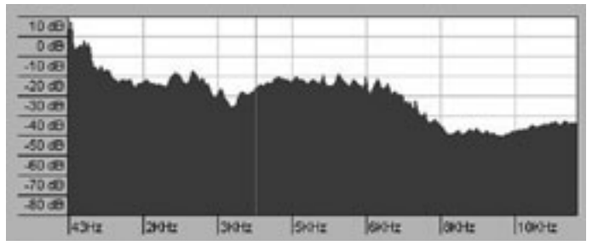
Z pojęciem wysokości dźwięku są związane **interwały muzyczne,** czyli odległości między dźwiękami na skali muzycznej. Określone są stosunkiem częstotliwości sygnałów. Oktawa jest to interwał określający dźwięki, których stosunek częstotliwości jest równy 2:1. Człowiek jest w stanie interpretować poprawnie interwały muzyczne dla tonów o częstotliwości max. ok. 5 kHz. Powyżej 2,5 kHz występują znaczne błędy. Natomiast powyżej częstotliwości 5 kHz występuje brak wrażenia melodii chociaż spostrzegane są różnice częstotliwości.

Bardzo często w analizie sygnału dźwiękowego korzysta się z jego częstotliwościowej reprezentacji. Mówimy wtedy o tzw. **widmie** sygnału dźwiękowego. Widmo sygnału dźwiękowego umożliwia zobrazowanie, jakie składowe sinusoidalne, będące funkcjami czasu, i o jakich częstotliwościach i amplitudach, tworzą dany dźwięk. Rysunek 3 przedstawia przykładowe widmo sygnału dźwiękowego. Oś *Ox* reprezentuje częstotliwość składowych sinusoidalnych, w tym przypadku w zakresie od 43 Hz do 12 000 Hz. Na osi *Oy* można odczytać pośrednio informacje o amplitudach składowych sinusoidalnych.

Barwa dźwięku to cecha wrażenia słuchowego, dzięki której rozróżniamy dźwięki o tej samej głośności i częstotliwości. Barwa dźwięku zależy głównie od jego struktury widmowej, natężenia dźwięku i przebiegu czasowego dźwięku. I tak, interesujące eksperymenty pokazują, że w przypadku niektórych instrumentów ważniejszą rolę odgrywa struktura widmowa (klarnet, trąbka), a innych – czasowa (flet). Kluczową rolę odgrywa też proces narastania i trwania dźwięku.



Rysunek 2.
Dwa sygnały sinusoidalne o tych samych amplitudach, przy czym częstotliwość pierwszego sygnału jest dwa razy mniejsza niż drugiego



Rysunek 3.
Widmo sygnału dźwiękowego

Słuch ludzki charakteryzuje pewna niesymetryczność w odbiorze wysokości dźwięków w uchu lewym i prawym. U zdrowego człowieka różnice nie przekraczają zwykle 3%. Osoby o słuchu muzycznym potrafią określić wysokość dźwięku z dokładnością do 0,3-1%.

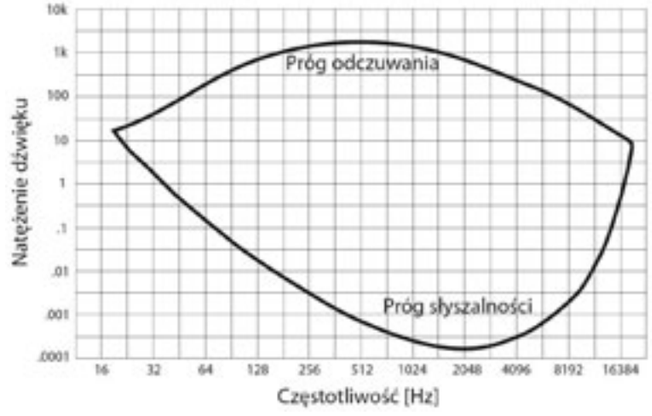
1.2 ZAKRES SŁYSZALNOŚCI

Głośność to taka cecha wrażenia słuchowego, która umożliwia uszeregowanie dźwięków na skali głośno-cicho. Teoretycznie ucho ludzkie potrafi odebrać i przetworzyć drgania o częstotliwości 16 Hz do 20 kHz. Jest to jednak duże uproszczenie niemające wiele wspólnego z rzeczywistością. Okazuje się, że powyższy zakres jest słyszalny tylko wtedy, gdy energia dźwięku jest duża. Przy cichych dźwiękach czułość ucha drastycznie maleje w obszarze częstotliwości poniżej 200 Hz oraz powyżej 8 kHz. W tych zakresach trudniej jest również rozróżniać wysokość dźwięku. Zakres częstotliwościowy percepcji dźwięków maleje też wraz z wiekiem.

Na wrażenie głośności dźwięku wpływa wiele dodatkowych czynników, np. czas trwania dźwięku. Dla krótkich czasów trwania dźwięków występuje efekt czasowego sumowania głośności. Natomiast dla czasów od ok. 1 s do ok. 3 min, dla dźwięków o niskim poziomie lub wysokiej częstotliwości, głośność maleje ze wzrostem czasu trwania. Jest to efektem adaptacji głośności. W wyniku efektu sumowania głośności powiększenie szerokości pasma częstotliwościowego szumu białego powoduje wzrost głośności. Głośność szumu (i dźwięków złożonych) jest wyższa niż tonów (sinusoidalnych) o takim samym natężeniu dźwięku.

Próg słyszalności (próg absolutny, próg detekcji sygnału) jest to najmniejszy poziom ciśnienia akustycznego dźwięku, który wywołuje zaledwie wyczuwalne wrażenie słuchowe wobec braku innych dźwięków. Najniższa wartość ciśnienia akustycznego (przy częstotliwości 1000 Hz) wykrywanego przez ucho ludzkie wynosi średnio 20μPa (rys. 4). **Próg bólu** jest to wartość ciśnienia akustycznego, przy której ucho odczuwa wrażenie bólu. Jest ono prawie niezależne od częstotliwości i wynosi 140 dB dla dźwięków sinusoidalnych oraz 120 dB dla szumów. Wrażenie bólu wywołane jest reakcją mięśni bębenka i kosteczki ucha środkowego na impulsy wysokiego ciśnienia akustycznego. Reakcja ta ma na celu ochronę aparatu słuchowego przed ewentualnymi uszkodzeniami.

Okazuje się, że człowiek nie wszystkie dźwięki o tym samym poziomie głośności słyszy jednakowo dobrze. Dźwięki bardzo niskie i bardzo wysokie są słyszane słabo, za to tony o częstotliwościach od 1 kHz do 5 kHz (mniej więcej zakres mowy ludzkiej) są słyszane wyjątkowo dobrze. Na przykład ton 10 dB mający częstotliwość 1000 Hz będzie przez większość ludzi świetnie słyszalny, ale ton 10 dB o częstotliwości 25 Hz chyba wszyscy odbierzemy jako ciszę. Uświadomienie sobie faktu, że nie wszystkie dźwięki o tej samej energii są przez ludzkie ucho rozpoznawane jako tak samo głośne, to dopiero początek problemów związanych z pojęciem głośności. Następnym problemem jest to, że ucho działa nieliniowo. Oznacza to, że dwa razy większe natężenie dźwięku wcale nie jest przez nas odbierane jako dwa razy głośniejszy dźwięk. Ucho dokonuje silnego spłaszczenia odczuwania głośności – dźwięk, który odczuwamy jako kilka razy głośniejszy od początkowego, ma w rzeczywistości energię dziesiątki, a nawet setki razy większą.



Rysunek 4.
Zakres słyszalności człowieka

1.3 OCENA JAKOŚCI DŹWIĘKU

Układ słuchowy, tak jak wzrokowy, jest instrumentem nieliniowym, a odbierane przez niego dźwięki są interpretowane w różny sposób przez różne osoby. Wpływ na sklasyfikowanie odbieranego dźwięku mają między innymi wspomnienia, wiedza, doświadczenie i uszkodzenia narządu słuchowego. Ocena jakości dźwięku przeprowadzona przez dwie osoby może dać zatem bardzo różne wyniki.

2 FORMATY ZAPISU I PRZECHOWYWANIA PLIKÓW MULTIMEDIALNYCH

Pliki przechowujące materiały multimedialne często muszą umożliwić zapis i przechowywanie różnego rodzaju danych: dźwięków, obrazów, filmów, napisów itp. Potrzebny jest do tego specjalny format zapisu danych, który będzie umożliwiał poprawne wyświetlenie lub synchronizację danych w celu ich jednoczesnego odtworzenia. Taki format zapisu nazywa się **kontenerem multimedialnym**. Istnieją 3 typy kontenerów multimedialnych:

- kontenery audio;
- kontenery audio-wideo;
- kontenery obrazkowe.

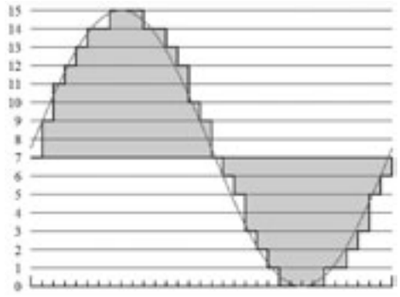
Przykładami kontenerów multimedialnych są:

AVI (ang. *Audio Video Interleave*) jest kontenerem multimedialnym stworzonym przez firmę Microsoft w roku 1992 jako część projektu Video for Windows. W kontenerze tym mogą być zawarte zarówno strumienie audio-wizualne, jak i dane służące do ich synchronizacji.
OGG jest bezpłatnym otwartym kontenerem dla multimediów wysokiej jakości. Wyróżniamy następujące rozszerzenia plików OGG, które są związane o określonym typem danych multimedialnych: .oga – pliki zawierające muzykę, .ogv – pliki zawierające wideo, .ogx – pliki zawierające aplikacje, .ogg – pliki zawierające muzykę w formacie Vorbis.

MPEG-4, wprowadzony pod koniec roku 1998, jest oznaczeniem grupy standardów kodowania audio i wideo wraz z pokrewnymi technologiami, opracowanej przez grupę MPEG (ang. *Moving Picture Experts Group*). Główne zastosowania MPEG-4 to: media strumieniowe w sieci Web (technika dostarczania informacji multimedialnej na życzenie, najpopularniejsze media strumieniowe opierają się na transmisji skompresowanych danych multimedialnych poprzez Internet), dystrybucja CD, DVD, wideokonferencje, telewizja. Oficjalne rozszerzenie pliku to .mp4. MPEG-4 może przechowywać zarówno dane audio-wideo, jak i teksty i obrazki. Może przechowywać dane zachowane praktycznie w każdym formacie.

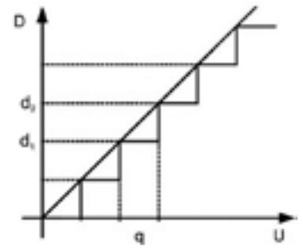
Dźwięk przechowywany w kontenerze multimedialnym musi być zapisany w postaci cyfrowej. Jedną z najpopularniejszych metod zapisu sygnału dźwiękowego jest **PCM** (ang. *Pulse Code Modulation*). Ta metoda jest używana w telekomunikacji, w cyfrowej obróbce sygnału (np. w procesorach dźwięku), do zapisu na płytach CD (CD-Audio) i w wielu zastosowaniach przemysłowych.

Metoda PCM polega na reprezentacji wartości chwilowej sygnału (**próbkowaniu**) w określonych (najczęściej równych) odstępach czasu (rys. 5), czyli z określoną częstością (tzw. **częstotliwością próbkowania**).



Rysunek 5.
Ilustracja zapisu dźwięku w formacie PCM przy 4-bitowym kodowaniu [źródło: http://pl.wikipedia.org/wiki/PCM]

Wartość chwilowa sygnału jest przedstawiana za pomocą słowa kodowego, którego wartości odpowiadają wybranym przedziałom kwantyzacji sygnału wejściowego. Przydział zakresu wartości analogowej jednej wartości cyfrowej jest nazywany **kwantyzacją sygnału,** prowadzi to do pewnej niedokładności (błąd kwantyzacji). Ilustracja kwantyzacji jest przedstawiona na rysunku 6. Z konkretnego przedziału kwantyzacji *q* wartości analogowe z przedziału od d_1 do d_2 zostaną zastąpione jedną wartością zapisaną cyfrowo, najbliższą liczbie d_1. Liczba poziomów kwantyzacji jest zazwyczaj potęgą liczby 2 (ponieważ do zapisu próbek używane są słowa binarne) i wyraża się wzorem 2^n, gdzie *n* to liczba bitów przeznaczona na pojedynczą próbkę. Im większa częstotliwość próbkowania i im więcej bitów słowa kodowego reprezentuje każdą próbkę, tym dokładność reprezentacji jest większa, a tak zapisany sygnał jest wierniejszy oryginałowi. Dobór częstotliwości próbkowania w taki sposób, aby połowa częstotliwości próbkowania (częstotliwość Nyquista) była większa od najwyższej częstotliwości składowej sinusoidalnej występującej w sygnale dźwiękowym (analiza widmowa), umożliwia bezstratną informacyjnie zamianę sygnału ciągłego na dyskretny.



Rysunek 6.
Kwantyzacja sygnału

Dźwięk w formacie PCM może być zapisywany z różną częstotliwością próbkowania, najczęściej jest to 8 kHz (niektóre standardy telefonii), 44,1 kHz (płyty CD-Audio) oraz różną rozdzielczością, najczęściej 8, 16, 20 lub 24 bity na próbkę, może reprezentować 1 kanał (dźwięk monofoniczny), 2 kanały (stereofonia dwukanałowa) lub więcej (stereofonia dookólna). Reprezentacja dźwięku próbkowana z częstotliwością 44,1 kHz i w rozdzielczości 16 bitów na próbkę (65 536 możliwych wartości amplitudy fali dźwiękowej na próbkę) jest uważana za bardzo wierną swemu oryginałowi, ponieważ z matematycznych wyliczeń wynika, iż pokrywa cały zakres pasma częstotliwości słyszalnych przez człowieka oraz prawie cały zakres rozpiętości dynamicznej słyszalnych dźwięków. Taki format kodowania zastosowano na płytach CD-Audio.

Inne formy cyfrowego kodowania dźwięku są zazwyczaj dużo bardziej złożone. Często wykorzystują różne metody kompresji danych w celu zredukowania ich liczby. Istnieją 2 rodzaje kompresji:

- **kompresja bezstratna** – algorytm upakowania informacji do postaci zawierającej mniejszą liczbę bitów w taki sposób, aby informację dało się odtworzyć do postaci identycznej z oryginałem,
- **kompresja stratna** – algorytm zmniejszania liczby bitów potrzebny do wyrażenia danej informacji, przy czym nie ma gwarancji, że odtworzona informacja będzie identyczna z oryginałem. Dla niektórych danych algorytm kompresji stratnej może odtworzyć informację prawie idealnie.

Przetworzenie pliku dźwiękowego do określonego formatu cyfrowego wymaga specjalnego programu, tzw. **kodeka**, w którym zaimplementowane są zaawansowane algorytmy cyfrowego przetwarzania sygnałów dźwiękowych. Poniżej krótko opisano najpopularniejsze kodeki dźwięku. W dalszej części szerzej będzie opisany sposób kodowania MP3.

Ogg Vorbis jest kodekiem ogólnego zastosowania. Najlepiej sprawdza się w tworzeniu plików o dużym stopniu kompresji (od 48 do 128 kbps). Uznaje się, że średnia jakość dźwięku zakodowanego w formacie Ogg Vorbis jest porównywalna do AAC i wyższa niż MP3 o tej samej **przepływności** (czyli szybkości transmisji danych

mierzonej w bitach na jednostkę czasu). W odróżnieniu od MP3, format Ogg Vorbis nie jest opatentowany i pozostaje bezpłatny, zarówno do celów prywatnych, jak i komercyjnych. Dekodowanie plików zapisanych w tym formacie wymaga większego zapotrzebowania na moc obliczeniową procesora niż MP3 (w przenośnych odtwarzaczach szczególnie uwidacznia się to poprzez skrócenie czasu pracy). Jest kodekiem z natury typu VBR (czyli dźwięk jest kodowany ze zmienną w czasie szybkością przepływu danych).

MPEG-4 Part 14 został utworzony w oparciu o format kontenera Apple QuickTime i jest właściwie identyczny z formatem MOV, ale wspiera wszystkie właściwości standardu MPEG. Pliki z zakodowanym dźwiękiem mają często rozszerzenie .mp4, nie istnieje natomiast coś takiego jak format kompresji dźwięku MP4.

AAC (ang. *Advanced Audio Coding*) to z kolei algorytm stratnej kompresji danych dźwiękowych, którego specyfikacja została opublikowana w 1997 roku. Format AAC zaprojektowany został jako następca MP3, oferujący lepszą jakość dźwięku przy podobnym rozmiarze danych.

Kompresja AAC jest modularna i oferuje standardowo cztery profile:

- Low Complexity (LC) – najprostszy, najszerzej stosowany i odtwarzany przez wszystkie odtwarzacze obsługujące format AAC;
- Main Profile (MAIN) – rozszerzenie LC;
- Sample-Rate Scalable (SRS) lub Scalable Sample Rate (AAC-SSR) – zakres częstotliwości dzielony jest na cztery kompresowane niezależnie pasma, jakość jest przez to nieco niższa niż pozostałych profili;
- Long Term Prediction (LTP) – rozszerzenie MAIN wymagające mniejszej liczby obliczeń.

Usprawnienia AAC w stosunku do poprzednich algorytmów kompresji dźwięku:

- próbkowanie 8–96 kHz (MP3 16–48 kHz);
- do 48 kanałów (MP3 – 2 kanały w standardzie MPEG-1 i 5,1 w standardzie MPEG-2);
- skuteczniejszy i wydajniejszy;
- lepsze przenoszenie częstotliwości ponad 16 kHz;
- lepszy tryb kompresji sygnału stereofonicznego joint stereo.

3 PSYCHOAKUSTYKA I PODSTAWY KOMPRESJI SYGNAŁÓW DŹWIĘKOWYCH

Psychoakustyka to współczesna dziedzina wiedzy zajmująca się związkiem obiektywnych (fizycznych) cech dźwięku z jego cechami subiektywnymi, z wrażeniem jakie w mózgu słuchacza wywołują bodźce dźwiękowe. Psychoakustyka próbuje przewidzieć zachowanie się słuchu człowieka w określonych warunkach fizycznych.

Modelami psychoakustycznymi nazywamy modele systemu słyszenia, które uwzględniają ograniczenia i tolerancje mechanizmów percepcji przeciętnego słuchacza, są to modele matematyczne mówiące, jakie dźwięki są rozpoznawalne przez ludzkie ucho, jakie natomiast nie są. Modele psychoakustyczne są podstawą między innymi kompresji dźwięku, algorytmów oceny jakości transmisji mowy, systemów automatycznie rozpoznających mowę oraz systemów rozpoznających mówców.

Wytyczne do modelowania pochodzą z pomiarów psychoakustycznych (odsłuchowych), w których słuchacze oceniają wrażenia wywołane różnymi sygnałami testowymi prezentowanymi w określonym kontekście (np. czy słyszą ton sinusoidalny prezentowany na tle szumu). Model przetwarza sygnał w taki sposób, aby jego wyjście stanowiło predykcję subiektywnych ocen słuchaczy. Najprostszym faktem psychoakustycznym jest różna czułość ludzkiego ucha na dźwięki o różnych częstotliwościach (niektórych częstotliwości np. bardzo wysokich lub bardzo niskich nie słyszymy w ogóle). Modele psychoakustyczne przewidują zwykle zakres

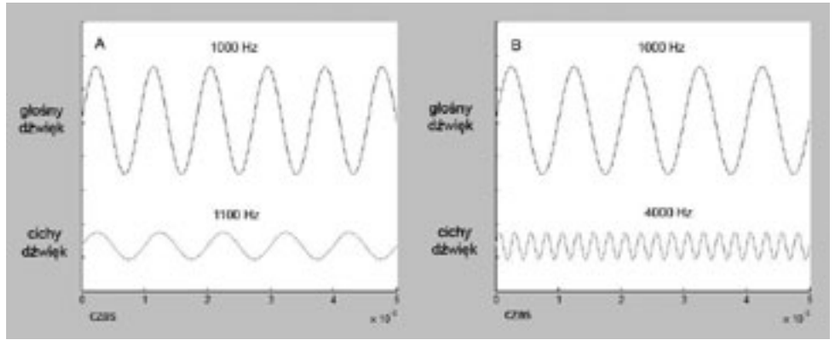
słyszalności od 20 Hz do 20 kHz (dlatego właśnie większość współczesnych odtwarzaczy muzyki zapisanej cyfrowo ma takie pasmo przenoszenia) i maksymalną czułość w zakresie od 2 kHz do 4 kHz.

Innym szeroko stosowanym faktem psychoakustycznym jest **maskowanie dźwięków**. Najogólniej, maskowanie polega na przysłanianiu sygnałów słabszych sąsiadujących z sygnałami znacznie głośniejszymi, które je zagłuszają.

Rozróżniamy 2 rodzaje maskowania:

- maskowanie równoczesne,
- maskowanie czasowe.

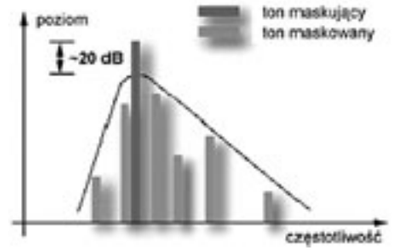
Efekt maskowania równoczesnego opiera się na tym, że człowiek nie jest w stanie odróżnić dwóch dźwięków o zbliżonej częstotliwości, jeśli jeden z nich jest znacznie głośniejszy od drugiego (rys. 7, przypadek A). Możliwe jest to dopiero wtedy, gdy sygnały mają zupełnie różne częstotliwości (przypadek B).



Rysunek 7.
Efekt maskowania równoczesnego

Najprościej mówiąc, maskowanie równoczesne polega na tym, że ciche dźwięki o częstotliwościach zbliżonych do częstotliwości dźwięku głośnego nie są słyszalne. Wszystkie standardy MPEG audio (a więc również MP3) wykorzystują tę właściwość ucha ludzkiego, bazują one na usuwaniu słabszych dźwięków, które nie docierają do mózgu człowieka.

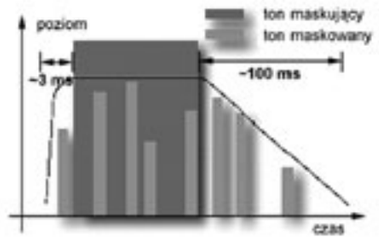
Maskowanie czasowe polega na eliminacji składowych o mniejszym natężeniu, które mają zbliżoną częstotliwość do dźwięku o większym natężeniu i występują razem w pewnym przedziale czasu.



Rysunek 8.
Efekt maskowania czasowego [źródło: 6]

Na rysynku 8 jest pokazany efekt maskowania czasowego, czarną linią zaznaczono próg słyszalności. Można w tym przypadku wyróżnić dwa typy maskowania:

- maskowanie dźwięków następujących (maskowanie pobodźcowe) – głośny dźwięk potrafi zagłuszyć cichsze dźwięki następujące zaraz po nim,
- maskowanie dźwięków poprzedzających (maskowanie wsteczne) – cichy dźwięk poprzedzający w krótkim czasie dźwięk głośny nie jest słyszalny. Ta własność układu słuchowego jest szczególnie ciekawa, gdyż nie da się jej wyjaśnić na gruncie adaptacji krótkoterminowej układu słuchowego. Równocześnie pokazuje ona, że układ słuchowy nosi pewne cechy układu nieprzyczynowego (tzn. skutek wywołany przez jakiś bodziec występuje przed wystąpieniem bodźca).



Rysunek 9.
Całkowity efekt maskowania [źródło: 6]

Na rysunku 9 zilustrowano efekt maskowania równoczesnego i czasowego jednocześnie. Czarna linia oznacza próg słyszalności. Słabe dźwięki (tony maskowane), które są maskowane przez dźwięk silniejszy, mogą zostać podczas kompresji usunięte. Pozostanie tylko dźwięk słyszalny (ton maskujący).

4 IDEA KOMPRESJI MP3

W 1987 roku w niemieckim instytucie Fraunhofera rozpoczęto prace nad radiofonią cyfrową. Jednym z kluczowych elementów było opracowanie systemu kompresji danych umożliwiającego skuteczny zapis sygnałów dźwiękowych. Algorytmy tam opracowane stały się później podstawą systemu MP3.

Należy zaznaczyć, że algorytm stosowany przy kompresji MP3 wykorzystuje kompresję stratną – przy odtwarzaniu, dźwięk nie odpowiada dokładnie dźwiękowi sprzed kompresji. Kompresja powoduje nawet ponad dziesięciokrotne zmniejszenie wielkości miejsca na dysku w stosunku do objętości dźwięku, który kompresji nie podlegał. Osoby z bardziej wrażliwym słuchem odbierają dźwięk skompresowany jako gorszy pod względem jakości.

W roku 1991 ukończone zostały prace w instytucie Fraunhofera nad algorytmem kodowania MPEG-1 – Layer3. Opracowany algorytm stał się najbardziej optymalnym sposobem kodowania sygnałów audio w rodzinie określanej przez międzynarodowe normy ISO-MPEG. Używając tego algorytmu – znanego powszechnie w Internecie jako **MP3**, ze względu na rozszerzenie – do kodowania plików audio, jakość „prawie CD", tj. stereo, 44 kHz, 16 bitów, można uzyskać przy przepływności 112–128 kbps (stopień kompresji 11:1–13:1).

Kompresja MP3 jest oparta na matematycznym modelu psychoakustycznym ludzkiego ucha.

- Idea kompresji MP3 polega na wyeliminowaniu z sygnału tych danych, które są dla człowieka niesłyszalne lub które słyszymy bardzo słabo.
- Kompresja MP3 jest połączeniem metody kompresji stratnej z kompresją bezstratną.
- Etap 1 – koder eliminuje z sygnału składowe słabo słyszalne i niesłyszalne dla człowieka (kompresja stratna).
- Etap 2 – uzyskane dane są poddawane dodatkowej kompresji w celu eliminacji nadmiarowości (kompresja bezstratna).

Algorytm operuje na dźwięku próbkowanym z jakością: 16; 22,5; 24; 32; 44,1 oraz 48 kHz. Jest optymalizowany pod wyjściową przepustowość 128 kbps dla sygnału stereo, aczkolwiek dostępne są przepustowości od 32 kbps do 320 kbps.

Algorytm kodowania MP3 może operować na 4 rodzajach dźwięku wejściowego:

- mono;
- stereo – kompresja dwóch oddzielnych strumieni;
- joint stereo – badane jest podobieństwo sygnałów w obu kanałach; jeśli w obu kanałach jest ten sam sygnał, to koder przełącza się do trybu mono; umożliwia to kodowanie dźwięku z większą dokładnością;
- dual channel – zawiera dwa niezależne kanały, jest stosowany np. przy tworzeniu kilku różnych wersji językowych dla filmu.

W procesie kodowania MP3 występuje kilka procesów, które wymagają dodatkowego wyjaśnienia. Należą do nich dyskretna transformacja kosinusowa, kwantyzacja oraz kodowanie Huffmana.

Dyskretna transformacja kosinusowa (DCT) pomaga rozdzielić sygnał na części, przekształcając dane do postaci umożliwiającej zastosowanie efektywnych metod kompresji. DCT przetwarza sygnał określony w dziedzinie czasu na sygnał określony w dziedzinie częstotliwości. W wyniku działania transformaty na sygnale wejściowym powstają odpowiadające mu współczynniki transformaty. Transformata kosinusowa jest odwracalna, to znaczy, że dysponując tylko współczynnikami transformaty można odtworzyć odpowiadający im sygnał bez żadnych strat. Zaletą transformaty DCT jest to, że większość współczynników jest zwykle bliska zeru, a zatem po procesie kwantyzacji współczynniki te można pominąć, co umożliwia lepszą kompresję danych.

Kwantyzacja jest to proces ograniczenia zbioru wartości sygnału w taki sposób, aby można go było zapisać na skończonej liczbie bitów. Polega na przypisaniu wartości analogowych do najbliższych poziomów reprezentacji, co oznacza nieodwracalną utratę informacji (rys. 6). Kwantyzacja polega na przeskalowaniu współczynników DCT poprzez podzielnie ich przez właściwy współczynnik znajdujący się w tabeli kwantyzacji, a następnie zaokrągleniu wyniku do najbliższej liczby całkowitej. Tablice kwantyzacji dobierane są doświadczalnie.

Kodowanie Huffmana to bezstratna metoda kodowania, przedstawiona przez Davida Huffmana w roku 1952. Kodowanie Huffmana stanowi jedną z najprostszych i jednocześnie łatwych w implementacji metod kompresji bezstratnej. W algorytmie jest wykorzystywany fakt, że pewne wartości danych występują częściej niż inne. Jeżeli zatem zakodujemy częściej występujące wielkości za pomocą krótszych słów kodowych, a rzadziej występujące – za pomocą dłuższych, to sumarycznie długość zakodowanych danych będzie krótsza niż przed kodowaniem.

4.1 KODOWANIE DŹWIĘKU W STANDARDZIE MP3

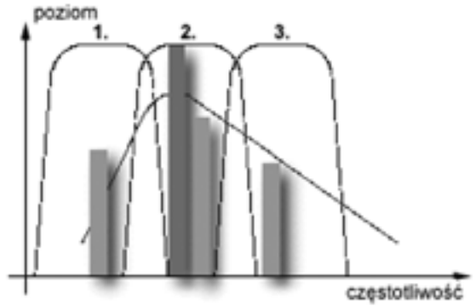
Poniżej podane są najważniejsze etapy kodowania dźwięku w standardzie MP3.

- Sygnał wejściowy jest dzielony na mniejsze fragmenty zwane **ramkami** o czasie trwania ułamka sekundy.
- Dla sygnału dźwiękowego określonego w czasie jest wyliczana jego reprezentacja częstotliwościowa, czyli wyliczane jest widmo sygnału dźwiękowego.
- Widmo sygnału dla każdej ramki jest porównywane z matematycznym modelem psychoakustycznym. W wyniku tego porównania koder określa, które ze składowych dźwięku jako najlepiej słyszalne muszą zostać odwzorowane najwierniej, a które można zakodować w przybliżeniu lub w ogóle pominąć.
- Ustalany jest optymalny przydział bitów na poszczególne częstotliwości pasma akustycznego tak, aby zapewnić możliwie najwierniejsze zakodowanie sygnału.
- Strumień bitów jest poddawany ponownej kompresji poprzez kodowanie Huffmana. Celem tej operacji jest usunięcie nadmiarowości z danych przetworzonych w pierwszym etapie, czyli dodatkowa kompresja bezstratna.

Kolejne ramki poprzedzone nagłówkami są składane w pojedynczy ciąg bitów (strumień bitowy). Nagłówki zawierają metainformacje określające parametry poszczególnych ramek.

Kompresja MP3 rozpoczyna się rozdzieleniem sygnału wejściowego na małe fragmenty (ramki) trwające ułamek sekundy, a następnie ramki są dzielone według pasma na 576 części – najpierw 32 w wielofazowym banku filtrów, a następnie podpasma przekształcane są dyskretną transformatą kosinusową, która generuje 18 współczynników dla każdego podpasma. Zwiększa to szanse na usunięcie niepotrzebnych informacji, sygnał może też być lepiej kontrolowany w celu śledzenia progów maskowania (rys. 11).

Na rysunku 10 zobrazowano ideę działania banku filtrów. Linie pod numerami 1, 2 i 3 oznaczają podział sygnału dźwiękowego na pasma częstotliwościowe 1, 2 i 3. Czwarta linia wyznacza poziom progu słyszalności wyliczony na podstawie modelu psuchoakustycznego. Dwa sygnały oznaczone słupkami po prawej stronie znajdują się poniżej poziomu słyszalności, można więc usunąć sygnał w trzecim podzakresie. Sygnał najbardziej z lewej strony jest słyszalny, można jednak podnieść dopuszczalny poziom szumów, czyli zapisać go mniejszą liczbą bitów. Jeśli kwantowany dźwięk da się utrzymać poniżej progu maskowania, to efekt kompresji powinien być nieodróżnialny od oryginalnego sygnału.



Rysunek 10.
Idea działania banku filtrów [źródło: 6]

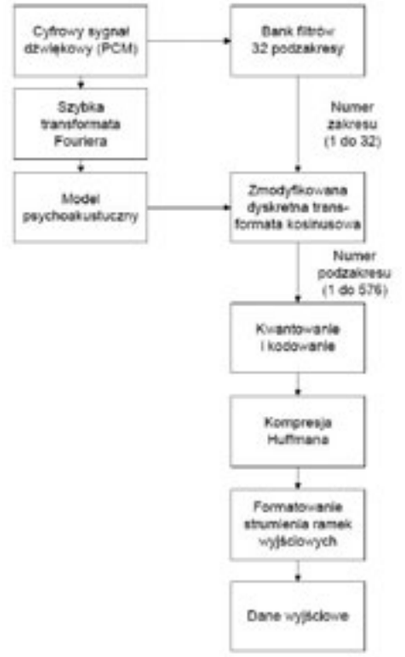
Proces kwantyzacji w kompresji MP3 jest realizowany na zasadzie dwóch pętli, jedna zagnieżdżona w drugiej (rys. 11). Zawiera on także część procesu formowania dźwięku.

Pierwsza z pętli, wewnętrzna, to pętla kontroli współczynnika kompresji. Przeprowadzany jest w niej proces kwantyzacji dla poszczególnych pasm częstotliwościowych, następnie symulowane jest kodowanie skwantowanych współczynników. Jeżeli po kodowaniu okaże się, że jest przekroczony limit przepływności, czyli plik po kompresji byłby zbyt duży, to wskaźnik przyrostu jest dopasowywany do danych i cała pętla jest powtarzana od nowa.

Druga pętla, zewnętrzna, pętla kontroli zniekształceń rozpoczyna się od ustawienia indywidualnych współczynników kwantyzacji na 1, po czym obliczany jest błąd kwantyzacji. Jeśli błąd ten przekracza oszacowany przez model psychoakustyczny próg percepcji, to jest odpowiednio zmieniany współczynnik kwantyzacji i obliczenie błędu odbywa się ponownie. Gdy nie jest możliwe uzyskanie żądanej przepływności i spełnienie wymagań modelu psychoakustycznego, to dźwięk jest kodowany mimo niespełnienia wymagań.

Po procesie kwantyzacji następuje proces kompresji algorytmem Huffmana. W celu dopasowania procesu kompresji do fragmentu danych źródłowych wybierana jest najbardziej pasująca tablica kodów Huffmana z całego zestawu. W celu otrzymania lepszego dopasowania, różne tablice kodów Huffmana są wybierane dla różnych części widma. Jest to proces usuwania nadmiarowych danych bez utraty informacji. Bazuje on na słowie kodowym – kluczu o zmiennej długości, w której klucze krótkie przypisane są do często występujących wzorców, a długie do rzadko występujących. Algorytm rozpoczyna działanie od utworzenia histogramu

(tablicy częstości występowania danych w pliku). W drugim kroku tworzy listę drzew binarnych, które w węzłach przechowują symbol i częstość jego wystąpienia. Następnie w pętli, dopóki jest jeszcze więcej niż jedno drzewo na liście, usuwane są dwa drzewa, które mają w korzeniu zapisane najmniejsze zsumowane częstości, i wstawiane jest nowe drzewo, którego korzeń zawiera sumę częstości usuniętych drzew.



Rysunek 11.
Kodowanie MP3

Końcowym etapem procesu kompresji jest formatowanie ramek wyjściowych i zapis do strumienia wyjściowego. Niektóre pliki MP3 dodatkowo zawierają sumy kontrolne. Suma kontrolna to 16-bitowa liczba, która jest zapisywana w każdej ramce oddzielnie i służy do weryfikacji poprawności strumienia MP3.

4.2 STRUMIEŃ BITOWY

Gęstość strumienia bitowego (ang. *bitrate*) określa współczynnik kompresji sygnału algorytmem MP3. Wyznacza on liczbę bitów przypadającą na sekundę finalnego zapisu. Ustawienie odpowiedniej wartości strumienia bitowego jest kompromisem między jakością a rozmiarem pliku wynikowego (rys. 12).

Rysunek 12.
Ilustracja pojęcia strumienia bitowego [źródło: 8]

Kompresja MP3 może przebiegać:
- ze stałą gęstością strumienia bitowego (ang. *constant bitrate*) – tryb CBR,
- ze zmienną gęstością strumienia bitowego (ang. *variable bitrate*) – tryb VBR.

tryb CBR – każda sekunda dźwięku jest skompresowana za pomocą tej samej liczby bitów, co powoduje jednak, że różne fragmenty utworu mają niejednakową jakość (spokojny fragment wykonany na instrument solo brzmi lepiej niż mocne uderzenie całej orkiestry wspomaganej chórem),
tryb VBR – koduje sygnał uwzględniając jego dynamikę, dzięki czemu przydziela więcej bitów fragmentom sygnału, który zawiera dużo ważnych informacji oraz mniej bitów dla części sygnału, które są mniej złożone. Każda sekunda dźwięku skompresowana jest za pomocą odpowiednio dobranej liczby bitów, dzięki czemu cały utwór ma stałą jakość. W tym przypadku spokojny fragment wykonany na instrument solo (dający się mocniej skompresować) brzmi tak samo dobrze, co mocne uderzenie całej orkiestry wspomaganej chórem (wymagające mniejszego stopnia kompresji). Kompresja w trybie VBR wymaga podania przedziału tolerancji, w jakim może się zmieniać gęstość strumienia bitowego.

Ponieważ zadana gęstość strumienia bitowego obowiązuje dla każdej ramki, w przypadku bardzo złożonych fragmentów może okazać się niewystarczająca i program kodujący nie będzie w stanie zapewnić żądanej jakości zapisu w ramach przydzielonej liczby bitów. Aby zapobiec temu zjawisku standard MP3 zapewnia możliwość skorzystania z dodatkowej rezerwy umożliwiającej zapisanie nadmiarowych danych, tzw. **rezerwy bitowej**. Rezerwa ta powstaje w miejscu pustych fragmentów ramek, w których po zakodowaniu sygnału zostało trochę miejsca.

4.3 ŁĄCZENIE KANAŁÓW ZAPISU STEREOFONICZNEGO

Jak wiemy, sygnał stereo składa się z dwóch odseparowanych od siebie kanałów. Przez znaczną część czasu kanały te jednak przenoszą jeśli nie identyczne to bardzo zbliżone do siebie informacje. Jeśli tak jest, to wtedy koder MP3 wykorzystuje tzw. **algorytm joint stereo**, który powtarzające się dźwięki w obu kanałach zapisuje jako jeden.

Dodatkową możliwością podczas kodowania sygnału z funkcją joint stereo jest **stereofonia różnicowa**. Polega ona na zapisaniu dwóch ścieżek – kanału środkowego będącego sumą sygnałów R i L oraz kanału bocznego, będącego ich różnicą, służącego później do rekonstrukcji sygnału oryginalnego podczas odtwarzania pliku. Warto dodać, że algorytm joint stereo jest bardzo efektywny – powoduje redukcję do 50% liczbę potrzebnych danych.

Ogólnie algorytm MP3 umożliwia skompresowanie dźwięku do postaci:

dual channel – kanały lewy i prawy są traktowane jako dwa niezależne kanały mono, każdy z nich otrzymuje dokładnie połowę dostępnej przepływności; w praktyce jest nieekonomiczny, nie jest więc używany;

stereo – kanały lewy i prawy są traktowane jako stereo, przepływność dzielona jest pomiędzy kanały dynamicznie (np. jeżeli w lewym kanale akurat jest cisza, to prawy dostaje większą część dostępnej przepływności – daje to lepszą jakość dźwięku w prawym kanale) – używany do kompresji w wysokich przepływnościach (192 kbps i więcej);

joint stereo (stereofonia różnicowa) – kanały lewy i prawy są rozbijane na kanały mid/side (*mid* – środek, czyli to, co jest identyczne w obu kanałach i *side* – otoczenie, czyli to, czym różnią się oba kanały) – używany do kompresji w średnich przepływnościach (128–192 kbps);

intensity stereo – kanały lewy i prawy są zamieniane na jeden kanał mono, do którego jest dodawana informacja o uśrednionym kierunku, z którego dźwięk dochodzi (dzięki czemu podczas odsłuchu dźwięk nie dochodzi ze środka tylko z jakiegoś kierunku) – używany do kompresji w niskich przepływnościach (128 kbps i mniej);

mono – kanały lewy i prawy są zamieniane na jeden kanał mono, który jest potem kompresowany, dźwięk odtwarzany jest jako mono – używany do bardzo niskich przepływności (32 kbps i mniej), głównie do kompresji głosu.

Ciekawostką jest to, że specyfikacja formatu MP3, zawarta w dokumencie ISO/IEC 11172-3, nie określa dokładnie sposobu samego kodowania, a jedynie prezentuje ogólny zarys techniki i podaje wymagany poziom zgodności zapisu z normą. Innymi słowy, ustala ona kryteria, jakie musi spełniać struktura pliku, by można było go sklasyfikować jako zgodny ze standardem MP3. Podejście takie ma na celu promowanie różnorodności implementacji programów kodujących i dekodujących dźwięk w standardzie MP3 realizowanych przez różnych producentów. Specyfikacja ISO pełni jedynie rolę bazowego zestawu reguł, określających sposób funkcjonowania standardu tak, aby za pomocą dowolnego kodera można było wygenerować plik odtwarzany przez dowolny dekoder.

4.4 ZALETY I WADY STANDARDU MP3

Niewątpliwie standard kodowania dźwięku MP3 ma wiele zalet. Do najważniejszych należą:
- duży stopień kompresji – stosując kompresję MP3 uzyskujemy plik wynikowy o rozmiarze ok. 10 razy mniejszym od oryginału;
- możliwość sterowania stopniem kompresji i tym samym dostosowania jakości dźwięku do indywidualnych potrzeb;
- metoda ta umożliwia uzyskanie sygnałów o stosunkowo dobrej jakości;
- dekompresja wymaga znacznie mniejszej mocy obliczeniowej niż kompresja;
- twórcy standardu bezpłatnie udostępnili kod źródłowy programów kodujących i dekodujących, dzięki czemu standard ten stał się niezwykle popularny.

Warto jednak pamiętać, że MP3 to metoda kompresji stratnej, a tym samym uniemożliwia zrekonstruowanie sygnału oryginalnego. Ocena jakości dźwięku odtworzonego z pliku MP3 jest bardzo indywidualnym doznaniem. Ponieważ algorytm opiera się na matematycznym modelu percepcji słuchowej przeciętnego człowieka, to siłą rzeczy zawsze będzie grupa ludzi, która usłyszy brakujące, wycięte dźwięki. Oczywiście bardzo duże znaczenie będą miały tu parametry dobrane przez twórcę pliku. Osoba nadzorująca proces kompresji MP3 nie ma co prawda bezpośredniego wpływu na współczynnik kompresji lub też na poziom stratności, może jednak ustalać liczbę bitów przypadających na sekundę docelowego zapisu tzw. przepływność. A to przekłada się bezpośrednio na jakość.

LITERATURA

1. Barański J., *MP3 – internetowy standard zapisu dźwięku,* „Magazyn Elektroniki Użytecznej” maj 2000
2. Beach A., *Kompresja dźwięku i obrazu wideo*, Helion, Gliwice 2009
3. Butryn W., *Dźwięk cyfrowy*, WKiŁ, Warszawa 2002
4. Butryn W., *Dźwięk cyfrowy. Systemy wielokanałowe*, WKiŁ, Warszawa 2004
5. Czyżewski A., *Dźwięk cyfrowy. Wybrane zagadnienia teoretyczne, technologia, zastosowania*, Exit, Warszawa 2001
6. Kołodziej P., *Komputerowe studio muzyczne i nie tylko. Przewodnik*, Helion, Gliwice 2007
7. Nasiłowski D., *Jakościowe aspekty kompresji obrazu i dźwięku. Poglądowo o DivX*, Mikom, Warszawa 2004
8. Rak R., Skarbek W. (red.), *Wstęp do inżynierii multimediów*, Politechnika Warszawska, Warszawa 2004

Bazy danych

Dokumenty XML w relacyjnych bazach danych – czyli wojna światów

Optymalizacja zapytań SQL

Tworzenie interfejsów do baz danych z wykorzystaniem technologii ADO.Net

Hurtownie danych – czyli jak zapewnić dostęp do wiedzy tkwiącej w danych

Dokumenty XML w relacyjnych bazach danych – czyli wojna światów

Andrzej Ptasznik
Warszawska Wyższa Szkoła Informatyki
aptaszni@wwsi.edu.pl

Streszczenie

Przedmiotem wykładu jest wykorzystanie dokumentów XML w relacyjnych bazach danych. W pierwszej części wykładu opowiedziana zostanie krótka historia standardu XML i podstawowe zasady tworzenia dokumentów XML. Następnie omówione zostaną sposoby przekształcania danych relacyjnych do postaci XML oraz zasady zapytań pobierających dane z dokumentu XML. Prezentowane będą też przykłady wykorzystania typu danych XML na etapie projektowania baz danych, a także przykłady zastosowania XML w rozwiązywaniu konkretnych problemów. Przedstawione zostaną również elementy schematów XSD i ich znaczenie w procesie zapewnienia poprawności danych zapisanych w dokumentach XML.

Spis treści

1 WPROWADZENIE

Relacyjne bazy danych towarzyszą twórcom aplikacji już od wielu lat i ciężko znaleźć projektanta lub programistę, który nie zetknął się z tą problematyką. Podobna sytuacja występuje w obszarze związanym z językiem XML. Jego intensywny rozwój, a właściwie dynamiczne rozszerzanie zastosowań oraz rozbudowywanie technologii „z otoczki" XML możemy obserwować od ponad dziesięciu lat. W takiej sytuacji jest dość oczywiste, że język ten pojawił się w otoczeniu relacyjnych baz danych i rozpoczęła się ich „współpraca". Przejawami tej współpracy i wzajemnego przenikania się obu technologii zajmiemy się na niniejszym wykładzie.

W ramach wykładu zostanie zaprezentowana geneza oraz podstawy języka XML, co ułatwi lepsze poruszanie się po omawianej problematyce. Zasadniczą częścią wykładu będzie jednak związek języka XML z systemem SQL Server 2008 i wsparcie oferowane przez to narzędzie w zakresie obsługi danych w formacie XML. Zaprezentowane zostaną polecenia służące do zwracania rezultatów zapytań w postaci dokumentów XML, przedstawione będą możliwości typu danych XML, który służy do przechowywania dokumentów XML w bazie w postaci natywnej, a także zostaną opisane przykłady stosowania XML Schema do definiowania dopuszczalnej struktury dokumentów XML.

2 JĘZYK XML

Wraz z intensywnym rozwojem Internetu pojawił się (a właściwie nabrał większego znaczenia) problem przekazywania danych pomiędzy różnymi systemami. Problem ten wystąpił w momencie pierwszych prób łączenia komputerów w sieć. Od samego początku istniały dwa podejścia: korzystanie z binarnego zapisu danych oraz z postaci tekstowej. W okresie, gdy komputery miały bardzo ograniczone zasoby (pamięć operacyjną, pojemność dysków, transfer w sieci) popularniejsze były rozwiązania korzystające z postaci binarnej. Istniejące protokoły binarne powodowały jednak konieczność „tłumaczenia" danych pomiędzy heterogenicznymi systemami (np. w jednym systemie liczby całkowite są cztero-, a w drugim – dwubajtowe), co powodowało wzrost poziomu komplikacji systemów rozproszonych. Równocześnie obserwowano burzliwy rozwój usługi WWW, dla której pokonanie bariery pomiędzy różnymi architekturami nie stanowiło problemu, gdyż stosowanym rozwiązaniem było przekazywanie danych w postaci tekstowej, czego przejawem stał się język HTML. Jego sukces i rosnąca popularność ujawniły jednak wiele niedoskonałości. Wystarczy wspomnieć o tzw. wojnie przeglądarek, polegającej z grubsza na różnym sposobie interpretowania (czyli renderowania) dokumentu HTML przez różne przeglądarki oraz na dodawaniu przez producentów przeglądarek nowych elementów (spoza specyfikacji HTML), które były interpretowane przez ich produkt. W efekcie idea HTML – jako języka służącego do przekazywania treści wraz z informacjami nadającymi znaczenie poszczególnym fragmentom tekstu – została wypaczona, a sam język sprowadzono do poziomu języka opisu układu (ang. *layout*) dokumentu.

Przez kilka lat taka sytuacja była obowiązującym standardem, a twórcy stron WWW używali w dokumentach HTML wielu karkołomnych sztuczek dla osiągania konkretnych efektów wizualnych. Koniecznością było w takiej sytuacji tworzenie kilku wersji strony WWW, które działały poprawnie w konkretnych przeglądarkach. Jednocześnie rosły możliwości komputerów (graficzne, obliczeniowe) oraz wymagania stawiane stronom WWW. Prowadziło to do pogłębiania się chaosu w świecie technologii internetowych.

I tak w 1996 roku powstał pomysł stworzenia nowego języka dla potrzeb Internetu. Miał to być prosty język, z prostymi regułami dotyczącymi jego składni, czytelny zarówno dla człowieka, jak i dla maszyn, oraz możliwie uniwersalny jako nośnik danych i informacji o ich strukturze. W efekcie tych prac powstał **język XML** (ang. *Extensible Markup Language*). Wizualnie zbliżony do HTML (stosowanie znaczników i elementów atrybutów), choć od samego początku rygorystycznie podchodzący do poprawnego formułowania dokumentu. Mimo że XML jest tylko podzbiorem istniejącego od dawna **języka SGML** (ang. *Standarized General Markup Language*) o potężnych możliwościach, szybko okazało się, że z racji swojej prostoty, XML wyparł SGML z większości zastosowań.

Wraz ze wzrostem popularności języka XML pojawiły się narzędzia służące do jego „obróbki". Dziś trudno znaleźć jakikolwiek język programowania, w którym nie byłoby bibliotek obsługujących XML (w zakresie odczytu, tworzenia i weryfikacji poprawności). Właśnie łatwość sprawdzenia poprawności syntaktycznej i semantycznej dokumentów XML stanowi jeden z głównych argumentów przemawiających za jego stosowaniem. Po co tworzyć dla każdego systemu osobne formaty plików z danymi oraz mechanizmy sprawdzające ich poprawność, skoro można użyć w każdym przypadku jednego i tego samego narzędzia?

Reguły określone dla struktury dokumentu XML są proste i przejrzyste, i nie zawierają wyjątków. Aby dokument XML mógł być uznany za **poprawnie sformułowany** (tzn. zgodny z wszystkimi wymaganiami co do syntaktyki) musi spełniać następujące reguły:

- składać się z elementów, przy czym jeden z elementów jest elementem głównym i zawiera w sobie pozostałe;
- element składa się (podobnie jak w HTML) ze znaczników: otwierającego i zamykającego. Język HTML dopuszczał brak znacznika zamykającego w miejscach, w których można się było domyślić, że element powinien być zamknięty (np. przy definiowaniu komórek w tabeli znacznik ‹td› nie musiał być zamykany, bo wiadomo, że gdy po nim pojawiał się kolejny ‹td› to oznaczał początek kolejnej komórki tabeli). W dokumencie XML każdy element musi być zamknięty. Jeżeli element nie ma zawartości to można zastosować formę skróconą (zamiast znacznika zamykającego umieszcza się po nazwie znacznika otwierającego znak /. Na przykład: `<br />`);
- elementy muszą być poprawnie zagnieżdżone – zakres objęty elementem musi być jednoznacznie określony. Innymi słowy: jeden element może zawierać w sobie inny, ale tylko w całości (wraz ze znacznikiem zamykającym);
- każdy element może zawierać atrybuty. Są one definiowane wewnątrz znacznika otwierającego po nazwie elementu. Każdy atrybut musi mieć wartość – w przeciwieństwie do HTML, gdzie takiej konieczności nie było. Wartość atrybutu musi być ujęta w apostrofy lub cudzysłów;
- w dokumentach XML jest rozróżniana wielkość liter i jest ona istotna. Elementy `<dane />`, `<Dane />` i `<DANE />` to trzy różne elementy;
- podobnie jak w dokumencie HTML, jeżeli w treści dokumentu chcemy umieścić znak, który mógłby powodować niejasności w interpretacji dokumentu – zastępujemy go odpowiednią **encją**. Na przykład, w przypadku:

```
<wyrazenie>Ala<Ela</wyrazenie>
```

Znak ‹ pomiędzy słowami Ela i Ala mógłby zostać błędnie zinterpretowany jako początek elementu o nazwie Ala. Po zastąpieniu go encją otrzymujemy:

```
<wyrazenie>Ala&lt;Ela</wyrazenie>
```

Co jest już poprawnie interpretowane.

Poprawność dokumentów XML można rozpatrywać na dwóch poziomach: syntaktycznym i semantycznym. Pierwszy z nich jest określony przez opisane wyżej reguły dotyczące tworzenia dokumentów XML. Jeżeli dokument pod względem składniowym spełnia te reguły, to jest **dokumentem poprawnie sformułowanym** (ang. *well formed*).

Jeżeli ten poziom weryfikacji nie wystarcza, można sięgnąć po dodatkowe narzędzia, aby nałożyć ograniczenia semantyczne (np. w każdym elemencie „osoba" musi wystąpić co najmniej jeden element „imie", wartością atrybutu „pesel" musi być ciąg 11 cyfr). Najczęściej stosowane są rozwiązania: **DTD** (ang. *Document Type Definition*) i XML Schema.

DTD jest rozwiązaniem starszym i o ograniczonych możliwościach. Obecnie znacznie częściej stosuje się **XML Schema**, co nie znaczy że zapewnia ono możliwość zdefiniowania dowolnych reguł dotyczących dopuszczalnej zawartości dokumentu XML. Jako alternatywa dla XML Schema wymieniane jest też Relax NG. Jest to rozwiązanie porównywalne z XML Schema, o nieco innej specyfice i możliwościach. Zależnie od reguł, które chcemy wymusić, w dokumencie można stosować konkretne rozwiązania. Najistotniejsze jest jednak to, że niezależnie od platformy sprzętowej i programowej, raz określone reguły mogą być weryfikowane w taki sam sposób za pomocą narzędzi dostępnych dla konkretnej platformy.

Jeżeli dokument jest poprawnie sformułowany oraz spełnia reguły opisane w DTD lub XML Schema, to jest **dokumentem poprawnym** (ang. *valid*). Istnieje wiele narzędzi służących do sprawdzania poprawności dokumentów XML, są to tzw. **parsery walidujące**. Połączenie XML z XML Schema, przy wsparciu ze strony narzędzi, umożliwia bardzo łatwe rozwiązywanie problemów z projektowaniem formatów przekazywania czy przechowywania danych. Wystarczy opracować dla konkretnego problemu dokument XML Schema i przekazać zainteresowanym stronom. Twórcy dokumentów XML będą mieli wszystkie informacje, niezbędne do utworzenia dokumentów w postaci zgodnej z założeniami. Z kolei, twórcy oprogramowania, które ma takie dane interpretować, będą wiedzieli czego mogą spodziewać się w dokumencie oraz będą mieli możliwość łatwego sprawdzenia poprawności dokumentu przed rozpoczęciem jego przetwarzania. Taki schemat sprzyja rozkwitowi technologii związanych ze stosowaniem XML. Można tu wymienić dla przykładu **język SVG** (ang. *Scalable Vector Graphics)* czy **MathML**. Oba definiują postać dokumentu XML, który następnie jest przetwarzany na grafikę wektorową lub równanie matematyczne.

Podsumowując kwestię weryfikacji dokumentów XML: dla dowolnego problemu, dla którego specyfikuje się format dokumentu XML, można za pomocą standardowych narzędzi zdefiniować dodatkowe reguły semantyczne i weryfikować automatycznie ich spełnienie. Dzięki temu można ujednolicić wstęp do procesu przetwarzania danych z dokumentu XML do postaci sprawdzenia poprawności sformułowania i poprawności dokumentu. Dalsze przetwarzanie odbywa się już przy założeniu poprawności dokumentu, co upraszcza ten proces ze względu na gwarancję co to struktury i zawartości przetwarzanego dokumentu.

3 JĘZYK XML ALTERNATYWĄ DLA RELACYJNYCH BAZ DANYCH

Skoro język XML umożliwia łatwe budowanie dokumentów o hierarchicznej strukturze, to czy może stanowić alternatywę dla relacyjnych baz danych? Można przecież zastąpić relacje odpowiednim zagnieżdżaniem elementów. Teoretycznie jest to jak najbardziej możliwe, ale praktycznie raczej nie.

W dokumencie XML, co prawda, można zastąpić relacje odpowiednim zagnieżdżaniem elementów oraz stosowaniem atrybutów, ale to podejście sprawdza się wyłącznie przy małej liczbie danych. Przy niewielkim rozmiarze dokumentu może on śmiało rywalizować z bazą danych, lecz wraz ze wzrostem rozmiaru dokumentu, XML szybko zostaje w tyle i osiąga „masę krytyczną", co powoduje ogromny spadek wydajności przy jego przetwarzaniu. Podobnie wygląda definiowanie ograniczeń stosowanych do zawartości i struktury dokumentu XML. Narzędzie XML Schema, mimo rozbudowanych możliwości, okazuje się jednak niewystarczające i powstaje konieczność dobudowywania własnych mechanizmów służących zapewnieniu spójności danych. To wszystko powoduje, że stosowanie języka XML w charakterze bazy danych staje się coraz bardziej złożone, co szybko przerasta stopień złożoności prawdziwej bazy relacyjnej zastosowanej do rozwiązania tego samego problemu, a co za tym idzie stosowanie XML w tym przypadku staje się ekonomicznie nieuzasadnione.

Obszarem, w którym XML sprawdza się jednak bardzo dobrze w roli bazy danych jest pełnienie funkcji magazynu danych *off-line*. Polega to na tworzeniu aplikacji, które łącząc się z bazą danych, pobierają dane niezbędne do pracy aplikacji i przechowują je na komputerze użytkownika w postaci XML. Dalsza praca odbywa się na danych z XML już bez połączenia z bazą danych i dopiero w momencie synchronizacji dane są uaktualniane w bazie. Przykładem tego typu rozwiązania jest klasa DataSet z .NET Framework. Ma ona bardzo rozbudowane możliwości przechowywania danych i śledzenia ich modyfikacji.

Zasadniczo jednak język XML należy raczej rozpatrywać jako pewnego rodzaju uzupełnienie funkcjonalności baz danych, które może być stosowane w przypadkach wymagających tworzenia rozbudowanej struktury tabel, aby zamodelować zbiór cech informacyjnych o zróżnicowanej strukturze. Tu XML sprawdza się znakomicie, co potwierdza coraz większe wsparcie dla korzystania z XML w relacyjnych bazach danych oferowane przez poszczególnych producentów. W ramach niniejszego wykładu prezentowane będą takie właśnie mechanizmy w odniesieniu do systemu SQL Server 2008.

4 JĘZYK XML W MICROSOFT SQL SERVER 2008

Klauzula FOR XML polecenia SELECT

Pierwszym zagadnieniem dotyczącym zastosowania języka XML w bazach danych było zwracanie wyników zapytań w formie dokumentów XML. Jest to wygodny mechanizm, szczególnie gdy aplikacja korzysta z danych o bardziej złożonej strukturze. Zbudowanie dokumentu XML umożliwia stworzenie struktury adekwatnej do wymagań i pozwala uniknąć np. wielokrotnego komunikowania się z bazą danych w celu pobierania różnych fragmentów potrzebnych danych. Baza w odpowiedzi na jedno zapytanie zwraca odpowiedni dokument XML, a aplikacja zajmuje się jego przetworzeniem, co zwykle odbywa się z wykorzystaniem gotowych narzędzi i rozwiązań (np. jako transformacja dokumentu XML do postaci strony HTML lub do dokumentu PDF czy XLS).

SQL Server począwszy od wersji 2000 dopuszcza zwracanie rezultatu wykonania polecenia SELECT w postaci fragmentu lub kompletnego dokumentu XML. Żeby skorzystać z tej możliwości należy dobrać odpowiedni wariant klauzuli FOR XML. Występuje ona w czterech wariantach:

- RAW
- AUTO
- EXPLICIT
- PATH

W prostszych przypadkach mogą to być tryby RAW albo AUTO. Służą one do tworzenia prostej reprezentacji zbioru wynikowego jako fragmentu dokumentu XML z opcją ujęcia go w zdefiniowany element główny (ROOT). W przypadku istnienia wymagań generowania bardziej złożonej struktury dokumentu XML, mamy do dyspozycji klauzulę FOR XML w wariantach EXPLICIT oraz PATH. Wariant PATH, podobnie jak RAW czy AUTO, można zastosować do większości zapytań, gdyż bardzo łatwo dostosować zapytanie do postaci wymaganej przez ten wariant klauzuli FOR XML. Inaczej wygląda sytuacja w przypadku wariantu EXPLICIT, który ma ściśle określone wymagania co do postaci zbioru wynikowego, który ma być przekształcony na postać XML, tzw. **tablica uniwersalna**. Taka tablica jest zwykle generowana za pomocą wielu zapytań łączonych klauzulą UNION. Poszczególne zapytania zwracają wartości różnych kolumn zbiorczego wyniku zapytania.

Najprostszym do zastosowania jest tryb RAW. Jego działanie sprowadza się do wygenerowania dla każdego wiersza ze zbioru wynikowego jednego elementu XML o domyślnej nazwie ROW. Wartości poszczególnych kolumn dla wiersza stają się wartościami atrybutów elementu ROW. Można także określić własną nazwę dla elementu row oraz zamiast atrybutów wybrać generowanie wartości z kolumn jako elementów XML o takiej nazwie jak nazwa kolumny. W ramach trybu RAW można stosować dodatkowe opcje:

- Opcja ROOT powoduje dodanie do rezultatu zapytania elementu głównego o domyślnej nazwie ROOT bądź dowolnej innej określonej w ramach opcji.
- Korzystanie z opcji ROOT jest powszechną praktyką, gdyż tylko wtedy zwrócony dokument XML spełnia reguły składni i jest poprawnie sformułowany, co umożliwia jego dalsze przetwarzanie.
- Opcja ELEMENTS powoduje, że zamiast atrybutów, do umieszczenia zawartości kolumn są wykorzystane elementy o nazwach takich, jak poszczególne kolumny.

W tym momencie nasuwa się pytanie: Czy stosować elementy czy atrybuty? Jest ono jednym z częściej zadawanych pytań dotyczących planowania struktury dokumentów. Niestety nie ma na nie jednoznacznej odpowiedzi, warto natomiast zdawać sobie sprawę, że korzystanie z atrybutów:

- zmniejsza rozmiar wynikowego dokumentu;
- uniemożliwia nadawanie struktury zawartości atrybutu;
- ogranicza liczebność wystąpień atrybutów o tej samej nazwie w ramach elementu do jednego.

Korzystanie z elementów natomiast:

- zwiększa rozmiar wynikowego dokumentu;
- umożliwia w razie potrzeby nadawanie struktury wartościom elementu.

Dla przykładu weźmy dwa elementy:

```
<dane osoba='Jan Nowak'/>
```

i

```
<dane><osoba>Jan Nowak</osoba></dane>
```

Widać, że pierwszy wariant jest bardziej zwięzły. Nie można natomiast wykonać w nim dodania drugiej „osoby" do elementu `dane`, chyba że w karkołomny sposób:

```
<dane osoba='Jan Nowak'  osoba2='Tomasz Kowalski' />
```

W drugim wariancie nie ma takiego problemu:

```
<dane><osoba>Jan Nowak</osoba><osoba>Tomasz Kowalski
</osoba></</dane>
```

Podobnie, gdy chcemy nadać zawartości atrybutu czy elementu `osoba` jakąś strukturę, to pierwszy wariant skutecznie to uniemożliwia.

Drugi wariant umożliwia natomiast swobodne wykonanie:

```
<dane>
   <osoba>
     <imie>Jan</imie>
     <nazwisko>Nowak</nazwisko>
  </osoba>
</dane>
```

Kolejnym trybem dostępnym w ramach klauzuli FOR XML jest tryb AUTO. Ma on możliwości zbliżone do RAW z tą różnicą, że potrafi budować proste hierarchie w dokumencie XML. Proces budowania hierarchii jest oparty na **heurystykach**. Analizowane są kolejne wiersze i wartości kolumn. Umożliwia to przy odpowiednim skonstruowaniu zapytania (sortowanie, kolejność złączeń) na sterowanie postacią wyjściowej hierarchii dokumentu XML. Umożliwia także (podobnie jak RAW) osadzenie w wyjściowym dokumencie XML wygenerowanego dla niego dokumentu XML Schema, opisującego postać wyjściowego dokumentu XML.

W przypadku pojawienia się w wyniku zapytania pól binarnych, są one domyślnie kodowane metodą URL Encode. Jeżeli nie jest to odpowiednie rozwiązanie, można skorzystać z opcji BINARY BASE64.

Klauzula FOR XML w wariancie EXPLICIT ma największe możliwości, ale jest też najbardziej złożona i trudna w stosowaniu. Nie nadaje się w przeciwieństwie do RAW i AUTO do zastosowania w dowolnym zapytaniu. Żeby skorzystać z trybu EXPLICIT, tabela wejściowa musi być tzw. **tabelą uniwersalną**. Składa się ona z kolumn o ustalonej kolejności, nazwach i znaczeniu:

Tabela 1.
Przykład tabeli uniwersalnej

Tag	Parent	Customer!1!cid	Customer!1!name	Order!2!id	Order!2!date	OrderDetail!3!id!id	OrderDetail!3!pid!idref
1	NULL	C1	"Janine"	NULL	NULL	NULL	NULL
2	1	C1	NULL	01	1/20/1996	NULL	NULL
3	2	C1	NULL	01	NULL	OD1	P1
3	2	C1	NULL	01	NULL	OD2	P2
2	1	C1	NULL	02	3/29/1997	NULL	NULL

Przykładowo (patrz tab. 1) pierwszą kolumną musi być kolumna `Tag`, która zawiera unikatowy numer dla każdego elementu, który będzie generowany. Druga kolumna ma nazwę `Parent` i określa wartość Tag dla rodzica elementu. Kolejne kolumny definiują składowe elementów XML. Nazwy tych kolumn są tworzone w opar-

ciu o schemat *nazwaElementu!Tag!NazwaAtrybutu!Dyrektywa*. Umożliwia to dokładne sterowanie procesem definiowania struktury elementów i atrybutami. Dokładne omówienie mechanizmu działania trybu EXPLICIT wykracza poza zakres niniejszego opracowania. Klauzula FOR XML EXPLICIT pomimo dużych możliwości definiowania struktury wynikowego dokumentu XML jest obarczona wadami związanymi z trudnościami w przygotowaniu odpowiedniego zapytania wejściowego. Szczególnie nieprzyjemne jest modyfikowanie już istniejącej struktury. Jest to proces żmudny i podatny na błędy, często skutkujący pisaniem zapytania od nowa.

Wraz z SQL Server 2005 pojawił się tryb PATH. Jest on rozsądnym kompromisem pomiędzy możliwościami w zakresie definiowania struktury XML, a łatwością zapisu tych reguł. W uproszczeniu, tryb PATH bazuje na nazwach nadawanych kolumnom zapytania. Mają one postać zbliżoną do wyrażeń języka XPath (znanego każdemu, kto uważa się za eksperta od XML) wskazujących elementy bądź atrybuty. Na ich podstawie budowana jest wyjściowa struktura XML. Można ją dodatkowo komplikować poprzez zagnieżdżanie zapytań. Pamiętać należy jedynie o odpowiedniej kolejności występowania kolumn w zapytaniu – te definiujące atrybuty elementu muszą wystąpić przed definiującymi kolejne lub zagnieżdżone elementy.

Jeżeli kolumna w wyniku zapytania ma być elementem, to domyślną jego nazwą będzie nazwa kolumny. Jeżeli ma być atrybutem – należy nadać kolumnie alias zaczynający się od znaku @ (małpki). Alias może zawierać także znaki ukośnika, które określają kolejne poziomy zagnieżdżenia. Z kolei znak * (gwiazdka) powoduje, że w przypadku kolumny typu XML, jej zawartość będzie osadzona w wyniku zapytania wprost. Dla kolumn innych typów wstawiony będzie węzeł tekstowy z zawartością.

Wszystkie opisane warianty klauzuli FOR XML mają jeszcze kilka opcji, z którymi warto się zapoznać. Jako przykład można podać opcję TYPE powodującą, że zwrócona wartość jest traktowana jak zmienna typu XML, co umożliwia wygodne dalsze jej przetwarzanie. Równie ciekawa jest opcja XSNIL, która umożliwia umieszczenie w wynikowym dokumencie XML elementów, które mają w wejściowym zapytaniu wartość null i domyślnie nie byłyby umieszczone w wyniku. Jest to szczególnie istotne przy przetwarzaniu zwróconych danych w aplikacji, która niekoniecznie ma skąd wziąć pełną listę dopuszczalnych elementów, które można w ramach przetwarzania uzupełnić. Oczywiście można próbować ten problem rozwiązać za pomocą osadzania XML Schema, ale jest to bardziej złożone i pracochłonne.

Opisane pokrótce możliwości SQL Server 2008 w zakresie zwracania wyników zapytań w postaci XML nie wyczerpuje wszystkich możliwości istniejących w tym narzędziu, sygnalizują jedynie podstawowe mechanizmy i ich zastosowanie.

Typ danych XML

Kolejnym obszarem zastosowania XML w SQL Server 2008 jest typ danych XML. Służy on do przechowywania dokumentów lub fragmentów dokumentów XML bezpośrednio w bazie danych oraz do wygodnego manipulowania nimi i walidowania z zastosowaniem XML Schema. Dane XML w bazie mogą występować w dwóch wariantach:

- skojarzone z kolekcją dokumentów XML Schema (ang. *typed XML*),
- nieskojarzone z XML Schema (ang. *untyped XML*).

Skojarzenie kolumny typu XML ze schemą powoduje nadanie ograniczeń strukturze dokumentów XML, które mogą być umieszczone w tej kolumnie. Ograniczenia te są weryfikowane automatycznie przy każdej operacji dodania czy modyfikacji zawartości kolumny XML.

Zanim skorzysta się z typu danych XML warto zapoznać się z jego dokumentacją. Szczególnie chodzi tu o sposób przechowywania dokumentu oraz o ograniczenia związane z samym typem danych. Warto również zastanowić się nad korzystaniem z kolekcji XML Schema oraz indeksów XML. Istotną informacją jest to, że dokument nie jest zapisywany w bazie wprost, tylko przechodzi proces normalizacji (modyfikacja dokumentu, kodowanie w Unicode, eliminowanie niepotrzebnych ciągów i znaków itp.). Eliminuje to możliwość korzystania z tego typu danych wszędzie tam, gdzie ważna jest oryginalna postać dokumentu (np. kwestie podpisu elektronicznego).

Deklarowanie kolumn typu XML nie odbiega od deklarowania kolumn każdego innego typu. Jedyną specyficzną rzeczą jest – w przypadku kolumny ze skojarzoną kolekcją dokumentów XML Schema – umieszczenie w nawiasie w deklaracji typu nazwy tej kolekcji.

Sama kolekcja dokumentów XML Schema zawierać może jedną bądź wiele pojedynczych schem, które zapisuje się jedna pod drugą. Po skojarzeniu kolekcji z kolumną typu XML, każda wartość wpisywana do tej kolumny będzie walidowana pod kątem zgodności z którąś ze schem z kolekcji. W przypadku braku zgodności – operacja zapisu zostanie anulowana.

Tworzenie dokumentów XML Schema (zwane też modelowaniem dopuszczalnej struktury dokumentów XML) jest zagadnieniem bardzo rozbudowanym i wykracza poza ramy niniejszego wykładu. Przy założeniu, że mamy już określoną postać dokumentu XML Schema, stworzenie kolekcji schem jest proste i ogranicza się do wykonania jednego polecenia. Od tego momentu można używać zdefiniowanej w ten sposób kolekcji przy deklaracjach kolumn typu XML.

Kiedy w praktyce należy stosować typ danych XML? W tej kwestii zdania są podzielone. Jedni z definicji odrzucają XML traktując go jako niepotrzebny, a wręcz szkodliwy wodotrysk (stawiając m.in. zarzuty co do kiepskiej wydajności), drudzy używają go, gdzie tylko się da, zastępując jedną kolumną XML strukturę kilku tabel lub tworząc procedury składowane, którym przekazuje się tylko jeden parametr typu XML, z którego są potem „wydłubywane” konkretne wartości. W skrajnych przypadkach cała komunikacja z bazą danych sprowadza się do wymiany dokumentów XML. Z aplikacji przychodzi żądanie z parametrami w postaci XML, na które baza odpowiada zwracając dokument XML z odpowiednią strukturą danych. Sprowadza to komunikację z bazą danych do postaci zbliżonej do korzystania z usług sieciowych (ang. *webservices*), które stają się w ostatnich latach coraz bardziej popularne.

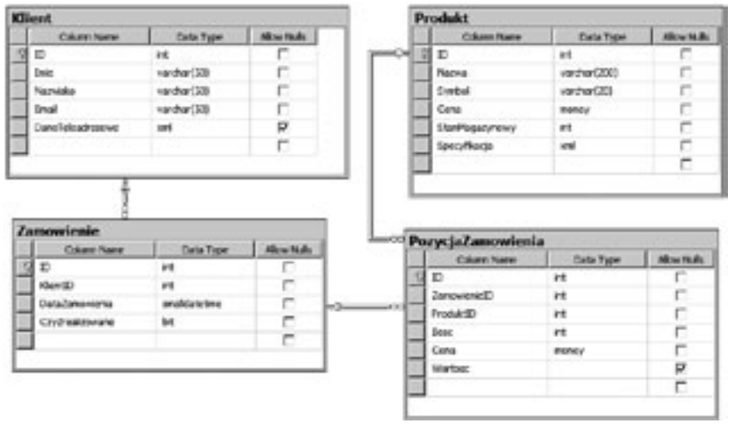
W przypadku niniejszego wykładu, z typu danych XML skorzystaliśmy w dwóch tabelach w przykładowej bazie danych (patrz rysunek 1).

W pierwszej tabeli (Klient) kolumna XML służy do przechowywania danych teleadresowych klienta (różne rodzaje adresów, identyfikatorów z komunikatorów itp.). W drugiej tabeli (Produkt), XML zostało zastosowane do przechowywania specyfikacji produktu. Ze względu na to, że produkty różnych kategorii mogą być opisywane zupełnie innym zbiorem cech – zastosowanie XML w połączeniu z kolekcją schem zapewni wygodny mechanizm przechowywania specyfikacji produktów. Podobnie wygląda sytuacja w przypadku pierwszej kolumny – tu kolekcja schem ma zapewnić opisanie i wymuszenie odpowiedniej postaci listy adresów/kontaktów klienta. W skład takiej listy mogą wchodzić adresy pocztowe, email czy identyfikatory różnych komunikatorów internetowych.

Skorzystanie z typu danych XML przyczyniło się do znacznego uproszczenia struktury bazy danych. W przypadku obu tabel przechowywane dane byłyby w postaci zmiennego zbioru cech i ich wartości przypisanych do konkretnego klienta czy produktu. Przy takim rozumieniu problemu jedna kolumna XML eliminuje ze struktury bazy co najmniej dwie tabele (słownik cech oraz przypisanie cechy i jej wartości do klienta/produktu). Dodatkowo, ze względu na niewielki rozmiar danych XML i sposób korzystania z nich, nie ma co się zbytnio przejmować spadkiem wydajności. W innych przypadkach jest to jednak bardzo istotne kryterium, które często staje się jedną z głównych przyczyn zarzucenia stosowania typu danych XML w konkretnym projekcie.

Manipulowanie danymi typu XML

Oprócz samej możliwości przechowywania dokumentów XML w kolumnie w bazie danych, typ danych XML oferuje jeszcze inne możliwości. Są one udostępniane jako metody, które można wywoływać na rzecz kolumny. Umożliwiają zaawansowane odpytywanie dokumentu XML oraz manipulowanie jego strukturą. Mechanizm ten działa w oparciu o język XQuery oraz wyrażenia XPath. Warto zaznaczyć, że implementacja XQuery i XPath w SQL Server 2008 nie zawiera wszystkich możliwości wynikających z ich specyfikacji.



Rysunek 1.
Tabele przykładowej bazy danych

Język XQuery, jak i wspomniany już wcześniej XPath, są rozwiązaniami stworzonymi i rozwijanymi przez organizacje W3C (ang. *World Wide Web Consortium*). Zastosowanie ich w miejsce własnych (Microsoftu) rozwiązań jest przejawem przykładania coraz większej wagi do otwartych technologii, co jest cechą charakterystyczną całego świata XML. Język XPath jest zaprojektowany do wskazywania (wybierania) odpowiednich węzłów dokumentu XML. Wyrażenia XPath z reguły nie funkcjonują samodzielnie, są natomiast szeroko stosowane w innych rozwiązaniach (np.: XSLT, XQuery).

Rola XQuery, jak sama nazwa wskazuje, polega na odpytywaniu dokumentu XML. Oprócz prostego wariantu bazującego na wyrażeniach XPath, można stosować rozbudowane wyrażenia FLWOR (nazwa pochodzi od angielskich słów: *For*, *Let*, *Where*, *Order by*, *Return*). XQuery bywa często porównywany do polecenia SELECT znanego z języka SQL, i mówi się, że XQuery jest dla XML tym, czym SELECT dla SQL.

Wróćmy jednak do możliwości typu danych XML. Otóż zawiera on pięć metod, które mają za zadanie ułatwienie korzystania z danych zawartych wewnątrz dokumentu XML oraz modyfikowanie samego dokumentu. Dodatkowo dają one możliwość wykonania operacji odwrotnej do działania klauzuli FOR XML – czyli do przetransformowania danych z XML w wiersz zbioru wyników (ROWSET).

Aby korzystać z metod typu XML, należy opanować dodatkowo podstawy wspomnianego języka XQuery oraz XPath, gdyż wyrażenia zbudowane w oparciu o nie są stosowane w parametrach wywołania metod typu XML.

Krótki opis metod typu XML zawiera poniższe zestawienie:

- Metoda `value(xquery,typ)` służy do wskazania poprzez wyrażenie XPath elementu lub atrybutu, którego wartość będzie pobrana z dokumentu XML, a następnie skonwertowana do typu wskazanego w drugim parametrze wywołania metody `value()`.
- Metoda `exist(xquery)` stosowana jest do sprawdzenia, czy kolumna XML zawiera w swojej wartości element lub atrybut wskazany przez wyrażenie XQuery. Podobny efekt da się osiągnąć za pomocą metody `value()`. Jeżeli jednak nie ma konieczności pobierania wartości z XML, a tylko sprawdzenia jej istnienia, to metoda `exist()` jest zalecana ze względu na szybsze działanie.
- Metoda `query(xquery)` służy do wskazania przez wyrażenie `xquery` zbioru węzłów z dokumentu XML, które są następnie zwracane także jako zmienna typu XML.
- Metoda `nodes(xquery)` jest używana do przetworzenia danych zawartych w dokumencie XML na postać relacyjną. Z jej pomocą (oraz z wykorzystaniem operatora CROSS APPLY) można wybrać węzły dokumentu, które będą tworzyły kolumny wiersza danych w zbiorze wynikowym zapytania SELECT. Rezultatem działania metody `nodes()` jest zbiór wierszy zawierający logiczne kopie węzłów dokumentu XML wybranych przez wyrażenie `xquery`. Funkcja ta jest szczególnie użyteczna i wygodna, gdy tworzymy zapytanie, które ma zawierać w kolumnach poszczególne informacje zaszyte w strukturze elementów i atrybutów dokumentu XML.
- Metoda `modify(xmldml)` służy do modyfikowania zawartości dokumentu XML. Modyfikacje te są realizowanie za pomocą poleceń języka XML DML (ang. *XML Data Manipulation Language*). W skład XML DML wchodzą trzy polecenia:
  - `insert` – służące do dodawania nowych węzłów (elementów, atrybutów, węzłów tekstowych itp.) do dokumentu XML
  - `delete` – stosowane do usuwania węzłów z dokumentu
  - `replace value of` – służące do zastępowania zawartości węzła dokumentu inną zawartością

    Możliwości tych trzech poleceń są dość ograniczone i łatwe do zastosowania wyłącznie w przypadku prostych modyfikacji operujących z reguły na wartościach podawanych w postaci stałych łańcuchów znaków. Gdy potrzebne są możliwości dynamicznego budowania wartości, która ma być wstawiona do dokumentu, to szybko okazuje się, że jest to trudne bądź wręcz niemożliwe. Dlatego, gdy wymagane są bardziej złożone operacje na dokumencie XML, to są realizowane one po stronie aplikacji, a ich gotowy wynik jest przekazywany do bazy.

Możliwości manipulowania danymi zapisanymi w kolumnach lub zmiennych typu XML są istotnym elementem składowym funkcjonalności obsługi XML w bazie danych. Umożliwiają realizowanie typowych operacji na danych XML w sposób zbliżony do znanego ze świata XML. Jest to bardzo istotne ze względu na łatwość zastosowania tych rozwiązań w praktyce.

PODSUMOWANIE

W ramach niniejszego wykładu zaprezentowany został pokrótce język XML oraz możliwości korzystania z niego w relacyjnych bazach danych. Z racji rosnącej popularności tego języka oraz coraz bogatszego wsparcia ze strony producentów serwerów baz danych, warto się nim zainteresować, aby móc rozważać go jako jedną z opcji przy projektowaniu baz danych. Jako przykład serwera baz danych, oferującego wsparcie dla XML, wykorzystany został SQL Server 2008. W ramach omawiania jego możliwości w kontekście XML, zasygnalizowane zostały główne obszary, w których serwer oferuje narzędzia służące do obsługi dokumentów XML. Przyjrzeliśmy się klauzuli FOR XML stosowanej do zwracania wyników zapytania w postaci dokumentów lub fragmentów dokumentów XML. Zapoznaliśmy się także z typem danych XML, który służy nie tylko do przechowywania danych w tym formacie, ale także do zaawansowanego odpytywania dokumentów XML oraz manipulowania ich zawartością. Dodatkowo, wspomnieliśmy o wykorzystywanych w ramach SQL Server 2008 innych technologiach i narzędziach wspierających XML. Warto tu wymienić choćby XQuery, XPath oraz XML Schema. Wszystkie te rozwiązania są otwarte, nie stanowią własności żadnej firmy, są rozwijane przez konsorcjum W3C i szeroko adoptowane w świecie IT. Umożliwia to ekspertom od XML na łatwe wykorzystanie nabytej wiedzy również przy używaniu z serwerów baz danych.

Jest jednak także druga strona medalu. Jeśli zachłyśniemy się wsparciem dla XML w SQL Server 2008, to bardzo szybko może nas czekać rozczarowanie. Otóż mechanizmy wspierające XML na pierwszy rzut oka wyglądają na potężne i wygodne. Faktycznie jest tak, ale tylko w zakresie przewidzianym przez ich twórców. Okazuje się, że nie zaimplementowali oni w pełni standardów XQuery i XPath (można szybko natrafić na wiele nieobsługiwanych funkcji bądź ograniczeń). Podobnie jest w przypadku XML Schema. Nie wszystkie możliwości tej technologii są dopuszczalne w ramach SQL Server 2008. W zakresie manipulowania strukturą dokumentów XML również okazuje się, że metoda `modify()` z poleceniami XML DML ma bardzo dużo ograniczeń, szczególnie gdy chce się polecenia tworzyć bardziej dynamicznie. To wszystko nie umniejsza jednak faktu, że wsparcie XML w relacyjnych bazach danych jest ciekawą możliwością, z której warto korzystać, po uprzednim gruntownym zapoznaniu się z dokumentacją, aby uniknąć przykrych niespodzianek w trakcie realizacji projektu.

LITERATURA

1. Liberty J., Kraley M., *XML od podstaw*, Translator, Warszawa 2001
2. Rizzo T., Machanic A., Dewson R., Walters R., Sack J., Skin J., *SQL Server 2005*, WNT, Warszawa 2008
3. Vieira R., *SQL Server 2005. Programowanie. Od Podstaw,* Helion, Gliwice 2007
4. Walmsley P., *Wszystko o XML Schema*, Helion, Gliwice 2007

Optymalizacja zapytań SQL

Andrzej Ptasznik
Warszawska Wyższa Szkoła Informatyki
aptaszni@wwsi.edu.pl

Streszczenie

Wykład zapoznaje słuchaczy z problematyką wydajności i optymalizacji zapytań SQL. Omówiona zostanie fizyczna organizacja przechowywania danych i wprowadzone zostaną pojęcia indeksów zgrupowanych i niezgrupowanych. Zaprezentowane zostaną przykłady planów wykonania zapytań generowane przez optymalizator SQL. Na bazie przykładu omówione będą problemy wyboru strategii wykonania zapytania w zależności od zawartości tabel i zdefiniowanych indeksów. Wykład wprowadzi pojęcie statystyk indeksów i ich znaczenie przy wyborze strategii realizacji zapytania.

Spis treści

1 WPROWADZENIE

W każdym projekcie informatycznym, wykorzystującym relacyjne bazy danych, prędzej czy później pojawia się problem związany z wydajnością. Jeśli „prędzej" oznacza „przed wdrożeniem", to nie jest jeszcze tak źle. Można wtedy podjąć decyzje wiążące się z dokonywaniem zmian w projekcie bazy danych i nie będą one się wiązać z koniecznością dbania o już istniejące dane. Gorszym wariantem jest praca na „żywym organizmie". Nie dość, że możliwości modyfikacji są ograniczone, to jeszcze trzeba starać się nie zakłócać normalnej pracy użytkowników. Gdy dodamy do tego presję czasu i stres – pojawia się obraz pracy nie do pozazdroszczenia. W każdym jednak przypadku istotne jest, żeby wiedzieć, jakie kroki podjąć, co sprawdzić, na co zwrócić szczególną uwagę, jakich narzędzi użyć i w jaki sposób, aby osiągnąć cel – wzrost wydajności bazy danych do akceptowalnego poziomu. Może nam się wydawać, że takie problemy dotyczą tylko dużych projektów i baz danych, więc nie ma się co martwić na zapas. Bardzo szybko jednak można natrafić na podobne problemy nawet w prostych aplikacjach.

W ramach niniejszego wykładu postaramy się przedstawić podstawy wiedzy potrzebnej do poruszania się w dziedzinie zagadnień związanych z wydajnością baz danych, a dokładniej – zapytań na nich wykonywanych. Zanim zaczniemy jednak wkraczać do problematyki optymalizacji zapytań SQL, postaramy się odpowiedzieć na pytanie: a po co w ogóle optymalizować? Odpowiedź na to pytanie nie jest, wbrew pozorom, taka oczywista. Niejako przy okazji zaprezentowany zostanie też ogólny model optymalizacji wydajności stosowany w praktyce przy realizacji zadań związanych z zapewnieniem wymaganego poziomu wydajności bazy danych.

2 MODEL OPTYMALIZACJI WYDAJNOŚCI BAZY DANYCH

W świecie systemów informatycznych i komputerów od wielu lat utrzymuje się stały trend wzrostu mocy obliczeniowej, pojemności pamięci operacyjnej, pojemności i szybkości dysków twardych itp. W związku z tym, jeśli mamy do czynienia ze zbyt niską wydajnością bazy danych, to pierwszym pomysłem może być rozbudowa systemu od strony sprzętowej – a nuż, ten dodatkowy procesor lub 4 GB pamięci dodadzą bazie skrzydeł. Niestety nie zawsze to działa, lub przewidywane koszty rozbudowy są zdecydowanie nieakceptowalne. Osiągnięty efekt może także nie być zbyt długotrwały i po kolejnym miesiącu uzupełniania danych w bazie wracamy do punktu wyjścia – działa za wolno!

W takiej sytuacji warto zrobić to, od czego tak naprawdę należało zacząć – przeanalizować bazę danych pod kątem możliwości optymalizacji jej wydajności. Okazuje się, że tą drogą można otrzymać bardzo dobre rezultaty. Niestety wymaga to znacznej wiedzy i umiejętności, a także sporej dozy wyczucia, którego ot tak nie da się nauczyć. Istnieją sprawdzone w praktyce podejścia (modele) optymalizacji wydajności baz danych, lecz ich rola polega raczej na wyznaczeniu ogólnych ram i sekwencji czynności, których wykonanie należy wziąć pod uwagę przy prowadzeniu optymalizacji, niż na dostarczeniu gotowej recepty. Proces optymalizacji wydajności według przyjętego przez nas modelu składa się z kilku obszarów:

- Struktura (projekt) bazy danych
- Optymalizacja zapytań
- Indeksy
- Blokady
- Tuning serwera

Całość modelu jest przedstawiona na diagramie na rysunku 1.

Kolejność realizacji zadań powinna przebiegać od dołu diagramu do góry. Podobnie, liczba możliwych do osiągnięcia usprawnień jest tym większa, im niżej znajdujemy się na diagramie. Sekwencja ta nie jest przypadkowa i wzajemne zależności pomiędzy blokami powodują, że założona kolejność realizacji umożliwia uzyskanie najlepszych efektów najmniejszym kosztem. W ramach wykładu skupimy się na wyróżnionych blokach – optymalizacji zapytań i indeksach.

Pierwszym i najistotniejszym zadaniem jest jednak optymalizacja struktury bazy danych. Osiąga się ją zwykle poprzez normalizację (doprowadzenie do trzeciej postaci normalnej). Taka postać cechuje się większą liczbą tabel, krótszymi rekordami w tabelach, mniejszą podatnością na blokowanie, łatwiejszym tworzeniem zapytań bazujących na zbiorach itp. Czas poświęcony na tym etapie zwraca się bardzo szybko, podobnie jak błędy tu popełnione okrutnie mszczą się przy dalszych próbach optymalizacji wydajności.

Rysunek 1.
Model procesu optymalizacji

Bardzo istotną rzeczą jest pamiętanie o tym, że baza nie istnieje sama dla siebie. Z reguły współpracuje z jakąś aplikacją lub aplikacjami. Jakiekolwiek modyfikacje struktury bazy danych mogą wiązać się z koniecznością wprowadzania modyfikacji kodu aplikacji, a to nie jest już tak miłe. Z tego względu zalecane jest podejście zakładające utworzenie w bazie danych „warstwy abstrakcji danych", której rola polega na odcięciu aplikacji od szczegółów struktury bazy danych. Zwykle realizowane jest to za pomocą widoków, procedur składowanych czy funkcji użytkownika. Aplikacje „widzą" i korzystają tylko z tych obiektów, nie kontaktując się bezpośrednio z tabelami. Dzięki temu, w przypadku modyfikowania struktury bazy, można ukryć ten fakt przed aplikacjami – wystarczy zmodyfikować kod procedury czy widoku, aby pasował do nowej struktury tabel, a aplikacje jak z nich korzystały tak będą korzystać – zupełnie nieświadome, że dane pobierane wcześniej z dwóch tabel obecnie są rozrzucone po pięciu tabelach.

Kolejnym etapem jest optymalizacja zapytań. Głównym zagadnieniem jest tu oderwanie się od starych nawyków pisania zapytań iteracyjnych (często korzystając z kursorów) na rzecz zapytań bazujących na zbiorach. Są one bardziej wydajne oraz łatwiej skalowalne. Indeksy łączą się z poprzednim etapem pełniąc rolę pomostu pomiędzy zapytaniem a danymi. Dobrze napisane zapytania z odpowiednio dobranymi indeksami potrafią czynić cuda, a z drugiej strony żaden indeks nie naprawi kardynalnych błędów w zapytaniach czy strukturze danych. Z kolei kwestie blokad wiążą się nierozerwalnie z korzystaniem z bazy przez wielu użytkowników jednocześnie. Często zdarza się, że pozornie wydajnie działająca baza szybko traci wigor przy kolejnych jednocześnie pracujących użytkownikach. Jakiekolwiek próby na tym poziomie nie dadzą nic, jeżeli na poprzednich etapach przeoczyliśmy jakieś problemy.

Ostatni poziom to wspominane już wcześniej „rozszerzanie" serwera. Zwiększanie mocy procesora/ów, ilości pamięci czy szybkości i pojemności dysków umożliwiają osiągnięcie szybkiego efektu wzrostu wydajności. Nie pomoże to jednak w przypadku błędów popełnionych na poprzednich etapach i efekt końcowy może być mizerny, szczególnie wziąwszy pod uwagę koszty.

3 FIZYCZNA ORGANIZACJA DANYCH W SQL SERVER 2008

Skoro przekonaliśmy się już co do konieczności zwrócenia uwagi na zagadnienia w ramach optymalizowania wydajności, to możemy przejść do rzeczy i rozpocząć zgłębianie tej dziedziny. Nie uda się nam to w żaden sposób, jeśli nie zrozumiemy mechanizmów leżących u podstaw działania SQL Servera. Jednym z istotnych zagadnień jest tu sposób, w jaki dane są fizycznie przechowywane w bazie danych. Gdy myślimy o tabeli, to od razu przedstawiamy sobie coś na kształt zbioru wierszy składających się z kolumn zawierających dane różnego typu (patrz rys. 2).

ContactID	Title	FirstName	LastName	EmailAddress	Phone
1	Mr.	Gustavo	Achong	gustavo0@adventure-works.com	398-555-0132
2	Ms.	Catherine	Abel	catherine0@adventure-works.com	747-555-0171
3	Ms.	Kim	Abercrombie	kim2@adventure-works.com	334-555-0137
4	Sr.	Humberto	Acevedo	humberto0@adventure-works.com	599-555-0127

Rysunek 2.
Tabela w bazie danych

Nie zastanawiamy się, jak te dane są przechowywane fizycznie na dysku ani jaki wpływ na wydajność mogą mieć nasze decyzje podjęte przy projektowaniu tabeli. Warto jednak zadać sobie nieco trudu i zapoznać się z fizycznym sposobem przechowywania danych w bazie. Zrozumienie podstaw ułatwi później wyjaśnienie, dlaczego w takiej czy innej sytuacji wykonanie zapytania czy modyfikacji danych trwa tak długo.

Strony i obszary

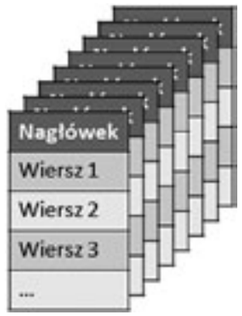
Najmniejszą jednostką przechowywania danych jest w SQL Serverze **strona** (ang. *page*). Jest to 8 KB blok składający się z nagłówka i 8060 bajtów na dane z wiersza (lub wierszy). Przy założeniu, że wiersz tabeli musi się zmieścić na stronie jasno widać, że maksymalny rozmiar wiersza to 8060 bajtów. Trochę mało? Niekoniecznie. Część danych o rozmiarze przekraczającym 8 KB jest zapisywana na innych stronach, a w samym wierszu umieszczany jest tylko wskaźnik do pierwszej z tych stron. SQL Server rozróżnia 9 rodzajów stron przechowujących informacje o rozmaitym znaczeniu:

- Strony danych (ang. *data*) zawierają wszystkie dane z wiersza, z wyjątkiem kolumn typów: text, ntext, image, nvarchar(max), varchar(max), varbinary(max), xml.
- Jeżeli wiersz nie mieści się w limicie długości 8060 bajtów, to najdłuższa z kolumn jest przenoszona do tzw. **strony przepełnienia** (strona danych), a w jej miejscu w wierszu zostaje 24 bajtowy wskaźnik.
- Strony indeksów (ang. *index*) zawierają poszczególne wpisy indeksu. W ich przypadku istotny jest limit długości klucza indeksu – 900 bajtów.
- Strony obiektów BLOB/CLOB (ang. *Binary/Character Large Object*) (ang. *text/image*) służą do przechowywania danych o rozmiarze do 2 GB.
- Strony GAM, SGAM i IAM – wrócimy do nich w dalszej części wykładu, gdy poznamy kolejne pojęcia dotyczące fizycznego przechowywania danych.

Wymieniliśmy tylko 6 rodzajów stron, żeby niepotrzebnie nie komplikować dalszych rozważań. Dla uporządkowania warto wspomnieć o pozostałych trzech: Page Free Space, Bulk Changed Map, Differential Changed Map. Pierwsza zawiera informacje o zaalokowanych stronach i wolnym miejscu na nich. Pozostałe dwa rodza-

je są wykorzystywane do oznaczania danych zmodyfikowanych w ramach operacji typu bulk oraz do oznaczania zmian od ostatnio wykonanej kopii zapasowej.

Podstawową jednostką alokacji nie jest jednak w SQL Serverze strona, tylko zbiór ośmiu stron zwany **obszarem** (ang. *extent*) – rysunek 3.

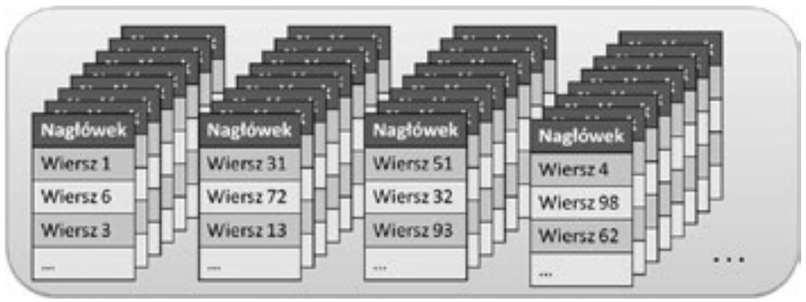


Rysunek 3.
Obszar

Jest tak ze względu na fakt, iż 8 KB to trochę za mało jak na operacje w systemie plików, a 64 KB to akurat jednostka alokacji w systemie plików NTFS. Obszary mogą zawierać strony należące do jednego obiektu (tabeli czy indeksu) – nazywamy je wtedy **jednolitymi** (ang. *uniform*), lub do wielu obiektów – stają się wtedy **obszarami mieszanymi** (ang. *mixed*). Jeżeli SQL Server alokuje miejsce na nowe dane, to najmniejszą jednostką jest właśnie obszar.

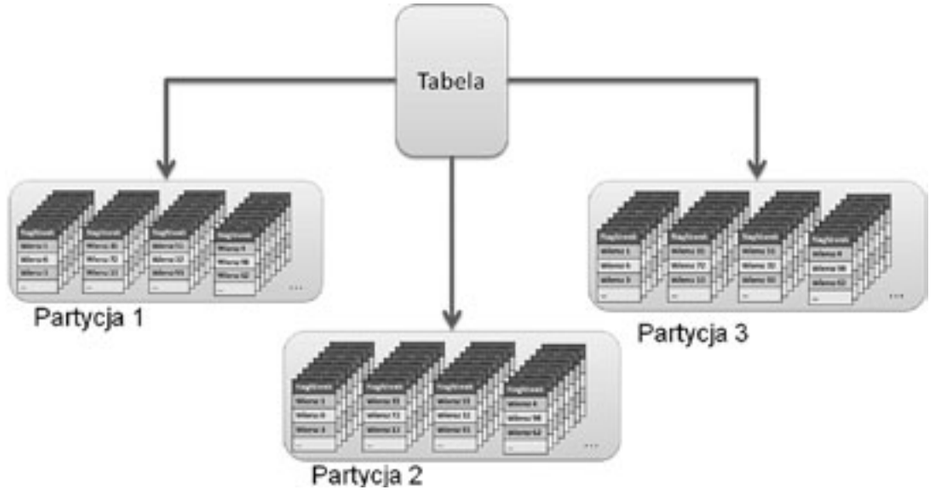
Sterty

Jeżeli tabela nie zawiera żadnego indeksu, to jej dane tworzą **stertę** – nieuporządkowaną listę stron należących do tej tabeli. Wszelkie operacje wyszukiwania na stercie odbywają się wolno, gdyż wymagają zawsze przejrzenia wszystkich stron. Inaczej w żaden sposób serwer nie jest w stanie stwierdzić, czy np. odnalazł już wszystkie wiersze zawierające dane klientów o nazwisku Kowalski. Stertę można wyobrazić sobie jak na rysunku 4.



Rysunek 4.
Przykładowa sterta

Dodatkowo tabela może zostać podzielona na partycje (względy wydajnościowe – zrównoleglenie operacji wejścia/wyjścia). W takim przypadku każda z partycji zawiera własną stertę. Wszystkie razem tworzą zbiór danych tabeli (rys. 5).



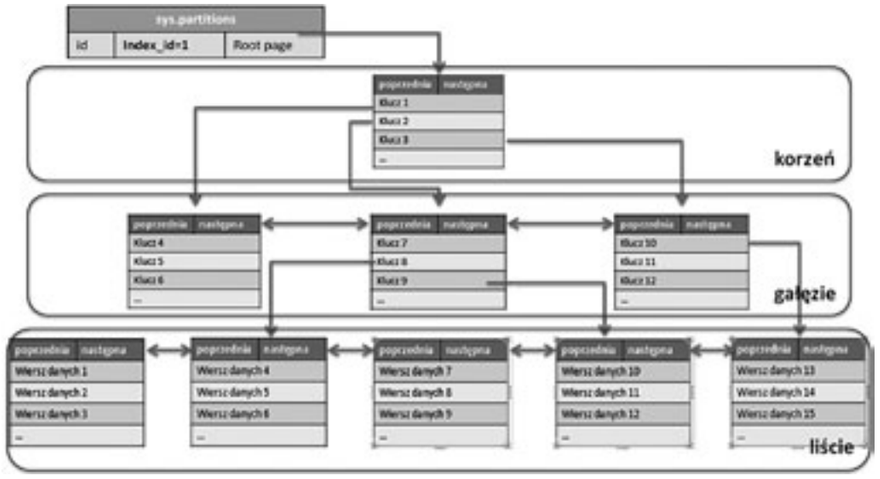
Rysunek 5.
Tabela, partycje, sterty

Gdy SQL Server alokuje miejsce w plikach bazy danych, wypełnia je obszarami, które wstępnie są oznaczone jako wolne. Podobnie wszystkie strony w obszarach są oznaczone jako puste. W jaki sposób przechowywane są informacje na temat tego, czy dany obszar lub strona są wolne lub należą do jakiegoś obiektu? Służą do tego specjalne strony – GAM, SGAM i IAM. Zawierają one informacje o zajętości poszczególnych obszarów w postaci map bitowych (GAM, SGAM) lub o przynależności obszarów do obiektów (tabel, indeksów) – IAM. Kluczem do uzyskania dostępu do danych z tabeli jest możliwość dostania się do strony IAM tej tabeli. Informacje na temat lokalizacji stron IAM dla poszczególnych obiektów znajdują się we wpisach w tabelach systemowych. Jako że nie zaleca się „szperania" bezpośrednio w tych tabelach, zostały udostępnione specjalne widoki, które zawierają potrzebne nam dane. W przypadku stron IAM jest to widok sys.partitions. Wpisy w nim zawarte składają się m.in. z kolumny index_id określającej rodzaj obiektu (sterta, indeks zgrupowany, indeks niezgrupowany, obiekty LOB), kolumn wskazujących id obiektu i partycji oraz wskaźnika do strony IAM obiektu.

Indeksy zgrupowane i niezgrupowane

Poznaliśmy już w zarysie sposób przechowywania danych w tabeli, dla której nie stworzono indeksów. Cechą charakterystyczną był fakt nieuporządkowania stron i wierszy należących do jednej tabeli, co wymuszało przy każdej operacji wyszukiwania danych w tabeli przeszukanie wszystkich wierszy. Taka operacja nosi nazwę **skanowania tabeli** (ang. *table scan*). Jest ona bardzo kosztowna (w sensie zasobów) i wymaga częstego sięgania do danych z dysku, tym częściej im więcej danych znajduje się w tabeli. Taki mechanizm jest skrajnie nieefektywny, więc muszą istnieć jakieś inne, bardziej efektywne mechanizmy wyszukiwania. Rzeczywiście istnieją – są to **indeksy**, występujące w dwóch podstawowych wariantach jako indeksy: **zgrupowane** (ang. *clustered*) i **niezgrupowane** (ang. *nonclustered*)

Indeks zgrupowany ma postać drzewa zrównoważonego (ang. *B-tree*). Na poziomie korzenia i gałęzi znajdują się strony indeksu zawierające kolejne wartości klucza indeksu uporządkowane rosnąco. Na poziomie liści znajdują się podobnie uporządkowane strony z danymi tabeli. To właśnie jest cechą charakterystyczną indeksu zgrupowanego – powoduje on fizyczne uporządkowanie wierszy w tabeli, rosnąco według wartości klucza indeksu (wskazanej kolumny lub kolumn). Z tego względu oczywiste jest ograniczenie do jednego indeksu zgrupowanego dla tabeli.



Rysunek 6.
Indeks zgrupowany

Specyfika indeksu zgrupowanego polega na fizycznym porządkowaniu danych z tabeli według wartości klucza indeksu. W związku z tym jasne jest, że indeks ten będzie szczególnie przydatny przy zapytaniach operujących na zakresach danych, grupujących dane, oraz korzystających z danych z wielu kolumn. W takich przypadkach indeks zgrupowany zapewnia znaczny wzrost wydajności w stosunku do sterty lub indeksu niezgrupowanego.

Istotną rzeczą przy podejmowaniu decyzji o utworzeniu indeksu zgrupowanego jest wybranie właściwej kolumny (kolumn). Długość klucza powinna być jak najmniejsza, co umożliwia zmieszczenie większej liczby wpisów indeksu na jednej stronie, co z kolei przenosi się na zmniejszenie liczby stron całości indeksu i w efekcie mniej operacji wejścia/wyjścia do wykonania przez serwer. Żeby indeks zgrupowany korzystnie wpływał na wydajność przy dodawaniu nowych wierszy do mocno wykorzystywanej tabeli, klucz powinien przyjmować dla kolejnych wpisów wartości rosnące (zwykle stosowana jest tu kolumna z cechą *identity*). Indeks daje duży zysk wydajności, gdy jego klucz jest możliwie wysoko selektywny (co oznacza mniejszą liczbę kluczy o tej samej wartości – duplikatów). Istotny jest także fakt, że kolumny klucza indeksu zgrupowanego nie powinny być raczej modyfikowane, gdyż pociąga to za sobą konieczność modyfikowania nie tylko stron indeksu, ale także porządkowania stron danych.

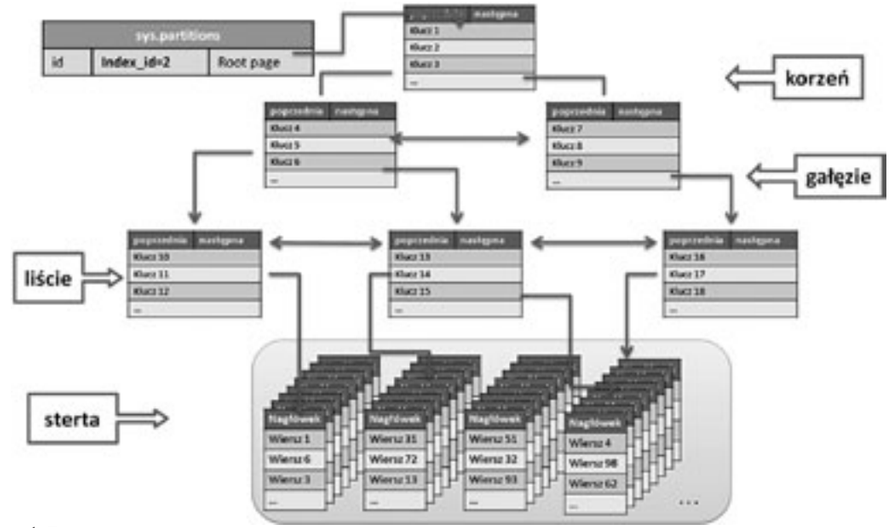
Indeksy zgrupowane nie wyczerpują możliwości budowania tego typu struktur w SQL Serverze 2008. Drugim typem indeksów są **indeksy niezgrupowane**. Ich budowa odbiega nieco od budowy indeksu zgrupowanego, a do tego indeksy niezgrupowane mogą być tworzone na bazie sterty lub istniejącego indeksu zgrupowanego. Dla jednej tabeli można utworzyć do 248 indeksów niezgrupowanych. Indeks niezgrupowany różni się od zgrupowanego przede wszystkim tym, że w swojej strukturze na poziomie liści ma także strony indeksu (a nie strony danych).

W przypadku budowania indeksu niezgrupowanego na stercie, strony te oprócz wartości klucza indeksu zawierają wskaźniki do konkretnych stron na stercie, które dopiero zawierają odpowiednie dane.

Indeksy niezgrupowane mają strukturę zbliżoną do zgrupowanych. Zasadnicza różnica polega na zawartości liści indeksu. O ile indeksy zgrupowane mają w tym miejscu strony danych, to indeksy niezgrupowane – strony indeksu. Strony te zależnie od wariantu indeksu niezgrupowanego zawierają oprócz klucza różne informacje. Indeksy niezgrupowane mogą być tworzone w oparciu o stertę. Jest to możliwe tylko wtedy, gdy

tabela nie ma indeksu zgrupowanego. W takim przypadku liście indeksu zawierają wskaźniki do konkretnych stron na stercie.

Indeks niezgrupowany tworzony na tabeli zawierającej już indeks zgrupowany, jest tworzony nieco inaczej. Korzeń, gałęzie i liście zawierają strony indeksu, ale liście zamiast wskaźników do stron na stercie zawierają wartości klucza indeksu zgrupowanego. Każde wyszukanie w oparciu o indeks niezgrupowany po dojściu do poziomu liści zaczyna dalsze przetwarzanie od korzenia indeksu zgrupowanego (wyszukiwany jest klucz zawarty w liściu).

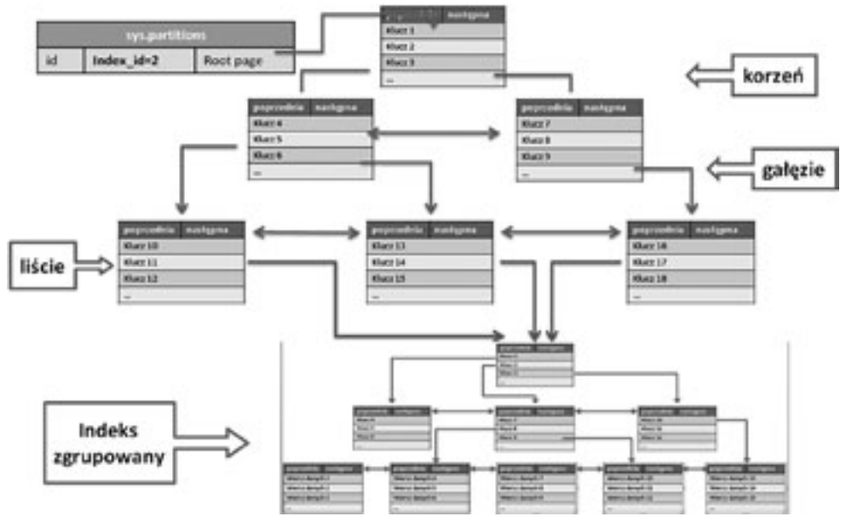


Rysunek 7.
Indeksy niezgrupowane – sterty w liściach

W przypadku budowania indeksów niezgrupowanych, szczególnie przy dużych tabelach, warto dobrze zaplanować tę czynność, szczególnie gdy planowane jest też utworzenie indeksu zgrupowanego. Niewzięcie tego pod uwagę może powodować konieczność przebudowywania indeksów niezgrupowanych w związku z dodaniem lub usunięciem indeksu zgrupowanego.

W sporym uproszczeniu rola indeksów sprowadza się do ograniczenia liczby operacji wejścia/wyjścia niezbędnych do realizacji zapytania. SQL Server nie odczytuje poszczególnych obszarów potrzebnych do realizacji zapytania z dysku za każdym razem. Zawiera rozbudowany bufor pamięci podręcznej, do której trafiają kolejne odczytywane z dysku obszary. Ze względu na ograniczony rozmiar bufora, strony nieużywane lub używane rzadziej są zastępowane tymi, z których zapytania korzystają częściej.

Przy korzystaniu z indeksów niezgrupowanych istnieje jeszcze jedna możliwość dalszego ograniczania liczby operacji wejścia/wyjścia. Polega ona na tym, że do indeksu (dokładnie do stron liści indeksu) dodawane są dodatkowe kolumny. Jeżeli liście indeksu niezgrupowanego zawierają wszystkie kolumny zwracane przez zapytanie, to nie ma w ogóle konieczności sięgania do stron z danymi. W takim przypadku mamy do czynienia z tak zwanym **indeksem pokrywającym**. Dodawanie kolumn do indeksu niezgrupowanego może polegać na dodawaniu kolejnych kolumn do klucza (występuje tu ograniczenie do 16 kolumn w kluczu i 900 bajtów długości klucza) albo na dodawaniu kolumn „niekluczowych” do indeksu (nie wliczają się one do długości klucza). Trzeba jednak pamiętać, że tworzenie indeksów pokrywających dla kolejnych zapytań nie prowadzi do niczego dobrego, gdyż po pierwsze rośnie liczba danych (wartości kolumn są przecież kopiowane do stron indeksu), a po drugie drastycznie spada wydajność modyfikowania danych (pociąga za sobą konieczność naniesienia zmian we wszystkich indeksach).



Rysunek 8.
Indeksy niezgrupowane – indeksy zgrupowane w liściach

Indeksy pokrywające

Żeby zademonstrować sposób działania indeksów pokrywających, załóżmy następującą sytuację. W bazie istnieje tabela zawierająca dane klientów. W jej skład wchodzi kilka kolumn: ID, Nazwisko, Imie, Email, DataOstatniegoZamowienia. Na tabeli został stworzony indeks zgrupowany na kolumnie ID oraz indeks niezgrupowany na kolumnie Nazwisko. Jeżeli w takim przypadku realizowane będzie zapytanie, które co prawda w klauzuli WHERE zawiera warunek tylko dla kolumny Nazwisko (zawartej w indeksie niezgrupowanym), ale na liście kolumn wyjściowych zawiera także inne kolumny (w naszym przypadku kolumna Email), to indeks niezgrupowany nie zostanie wykorzystany, gdyż wartości kolumn spoza indeksu muszą zostać pobrane ze stron danych. Zapytanie zostanie zrealizowane poprzez skanowanie indeksu zgrupowanego. Jeżeli usuniemy z listy kolumn wyjściowych kolumnę Email i wykonamy zapytanie ponownie, to tym razem indeks niezgrupowany okaże się przydatny i zostanie na nim wykonana operacja wyszukiwania w indeksie (ang. *index seek*). Będzie ona mniej kosztowna od skanowania indeksu zgrupowanego, gdyż nie wymaga dostępu do stron danych. Żeby osiągnąć ten sam efekt z kolumną Email na liście wyjściowej należy dodać ją do indeksu niezgrupowanego (jako część klucza lub nie). Po takiej modyfikacji osiągniemy założony cel – zapytanie zostanie zrealizowane z wykorzystaniem operacji wyszukiwania w indeksie niezgrupowanym.

Mechanizm indeksów pokrywających wygląda bardzo fajnie i nie jest trudny w zastosowaniu. Jest jednak druga strona medalu. Zwykle zapytań jest więcej niż jedno i zwracają więcej kolumn. Rozbudowywanie indeksów (zarówno ich liczba, jak i liczba kolumn w nich zawartych) prowadzi do znacznego wzrostu rozmiaru bazy danych oraz spadku wydajności przy modyfikowaniu danych z tej tabeli. W skrajnych przypadkach tworzymy przecież kopie poszczególnych wierszy na stronach indeksu, a co za tym idzie liczba operacji wejścia/wyjścia staje się zbliżona do tej potrzebnej do skanowania tabeli czy indeksu zgrupowanego.

4 PLANY WYKONANIA ZAPYTANIA

Gdy zlecamy serwerowi wykonanie zapytania, rozpoczyna się dość złożony proces prowadzący do określenia sposobu realizacji zapytania. Zależnie od konstrukcji samego zapytania, rozmiarów tabel, istniejących indeksów, statystyk itp. serwer tworzy kilka planów wykonania zapytania. Następnie spośród nich wybierany jest ten, który cechuje się najniższym kosztem wykonania (wyrażanym przez koszt operacji wejścia/wyjścia oraz czasu procesora). Tak wybrany plan jest następnie kompilowany (przetwarzany na postać gotową do wykonania przez silnik bazodanowy) i przechowywany w buforze, w razie gdyby mógłby się przydać przy kolejnym wykonaniu podobnego zapytania. W ramach tego punktu zajmiemy się nieco dokładniej procesem wykonania zapytania przez SQL Server.

Cały proces, przebiegający od momentu przekazania zapytania do wykonania i odebrania jego rezultatów, jest dość złożony i może stanowić temat niejednego wykładu. Postaramy się choć z grubsza zasygnalizować najistotniejsze etapy tego procesu.

- **Parsowanie zapytania**. Polega na zweryfikowaniu składni polecenia, wychwyceniu błędów i nieprawidłowości w jego strukturze. Jeżeli takie błędy nie występują, to efektem parsowania jest tak zwane **drzewo zapytania** (postać przeznaczona do dalszej obróbki).
- **Standaryzacja zapytania**. Na tym etapie drzewo zapytania jest doprowadzane do postaci standardowej – usuwana jest ewentualna nadmiarowość, standaryzowana jest postać podzapytań itp. Efektem tego etapu jest ustandaryzowane drzewo zapytania.
- **Optymalizacja zapytania**. Polega na wygenerowaniu kilku planów wykonania zapytania oraz przeprowadzeniu ich analizy kosztowej zakończonej wybraniem najtańszego planu wykonania.
- **Kompilacja.** To przetłumaczenie wybranego planu wykonania do postaci kodu wykonywalnego przez silnik bazodanowy.
- **Określenie metod fizycznego dostępu do danych.** To skanowanie tabel, skanowanie indeksów, wyszukiwanie w indeksach itp.

Proces optymalizacji zapytania składa się z kilku etapów. W ich skład wchodzą: analizowanie zapytania pod kątem kryteriów wyszukiwania i złączeń, dobranie indeksów mogących wspomóc wykonanie zapytania oraz określenie sposobów realizacji złączeń. W ramach realizacji poszczególnych etapów optymalizator zapytań może korzystać z istniejących statystyk indeksów, generować je dla wybranych indeksów lub wręcz tworzyć nowe indeksy na potrzeby wykonania zapytania. Efektem tego procesu jest plan wykonania o najniższym koszcie, który jest następnie przekazywany do kompilacji i wykonania. Plan wykonania dla zapytania można podejrzeć w formie tekstowej, XML bądź zbioru wierszy. Realizuje się to za pomocą ustawienia na „ON” jednej z opcji SHOWPLAN_TEXT, SHOWPLAN_XML, SHOWPLAN_ALL. SQL Server, a właściwie narzędzie SQL Server Management Studio, umożliwia podejrzenie graficznej reprezentacji planu wykonania dla zapytania

Opcja prezentacji graficznej postaci planu wykonania dla zapytania jest dostępna w dwóch wariantach: *Estimated Execution Plan* oraz *Actual Execution Plan*. Pierwszy z nich polega na wygenerowaniu planu wykonania dla zapytania bez jego wykonywania. Powoduje to, że część informacji w planie wykonania jest szacunkowa lub jej brakuje (np. liczba wierszy poddanych operacjom, liczba wątków zaangażowanych w wykonanie itp.). Zaletą tego wariantu jest na pewno szybkość działania. Jest to szczególnie odczuwalne przy zapytaniach, które wykonują się dłużej niż kilkanaście sekund.

Drugi wariant zawiera pełne dane na temat wykonania zapytania. Jest on zawsze wiarygodny i mamy gwarancję, że dokładnie tak zostało wykonane zapytanie. W praktyce lepiej jest pracować z faktycznymi planami wykonania, chyba że czas potrzebny na ich uzyskanie jest przeszkodą.

Na diagramach reprezentujących plany wykonania zapytań może znajdować się kilkadziesiąt różnych symboli graficznych reprezentujących różne operatory (logiczne i fizyczne) oraz przebieg wykonania zapytania. Nie sposób omówić ich choćby pobieżnie w ramach tego wykładu.

Wśród całej gamy informacji wyświetlanych w szczegółach wybranego operatora, dla nas najistotniejsze są te, związane z kosztem wykonania danego etapu. W dalszych przykładach będziemy się na nich opierać prezentując zmiany kosztu wykonania zapytania w zależności od podjętych kroków przy optymalizacji zapytania.

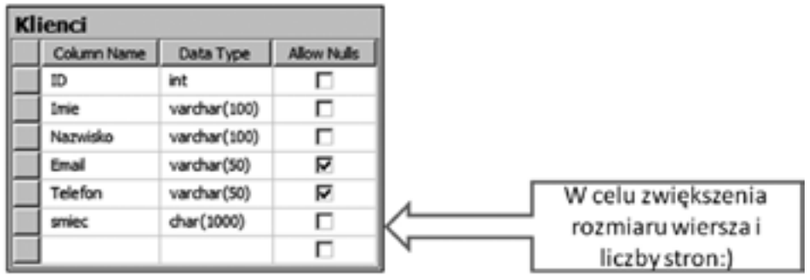
5 STATYSTYKI

Sam fakt istnienia takiego czy innego indeksu nie powoduje, że od razu staje się on kandydatem do skorzystania w ramach realizacji zapytania. W trakcie optymalizacji zapytania potrzebne są dodatkowe informacje na temat indeksów – **statystyki indeksów**. Sensowność skorzystania z indeksu można ocenić tylko w połączeniu z informacjami o liczbie wierszy w tabeli oraz o rozkładzie wystąpień poszczególnych wartości lub zakresów wartości w danych zawartych w kolumnie. Przykładowo mamy tabele klientów, w której 80% klientów nosi nazwisko Kowalski, a jedynie dwóch Nowak. Na podstawie samego faktu istnienia indeksu na kolumnie Nazwisko trudno ocenić, czy sensownie jest go wykorzystać przy wyszukiwaniu Kowalskich lub Nowaków. Po przejrzeniu statystyk może okazać się, że dla Kowalskiego nie ma co zaprzątać sobie głowy indeksami, natomiast w przypadku Nowaka może to znacznie poprawić wydajność.

Ponieważ dane zawarte w tabelach zwykle się zmieniają (pojawiają się nowe, istniejące są modyfikowane lub usuwane), istotne jest także aktualizowanie statystyk. Optymalizator zapytań podejmujący decyzje na podstawie nieaktualnych statystyk działa jak pilot samolotu, któremu przyrządy pokładowe pokazują wskazania sprzed 5 minut. Skutki mogą być opłakane. Z tego powodu, jeżeli mamy do czynienia z sytuacją, gdy do tej pory zapytanie wykonywało się zadowalająco szybko, a nagle wydajność spadła, pierwszym krokiem do wykonania jest właśnie uaktualnienie statystyk. Warto o tym pamiętać, bo może to nam oszczędzić sporo czasu.

6 OPTYMALIZACJA PRZYKŁADOWEGO ZAPYTANIA

Przejdźmy teraz do kilku przykładów wykonywania zapytań przy różnych kombinacjach istniejących indeksów. Za każdym razem spróbujemy przyjrzeć się kosztom wykonania zapytania i szczegółom przyjętych planów wykonania. Na kolejnych przykładach postaramy się zademonstrować wpływ indeksów na plan wykonania zapytania i jego całkowity koszt. Zapytania będą dotyczyły tabeli Klienci (rys. 9), w której nie ma żadnego indeksu.

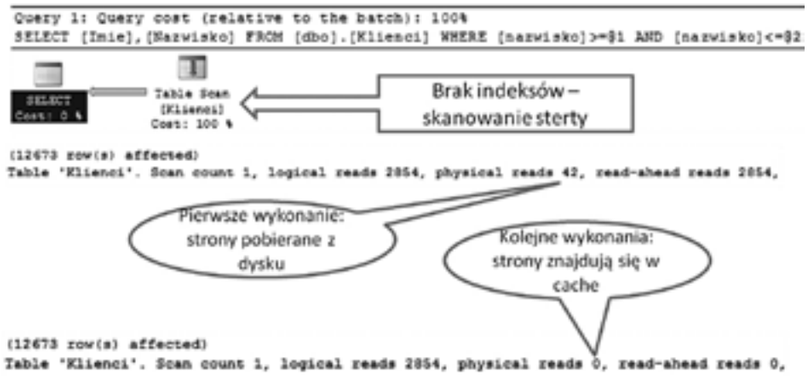


Rysunek 9.
Przykładowa tabela – Klienci

Z tego powodu można przewidywać, że operacją wykorzystaną do realizacji zapytania będzie skanowanie tabeli. Przykładowe zapytanie ma postać:

```
SELECT
  Imie
 ,Nazwisko
FROM
  dbo.Klienci
WHERE
  nazwisko BETWEEN 'F' AND 'T'
```

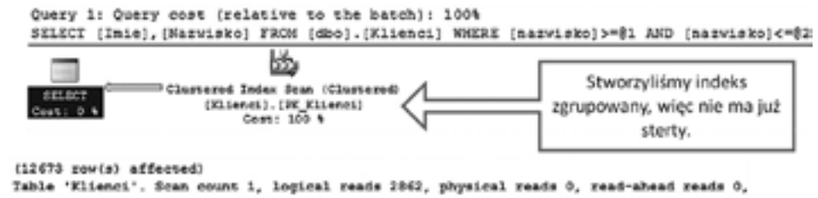
Po dwukrotnym wykonaniu zapytania uzyskaliśmy rezultaty, jak na rysunku 10.



Rysunek 10.
Efekty wykonania przykładowego zapytania

Zgodnie z oczekiwaniami, do realizacji zapytania została wykorzystana operacja skanowania tabeli. Przy pierwszym wykonaniu konieczne było pobranie stron danych z dysku (liczba fizycznych odczytów większa od 0). Każde następne wykonanie korzysta już ze stron umieszczonych w pamięci cache, czego przejawem jest zerowa wartość fizycznych odczytów. Całkowity koszt zapytania realizowanego według tego planu jest równy 2,1385.

Pierwszym etapem naszych działań jest utworzenie indeksu zgrupowanego na kolumnie ID. Nie przyczyni się to w znaczącym stopniu do zwiększenia wydajności, ale spowoduje zmianę planu wykonania. Skoro utworzenie indeksu zgrupowanego powoduje fizyczne uporządkowanie stron danych (i likwidacje sterty), to plan wykonania powinien zawierać wykonanie innej operacji niż skanowanie tabeli. Po wykonaniu zapytania stwierdzamy, że faktycznie tak jest (patrz rysunek 11).



Rysunek 11.
Efekty wykonania przykładowego zapytania po utworzeniu indeksu zgrupowanego

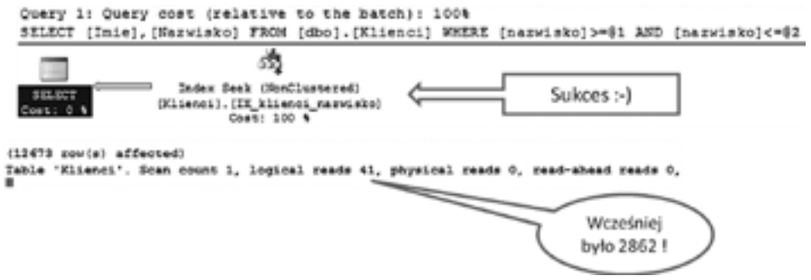
Tym razem serwer skorzystał z operacji skanowania indeksu zgrupowanego. Nie jest to żaden skok wydajnościowy, bo i tak przejrzane muszą być wszystkie strony danych, gdyż nasze kryterium wyszukiwania nie jest kolumną zawartą w indeksie.

Rozpocznijmy teraz działania ukierunkowane na obniżenie kosztów realizacji zapytania. Pierwszy pomysł – stwórzmy indeks niezgrupowany na kolumnie Nazwisko, która jest wykorzystywana jako kryterium wyszukiwania. Powinno to spowodować wykorzystanie tego indeksu do wyszukania wierszy z nazwiskami z określonego przez nas zakresu. Niestety po wykonaniu zapytania doznaliśmy rozczarowania – patrz rysunek 12.

Rysunek 12.
Efekt utworzenia indeksu niezgrupowanego

Plan wykonania się nie zmienił! Dlaczego? Z powodu umieszczenia na liście kolumn wyjściowych kolumny Imie. Optymalizator zapytań stwierdził, że mimo istnienia indeksu niezgrupowanego na kolumnie, po której wyszukujemy, nie warto z niego korzystać, gdyż i tak trzeba sięgnąć do stron danych, żeby pobrać wartości kolumny Imie. Z tego powodu plan wykonania nie uległ zmianie.

Skoro przemyśleliśmy już mechanizm działania zapytania i role indeksu, doprowadźmy sprawę do końca. Usuńmy istniejący indeks niezgrupowany, utwórzmy go na nowo z dodaną kolumną Imie. Po wykonaniu zapytania po raz kolejny, okazuje się, że tym razem indeks został wykorzystany (patrz rys. 13).



Rysunek 13.
Przykład wykorzystania indeksu

Odpowiednie wiersze spełniające kryteria wyszukiwania zostały zlokalizowane bardzo łatwo dzięki indeksowi niezgrupowanemu. Dodatkowo nie było konieczności sięgania do stron danych, gdyż indeks zawierał także kolumnę Imie, która była potrzebna do realizacji zapytania. Efekt jest widoczny. Koszt realizacji zapytania spadł z 2,1385 do 0,0453!

Po zakończeniu „walki" z optymalizacją prostego zapytania przez utworzenie odpowiednich indeksów można zdać sobie sprawę, iż wykonywanie tego typu operacji na prawdziwych bazach danych jest procesem złożonym i żmudnym. Do tego często nie da się pogodzić ze sobą wydajności dwóch lub więcej zapytań, bo każda poprawa wydajności w jednym psuje wydajność drugiego. Dodatkowo każdy kolejny indeks to dodatkowy problem z jego utrzymaniem oraz więcej czynności do wykonania przy modyfikacji danych. Jak sobie z tym radzić? Nie ma jednej sprawdzonej i zawsze działającej recepty. Są pewne podejścia umożliwiające realizację czynności w określonym porządku, co może się przyczynić do uniknięcia błędów lub ułatwienia wychwycenia typowych problemów. Zawsze jednak optymalizowanie wydajności pozostanie po części sztuką :-).

7 NARZĘDZIA WSPOMAGAJĄCE OPTYMALIZACJĘ

Przy optymalizowaniu zapytań trzeba brać pod uwagę mnóstwo czynników. Jeśli dodać do tego pracę z wieloma zapytaniami, to szybko wyłania się obraz ogromu pracy do wykonania. Na szczęście istnieją narzędzia, które mogą choć trochę wspomóc nasze wysiłki. Narzędzie Database Engine Tuning Advisor jest w stanie

wygenerować i wykonać wiele czynności prowadzących do podniesienia wydajności bazy danych. Proces ten jest realizowany rzecz jasna w kontekście konkretnych zapytań, gdyż nie ma możliwości optymalizowania pod kątem dowolnych zapytań.

Punktem wejścia do procesu automatycznej optymalizacji jest określenie zapytań, które są wykonywane na bazie wraz z określeniem częstotliwości ich wykonywania. Najłatwiej zrobić to w ramach monitorowania działania aplikacji. Za pomocą narzędzia SQL Profiler można zebrać tzw. **ślad** zawierający informacje o wszystkich wykonywanych na bazie zapytaniach. Plik z takimi informacjami może stanowić dane wejściowe dla Database Engine Tuning Advisora. Na ich podstawie narzędzie jest w stanie określić zapytania najistotniejsze dla funkcjonowania aplikacji i skupić się na optymalizowaniu pod ich kątem. Narzędzie zawiera wiele opcji umożliwiających sterowanie procesem optymalizacji. Można na przykład określić zbiór mechanizmów, które mają być wykorzystane do zwiększenia wydajności (indeksy, widoki indeksowane itp.). Można również określić, czy optymalizacja ma pozostawić istniejące indeksy bez zmian, czy „zaorać" je i zaplanować wszystkie od początku.

Rezultatem pracy narzędzia jest lista poleceń do wykonania na bazie danych (służą one do tworzenia zaplanowanych indeksów, usuwania niepotrzebnych itp.). To co jest istotne, to fakt, że przedstawiony przez narzędzie plan z reguły przyczynia się do podniesienia wydajności. Często można na tym zakończyć dalsze prace. Jeśli jednak mamy więcej pomysłów na zwiększenie wydajności, to wynik prac narzędzia zawsze można traktować jako dobry punkt wyjścia do dalszej analizy prowadzonej już „ręcznie".

W ramach tego wykładu zaledwie rozpoczęliśmy omawianie zagadnień związanych z optymalizacją zapytań i optymalizacją wydajności SQL Servera jako taką. Celem było przedstawienie pewnych podstawowych zagadnień i mechanizmów niezbędnych do zrozumienia podstawowych zasad rządzących w dziedzinie sposobów realizacji zapytań przez SQL Server.

LITERATURA

1. Ben-Gan I., Kollar L., Sarka D., *MS SQL Server 2005 od środka. Zapytania w języku T-SQL*, APN PROMISE, Warszawa 2006
2. Delany K., *MS SQL Server 2005 od środka. Dostrajanie i optymalizacja zapytań*, APN PROMISE, Warszawa 2008
3. Rizzo T., Machanic A., Dewson R., Walters R., Sack J., Skin J., *SQL Server 2005*, WNT, Warszawa 2008
4. Vieira R., *SQL Server 2005. Programowanie. Od Podstaw,* Helion, Gliwice 2007

Tworzenie interfejsów do baz danych z wykorzystaniem technologii ADO.Net

Andrzej Ptasznik
Warszawska Wyższa Szkoła Informatyki
aptaszni@wwsi.edu.pl

Streszczenie

W ramach wykładu zostaną przedstawione podstawy technologii ADO.Net, zapewniającej dostęp do baz danych, podstawy wielowarstwowej architektury aplikacji korzystających z baz danych, a także podstawowe cechy technologii Data Access Application Block, która umożliwia uproszczenie obsługi baz danych z poziomu aplikacji. Wśród poruszanych tematów znajdują się elementy LINQ (ang. *Language Integrated Query*). Poszczególne elementy technologii zostaną natomiast omówione z wykorzystaniem przykładów napisanych w języku C#. W czasie wykładu zostaną zaprezentowane również przykładowe rozwiązania wykorzystujące omawiane technologie.

Spis treści

1 WPROWADZENIE

We współczesnym świecie trudno wyobrazić sobie jakikolwiek system informatyczny, który nie korzystałby z jakiegoś źródła danych. W charakterze źródła danych występują zarówno proste pliki tekstowe, pliki XML, zasoby umieszczone w Internecie (RSS, Atom), jak i relacyjne bazy danych. Te ostatnie są szczególnie popularne ze względu na dalece bardziej zaawansowane możliwości manipulowania przechowywanymi danymi. Zależnie od technologii wykorzystanej do budowy aplikacji zmieniają się także sposoby uzyskiwania dostępu do baz danych. W ramach niniejszego artykułu zajmiemy się tym zagadnieniem w kontekście .NET Framework 3.5.

W kolejnych rozdziałach są opisane zagadnienia dotyczące planowania budowy aplikacji i doboru właściwej architektury do konkretnych wymagań. Wymienione są również typowe błędy popełniane na tym etapie tworzenia aplikacji, a także ich konsekwencje. W dalszej części zajmujemy się sposobami tworzenia kodu dostępu do danych z wykorzystaniem trzech różnych podejść. W efekcie możemy zobaczyć, jakie są charakterystyczne cechy poszczególnych rozwiązań, ich wady i zalety oraz w jakich projektach można je bez obaw stosować. Ze względu na ograniczony czas wykładu, naszym celem jest jedynie zasygnalizowanie problemów oraz wskazanie przykładowych rozwiązań. Pozwoli to uświadomić słuchaczom podstawowe zagadnienia dotyczące budowania kodu dostępu do baz danych oraz uczulić ich na najpospolitsze błędy popełniane przy okazji tworzenia aplikacji bazodanowych.

2 ARCHITEKTURA APLIKACJI BAZODANOWYCH

W ciągu kilkunastu ostatnich lat burzliwy rozwój systemów informatycznych sprawił, że stają się one bardziej skomplikowane i spełniają coraz to bardziej złożone i zaawansowane wymagania. Co za tym idzie ich tworzenie nastręcza większych trudności. W związku z tym coraz istotniejsze staje się podejście do procesu wytwarzania systemu informatycznego, które ma zapewnić minimalizowanie możliwości błędnego działania systemu, a także umożliwia rozwijanie i rozbudowywanie jego funkcjonalności wraz z pojawiającymi się i zmieniającymi wymaganiami.

Kolejnym zagadnieniem jest kwestia doboru architektury rozwiązania, adekwatnej do wymagań. Pomimo istnienia dużej liczby rozwiązań, szablonów, koncepcji oraz wzorców, nie ma architektury idealnej. Do rozwiązania konkretnego problemu można zastosować wiele podejść, natomiast każde z nich niesie ze sobą swoją specyfikę, która w znacznym stopniu może rzutować na to, jak sprawdzi się przy realizacji konkretnego przedsięwzięcia. Bardzo często w praktyce okazuje się, że zbyt pochopne przyjęcie architektury rozwiązania może powodować powstawanie na różnych etapach zaawansowania projektu nieprzewidzianych problemów, które zwykle powodują powstawanie opóźnień przy realizacji harmonogramu, lub wręcz wymuszają zmianę koncepcji projektu już w trakcie jego realizacji. Tego typu zdarzenia potrafią doprowadzić nawet do upadku projektu, co nie należy do rzadkości, jeśli weźmie się pod uwagę, że w opinii wielu specjalistów tylko ok. 20-30% projektów kończy się sukcesem.

Nie należy wierzyć w istnienie jednej, najlepszej i gwarantującej sukces architektury systemu, jak i samego procesu wytwórczego. Gdyby takowe istniały, to nie byłoby problemu z realizacją projektów. Warto wspomnieć o paradoksie Cobb'a:

We know why projects fail, we know how to prevent their failure
– so why do they still fail?
[Wiemy, dlaczego projekty upadają, wiemy jak zapobiec tym upadkom
– więc dlaczego one ciągle upadają?][1]

[1] Za: http://stakeholdermenagment.wordpress.com/2011/03/18/cobbs-paradox/, tłumaczenie autora.

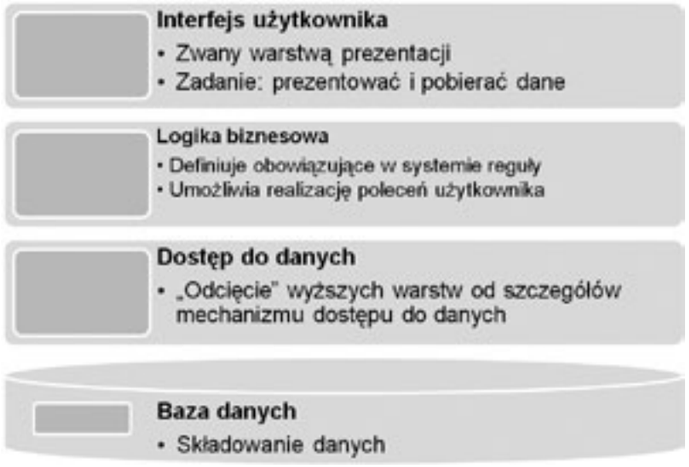
Zagadnienia związane z tym, jak podejść do realizacji konkretnego projektu, jaką zastosować metodykę, jaką architekturę rozwiązania są na tyle rozległe, że mogą stanowić temat na cały cykl publikacji, z których każda będzie przekonywać o tym, że ta konkretna metoda jest najlepsza.

Istotne także są względy ekonomiczne. Zwykle jednym z zasobów, którego jest wiecznie za mało w projekcie, jest czas. Nie zawsze da się poświęcić odpowiednią ilość czasu na dopracowanie szczegółów architektury, gdyż trzeba zaczynać realizację systemu, żeby udało się go ukończyć w rozsądnym terminie. W takiej sytuacji szuka się kompromisu i upraszcza niektóre aspekty architektury, zyskując na czasie, co jednak potrafi się okrutnie zemścić na dalszych etapach realizacji i rozwoju systemu.

W czasie tego wykładu skupimy uwagę jedynie na niewielkim wycinku architektury systemu, jakim jest kod dostępu do baz danych. Jest to jednak na tyle istotny wycinek, że z pewnością zasługuje na dokładniejsze omówienie. Dobrze zaprojektowany i zaimplementowany system jest w stanie zapewnić aplikacji wysoką stabilność i wydajność oraz odporność na zmiany wymagań (w postaci łatwości nanoszenia zmian). Dla odmiany, zaprojektowany źle – potrafi pogrzebać cały projekt.

3 ARCHITEKTURA WIELOWARSTWOWA

Typowa struktura aplikacji bazodanowej może zostać opisana modelem warstwowym, jak na rysunku 1.



Rysunek 1.
Typowa architektura aplikacji bazodanowej

Istotnym mechanizmem w ramach tej architektury jest sposób, w jaki warstwy komunikują się między sobą. Polega to na tym, że wyższa warstwa komunikuje się jedynie z leżącą bezpośrednio pod nią – czyli warstwa prezentacji korzysta z warstwy logiki biznesowej, a nie ma możliwości sięgania do warstw niższych, a nawet nie wie nic o ich istnieniu. Podobnie warstwa biznesowa korzysta z warstwy dostępu do danych nie wiedząc nic na temat samej bazy danych. Warstwa dostępu do danych korzysta z bazy danych i przekazuje jej polecenia do wykonania oraz odbiera wyniki.

Role poszczególnych warstw są ściśle określone i można je ogólnie opisać następująco:

- Warstwa prezentacji:
  - jej celem jest przedstawianie danych użytkownikowi oraz umożliwienie mu modyfikowania tych danych, a także sterowania zachowaniem aplikacji;
  - może zawierać tzw. logikę aplikacji (np. regułę, że jeżeli opcja x jest zaznaczona, to użytkownik nie widzi pól y i z).
- Warstwa logiki biznesowej:
  - zawiera kod odpowiedzialny za realizację poszczególnych operacji biznesowych (np. złożenie zamówienia, przyznanie rabatu itp.);
  - zawiera także reguły biznesowe (np. pracownik może przyznać rabat do 5%, a za zgodą menedżera do 20%, kwota na fakturze musi być większa od zera).
- Warstwa dostępu do danych:
  - zawiera kod realizujący operacje komunikowania się z bazą danych i przekazywania jej poleceń pobrania/dodania/modyfikacji/usuwania danych;
  - może zawierać także mechanizmy umożliwiające konfigurowanie dostępu do danych (wybór serwera, sposobu łączenia się z bazą itp.).
- Warstwa danych:
  - zwykle jest to serwer baz danych;
  - baza danych zawiera tabele;
  - baza może także zawierać widoki, procedury składowane, funkcje użytkownika, wyzwalacze, które służą do ukrycia szczegółów implementacyjnych bazy przed aplikacjami z niej korzystającymi, bądź do zaimplementowania części lub całości logiki biznesowej.

Zależnie od stopnia złożoności projektu, architektura może ulegać modyfikacjom polegającym na wchłanianiu wewnętrznych warstw przez skrajne. W takim przypadku warstwa logiki biznesowej bywa wplatana w kod warstwy prezentacji lub wprost w bazę danych (w postaci procedur składowanych, widoków, funkcji użytkownika i wyzwalaczy). Podobnie warstwa dostępu do danych może przestać być osobnym tworem i zostaje pofragmentowana i wkomponowana w kod warstwy prezentacji.

Takie zjawisko prowadzi jednak do powstawania aplikacji bardzo trudnych w utrzymaniu i rozwijaniu. Co z tego, że mamy jasno sformułowane wymaganie, skoro jego implementacja w aplikacji jest podzielona na kilka fragmentów osadzonych w różnych metodach obsługi zdarzeń, ewentualnie mamy do czynienia z powielaniem tego samego kodu w kilku miejscach. To wszystko razem powoduje, że bardzo trudno jest oszacować, ile czasu będzie potrzeba na realizację zadania, a także zapewnić stabilne i poprawne działanie aplikacji.

4 PLANOWANIE APLIKACJI BAZODANOWEJ

Przy planowaniu aplikacji bazodanowej, a w szczególności jej warstwy biznesowej i dostępu do danych, bardzo ważne jest wcześniejsze zebranie wymagań. Znakomicie ułatwia to proces definiowania operacji, które będą wykonywane na danych, co z kolei staje się podstawą do zdefiniowania interfejsu warstwy dostępu do danych. Interfejs ten powinien zawierać metody, które są niezbędne do wykonania wszystkich wynikających z wymagań operacji na danych. Interfejs ten będzie odtąd pełnił rolę kontraktu zawieranego pomiędzy warstwą biznesową a warstwą dostępu do danych i będzie definiował listę operacji, które będą udostępniane przez warstwę dostępu do danych, a z których będzie korzystała warstwa biznesowa.

Kolejnym krokiem jest zaprojektowanie encji biznesowych – klas, które będą nośnikiem danych. Będą zawierały wszystkie cechy informacyjne wynikające z wymagań. Dane pobierane z bazy będą przetwarzane do postaci encji biznesowych, a na nich z kolei będą operować wyższe warstwy. Przy tej okazji warto wspo-

mnieć o narzędziach, które w większym lub mniejszym stopniu automatyzują generowanie klas encji biznesowych oraz mechanizmu zasilania ich danymi z bazy oraz przenoszenia modyfikacji z encji do bazy. Jedno z takich narzędzi zostanie opisane w dalszej części rozdziału.

5 PODSTAWY ADO.NET

W systemie .NET Framework 3.5 biblioteką odpowiedzialną za udostępnianie klas, służących do organizowania dostępu do danych, jest ADO.NET. W jej ramach dostajemy do dyspozycji sporą liczbę klas, które umożliwiają realizowanie nawet bardzo złożonych mechanizmów współpracy z bazą danych. Dodatkowo, całość jest zaprojektowana w sposób ułatwiający rozszerzanie istniejących mechanizmów o nowe implementacje bez konieczności znacznych modyfikacji kodu aplikacji (model Dostawców – ang. *Providers*). Z grubsza polega on na zdefiniowaniu ogólnych interfejsów dla usług oferowanych przez dostawcę (np. połączenie z bazą powinno umożliwiać otwarcie połączenia oraz jego zamknięcie, a także zwracać informacje o jego aktualnym stanie – interfejs IDbConnection zawiera zdefiniowane metody Open(), Close() oraz właściwość State).

Każdy mechanizm służący do nawiązywania połączenia z konkretną bazą danych zawiera klasy implementujące interfejsy dostawcy. Standardowo .NET Framework zawiera dostawców:

- ODBC Data Provider;
- OLEDB Data Provider;
- SQLClient Data Provider.

Każdy z nich zawiera zestaw klas umożliwiających dostęp do różnych rodzajów źródeł danych:

- ODBC – dostęp do źródeł z wykorzystaniem ODBC (ang. *Open DataBase Connectivity*);
- OLEDB (ang. *Object Linking and Embedding, Database*) – następca ODBC, ma większe możliwości;
- SQLClient – dedykowany dostawca dla SQL Servera.

Jeżeli musimy się łączyć z inną bazą danych lub dowolnym innym źródłem – wystarczy uzyskać odpowiedniego dostawcę od producenta bazy (np. Oracle) lub wręcz napisać kod własnego dostawcy danych. Korzystać z niego będziemy dokładnie tak samo jak z innych.

.NET Framework 3.5 oferuje wiele interfejsów pomocnych przy komunikowaniu się z bazami danych. Klasy implementujące te interfejsy pełnią następujące role:

- IDbConnection – odpowiedzialna za zdefiniowanie i zarządzanie połączeniem z bazą danych;
- IDbCommand – odpowiedzialna za zbudowanie polecenia, które będzie wysłane do bazy danych za pośrednictwem połączenia;
- IDataReader – umożliwia odbieranie rezultatu wykonania polecenia przez bazę danych;
- IDbParameter – ułatwia definiowanie parametrów polecenia przekazywanego do bazy danych, lub odbierania wartości parametrów wyjściowych;
- IDataAdapter – umożliwia zdefiniowanie operacji CRUD (*Create, Read, Update, Delete*) dla określonej tabeli w bazie danych. W jej skład wchodzą cztery zestawy klas implementujących interfejsy IDbConnection oraz IDbCommand, które odpowiadają operacjom wykonywanym na danych;
- DataSet – Uniwersalny nośnik danych – umożliwia zdefiniowanie modelu danych (encje, relacje między nimi). Ma szerokie możliwości manipulowania zawartymi w sobie danymi oraz śledzenia zmian. Zwykle jest napełniany i obsługiwany za pomocą DataAdaptera. Ze względu na rozbudowaną funkcjonalność nie jest zbyt wydajny.

6 TYPOWE SCENARIUSZE DOSTĘPU DO BAZ DANYCH

W dalszych rozważaniach przyjmiemy założenie, że pracujemy z dostawcą SQLClient. Proces uzyskania dostępu do danych składa się z kilku etapów. Można je opisać w następujący sposób:

1. Utworzenie obiektu SqlCommand i przekazanie mu ciągu definiującego połączenie z bazą (ang. *connection string*).
2. Utworzenie obiektu SqlCommand, reprezentującego polecenie do wykonania. Może to być polecenie DML, DDL, DCL lub wywołanie procedury składowanej. Jeżeli to konieczne, należy do stworzonego obiektu dodać definicje parametrów wykonania polecenia.
3. Otwarcie połączenia – wykonanie metody Open() obiektu SqlCommand.
4. Wykonanie polecenia za pośrednictwem obiektu SqlConnection skojarzonego z SqlCommand. Istnieją trzy warianty wykonania polecenia:
   - ExecutenonQuery() – stosowane, gdy nie spodziewamy się zwrotnie zbioru rekordów (ang. *recordset*).
   - ExecuteScalar() – używane, gdy zwrotnie dostajemy zbiór rekordów zawierający jeden rekord z jedną kolumną. Upraszcza mechanizm „wyłuskania" zwróconej wartości.
   - ExecuteReader() – wykorzystane, gdy spodziewamy się zwrotnie zbioru (lub zbiorów) rekordów. Metoda ta zwraca obiekt SqlDataReader, który można traktować jako prosty kursor read and forward only, czyli umożliwiający iterowanie po kolejnych rekordach ze zbioru zwróconego przez polecenie. Umożliwia także na przejście do kolejnego zbioru rekordów (o ile polecenie spowodowało zwrócenie takowego).
5. Przetworzenie wyników polega zwykle na utworzeniu pętli działającej dla każdego rekordu zwróconego w zbiorze wynikowym.
6. Istotnym zagadnieniem jest zamknięcie połączenia z bazą, gdyż w przeciwnym przypadku można szybko doprowadzić do niepotrzebnego zużycia zasobów i utrzymywania niepotrzebnych już połączeń.

Drugim typowym scenariuszem jest korzystanie z obiektu DataSet i DataAdapter. Odbywa się to za pomocą dwóch metod klasy SqlDataAdapter – Fill() i Update(). Wewnętrznie jednak korzystają one z tego samego podstawowego scenariusza (stosującego SqlDataReader), który został opisany wcześniej.

7 IMPLEMENTACJE KOMPONENTÓW DOSTĘPU DO BAZ DANYCH

Opisany w poprzednim rozdziale scenariusz uzyskania dostępu do danych można zaimplementować w najprostszej postaci, jak pokazano na rysunku 2.

```
string connectionString = @"server=.\sql2008;database=schooldatabase;integrated security=true";
string sqlText = "SELECT * FROM Student ORDER BY LastName";

SqlConnection connection = new SqlConnection(connectionString);
SqlCommand command = new SqlCommand(sqlText, connection);
connection.Open();
SqlDataReader dr = command.ExecuteReader();
while (dr.NextResult())  //przejście do kolejnego recordsetu
{
    while (dr.Read())
    {
        // ...przetworzenie rekordu.....
    }
}
connection.Close();
```

Rysunek 2.

Należy zauważyć, że nie jest to dobry przykład, jak w praktyce wykonywać operacje na bazie danych! Ten kod nie uwzględnia możliwości wystąpienia błędu, a w takim przypadku połączenie z bazą może pozostać otwarte, mimo że nie będziemy już z niego korzystać.

Kwestia sensownego przechowywania zawartości łańcucha połączenia i treści polecenia również nie jest tu istotna. W prawdziwych projektach tego typu informacje są przechowywane w plikach konfiguracyjnych aplikacji, często w postaci zaszyfrowanej. Zaprezentowany kod będzie poprawnie działał, jeżeli nie wystąpią żadne nieprzewidziane sytuacje i nie nastąpi błąd, który wygeneruje tzw. wyjątek. Każdy wyjątek spowoduje, że połączenie z bazą nie zostanie automatycznie zamknięte i będzie niepotrzebnie zużywać zasoby serwera.

W celu bezpiecznego skorzystania z opisywanego scenariusza należy nieznacznie zmodyfikować jego kod, aby uwzględnić możliwość wystąpienia błędu i w sposób bezpieczny zwolnić zaalokowane zasoby, patrz rysunek 3.

```
string connectionString = @"server=.\sql2008;database=schooldatabase;integrated sec
string sqlText = "SELECT * FROM Student ORDER BY LastName";

using (SqlConnection connection = new SqlConnection(connectionString))
{
    using (SqlCommand command = new SqlCommand(sqlText, connection))
    {
        connection.Open();
        SqlDataReader dr = command.ExecuteReader();

        while (dr.NextResult())  //przejście do kolejnego recordsetu
        {
            while (dr.Read())
            {
                // ...przetworzenie rekordu.....
            }
        }
    }
}
```

Rysunek 3.

Zastosowanie konstrukcji using gwarantuje, że w momencie wyjścia z bloku kodu:

```
using (Typ x = new Typ())
{
…
}
```

zostanie wywołana metoda Dispose() obiektu `x`, która zgodnie z przeznaczeniem powinna zawierać kod zwalniający zasoby używane przez obiekt `x`. W naszym przypadku będzie to zamknięcie połączenia z bazą. Metoda ta zostanie wywołana **ZAWSZE**. Nawet gdy wystąpi nieoczekiwany wyjątek. Wewnętrznie jest to realizowane poprzez zamianę bloku `using {…}` na blok `try{…} finally{…}`.

Wypadałoby jeszcze wspomnieć o kwestii zarządzania połączeniem z bazą danych. Tego typu zasoby są uznawane za dość kosztowne, więc należy zwrócić szczególną uwagę na sposób posługiwania się nimi. Mamy tu do czynienia z dwoma skrajnościami:

- Utworzenie połączenia, otwarcie go i trzymanie w tym stanie w trakcie działania aplikacji.
- Tworzenie połączenia za każdym razem, gdy chcemy przesłać polecenie do bazy, i niezwłoczne zamykanie go zaraz po odebraniu wyników.

Często uznaje się, że ciągłe otwieranie i zamykanie połączeń ma negatywny wpływ na wydajność ze względu na to, że utworzenie i nawiązanie połączenia z bazą jest dość kosztowne. Na szczęście SQL Server zawiera mechanizm (ang. *Connection Pooling*), który pozwala zapomnieć o problemach z wydajnością. Zamykane połączenie nie jest tak naprawdę niszczone tylko trafia do specjalnej puli, z której jest ponownie pobierane i wykorzystywane w razie potrzeby. Dzięki temu można przyjąć (dotyczy to szczególnie aplikacji typu WWW), że połączenie można śmiało tworzyć wyłącznie na czas wykonania polecenia. Do tego właśnie służy zaprezentowany kod.

Jak poradzić sobie w sytuacji, gdy trzeba zwrócić obiekt SqlDataReader do wykorzystania przez inne klasy? Co wtedy z połączeniem z bazą danych? W takim przypadku nie można zastosować konstrukcji using do zapewnienia bezpiecznego zamknięcia połączenia z bazą. Zastosowanie using powodowałoby, że SqlDataReader stałby się bezużyteczny, gdyż do pobierania kolejnych rekordów wymaga on otwartego połączenia z bazą. Jest to istotna cecha obiektu DataReader – potrzebuje on otwartego połączenia z bazą.

Aby móc bezpiecznie zamknąć połączenie po wykorzystaniu obiektu SqlDataReader w innej metodzie, należy skorzystać z parametru CommandBehavior.CloseConnection, który spowoduje automatyczne zamknięcie połączenia w momencie wywołania metody Close() lub Dispose() obiektu SqlDataReader.

```
string connectionString = @"server=.\sql2008;database=schooldatabase;integrated secu
string sqlText = "SELECT * FROM Student ORDER BY LastName";

SqlConnection connection = new SqlConnection(connectionString);
{
    using (SqlCommand command = new SqlCommand(sqlText, connection))
    {
        connection.Open();
        SqlDataReader dr = command.ExecuteReader(CommandBehavior.CloseConnection);
        return dr;
    }
}
```

Rysunek 4.

Często do polecenia wysyłanego do bazy trzeba wstawić wartości przekazane przez użytkownika. Można to zrealizować poprzez zbudowanie całego polecenia z fragmentów stałych przeplatanych wartościami podanymi przez użytkownika. Budowanie takiego polecenia wygląda na najprostszą metodę, lecz stanowi spore zagrożenie. Po pierwsze, przy bardziej złożonych poleceniach łatwo można popełnić błąd, który spowoduje, że serwer baz danych nie będzie mógł wykonać polecenia z powodu błędnej składni (niedomknięte apostrofy itp.). Także modyfikowanie zapytania stworzonego w ten sposób nastręcza wielu problemów.

Większym zagrożeniem są jednak ataki typu SQL Injection. Polegają one z grubsza na tym, że użytkownik aplikacji podaje specjalnie sformatowane wartości, które po wkomponowaniu w polecenie modyfikują jego działanie.

Aby ograniczyć ryzyko takich ataków, należy skorzystać z możliwości definiowania parametrów poleceń (SQLParameter). Polecenia tworzone w ten sposób mają postać szablonu z miejscami na wartości parametrów, do którego w następnej kolejności dodaje się kolekcje obiektów SQLParameter z przypisanymi do nich wartościami. Tak skonfigurowane polecenie wykonuje się analogicznie jak poprzednio (patrz rysunek 5).

```
string sqlText2 =
"INSERT INTO dbo.StudentNote (StudentID,SubjectID,NoteValue) VALUES ( @studentID,@subjectID,@noteValue)";

using (SqlConnection connection = new SqlConnection(connectionString))
{
    using (SqlCommand command = new SqlCommand(sqlText2, connection))
    {
        command.Parameters.Add("@studentID", SqlDbType.Int).Value = studentId;
        command.Parameters.Add("@subjectID", SqlDbType.Int).Value = subjectID;
        command.Parameters.Add("@noteValue", SqlDbType.Decimal).Value = noteValue;

        connection.Open();
        command.ExecuteNonQuery();
    }
```

Rysunek 5.

Najbardziej eleganckim rozwiązaniem tego problemu jest korzystanie z procedur składowanych. Upraszcza to tworzony kod oraz uodparnia go na zmiany w bazie (np. zmianę postaci zapytania). W takich przypadkach kod pozostaje niezmienny do momentu, w którym zmienia się parametry wywołania procedury składowanej.

Można wtedy dodać dodatkową warstwę abstrakcji, odcinającą kod dostępu do danych od szczegółów implementacyjnych samej bazy. Specjalnym efektem jest otrzymanie aplikacji, w której można część modyfikacji wprowadzić bez dokonywania jakichkolwiek zmian w kodzie – zmieniając jedynie kod procedur składowanych w bazie. Z drugiej strony – rosnąca ilość logiki zaszytej w bazie danych w postaci skomplikowanego kodu SQL powoduje wzrost nakładów pracy, potrzebnych na jego utrzymanie i weryfikowanie poprawności działania po naniesieniu modyfikacji. Przykładowy kod wywołujący procedurę składowaną może mieć postać, jak na rysunku 6.

```
string procedureName = "pAddStudentNote";

using (SqlConnection connection = new SqlConnection(connectionString))
{
    using (SqlCommand command = new SqlCommand(procedureName, connection))
    {
        command.CommandType = CommandType.StoredProcedure;

        command.Parameters.Add("@studentID", SqlDbType.Int).Value = studentId;
        command.Parameters.Add("@subjectID", SqlDbType.Int).Value = subjectID;
        command.Parameters.Add("@noteValue", SqlDbType.Decimal).Value = noteValue;

        connection.Open();
        command.ExecuteNonQuery();
    }
```

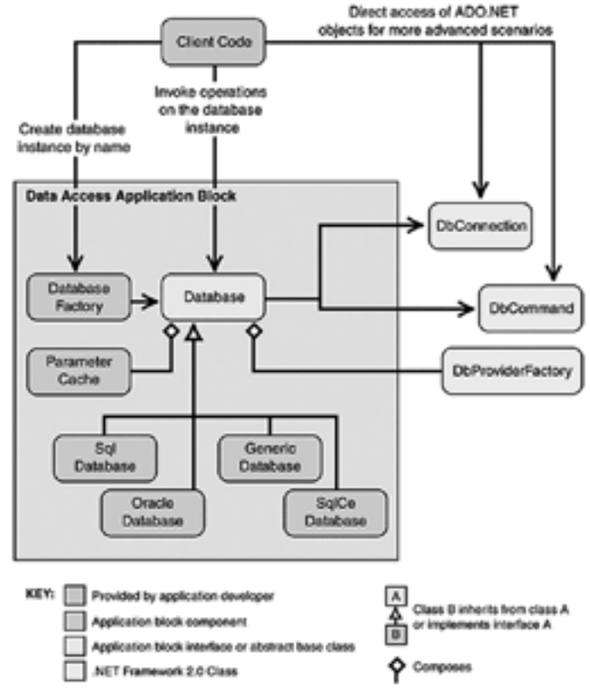
Rysunek 6.

W powyższym przykładzie zakładamy, że w bazie danych istnieje procedura składowana o nazwie pAddStudentNote, która wymaga podania wartości trzech parametrów @studentID, @subjectID oraz @noteValue. Istotne jest to, że z punktu widzenia aplikacji nieistotna jest wiedza o sposobie działania tej procedury i szczegółach jej operacji wykonywanych w bazie danych.

8 ZASTOSOWANIE DATA ACCESS APPLICATION BLOCK (DAAB)

Opisane w poprzednim rozdziale metody dostępu do danych dość dobrze ilustrują koncepcję, jaka przyświecała autorom ADO.NET, ale nie do końca nadają się do zastosowań praktycznych. Przede wszystkim ze względu na znaczną ilość kodu, który trzeba napisać, żeby zaimplementować pożądaną funkcjonalność.

Na szczęście istnieją inne rozwiązania, które choć wewnętrznie korzystają dokładnie z tych samych mechanizmów, to od strony programisty znacznie upraszczają pracę z bazą danych. Do tego typu rozwiązań należy Data Access Application Block z pakietu Enterprise Library. Jego ogólna koncepcja jest następująca (patrz rysunek 7):



Rysunek 7.
Schemat architektury DAAB [źródło: http://cdiban.wordpress.com/page/2/]

- kluczowym elementem jest obiekt typu Database, którego instancji nie tworzy sam programista, tylko specjalna klasa DatabaseFactory. To ona jest odpowiedzialna za odnalezienie w pliku konfiguracyjnym aplikacji informacji o sposobie łączenia się z bazą danych oraz o rodzaju bazy danych (SQL Server, Oracle itp.);
- obiekt typu Database udostępnia wiele metod służących do przekazywania poleceń do bazy danych i odbierania wyników;
- w razie potrzeby [bardziej wyrafinowane operacje na bazie: organizowanie transakcji, ustawianie parametrów połączenia (ang. *timeout*) i polecenia] można uzyskać dostęp do używanych wewnętrznie przez obiekt typu Database obiektów DbConnection, DbCommand;
- dodatkowo DAAB zawiera wbudowane mechanizmy powodujące wzrost wydajności (np.: cache dla obiektów SqlParameter wykorzystywanych przy wywoływaniu procedur i poleceń z parametrami).

Kod, który trzeba napisać, żeby wykonać polecenie bazodanowe przy użyciu DAAB, jest bardzo prosty i w sporej części przypadków składa się z jednego wiersza. Programista nie musi pilnować procesu nawiązywania połączenia z bazą oraz jego zamykania, a także obsługi błędów z tym związanych. Nie musi także definiować

parametrów poleceń – wystarczy przekazanie kolejnych wartości parametrów jako argumentów wywołania metody obiektu klasy Database. DAAB przed wykonaniem polecenia sam pobierze z bazy informacje na temat parametrów konkretnej procedury, stworzy odpowiednie obiekty SqlParameter i przypisze im wartości. Do tego utworzone obiekty zostaną umieszczone w pamięci cache i przy kolejnych wywołaniach tej procedury będą gotowe do użycia (wzrost wydajności). W tworzonym kodzie nie ma ŻADNYCH informacji o docelowej bazie danych i jej typie – zawiera on jedynie informacje na temat poleceń, które trzeba wykonać. Raz napisany i skompilowany kod może być wykorzystywany do komunikacji z dowolną bazą danych w dowolnej aplikacji. To właśnie aplikacja dostarcza informacji na temat typu i lokalizacji serwera baz danych, z którym będzie się komunikować. Przykładowy kod korzystający z DAAB jest przedstawiony na rysunku 8.

```
private Database database = DatabaseFactory.CreateDatabase("StudentNotesDBConnectionString");

public IDataReader GetStudents()
{
    return database.ExecuteReader(CommandType.Text, "SELECT * FROM Student ORDER BY LastName");
}

public void AddNote(int studentID, int subjectID, decimal noteValue)
{
    database.ExecuteNonQuery("pAddStudentNote", studentID, subjectID, noteValue);
}
```

Rysunek 8.

Na tym przykładzie łatwo zauważyć, jak bardzo zmniejsza się ilość tworzonego kodu. Dodatkowo programista koncentruje się na samym poleceniu i ewentualnych parametrach jego wywołania. Cała reszta jest załatwiana na poziomie konfigurowania aplikacji.

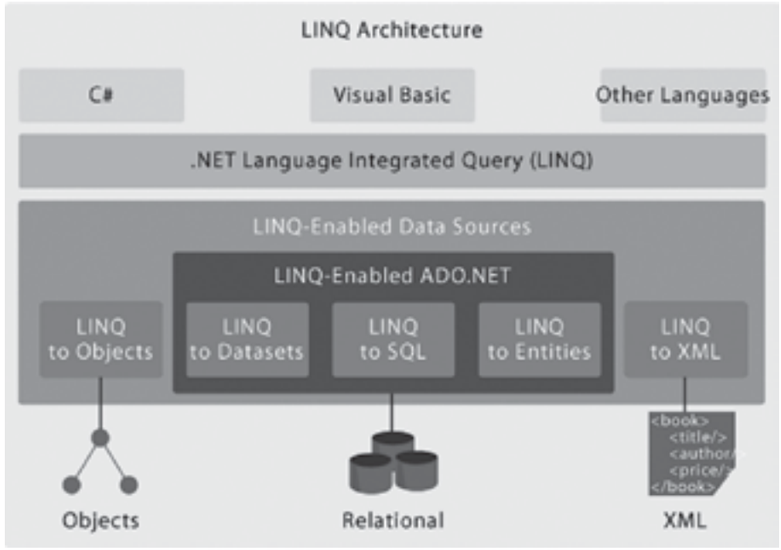
9 LINQ TO SQL

Przy komunikowaniu się z bazą danych zwykle występował problem budowania zapytań. Niezależnie od języka programowania, w którym tworzona była aplikacja, język zapytań do bazy (czy innego źródła danych) był zawsze odrębny i stanowił „wyspy” w kodzie aplikacji. To zjawisko dało się zauważyć w obu zaprezentowanych do tej pory przykładach w postaci poleceń SQL budowanych w mniej lub bardziej złożony sposób.

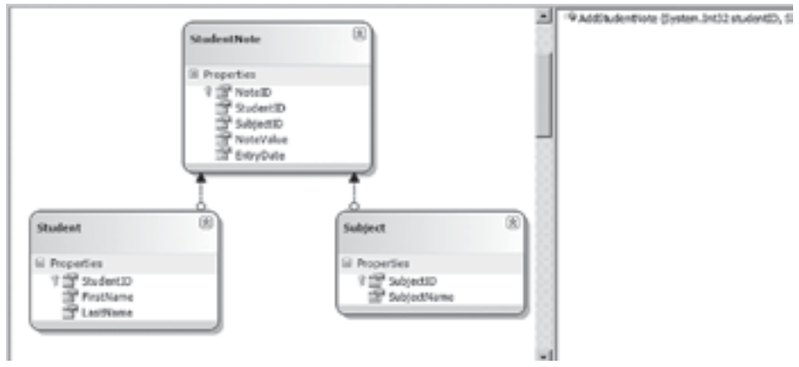
LINQ (ang. *Language Integrated Query*), jak sama nazwa wskazuje, stanowi odejście od tej koncepcji i wprowadza język budowania zapytań zintegrowany z językiem programowania. Występuje w kilku wariantach umożliwiających korzystanie z różnych źródeł danych (relacyjnych baz danych, XML, obiektów DataSet, innych obiektów).

W ramach wykładu skorzystamy z jednego z wariantów – LINQ to SQL, który można sklasyfikować jako O/RM (ang. *Object/Relational Mapping*). Rozwiązanie to znakomicie ułatwia i przyspiesza tworzenie warstwy dostępu do danych, a wręcz zwalnia programistę z tworzenia jakichkolwiek zapytań SQL.

Głównym elementem biblioteki stworzonej za pomocą LINQ to SQL jest **model**. Tworzy się go poprzez dodanie do projektu szablonu LINQ to SQL Classes. Dalsza praca z modelem polega na przeciąganiu i upuszczaniu obiektów z bazy danych (z narzędzia Server Explorer/Data Connections), co powoduje generowanie w tle potrzebnych klas. Modyfikacja modelu zwykle ogranicza się do dopasowania nazw wygenerowanych typów, uzupełnienia ich o dodatkowe metody i właściwości oraz skonfigurowania ich zachowania przy komunikacji z bazą danych (patrz rysunek 10).



Rysunek 9.
Architektura LINQ [źródło http://www.codeproject.com/KB/linq/UnderstandingLINQ.aspx]



Rysunek 10.
Tworzenie modelu LINQ

W wygenerowanym modelu można także tworzyć relacje pomiędzy encjami. Nie muszą one odpowiadać kluczom obcym w bazie. Można je budować w sposób dowolny. Klucze obce są natomiast podstawą do automatycznego wygenerowania relacji. Efektem istnienia relacji pomiędzy encjami jest pojawienie się w nich dodatkowej właściwości zawierającej referencję do encji znajdującej się na drugim końcu relacji. Przykładowo klasa Student będzie miała właściwość StudentNotes prowadzącą do listy ocen studenta, a każda ocena będzie miała (zależnie od tego czy sobie tego życzymy) referencję do encji Studenta, do którego ocena należy.

Utworzenie modelu daje efekt w postaci wygenerowanych klas. Podstawową jest klasa DataContext, której instancję tworzy się zawsze, gdy chce się skorzystać z danych. Należy zaznaczyć, że zawiera ona wiele dodatkowych metod, właściwości i zdarzeń pozwalających na operowanie na danych oraz śledzenie zmian.

```
public class StudentNotesDBLINQ
{
    private StudentNotesDataContext context = new StudentNotesDataContext();

    public IQueryable<Student> GetStudents()
    {
        return context.Students;
    }

    public void AddNote(int studentId, int subjectID, decimal noteValue)
    {
        context.AddStudentNote(studentId, subjectID, noteValue);
    }

    public decimal GetAverageNoteForStudent(int studentID)
    {

        Student selectedStudent = context.Students.Where(s => s.StudentID == studentID).SingleOrDefault();
        if (selectedStudent == null)
            return 0;
        return selectedStudent.StudentNotes.Average(sn => sn.NoteValue);
    }
```

Rysunek 11.

LINQ to SQL znakomicie upraszcza i przyspiesza proces tworzenia kodu organizującego dostęp do bazy danych. Korzysta z dobrych praktyk i jest zaprojektowane tak, aby zapewnić wysoką wydajność. Polecenia, które LINQ to SQL przekaże do bazy danych oraz momenty, w których dojdzie do takiego przekazania są dobierane w taki sposób, żeby zapewnić jak najlepszą wydajność oraz by sięgać do danych tylko, gdy są one bezpośrednio potrzebne. W powyższym przykładzie, wykonanie metody GetStudents() nie powoduje wykonania żadnego polecenia w bazie. Zwraca ono tylko obiekt, którego można dalej używać, wywołując na jego rzecz kolejne metody – tak jak ma to miejsce na przykładzie na rysunku 12:

```
public partial class StudentList : System.Web.UI.Page
{
    protected StudentNotesDBLINQ service = new StudentNotesDBLINQ();

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            studentsGrid.DataSource = service.GetStudents().ToList();
            studentsGrid.DataBind();
        }
    }

    protected void studentsGrid_Sorting(object sender, GridViewSortEventArgs e)
    {
        if (e.SortExpression == "LastName")
            studentsGrid.DataSource = service.GetStudents().OrderBy(s => s.LastName).ToList();
        if (e.SortExpression == "FirstName")
            studentsGrid.DataSource = service.GetStudents().OrderBy(s => s.FirstName).ToList();
        studentsGrid.DataBind();

    }
```

Rysunek 12.

Dopiero wykonanie metody ToList() powoduje scalenie wszystkich dotychczasowych wywołań i wygenerowanie polecenia SQL, które jest niezwłocznie przekazane do bazy.

Dzięki silnej kontroli typów w LINQ to SQL programiści mogą uniknąć popełniania błędów, które ujawnią się dopiero po uruchomieniu aplikacji. Przykładowo przy korzystaniu z SqlDataReadera, jeżeli popełnimy błąd literowy w nazwie kolumny lub wykonamy rzutowanie na niewłaściwy typ – efektem będzie powstanie wyjątku w trakcie działania aplikacji. W przypadku LINQ błąd pojawi się już na etapie kompilacji.

LINQ to SQL jest z powodzeniem stosowany przy tworzeniu prostych projektów RAD (ang. *Rapid Application Development*) dzięki swojej prostocie. Sprawdza się w większości prostych systemów. Ograniczenie do SQL Servera 2005 i 2008 zwykle nie powoduje problemów. Jeżeli natomiast trzeba łączyć się z innym serwerem baz danych lub chce się operować na bardziej abstrakcyjnym modelu, to można skorzystać ze „starszego brata" LINQ to SQL – ADO.NET Entity Framework. Jest on znaczniej bardziej skomplikowany, ale po nabraniu wprawy umożliwia równie szybkie tworzenie aplikacji.

PODSUMOWANIE

Zaprezentowane na tym wykładzie rozwiązania stanowią tylko część istniejących narzędzi służących do organizowania dostępu do danych. W Internecie można znaleźć wiele metod umożliwiających realizowanie tego celu w różny sposób. Są tam rozwiązania bardzo proste, ale nie brak też bardzo rozbudowanych i skomplikowanych. Każde z nich ma grupę swoich zwolenników i przeciwników, a Internet jest pełen informacji na temat tego, jak dobre/złe jest konkretne rozwiązanie.

Przy podejmowaniu decyzji o zastosowaniu takiego czy innego rozwiązania trzeba brać pod uwagę wiele czynników. Najważniejsze z punktu widzenia programisty są łatwość nauczenia się danego rozwiązania oraz dostępność dokumentacji i szeroko rozumianego wsparcia. Z punktu widzenia projektanta czy architekta brane pod uwagę są inne cechy – m.in. skalowalność, koszty, to czy projekt jest nadal rozwijany itd.
Wśród innych przykładów mechanizmów dostępu do danych można wymienić:

- **Subsonic** (http://subsonicproject.com)
- **nHibernate** (https://www.hibernate.org/343.html)
- **ADO.NET Entity Framework** (http://msdn.microsoft.com/en-us/library/bb399572.aspx)

Warto zapoznać się z ich możliwościami zanim podejmie się decyzję dotyczącą koncepcji mechanizmu organizowania dostępu do danych w realizowanym projekcie. Świadoma i przemyślana decyzja w tym zakresie może oszczędzić wielu godzin pracy programistom, a także przyczynić się w znacznym stopniu do sukcesu projektu.

LITERATURA

1. Chappell D., *Zrozumieć platformę .NET.*, Helion, Gliwice 2007
2. Matulewski J., Orłowski S., *ASP.Net i ADO.Net w Visual Web Developer,* Helion, Gliwice 2007
3. Matulewski J., *C# 3.0 i .Net 3.5 Technologia LINQ*, Helion, Gliwice 2008
4. Michelsen K., *Język C# Szkoła programowania*, Helion, Gliwice 2007
5. Vieira R., *SQL Server 2005. Programowanie. Od Podstaw,* Helion, Gliwice 2007

Hurtownie danych – czyli jak zapewnić dostęp do wiedzy tkwiącej w danych

Andrzej Ptasznik
Warszawska Wyższa Szkoła Informatyki
aptaszni@wwsi.edu.pl

Streszczenie

Przedmiotem wykładu są podstawy teorii hurtowni danych i aspekty ich wykorzystania. W pierwszej części zostaną omówione podstawowe cechy systemów OLTP (ang. *On-Line Transaction Processing*) oraz systemów OLAP (*ang. On-Line Analitycal Processing*). Omówione zostaną podstawowe pojęcia i przykłady projektów hurtowni danych. Przedstawione zostaną podstawowe zagadnienia związane z integracją danych oraz pojęcie analitycznej kostki wielowymiarowej. Zaprezentowane zostaną elementy technologii usług analitycznych i ich znaczenie w systemach typu *Business Intelligence*. W części końcowej wykładu omówione zostaną krótko podstawowe pojęcia związane z eksploracją danych (ang. *Data Mining*).

Spis treści

1 WPROWADZENIE

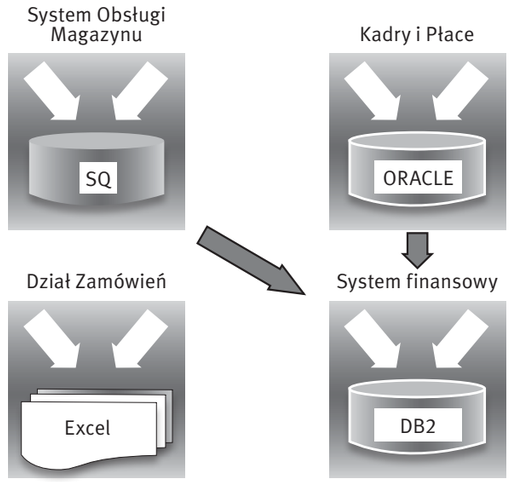
Burzliwy rozwój technologii informatycznych, a w szczególności baz danych, spowodował, że w każdej firmie czy instytucji gromadzone są różne dane na różnych etapach działalności. Bardzo często dane są gromadzone w różnorodny sposób – od plików tekstowych poprzez arkusze kalkulacyjne do baz danych. W okresie początkowego rozwoju systemy informatyczne wspomagające działalność firm koncentrowały się na wsparciu działalności operacyjnej. Powstawały rozmaite systemy ukierunkowane na konkretny aspekt działania, przykładowo:

- wystawianie faktur;
- obsługa magazynu;
- systemy kadrowe;
- systemy księgowe;
- obsługa klientów.

Zwykle systemy takie nie były z sobą w żaden sposób powiązane i tworzyli je różni producenci w odmiennych technologiach. Stosowanie technologii informatycznych w codziennej działalności firm i instytucji było związane z gromadzeniem danych na potrzeby konkretnego typu działania. Dane zbierane w różnych systemach, oprócz wspomagania codziennych działań, były wykorzystywane także do celów raportowania i informowania kierownictwa. Istniały jednak podstawowe problemy takiej działalności:

- dane po pewnym czasie stawały się niepotrzebne, ponieważ obsługa działalności codziennej nie musiała korzystać z danych historycznych (w systemie obsługi magazynu istotny był aktualny stan towaru w magazynie, a nie jaki był ten stan w zeszłym roku) – często w tego typu systemach usuwano starsze dane;
- wielokrotnie przetrzymywano te same dane w rożnych formatach;
- przetwarzanie danych na potrzeby inne niż wsparcie działalności codziennej znacząco wpływało na wydajność tych systemów.

Na rysunku 1 przedstawiony został schemat organizacji instytucji z wykorzystaniem różnych systemów informatycznych.



Rysunek 1.
Przykładowa organizacja firmy z wykorzystanie różnych systemów informatycznych

Duże ilości gromadzonych danych stają się kopalnią wiedzy, która może zostać wykorzystana do właściwego kierowania firmą i osiągnięcia przewagi konkurencyjnej na rynku.

2 SYSTEMY OLTP I OLAP

Tradycyjne systemy baz danych ukierunkowane są na realizację wielu małych i prostych zapytań i mają zapewnić wsparcie dla realizacji codziennych działań pracowników danej firmy lub instytucji. Dla tego typu systemów Edgar Frank „Ted" Codd (brytyjski informatyk, znany przede wszystkim ze swojego wkładu do rozwoju teorii relacyjnych baz danych) wprowadził pojęcie systemów **OLTP** (ang. *On-Line Transaction Processing*) i zdefiniował zbiór zasad, które powinny spełniać systemy tego typu. Podstawowe cechy systemów OLTP:

- przechowywane dane są zorientowane procesowo, np. wystawione faktury, otrzymane zamówienia, złożone reklamacje, wykonane przelewy itp.;
- stosunkowo niewielkie rozmiary baz danych (kilka gigabajtów);
- przechowywane są dane bieżące bez konieczności gromadzenia danych historycznych;
- realizowana jest duża liczba w miarę prostych zapytań;
- przechowywane są dane elementarne;
- realizowane są operacje wstawiania, modyfikowania i usuwania danych.

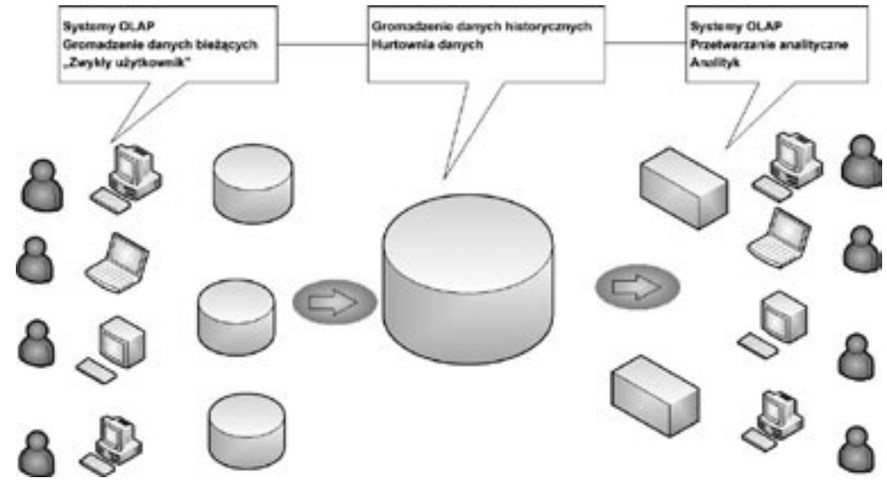
Zbiory danych tworzone w systemach OLTP stają się przydatne do pozyskiwania dodatkowych informacji potrzebnych kierownictwu firmy do podejmowania decyzji. Pojawiają się tu jednak pewne problemy:

- w ramach jednej firmy może istnieć wiele systemów typu OLTP;
- realizowanie dodatkowych czynności w ramach systemu OLTP wpływa na jego wydajność, tym bardziej że pozyskiwanie danych analitycznych wymaga wykonywania złożonych zapytań operujących na dużej liczbie danych;
- klasyczne zapytania SQL dostarczają danych w postaci dwuwymiarowych tabel, co często jest niewystarczające dla tego typu zastosowań.

Rozwiązaniem tych problemów stała się koncepcja wydzielonych systemów informatycznych świadczących usługi analityczne. Wspomniany wyżej Edgar Codd nazwał systemy tego typu **OLAP** (ang. *On-Line Analitycal Processing*) i również dla tych systemów sformułował zbiór zasad, które powinny spełniać. Podstawowe cechy systemów OLAP:

- przechowywane dane są zorientowane tematycznie, np. sprzedaż produktów, stany zapasów, wydatki, akcje promocyjne itp.;
- ogromne ilości gromadzonych danych (rzędu wielu terabajtów);
- przechowywane są dane bieżące i historyczne;
- realizowane są bardzo złożone zapytania operujące na wielkiej grupie danych;
- przechowywane są dane elementarne i zagregowane (sumy, średnie itp.);
- wykonywane są głównie operacje dopisywania nowych danych – praktycznie nie wykonuje się operacji modyfikowania danych.

Elementem łączącym systemy OLTP i OLAP są wyspecjalizowane bazy danych, gromadzące w specjalnie zaprojektowanych strukturach dane historyczne zwane **hurtowniami danych** (ang. *Data Warehouse*). Na rysunku 2 przedstawiono schemat architektury systemów OLTP i OLAP z hurtownią danych. Pokazuje on w sposób symboliczny ideę centralnej zbiornicy danych łączącej systemy OLTP i systemy OLAP.



Rysunek 2.
Schemat architektury powiązania systemów OLTP i OLAP

3 PODSTAWY HURTOWNI DANYCH

Potrzeba analizy danych dotyczących bieżącej i przyszłej działalności organizacji była podstawowym impulsem do powstania nowych systemów informatycznych. Analiza taka stanowi podstawę do podejmowania decyzji dotyczących zarządzania przedsiębiorstwem i wspomagania podejmowania decyzji. Istniejące dotychczas systemy informatyczne (głównie klasy OLTP) nie mogą dostarczyć potrzebnych danych, gdyż są oparte na operacyjnych bazach danych realizujących codzienne procesy, mogą być rozproszone (dane znajdują się w wielu różnych źródłach), niejednorodne, a często nie są z sobą powiązane. Struktury danych są dostosowane do działań operacyjnych, dane są poddawane operacjom modyfikacji. W operacyjnych bazach danych przechowuje się dane odzwierciedlające jedynie aktualny stan lub najnowszą historię, tymczasem do analiz i porównań potrzebne są długookresowe dane historyczne. Rozwiązaniem tego problemu okazała się **hurtownia danych**. Hurtownia danych jest wydzieloną centralną bazą danych zbierającą informacje służące do zarządzania organizacją. Jest ona odizolowana od baz operacyjnych, a jej struktura i użyte do jej budowy narzędzia powinny być zoptymalizowane pod kątem przetwarzania analitycznego. Prostą, najczęściej cytowaną, definicję pojęcia hurtowni danych zaproponował William H. Inmon (jeden z czołowych teoretyków hurtowni danych i systemów OLAP – autor książki *Building the Data Warehouse,* Wiley & Sons, New Jork 1996).

> *Hurtownia danych to zbiór zintegrowanych, nieulotnych, ukierunkowanych baz danych, wykorzystywanych w systemach wspomagania decyzji.*

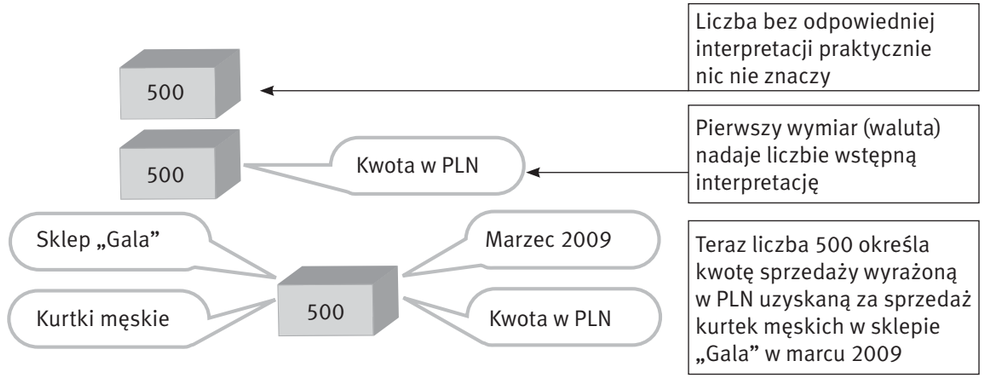
Podstawowe cechy hurtowni danych:

- **Jest scentralizowaną bazą danych** – gromadzi dane z różnych źródeł i przechowuje je w specjalnie zaprojektowanych strukturach.
- **Jest oddzielona od baz operacyjnych** – tym samym operacje wykonywane na danych gromadzonych w hurtowniach nie wpływają na wydajność operacji realizowanych w systemach OLTP.
- **Scala informacje z wielu źródeł** – ponieważ dane dotyczące jednego procesu mogą być w konkretnej firmie tworzone i przechowywane w różnych bazach danych lub nawet w plikach czy arkuszach kalkulacyjnych.

- **Jest zorientowana tematycznie** – gromadzi dane opisujące różne aspekty działalności firmy.
- **Przechowuje dane historyczne** – hurtownie mają niezaspokojony „apetyt" na dane, im dłuższa historia przechowywanych danych, tym większe możliwości analizy.
- **Utrzymuje wielką liczbę informacji** – w hurtowniach danych praktycznie nie wykonuje się operacji usuwania danych, czyli suma danych tylko rośnie wraz z dostarczaniem nowych porcji danych.
- **Agreguje informacje** – z punktu widzenia analizy najczęściej interesują nas podsumowania, obliczenia średnich i inne działania matematyczne wykonywane na grupach danych.

Najczęściej hurtownie danych są tworzone jako bazy relacyjne, w których są projektowane tabele faktów i tabele wymiarów. **Fakt** to pojedyncze zdarzenie będące podstawą analiz (np. sprzedaż produktów, udzielone kredyty itp.). Fakty są opisane przez wymiary i miary. **Miara** to wartość liczbowa dowiązana do danego faktu, np. kwota sprzedaży, liczba sztuk, a **wymiar** to cecha opisująca dany fakt, np. data, klient, produkt, lokalizacja. Dodatkowo, wymiary zawierają atrybuty, które są cechami wymiaru, np. dla wymiaru czas atrybutami mogą być miesiąc, kwartał i rok. Istotę pojęć miar i wymiarów omówimy na przykładzie. Podstawowymi elementami gromadzonymi w hurtowniach są wartości liczbowe, czyli miary pewnych faktów.

Jak pokazano na rysunku 3, wymiary są cechami opisującymi wartość miar, czyli nadają wartościom liczbowym odpowiedni sens. Najczęściej stosowanym wzorcem przy projektowaniu hurtowni jest tak zwany **schemat gwiazdy**. Na rysunku 4 przedstawiono przykładowy projekt hurtowni danych opisujący sprzedaż samochodów.

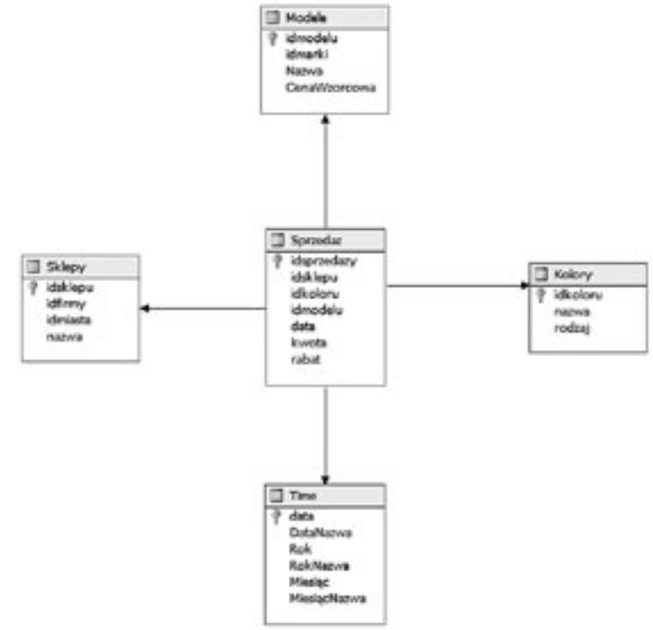


Rysunek 3.
Interpretacja miary

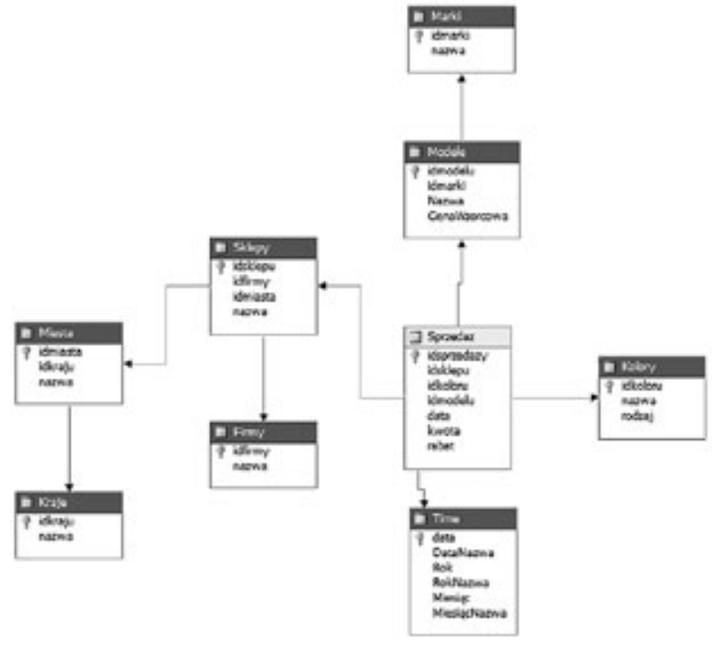
Centralną tabelą jest tabela o nazwie Sprzedaz, w której są zapisywane fakty opisujące kwoty uzyskane za sprzedaż samochodów. Tabela faktów łączy się z czterema tabelami opisującymi różne wymiary (kolor, model, sklep i czas). Połączenia tabel wymiarów z tabelą faktów są realizowane za pomocą odpowiednich kluczy obcych.

Do podstawowych cech schematu gwiazdy należy zaliczyć:
- prostą strukturę, dzięki czemu schemat jest łatwy do zrozumienia;
- dużą efektywność zapytań ze względu na niewielką liczbę połączeń tabel;
- dominującą strukturę dla hurtowni danych, wspieraną przez wiele narzędzi.

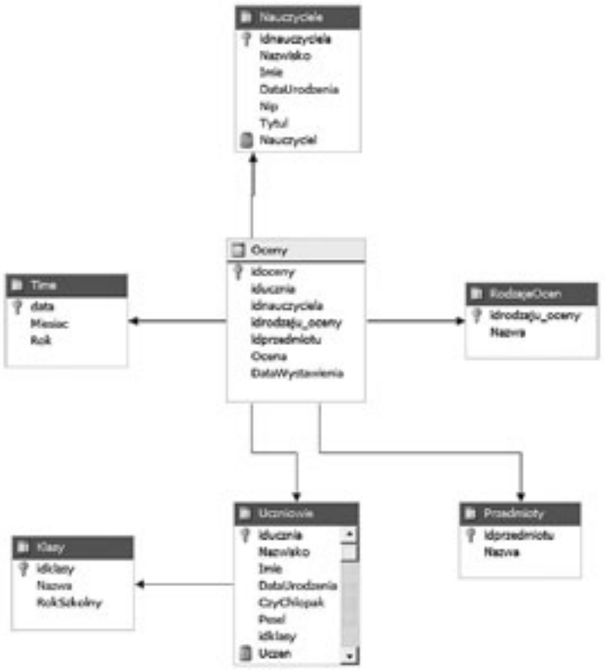
Rozwinięciem schematu gwiazdy jest schemat **płatka śniegu**, który występuje wtedy, gdy wymiary są powiązane z innymi tabelami. Na rysunku 5 przedstawiono przykładowy projekt hurtowni w schemacie płatka śniegu, który jest rozszerzeniem projektu z rysunku 4.



Rysunek 4.
Przykładowy projekt hurtowni danych w schemacie gwiazdy



Rysunek 5.
Przykładowy projekt hurtowni danych w schemacie płatka śniegu



Rysunek 6.
Przykładowy projekt hurtowni dla wystawianych ocen w szkołach

Podstawowe cechy schematu płatka śniegu:
- spadek wydajności zapytań w porównaniu ze schematem gwiazdy ze względu na większą liczbę połączeń tabel;
- struktura łatwiejsza do modyfikacji;
- wykorzystywany rzadziej niż schemat gwiazdy, gdyż efektywność zapytań jest ważniejsza niż efektywność ładowania danych do tabel wymiarów.

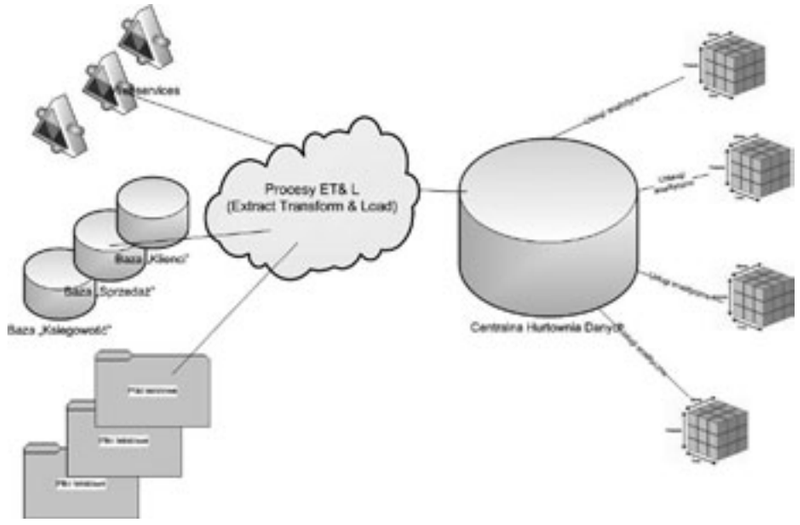
Hurtownie danych stanowią podstawowe źródło zasilające procesy analizy danych. Przedstawione przykłady projektów hurtowni są jedynie wycinkiem, gdyż w rzeczywistości hurtownie składają się z wielu podobnych struktur danych, opisujących różne fakty i korzystających z różnych wymiarów. Na rysunku 6 został przedstawiony jeszcze jeden przykład projektu struktury hurtowni danych, w którym faktami są oceny wystawione uczniom. Każda ocena jest charakteryzowana przez :
- datę jej wystawienia – wymiar Time;
- ucznia, który otrzymał ocenę – wymiar Uczniowie, który jest dodatkowo opisywany przez wymiar Klasy,
- nauczyciela, który ocenę wystawił – wymiar Nauczyciele;
- przedmiot, z którego ocena została wystawiona – wymiar Przedmioty;
- rodzaj wystawionej oceny – wymiar RodzajeOcen.

Tworzenie hurtowni danych dla jednej szkoły wydaje się niecelowe ze względu na stosunkowo niewielką liczbę danych, ale można sobie wyobrazić istnienie takiej hurtowni w skali kraju i wtedy stanowiłaby podstawę do analizy skuteczności nauczania.

Nie jesteśmy w stanie w ramach tego wykładu omówić wszystkich aspektów tworzenia hurtowni danych, gdyż są to zagadnienia złożone i praktycznie każdy projekt ma swoją specyfikę i może wyglądać zupełnie inaczej w zależności od swojego przeznaczenia i założeń, jakie dana firma przyjęła przy realizacji. Przestawione zasady stanowią punkt wyjścia przy realizowaniu konkretnego projektu.

4 PROBLEMY INTEGRACJI DANYCH

Hurtownie danych są zasilane danymi pobieranymi z systemów OLTP, które mogą być wykonane w różnych technologiach, oraz innych źródeł danych dostępnych w konkretnej firmie. Na bazie hurtowni są realizowane różne zadania analityczne. Na rysunku 7 został przedstawiony przykładowy schemat architektury takiego systemu.

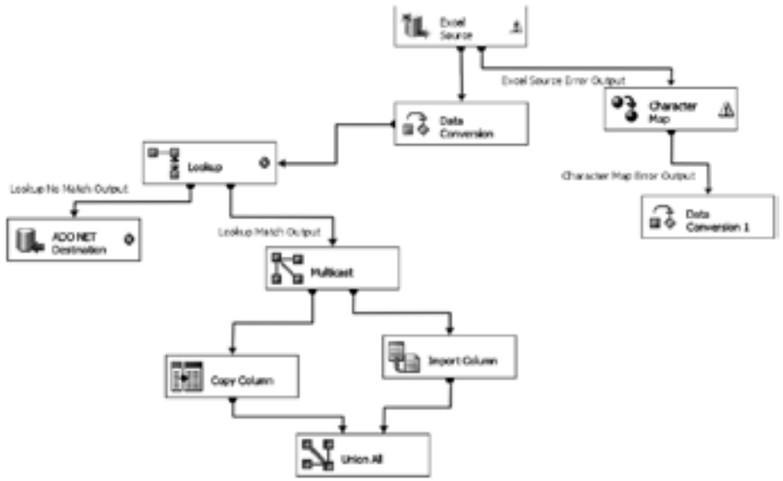


Rysunek 7.
Architektura otoczenia hurtowni danych

Przedstawiony na rysunku 7 schemat pokazuje warstwę (nazwaną procesami ET&L), która występuje pomiędzy systemami OLTP i innymi źródłami danych a hurtownią danych. Problemy związane z pozyskiwaniem danych dla hurtowni są jednymi z najtrudniejszych zadań przy jej tworzeniu. W ramach warstwy ET&L (ang. *Extract Transform & Load* – pobierz, przekształć i zapisz) są realizowane następujące zadania:
- standaryzacja danych – ponieważ dane pobierane mogą być z wielu różnego typu źródeł, to należy doprowadzić je do jednakowej postaci;
- konwersja typów danych – różne systemy mogą w inny sposób zapisywać dane i dlatego należy je doprowadzić do tego samego typu;
- transformacje danych – dane w systemach roboczych mogą być przechowywane w innej postaci niż postać ich zaprojektowana w hurtowni, dlatego należy je odpowiednio przekształcić;
- agregacja danych – w hurtowniach nie musimy zapisywać każdej elementarnej danej z systemów operacyjnych, a jedynie pewne zbiorcze wartości;
- integracja danych z różnych źródeł – dane tego samego rodzaju z punktu widzenia hurtowni (np. opis klienta) mogą być zapisywane w różnych źródłach danych i przed zapisaniem w hurtowni należy je odpowiednio powiązać;

- czyszczenie danych i kontrola poprawności – ponieważ w systemach operacyjnych mogą być przechowywane dane błędne, dlatego przed zapisaniem w hurtowni należy je sprawdzić i usunąć;
- dodatkowe przekształcenia, np. przeliczenie wartości różnych walut.

Zadania warstwy ET&L są wspierane przez różne technologie, w ramach których projektuje się i programuje działanie odpowiednich procesów. Na rysunku 8 został przedstawiony przykładowy fragment schematu procesu ET&L wykonany w MS SQL Server 2008 Integration Services.

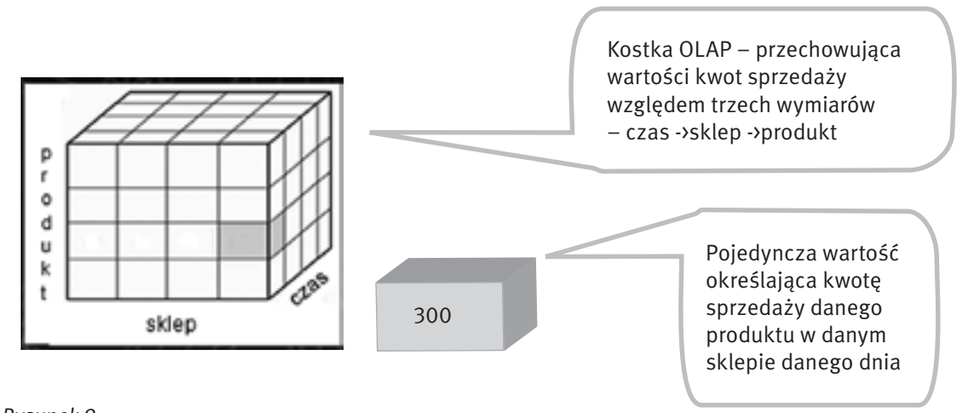


Rysunek 8.
Przykładowy pakiet usługi MS SQL Server 2008 Integration Services

Technologia MS SQL Server 2008 Integration Services umożliwia definiowanie złożonych procesów pozyskiwania, przekształcania i zapisywania danych z rożnych źródeł. Projektowany schemat przetwarzania prezentowany jest za pomocą ikon opisujących różne etapy i zadania procesu.

5 KOSTKA WIELOWYMIAROWA

Hurtownie danych stanowią punkt wyjścia do realizacji usług analitycznych. Najczęściej stosowanym elementem usług analitycznych jest **wielowymiarowa kostka OLAP**, która przechowuje dane w sposób bardziej przypominający wielowymiarowe arkusze kalkulacyjne niż tradycyjną, relacyjną bazę danych. Kostka umożliwia wyświetlanie i oglądanie danych z różnych punktów widzenia. Do jej budowy potrzeba dowolnego źródła danych opartego na tabelach relacyjnych – oznacza to, że najczęściej kostki wielowymiarowe buduje się w oparciu o hurtownie danych. Kostka składa się z miar, wymiarów oraz poziomów i jest zoptymalizowana pod kątem szybkiego i bezpiecznego dostępu do danych wielowymiarowych. **Miary** to wskaźniki numeryczne (ile?), natomiast **wymiary** reprezentują dane opisowe (kto? co? kiedy? gdzie? jak?). Wymiary są pogrupowane za pomocą **poziomów**, które odzwierciedlają hierarchię i umożliwiają użytkownikom zwiększanie lub zmniejszanie poziomu szczegółowości analizowanego wymiaru. Jak widać, kostka OLAP oparta jest na tych samych pojęciach (miary i wymiary) co schematy hurtowni danych. Trudno graficznie zaprezentować strukturę wielowymiarową – dlatego najczęściej kostka jest pokazywana w postaci sześcianu, czyli kostki złożonej z trzech wymiarów.

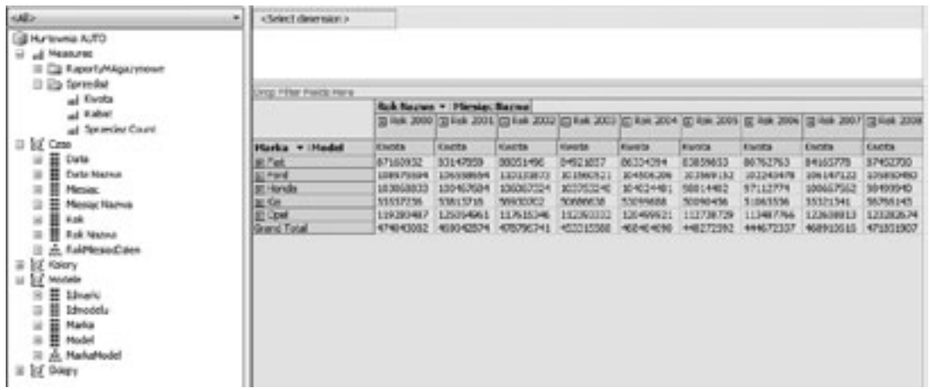


Rysunek 9.
Kostka OLAP

Podczas analizy z wykorzystaniem kostek wielowymiarowych, dane są poddawane typowym operacjom, do których zaliczamy m.in.:

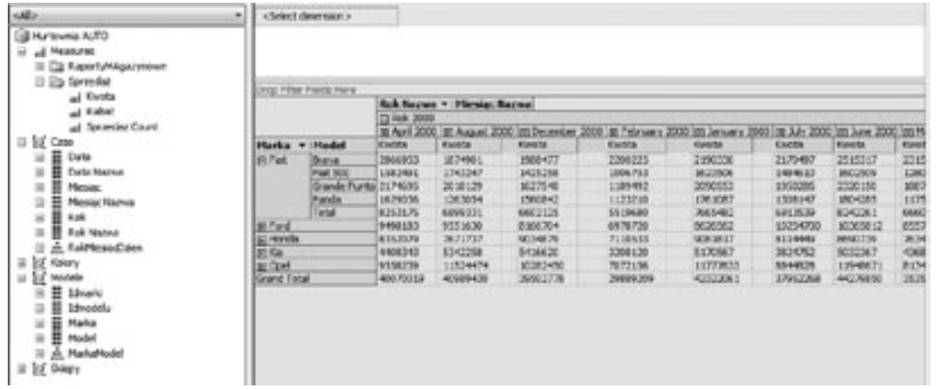
- **zwijanie** – podnoszenie poziomu agregacji, czyli uogólnianie danych;
- **rozwijanie** – zmniejszanie poziomu agregacji, dane stają się bardziej szczegółowe;
- **selekcję** – wybór interesujących elementów wymiarów;
- **projekcję** – zmniejszanie liczby wymiarów.

Obsługę tworzenia i eksploatacji kostek wielowymiarowych wspierają różne technologie, między innymi MS SQL Server 2008 Analysis Services. Na rysunku 10 przedstawiono przykładowe zestawienie na bazie kostki OLAP, opisującej sprzedaż samochodów. Zestawienie pokazuje wartość sprzedaży poszczególnych marek samochodów w kolejnych latach.



Rysunek 10.
Przykładowe zestawienie zbiorcze na bazie kostki OLAP

Kolejne zestawienie na rysunku 11 pokazuje elementy uszczegółowienia, polegające na rozbiciu kwot rocznych na poszczególne miesiące oraz rozbiciu kwot sprzedaży marki Fiat na poszczególne modele.



Rysunek 11.
Przykładowe zestawienie zbiorcze na bazie kostki OLAP z elementami uszczegółowienia

Do obsługi i pozyskiwania danych z kostek wielowymiarowych istnieje specjalny język MDX (ang. ***M**ulti **D**imensional e**X**pressions* – wyrażenia wielowymiarowe) – opis tego języka wykracza zdecydowanie poza ramy naszego wykładu. Wielowymiarowe kostki OLAP są przechowywane w specjalistycznych strukturach zoptymalizowanych pod kątem szybkości pozyskiwania danych.

6 SYSTEMY BUSINESS INTELLIGENCE

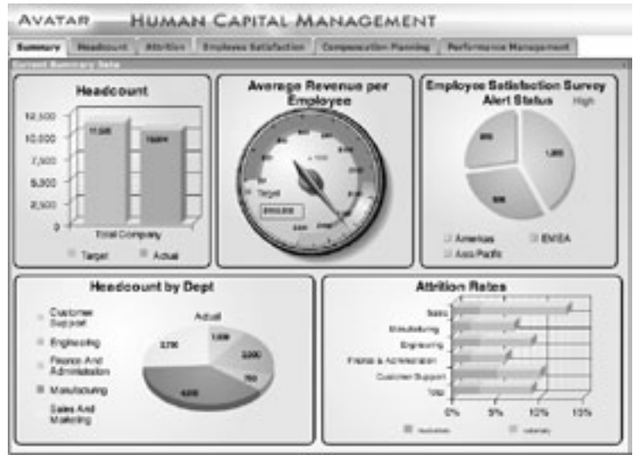
Business Intelligence (**BI**) – **analityka biznesowa** – jest pojęciem bardzo szerokim. Do dzisiaj nie istnieje powszechnie przyjmowana definicja systemów tej klasy. Najbardziej ogólnie można przedstawić je jako proces przekształcania danych w informacje, a informacji w wiedzę, która może być wykorzystana do zwiększenia konkurencyjności przedsiębiorstwa. Systemy BI są mocno uzależnione od utworzenia hurtowni danych, które umożliwiają ujednolicenie i powiązanie danych zgromadzonych z różnorodnych systemów informatycznych przedsiębiorstwa. Utworzenie hurtowni danych zwalnia systemy transakcyjne od tworzenia raportów i umożliwia równoczesne korzystanie z różnych systemów BI. System BI opiera się na następującej koncepcji:

- generuje standardowe raporty lub wylicza kluczowe wskaźniki efektywności działania przedsiębiorstwa (ang. *Key Performance Indicators*);
- na podstawie standardowych raportów i wskaźników stawia się hipotezy;
- postawione hipotezy weryfikuje się poprzez wykonywanie szczegółowych analiz danych z wykorzystaniem różnego rodzaju narzędzi analitycznych (np. OLAP, *Data Mining*).

Najczęściej spotykane odmiany systemów zaliczanych do BI to:

- EIS – systemy powiadamiania kierownictwa (ang. *Executive Information Systems*);
- DSS – systemy wspomagania decyzji (ang. *Decision Support Systems*);
- MIS – systemy wspomagania zarządzania (ang. *Management Information Systems*);
- GIS – systemy informacji geograficznej (ang. *Geographic Information Systems*).

Systemy BI są narzędziem dla menedżerów i specjalistów zajmujących się analizami i strategią. Dla menedżerów niższych szczebli, którzy oczekują informacji o aktualnym stanie procesów, przeznaczone są rozwiązania *Business Activity Monitoring* (**BAM**), umożliwiające przetwarzanie napływających na bieżąco danych. Techniki prezentacyjne są dobierane odpowiednio do potrzeb użytkownika. Jednym ze sposobów prezentowania wyników wstępnej analizy i sygnalizowania przekroczenia założonych wartości w działalności firmy jest koncepcja **kokpitu menedżera**. Idea kokpitu jest taka, aby bardzo szybko informować menedżera o wartościach podstawowych wskaźników oraz sygnalizować niekorzystne zjawiska zachodzące w jego dziedzinie odpowiedzialności. Do graficznej prezentacji takich faktów są używane proste gadżety (wskaźniki, sygnalizatory świetlne, liczniki). Elementy kokpitu powinny dać ogólny obraz procesów zachodzących w firmie. Na rysunku 12 został pokazany przykładowy kokpit menedżera.



Rysunek 12.
Przykładowa postać kokpitu menedżera [źródło: http://xelfin.pl/galeria/Galerie/1/]

Jeżeli z obrazu wskaźników kokpitu wynika problem, to należy uruchomić inne, przeważnie bardziej złożone procesy analizy.

7 EKSPLORACJA DANYCH

Eksploracja danych (spotyka się również określenie drążenie danych, pozyskiwanie wiedzy, wydobywanie danych, ekstrakcja danych) (ang. *Data Mining*) – jest jednym z etapów procesu, który bywa nazywany **odkrywaniem wiedzy z baz danych** (ang. *Knowledge Discovery in Databases*, **KDD**). Idea eksploracji danych jest oparta na wykorzystaniu komputerów i ogromnych zbiorów danych do znajdowania ukrytych dla człowieka prawidłowości w danych zgromadzonych w hurtowniach danych. Istnieje wiele technik eksploracji danych, które są oparte na zaawansowanej statystyce (statystyczna analiza wielowymiarowa) oraz technikach i metodach wywodzących się z obszaru badań nad sztuczną inteligencją. Główne przykłady stosowanych rozwiązań to:

- wizualizacje na wykresach;
- metody statystyczne;
- sieci neuronowe;
- metody uczenia maszynowego;
- metody ewolucyjne;
- logika rozmyta;
- zbiory przybliżone.

Motywację dla rozpatrywania tego typu narzędzi stanowi ciągły wzrost technicznych możliwości gromadzenia i analizy danych, w których ukryte są potencjalnie cenne informacje dopełniające wiedzę. Zastosowanie technik KDD daje szczególnie dobre wyniki w nowych dziedzinach, gdzie tak zwana wiedza ekspercka jest jeszcze w dużej mierze niepełna i nieugruntowana. Do takich dziedzin można przykładowo zaliczyć:

- analizę różnych aspektów ruchu internetowego;
- marketing z wykorzystaniem Internetu;
- rozpoznawanie obrazu, pisma, mowy itd.;
- wspomaganie diagnostyki medycznej;
- badania genetyczne;
- analizę historii operacji bankowych i zapobieganie wyłudzeniom;
- optymalizację działań (związanych z systemami CRM) zajmujących się zarządzaniem relacjami z klientami.

Proces odkrywania wiedzy z danych przebiega według poniższego schematu:

- **Zrozumienie dziedziny problemu** – złożoność danych, a także problemów stawianych przy okazji ich analizy, coraz częściej nie umożliwia natychmiastowego sformułowania pytań, na które użytkownik chce uzyskać odpowiedź. Trzeba dobrze zrozumieć problem, dla rozwiązania którego chcemy stosować techniki KDD.
- **Budowa roboczego zbioru danych** – określenie, z jakich zasobów danych będziemy korzystać w procesie KDD.
- **Oczyszczenie, przekształcanie i redukcja danych** – istotę tego problemu omówiliśmy w rozdziale poświęconym integracji danych.
- **Eksploracja danych** – realizacja procesu odkrywania wiedzy przy użyciu bardzo różnorodnych technik, opartych na statystyce, sztucznej inteligencji, czy też odwołujących się do metod uczenia maszynowego.

Rysunek 13.
Przykładowa postać kokpitu menedżera [źródło: www.shopfloorreporting.com]

Podstawowym problemem procesów odkrywania wiedzy tkwiącej w danych jest to, że różnych regularności jest w danych praktycznie nieskończenie wiele, zaś dla użytkownika interesujące będą tylko niektóre z nich i to w różnym stopniu. Osiągnięcie dobrych wyników w procesie eksploracji danych jest uzależnione nie tylko od danych i wykorzystywanych technologii, ale przede wszystkim od wiedzy i zaangażowania analityków wykonujących te zadania. Przykładowe postaci zobrazowań wyników, które można uzyskiwać w procesie eksploracji danych przedstawiono na rysunku 13.

Techniki i metody eksploracji danych są w stadium ciągłego rozwoju i należy się spodziewać nowych rozwiązań w tym zakresie.

PODSUMOWANIE

Hurtownie danych są wydzielonymi, specjalizowanymi bazami danych, przeznaczonymi do wspomagania usług analitycznych. Wdrożenie hurtowni danych może dostarczyć firmie wiele korzyści:

- **Odciążenie systemów transakcyjnych** – przygotowanie analiz i zestawień nie obciąża już systemów transakcyjnych, które mogą obsługiwać bieżące operacje. Zasilenie hurtowni danymi z systemów źródłowych wykonywane jest automatycznie i najczęściej odbywa się w cyklu dziennym, z reguły w nocy, gdy użytkownicy nie korzystają z systemu.
- **Poprawa jakości analizowanych danych** – analizując dane w hurtowni danych na zagregowanym poziomie dużo łatwiej wychwycić pewne nieprawidłowości w systemach źródłowych. W hurtowni danych bardzo dobrze widać np., czy koszty są przypisane do odpowiednich nośników, czy wszyscy klienci są przypisani do regionów sprzedaży lub handlowców itd.
- **Przechowywanie danych o długim horyzoncie czasowym** – dzięki temu, że w hurtowni danych mamy łatwy dostęp do danych wieloletnich możemy wykonywać bardziej trafne prognozy, czy też doszukiwać się określonych trendów.
- **Łączenie danych pochodzących z różnych systemów transakcyjnych –** hurtownia danych może pobrać dane z praktycznie każdego źródła danych. Dane te są następnie porządkowywane i dokonywana jest unifikacja pojęć i mierników. Dzięki temu możliwe staje się porównanie niejednorodnych danych.
- **Udostępnienie danych dla wszystkich potrzebujących –** w hurtowni danych możemy zdefiniować poszczególnym użytkownikom uprawnienia do odpowiedniego wycinka danych. Przy pomocy narzędzi analitycznych i wizualizacji danych, użytkownicy mogą wykonywać na ich bazie różne zestawienia, raporty i analizy.

LITERATURA

1. Hand D., Mannila H., Smyth P., *Eksploracja danych*, WNT, Warszawa 2002
2. Jarke M., Lenzerini M., Vassiliou Z., Vassiliadis P., *Hurtownie danych. Podstawa organizacji funkcjonowania*, WSiP, Warszawa 2003
3. Poe V., Klauer P., Brobst S., *Tworzenie hurtowni danych*, WNT, Warszawa 2000
4. Surma J., *Business Intelligence*. *Systemy wspomagania decyzji biznesowych*, WN PWN, Warszawa 2009
5. Todman Ch., *Projektowanie hurtowni danych*, WNT, Warszawa 2003

Sieci komputerowe

Podstawy działania sieci komputerowych

Podstawy działania sieci bezprzewodowych

Podstawy działania wybranych usług sieciowych

Podstawy bezpieczeństwa sieciowego

Podstawy działania sieci komputerowych

Dariusz Chaładyniak
Warszawska Wyższa Szkoła Informatyki
dchalad@wwsi.edu.pl

Streszczenie

Wykład prezentuje podstawowe informacje o budowie i działaniu sieci komputerowych, w tym m.in. najważniejsze fakty z historii sieci komputerowych i Internetu, mające istotny wpływ na obecny ich kształt i możliwości. Przedstawia główne zastosowanie, przeznaczenie i zasięg sieci komputerowych (LAN, MAN, WAN). Wyjaśnia budowę podstawowych modeli sieciowych (ISO/OSI, TCP/IP) i praktyczną interpretację ich poszczególnych warstw. Omawia podstawowe aktywne urządzenia sieciowe i ich zastosowanie przy budowie sieci komputerowych (karty sieciowe, koncentratory, przełączniki, mosty, routery). Opisano także najczęściej spotykane topologie sieciowe (magistrala, gwiazda, pierścień, siatka).

Spis treści

1 HISTORIA SIECI KOMPUTEROWYCH I INTERNETU

Rys historyczny

1957 4 października Związek Radziecki wystrzelił na orbitę okołoziemską Sputnika, pierwszego sztucznego satelitę Ziemi; w odpowiedzi na to w USA powołano agencję ARPA (ang. *Advanced Research Projects Agency*);

1964 Raport Paula Barana *On Distributed Communications* dla korporacji RAND, amerykańskiej agencji bezpieczeństwa narodowego (Paul Baran wyemigrował z Polski do USA w latach 30. XX wieku);

1967 Agencja ARPA zleciła firmie BBN (Bolt, Beranek, Newman) zbudowanie sieci ARPANET (ang. *Advanced Research Projects Agency Network*), opartej na wymianie pakietów zaproponowanej przez Barana;

1968 Pierwsza funkcjonująca sieć pakietowa w National Physical Laboratory w Wielkiej Brytanii;

1969 Uruchomienie pierwszych czterech węzłów sieci ARPANET o przepustowości 50 kbps:
- Sieciowe Centrum Pomiarowe Uniwersytetu Kalifornijskiego w Los Angeles;
- Sieciowe Centrum Informacyjne Instytutu Badawczego Stanforda;
- Instytut Interaktywnej Matematyki Cullera-Frieda Uniwersystetu Kalifornijskiego w Santa Barbara;
- Instytut Grafiki Uniwersytetu Utah.

Powstaje pierwszy dokument z serii RFC (Steve Crocker, *Host Software*);

1970 Wprowadzenie w węzłach sieci ARPANET protokołu NCP (ang. *Network Control Protocol*) – zapewniał on transmisję danych w pojedynczej sieci komputerowej i obsługiwał maksymalnie 255 maszyn;

1972 Pierwsza publiczna prezentacja funkcjonowania sieci ARPANET;
opracowanie Telnetu oraz programu do wymiany poczty elektronicznej (Ray Tomlinson);

1973 FTP (ang. *File Transfer Protocol*) – protokół transferu plików typu klient serwer

1974 Specyfikacja protokołu TCP (Vinton Cerf i Bob Kahn, *A Protocol for Packet Network Intercommunication*);

1977 Pierwsza demonstracja funkcjonowania zestawu protokołów TCP/IP;

1982 Początki właściwego Internetu (jako sieci sieci) w związku z przejściem sieci ARPANET na protokół TCP/IP;

1983 Wyodrębnienie z sieci ARPANET części militarnej – MILNET (ang. *Military Network*);
utworzenie DNS (ang. *Domain Name System* – Paul Mockapetris);

1988 Narodowa Fundacja Nauki w USA, NSF (ang. *National Science Foundation*) rozpoczyna zakładanie linii T1 o przepustowości 1,544 Mbps – powstaje sieć szkieletowa NSFNET;
opracowanie IRC (ang. *Internet Relay Chat*) – Jarkko Oikarinen;

1989 Opracowanie WWW (ang. *World Wide Web*) w Instytucie Fizyki Jądrowej CERN w Genewie przez Tima Bernersa-Lee (absolwent uniwersytetu Oxford w Wielkiej Brytanii);

1991 Wprowadzenie łączy T3 (45 Mbps) w sieci szkieletowej NSFNET;
stworzenie rozproszonego systemu wyszukiwania tekstów na zdalnych komputerach,

typu klient-serwer WAIS (ang. *Wide Area Information Servers*) w siedzibie firmy Thinking Machines Corporation przez Brewstera Kahle'a;
opracowanie Gopher w Uniwersytecie w Minnesocie przez Paula Lindnera i Marka McCahilla;

1993 Mosaic – pierwsza graficzna przeglądarka WWW;

1995 Zastąpienie sieci szkieletowej NSFNET kilkoma sieciami komercyjnymi;

1996 Konstrukcja sieci ATM (ang. *Asynchronous Transfer Mode*) o przepustowości 155 Mbps;

1999 Początek programu SETI@home (wspólne poszukiwanie cywilizacji pozaziemskich przez internautów).

Dokumenty RFC

Aby usprawnić technologię wykorzystywaną przez sieć ARPANET, zaprojektowano specjalny system obsługujący i ułatwiający wymianę korespondencji pomiędzy inżynierami pracującymi nad nową siecią.

RFC (ang. *Request for Comments*) to dokumenty tworzone przez inżynierów, zespoły inżynierów lub kogoś, kto miał po prostu lepszy pomysł na nową technologię albo jej usprawnienie. Proces powstawania RFC został zaprojektowany jako biuletyn dla zgłaszania koncepcji technologicznych. Po napisaniu i rozesłaniu RFC, może on być modyfikowany, krytykowany oraz wykorzystany przez innych inżynierów i wynalazców. Jeśli ktoś z nich chciał rozwinąć teorię, RFC zapewnia do tego celu otwarte forum.

RFC jest przedkładany do IETF (ang. *Internet Engineering Task Force*), gdzie zostaje mu przypisany numer, który jest automatycznie nazwą dokumentu RFC. RFC 1 został przekazany w 1969 roku przez Steve'a Crockera. Obecnie jest ponad 5500 dokumentów RFC – stan na czerwiec 2009.

Z dokumentami RFC można się zapoznać na oficjalnej stronie IETF – www.ietf.org.

2 ZASTOSOWANIA I PODZIAŁ SIECI KOMPUTEROWYCH

2.1 SIEĆ KOMPUTEROWA I JEJ MOŻLIWOŚCI

Rysunek 1.
Przykład sieci komputerowej

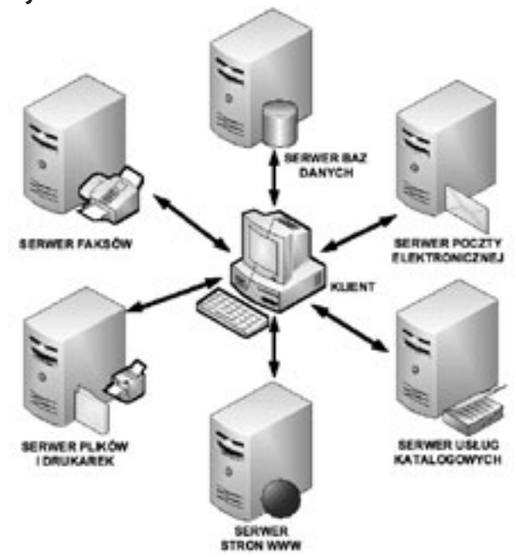
Siecią komputerową (ang. *computer network*) nazywamy zespół połączonych ze sobą komputerów, terminali, serwerów, drukarek za pomocą mediów transmisyjnych. Komunikacja w sieci jest możliwa dzięki odpowiednim protokołom.

Co umożliwia praca w sieci komputerowej

Praca w sieci komputerowej umożliwia:

- scentralizowanie administracji – z jednego (dowolnego) komputera w sieci można zarządzać i administrować wszystkimi urządzeniami połączonymi w sieć;
- udostępnianie danych – na serwerach bazodanowych, znajdujących się w sieci, można udostępniać informacje każdemu uprawnionemu użytkownikowi sieci;
- udostępnianie sprzętu i oprogramowania – użytkownikom sieci można udostępniać sprzęt komputerowy (drukarki, faksy, skanery, plotery, modemy itp.) przyłączony do sieci oraz oprogramowanie (edytory tekstu, arkusze kalkulacyjne, bazy danych, specjalistyczne aplikacje itp.) znajdujące się w komputerach w sieci.

Jaką rolę pełnią komputery w sieci

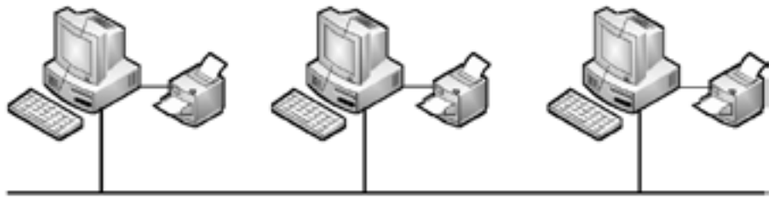


Rysunek 2.
Przykładowe role komputerów w sieci

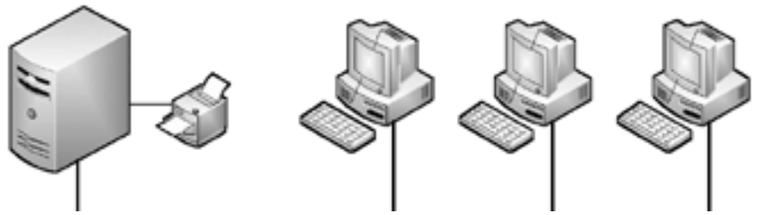
Jak pokazano na rysunku 2, komputery połączone w sieć mogą pełnić następujące role:

- serwera baz danych – do udostępniania dowolnych danych;
- serwera poczty elektronicznej – do przechowywania i zarządzania pocztą przychodzącą i wychodzącą z serwera;
- serwera usług katalogowych – do optymalnego zarządzania zasobami firmy;
- serwera stron WWW – do obsługi zasobów „globalnej pajęczyny", przeglądarek, wyszukiwarek;
- serwera plików i drukarek – do udostępniania dowolnych plików (na określonych zasadach) i drukarek;
- serwera faksów – do zarządzania i obsługi faksami;
- klienta – użytkownika komputera w sieci.

2.2 TYPY SIECI KOMPUTEROWYCH



Rysunek 3.
Sieć równorzędna



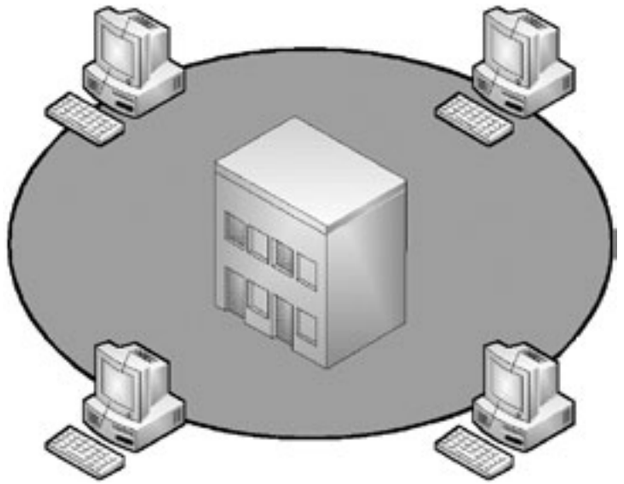
Rysunek 4.
Sieć typu klient-serwer

Sieć typu peer-to-peer (równorzędna)
Na kolejnym rysunku przedstawiono sieć **typu peer-to-peer** (p2p – równorzędna, partnerska). Jest to przykład rozwiązania bez wydzielonego urządzenia zarządzającego (serwera). Wszystkie podłączone do sieci urządzenia są traktowane jednakowo. Do zalet tego typu sieci należą: niski koszt wdrożenia, nie jest wymagane oprogramowanie do monitorowania i zarządzania, nie jest wymagane stanowisko administratora sieciowego. Natomiast wadami tego rozwiązania są: mniejsza skalowalność rozwiązania niższy poziom bezpieczeństwa i to, że każdy z użytkowników pełni rolę administratora.

Sieć typu klient-serwer
Sieć typu **klient-serwer** jest rozwiązaniem z wydzielonym serwerem zarządzającym. Komputery użytkowników są administrowane, monitorowane i zarządzane centralnie. Do zalet tego typu sieci należą: zdecydowanie wyższy poziom bezpieczeństwa, łatwiejsze zarządzanie i utrzymanie, prostsze i wygodniejsze tworzenie kopii zapasowych. Wadami tego rozwiązania są: wymóg specjalistycznego oprogramowania do monitorowania, administrowania i zarządzania, wyższy koszt urządzeń sieciowych, obecność wyszkolonego personelu administracyjnego.

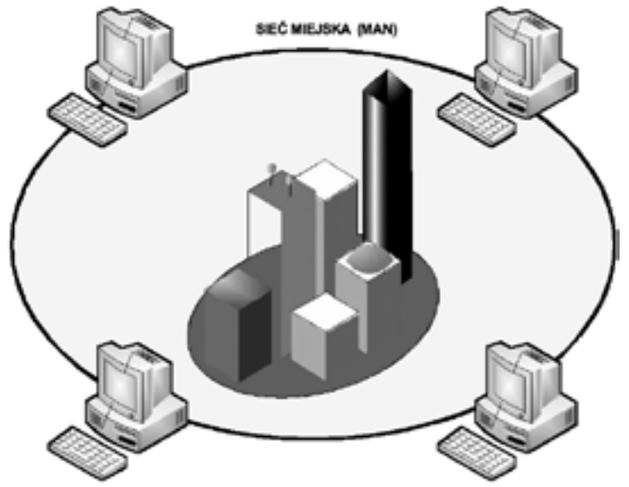
2.3 ZASIĘG SIECI KOMPUTEROWYCH

Sieć LAN
Sieć lokalna LAN (ang. *Local Area Network*) obejmuje stosunkowo niewielki obszar i zwykle łączy urządzenia sieciowe w ramach jednego domu, biura, budynku.



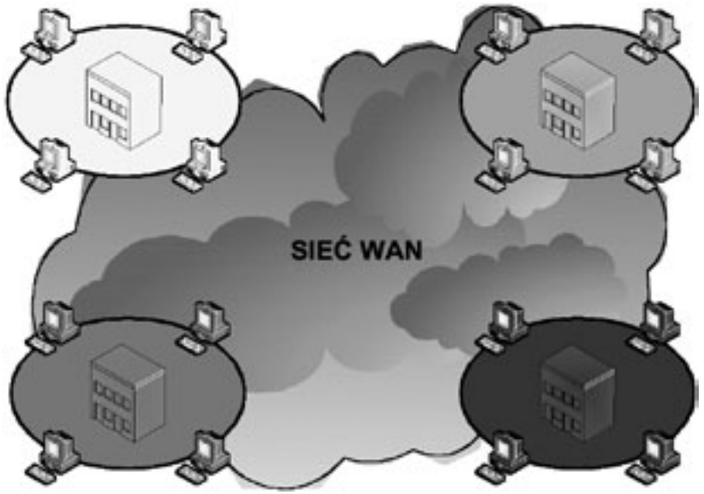
Rysunek 5.
Lokalna sieć komputerowa (LAN)

Sieć MAN
Sieć miejska MAN (ang. *Metropolitan Area Network*) jest siecią, która łączy sieci LAN i urządzenia komputerowe w obrębie danego miasta. Zasięg tej sieci zawiera się zwykle w przedziale od kilku do kilkudziesięciu kilometrów.



Rysunek 6.
Miejska sieć komputerowa (MAN)

Sieć WAN



Rysunek 7.
Rozległa sieć komputerowa (WAN)

Sieć rozległa WAN (ang. *Wide Area Network*) jest siecią o zasięgu globalnym. Łączy ona sieci w obrębie dużych obszarów, obejmujących miasta, kraje, a nawet kontynenty.

3 MODELE SIECIOWE

3.1 MODEL ODNIESIENIA ISO/OSI

Model odniesienia ISO/OSI (ang. *The International Organization for Standardization/Open Systems Interconnection*) został opracowany, aby określić wymianę informacji pomiędzy połączonymi w sieć komputerami różnych typów. Składa się on z siedmiu warstw.

1. **Warstwa fizyczna** (ang. *physical layer*) – definiuje elektryczne, mechaniczne, proceduralne i funkcjonalne mechanizmy aktywowania, utrzymywania i dezaktywacji fizycznego połączenia pomiędzy urządzeniami sieciowymi. Warstwa ta jest odpowiedzialna za przenoszenie elementarnych danych (bitów) za pomocą sygnałów elektrycznych, optycznych lub radiowych.
2. **Warstwa łącza danych** (ang. *data link layer*) – zapewnia niezawodne przesyłanie danych po fizycznym medium transmisyjnym. Warstwa ta jest odpowiedzialna za adresowanie fizyczne (sprzętowe), dostęp do łącza, informowanie o błędach i kontrolę przepływu danych.
3. **Warstwa sieci** (ang. *network layer*) – zapewnia łączność i wybór optymalnych ścieżek między dwoma dowolnymi hostami, znajdującymi się w różnych sieciach. Do podstawowych funkcji tej warstwy należy: adresowanie logiczne oraz wybór najlepszych tras dla pakietów.
4. **Warstwa transportu** (ang. *transport layer*) – odpowiedzialna jest za ustanowienie niezawodnego połączenia i przesyłania danych pomiędzy dwoma hostami. Dla zapewnienia niezawodności świadczonych usług, w tej warstwie są wykrywane i usuwane błędy, a także jest kontrolowany przepływ informacji.

5. **Warstwa sesji** (ang. *session layer*) – ustanawia, zarządza i zamyka sesje pomiędzy dwoma porozumiewającymi się ze sobą hostami. Ponadto warstwa ta synchronizuje komunikację pomiędzy połączonymi hostami i zarządza wymianą danych między nimi.
6. **Warstwa prezentacji** (ang. *presentation layer*) – odpowiedzialna jest za właściwą reprezentację i interpretację danych. Warstwa ta zapewnia, że informacje przesłane przez warstwę aplikacji jednego systemu będą czytelne dla warstwy aplikacji drugiego systemu.
7. **Warstwa aplikacji** (ang. *application layer*) – świadczy usługi sieciowe dla programów użytkowych (przeglądarek internetowych, wyszukiwarek, programów pocztowych itp.).

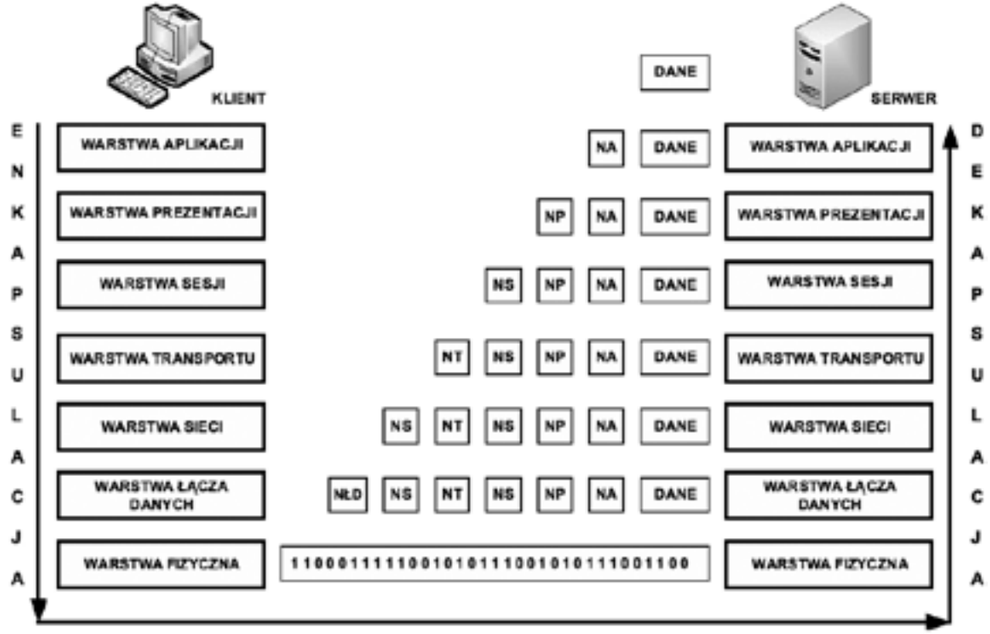
Rysunek 8.
Referencyjny model odniesienia ISO/OSI

Współpraca warstw w modelu ISO/OSI
Warstwy w modelu odniesienia ISO/OSI współpracują ze sobą zarówno w pionie, jak i w poziomie. Na przykład warstwa transportu klienta współpracuje z warstwami sesji i sieci klienta oraz warstwą transportu serwera.

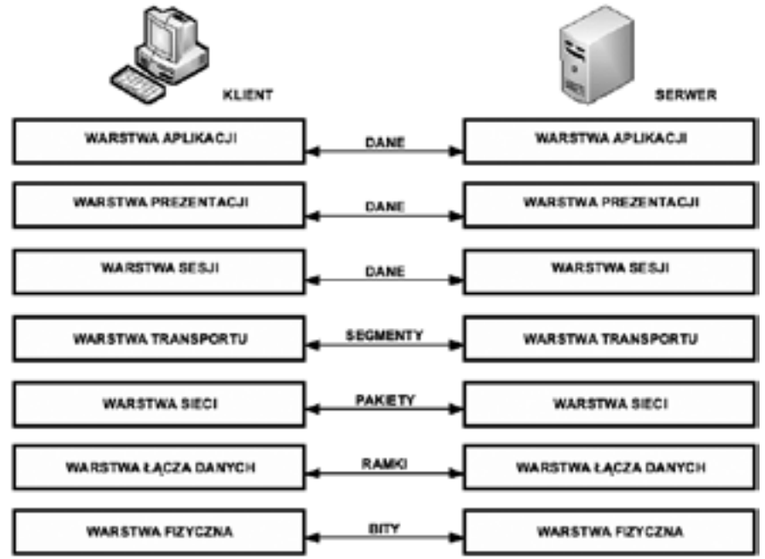
Enkapsulacja (dekapsulacja) danych
Enkapsulacja (dekapsulacja) danych jest procesem zachodzącym w kolejnych warstwach modelu ISO/OSI. Proces enkapsulacji oznacza dokładanie dodatkowej informacji (nagłówka) związanej z działającym protokołem danej warstwy i przekazywaniu tej informacji warstwie niższej do kolejnego procesu enkapsulacji. Proces dekapsulacji polega na zdejmowaniu dodatkowej informacji w kolejnych warstwach modelu ISO/OSI.

Dane, segmenty, pakiety, ramki, bity
W poszczególnych warstwach w modelu odniesienia ISO/OSI przechodzące dane noszą nazwę jednostek danych protokołu PDU (ang. *Protocol Data Unit*). Jednostki te mają różne nazwy w zależności od protokołu. I tak w trzech górnych warstwach mamy do czynienia ze **strumieniem danych**, w warstwie transportu są **segmenty**, w warstwie sieci są **pakiety**, w warstwie łącza danych – **ramki**, a w warstwie fizycznej – **bity** (zera i jedynki). Jednostki te w poszczególnych warstwach różnią się częścią nagłówkową.

Rysunek 9.
Przykład współpracy kolejnych warstw w modelu ISO/OSI

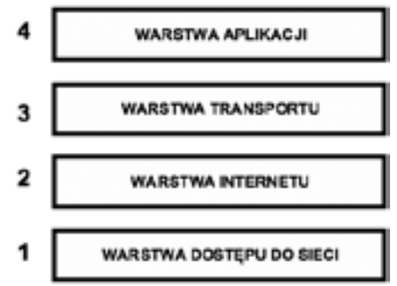


Rysunek 10.
Proces enkapsulacji i dekapsulacji danych



Rysunek 11.
Jednostki informacji w poszczególnych warstwach w modelu odniesienia ISO/OSI

3.2 MODEL TCP/IP



Rysunek 12.
Model sieciowy TCP/IP

Historycznie starszym modelem sieciowym jest **model TCP/IP** (ang. *Transmission Control Protocol/Internet Protocol*). Działanie sieci Internet opiera się właśnie na tym modelu sieciowym (patrz rys. 12). Opracowano go w połowie lat 70. XX wieku w amerykańskiej agencji DARPA (ang. *Defence Advanced Research Projects Agency*). Model TCP/IP składa się z czterech warstw.

1. **Warstwa dostępu do sieci** (ang. *network access layer*) – określa właściwe procedury transmisji danych w sieci, w tym dostęp do medium transmisyjnego (Ethernet, Token Ring, FDDI).
2. **Warstwa internetu** (ang. *internet layer*) – odpowiada za adresowanie logiczne i transmisję danych, a także za fragmentację i składanie pakietów w całość.
3. **Warstwa transportu** (ang. *transport layer*) – odpowiada za dostarczanie danych, inicjowanie sesji, kontrolę błędów i sprawdzanie kolejności segmentów.

4. **Warstwa aplikacji** (ang. *application layer*) – obejmuje trzy górne warstwy modelu odniesienia ISO/OSI, realizując ich zadania.

Porównanie modelu ISO/OSI i TCP/IP
Model ISO/OSI i model TCP/IP pomimo, że mają różną liczbę warstw i zostały opracowane w różnych czasach i przez inne organizacje, wykazują wiele podobieństw w funkcjonowaniu. Dwie dolne warstwy w modelu ISO/OSI pokrywają się z najniższą warstwą w modelu TCP/IP. Warstwa sieci w modelu ISO/OSI funkcjonalnie odpowiada warstwie Internetu w modelu TCP/IP. Warstwy transportowe występują w obu modelach i spełniają podobne zadania. Z kolei trzy górne warstwy w modelu odniesienia ISO/OSI pokrywają się z najwyższą warstwą w modelu TCP/IP.

4 PODSTAWOWE URZĄDZENIA SIECIOWE

Karta sieciowa

Rysunek 13.
Karta sieciowa [źródło: http://www.swiatkomputerow.pl]

Karta sieciowa (ang. *network interface card*), chociaż formalnie jest przypisana do warstwy łącza danych w modelu odniesienia ISO/OSI, funkcjonuje również w warstwie fizycznej. Jej podstawowa rola polega na translacji równoległego sygnału generowanego przez komputer do formatu szeregowego wysyłanego medium transmisyjnym.

Każda karta sieciowa ma unikatowy w skali całego świata **adres fizyczny (sprzętowy) MAC** (ang. *Media Access Control*), składający się z 48 bitów i przedstawiany przeważnie w postaci 12 cyfr w zapisie szesnastkowym. Pierwszych 6 szesnastkowych cyfr adresu MAC identyfikuje producenta OUI (ang. *Organizational Unique Identifier*), a ostatnie 6 szesnastkowych cyfr reprezentuje numer seryjny karty danego producenta.

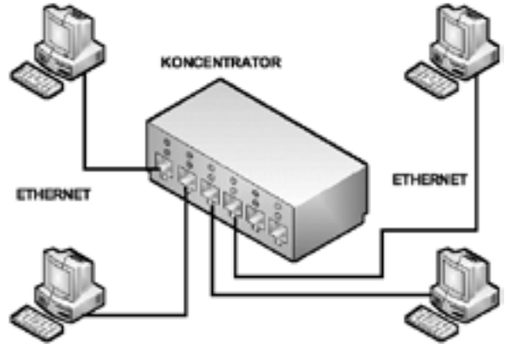
Każde urządzenie sieciowe musi zawierać kartę sieciową i tym samym ma adres MAC.

Wzmacniak

Rysunek 14.
Przykład wykorzystania wzmacniaka w sieci

Wzmacniak (ang. *repeater*) jest urządzeniem sieciowym pracującym w pierwszej warstwie modelu odniesienia ISO/OSI. Jest to najprostszy element sieciowy stosowany do łączenia różnych sieci LAN. Głównym jego zadaniem jest regeneracja (wzmocnienie) nadchodzących doń sygnałów i przesyłanie ich pomiędzy segmentami sieci. Wzmacniak może łączyć różne sieci, ale o jednakowej architekturze, używając tych samych protokołów, metod uzyskiwania dostępu oraz technik transmisyjnych. To urządzenie nieinteligentne, nie zapewnia izolacji między segmentami, nie izoluje też uszkodzeń i nie filtruje ramek, w związku z czym informacja, często o charakterze lokalnym, przenika do pozostałych segmentów, obciążając je bez potrzeby.

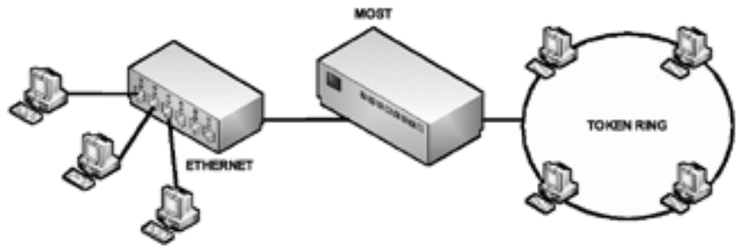
Koncentrator



Rysunek 15.
Przykład zastosowania koncentratora

Koncentrator (ang. *hub*), podobnie jak wzmacniak, pracuje w warstwie fizycznej modelu odniesienia ISO/OSI. Jest podstawowym urządzeniem sieciowym w topologii gwiazdy. Każde stanowisko sieciowe jest podłączone do koncentratora, który jest centralnym elementem sieci. Koncentratory zawierają określoną liczbę portów, z reguły od 4 do 48. Jeżeli jest więcej stanowisk niż portów koncentratora, to wtedy należy użyć dodatkowego koncentratora i połączyć je ze sobą. W przypadku dużych sieci jest możliwe kaskadowe łączenie koncentratorów. Niestety, większe sieci oparte wyłącznie na koncentratorach, są nieefektywne, gdyż wszystkie stacje w sieci współdzielą to samo pasmo. Jeżeli jedna stacja wyemituje jakąś ramkę, to pojawia się ona zaraz we wszystkich portach koncentratorów. Przy większym ruchu powoduje to kompletną niedrożność sieci.

Most

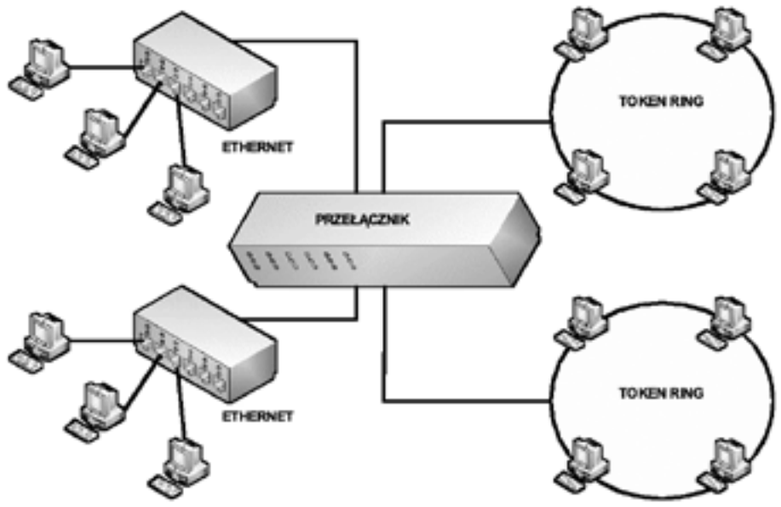


Rysunek 16.
Przykład zastosowania mostu

Most (ang. *bridge*) jest urządzeniem sieciowym działającym w drugiej warstwie modelu odniesienia ISO/OSI, czyli w warstwie łącza danych. Służy do wzajemnego łączenia sieci lokalnych. Mosty, podobnie jak wzmacniaki, pośredniczą pomiędzy dwoma sieciami, mają przy tym większe możliwości. Największą ich zaletą jest to, że filtrują ramki, przesyłając je z segmentu do segmentu wtedy, gdy zachodzi taka potrzeba. Na przykład, jeżeli komunikują się dwie stacje należące do jednego segmentu most nie przesyła ich ramek do drugiego segmentu. Wzmacniak w tym przypadku wysyłałby wszystko do drugiego segmentu, powiększając obciążenie zbędnym ruchem.

Mosty „wykazują zdolność" uczenia się. Zaraz po dołączeniu do sieci wysyłają sygnał do wszystkich węzłów z żądaniem odpowiedzi. Na tej podstawie oraz w wyniku analizy przepływu ramek, tworzą tablicę adresów fizycznych komputerów w sieci. Przy przesyłaniu danych most odczytuje z tablicy położenie komputera odbiorcy i zapobiega rozsyłaniu ramek po wszystkich segmentach sieci.

Przełącznik

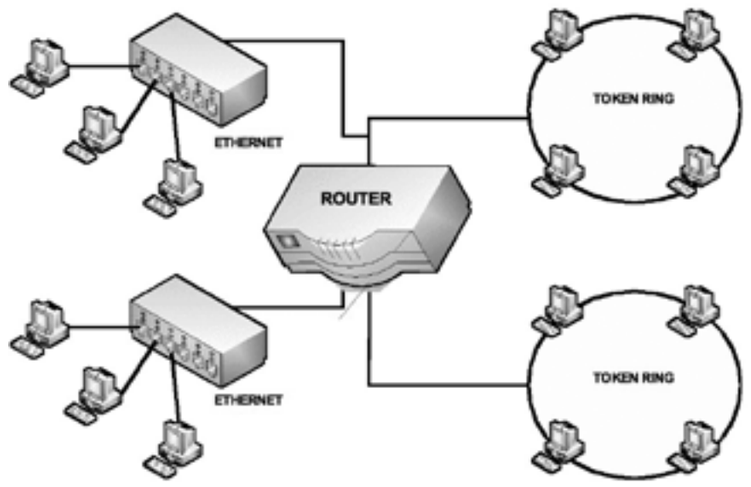


Rysunek 17.
Przykład zastosowania przełącznika

Przełącznik (ang. *switch*) jest urządzeniem sieciowym przypisanym do warstwy łącza danych modelu odniesienia ISO/OSI. Służy do podziału sieci na segmenty. Polega to na tym, że jeżeli w jakimś segmencie występuje transmisja danych angażująca jedynie stacje znajdujące się w tym segmencie, to ruch ten nie jest widoczny poza tym segmentem. Wydatnie poprawia to działanie sieci poprzez zmniejszenie natężenia ruchu i wystąpienia kolizji. Każdy przełącznik zawiera tablicę fizycznych adresów sieciowych MAC i na tej podstawie określa, czy dany adres docelowy znajduje się po stronie portu, z którego nadszedł, czy też jest przypisany innemu portowi. W ten sposób po inicjacji połączenia dane nie są rozsyłane w całej sieci, lecz są kierowane tylko do komunikujących się urządzeń. Użytkownikowi jest przydzielana wówczas cała szerokość pasma i na jego port są przesyłane wyłącznie dane skierowane do niego. W efekcie pracy przełącznika zawierającego np. 16 portów powstaje 16 niezależnych segmentów sieci, dysponujących całą szerokością pasma. Potencjalna przepustowość przełącznika jest określana przez sumaryczną przepustowość każdego portu. Szesnastoportowy przełącznik Fast Ethernet ma zatem zagregowaną przepustowość 1,6 Gb/s, podczas gdy wyposażony w szesnaście portów koncentrator Fast Ethernet – zaledwie 100 Mb/s.

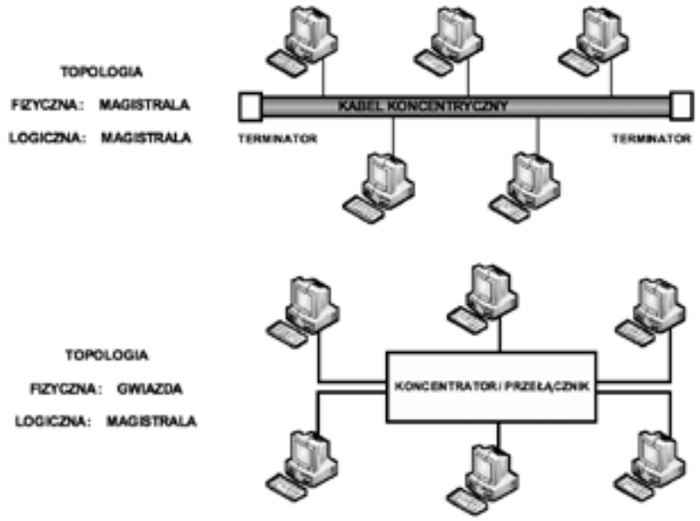
Router

Router (ang. *router*) jest urządzeniem sieciowym pracującym w trzeciej warstwie modelu odniesienia ISO/OSI, czyli warstwie sieci. Służy do zwiększania fizycznych rozmiarów sieci poprzez łączenie jej segmentów. Urządzenie to wykorzystuje logiczne adresy hostów w sieci. Ponieważ komunikacja w sieci jest oparta na logicznych adresach odbiorcy i nadawcy, przesyłanie danych i informacji jest niezależne od fizycznych adresów urządzeń. Oprócz filtracji pakietów pomiędzy segmentami, router określa optymalną drogę przesyłania danych po sieci między nadawcą i odbiorcą. Dodatkowo eliminuje on pakiety bez adresata i ogranicza dostęp określonych użytkowników do wybranych segmentów czy komputerów sieciowych. Router jest konfigurowalny, umożliwia sterowanie przepustowością sieci oraz zapewnia pełną izolację pomiędzy segmentami.



Rysunek 18.
Przykład zastosowania routera

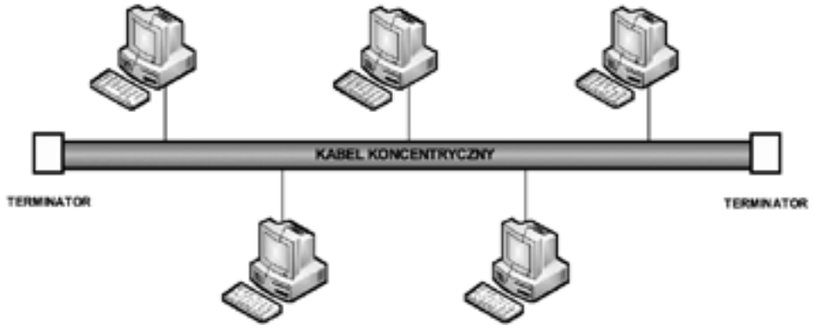
5 TOPOLOGIE SIECIOWE



Rysunek 19.
Porównanie topologii fizycznej i logicznej

Topologia fizyczna (ang. *physical topology*) jest związana z fizycznym (elektrycznym, optycznym, radiowym) łączeniem ze sobą urządzeń sieciowych. **Topologia logiczna** (ang. *logical topology*) określa standardy komunikacji, wykorzystywane w porozumiewaniu się urządzeń sieciowych.

Topologia magistrali



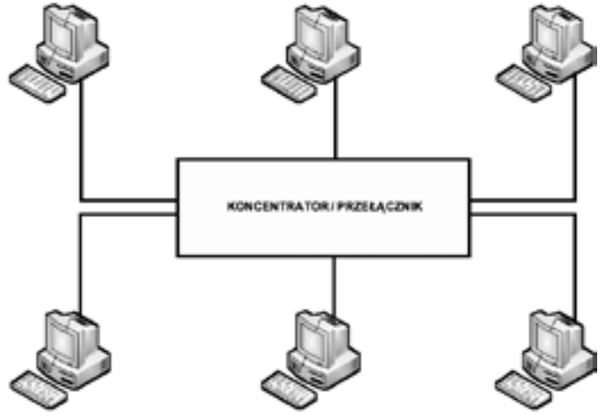
Rysunek 20.
Topologia magistrali

Topologia magistrali (**szyny**) (ang. *bus topology*) do niedawna była jedną z najpopularniejszych topologii sieciowych. Składa się z wielu komputerów przyłączonych do wspólnego kabla koncentrycznego (grubego lub cienkiego) zakończonego z obu stron terminatorem (opornikiem). Gdy dane zostają przekazane do sieci, w rzeczywistości trafiają do wszystkich przyłączonych komputerów. Wówczas każdy komputer sprawdza, czy

adres docelowy danych pokrywa się z jego adresem MAC. Jeżeli się zgadza, to komputer odczytuje (kopiuje) przekazywane informacje (ramki), a w przeciwnym przypadku przesyłka zostaje odrzucona. Do atutów topologii magistrali należą: niewielka długość kabla oraz prostota układu przewodów. Pojedyncze uszkodzenie (awaria komputera) nie prowadzi do unieruchomienia całej sieci. Słabością jest to, że wszystkie komputery muszą dzielić się wspólnym kablem.

Topologia gwiazdy

Sieć w **topologii gwiazdy** (ang. *star topology*) zawiera centralny koncentrator połączony ze wszystkimi komputerami użytkowników za pomocą kabli skrętkowych. Cały ruch w sieci odbywa się przez koncentrator lub przełącznik. W stosunku do pozostałych topologii, struktura gwiazdy ma parę zalet. Jedną z nich jest łatwość konserwacji i łatwiejsza diagnostyka. Na przykład łatwo odszukać uszkodzony odcinek kabla, gdyż każdemu węzłowi odpowiada tylko jeden kabel dołączony do koncentratora. Wadą tej topologii jest zwiększona całkowita długość okablowania, czyli koszty założenia sieci. Poważniejszy problem wynika z centralnego koncentratora lub przełącznika – ich awaria powoduje awarię całej sieci.



Rysunek 21.
Topologia gwiazdy

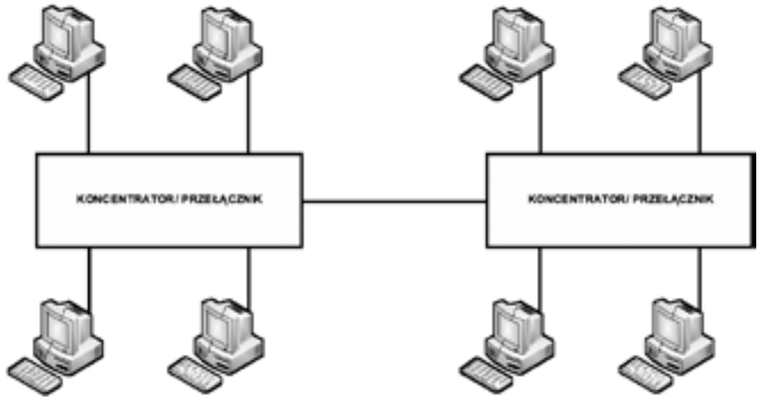
Topologia rozszerzonej gwiazdy

Topologia rozszerzonej gwiazdy (ang. *extended star topology*) to obecnie najczęściej stosowana topologia sieciowa. Umożliwia dużą skalowalność, zwłaszcza gdy są stosowane przełączniki jako węzły centralne.

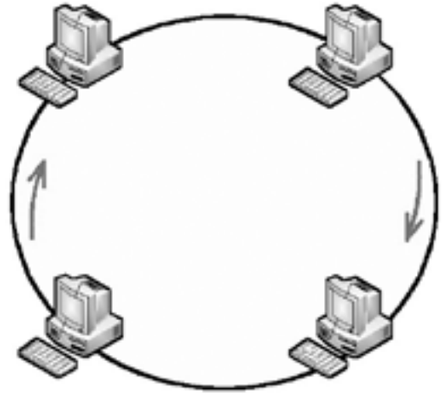
Topologia pierścienia

W **topologii pierścienia** (ang. *ring topology*) wiele stacji roboczych łączy się za pomocą jednego nośnika informacji w zamknięty pierścień. Okablowanie nie ma żadnych zakończeń, bo tworzy pełny krąg. Każdy węzeł włączony do pierścienia działa jak wzmacniak, wyrównując poziom sygnału między stacjami. Dane poruszają się w pierścieniu w jednym kierunku, przechodząc przez każdy węzeł. Jednym z plusów topologii pierścienia jest niewielka potrzebna długość kabla, co obniża koszty instalacji. Nie ma tu również centralnego koncentratora, gdyż tę funkcję pełnią węzły sieci.

Jednakże ponieważ dane przechodzą przez każdy węzeł, to awaria jednego węzła powoduje awarię całej sieci. Trudniejsza jest również diagnostyka, a modyfikacja (dołączenie, odłączenie urządzenia sieciowego) wymaga wyłączenia całej sieci.



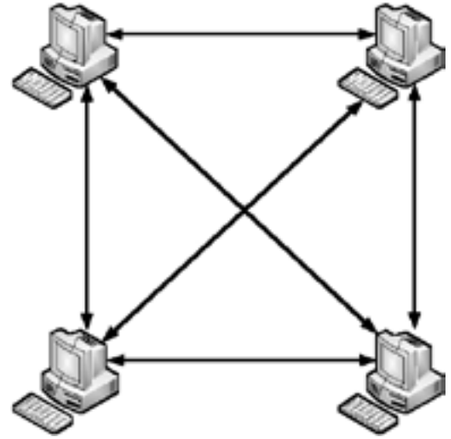
Rysunek 22.
Topologia rozszerzonej gwiazdy



Rysunek 23.
Topologia pierścienia

Topologia siatki

Topologia siatki (ang. *mesh topology*) jest stosowana w rozwiązaniach nadmiarowych (redundantnych), aby zapewnić bardzo wysoki poziom niezawodności. W topologii tej urządzenia sieciowe są połączone ze sobą każdy z każdym.



Rysunek 24.
Topologia siatki

LITERATURA

1. Dye M.A., McDonald R., Rufi A.W., *Akademia sieci Cisco. CCNA Exploration. Semestr 1*, WN PWN, Warszawa 2008
2. Krysiak K., *Sieci komputerowe. Kompendium*, Helion, Gliwice 2005
3. Mucha M., *Sieci komputerowe. Budowa i działanie*, Helion, Gliwice 2003
4. Odom W., Knot T., *CCNA semestr 1. Podstawy działania sieci*, WN PWN, Warszawa 2007
5. Pawlak R., *Okablowanie strukturalne sieci,* wydanie II, Helion, Gliwice 2008

Podstawy działania sieci bezprzewodowych

Dariusz Chaładyniak
Warszawska Wyższa Szkoła Informatyki
dchalad@wwsi.edu.pl

Streszczenie

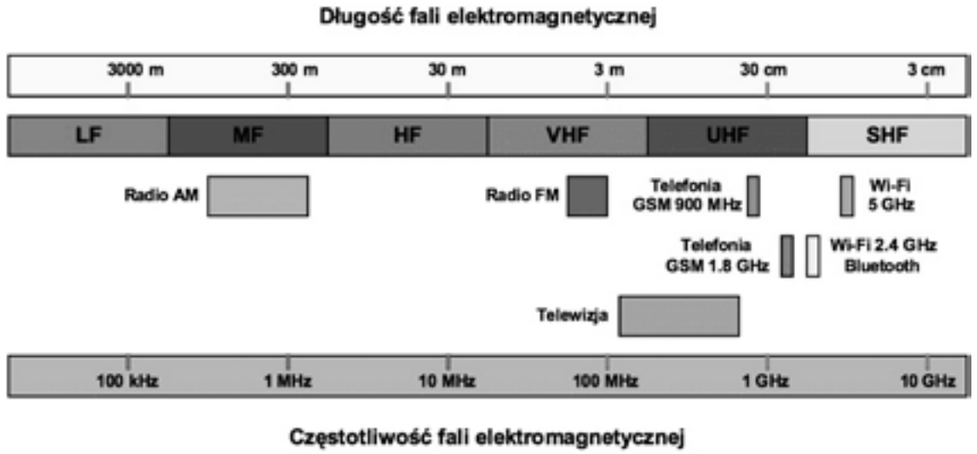
Wykład dostarcza podstawowych informacji, niezbędnych do zrozumienia działania sieci bezprzewodowych, będących bardzo ciekawą alternatywą dla klasycznych rozwiązań przewodowych. Chociaż te sieci raczej nie wyprą całkowicie tych drugich, to jednak mogą stanowić istotne ich uzupełnienie. Wykład przedstawia działanie i przeznaczenie typowych technologii bezprzewodowych (Wi-Fi, IrDA, Bluetooth, WiMAX). Bardzo istotnym zagadnieniem przy konfigurowaniu sieci bezprzewodowych jest ich właściwe bezpieczeństwo. Poświęca się temu zagadnieniu sporo miejsca. Omawia się także popularne zjawiska warchalkingu i wardrivingu. Wykład kończy opis konfiguracji punktu dostępu oraz sieciowej karty bezprzewodowej.

Spis treści

1 WSTĘP DO SIECI BEZPRZEWODOWYCH

Sieci bezprzewodowe (ang. *wireless networks*) są bardzo ciekawą alternatywą dla klasycznych sieci przewodowych. Wszędzie tam, gdzie te drugie są mało ekonomicznym rozwiązaniem stosuje się sieci WLAN. Sieci bezprzewodowe jako medium transmisyjne wykorzystują fale radiowe (elektromagnetyczne) albo fale podczerwone.

Spektrum fal elektromagnetycznych



Rysunek 1.
Spektrum fal elektromagnetycznych

Spektrum fal elektromagnetycznych (rys. 1), często występujące również pod pojęciem **widma fal**, jest przedstawieniem fal w zależności od ich częstotliwości lub długości. Widmo fal elektromagnetycznych obejmuje: fale radiowe, mikrofale, promieniowanie widzialne, promieniowanie podczerwone, ultrafioletowe, promieniowanie gamma, czy promieniowanie rentgenowskie.

W sieciach bezprzewodowych Wi-Fi i Bluetooth wykorzystuje się fale radiowe, a w sieciach IrDA – fale w kanale podczerwieni. Na rysunku 1 można zaobserwować, że dłuższym falom odpowiadają mniejsze częstotliwości i odwrotnie, krótszym falom odpowiadają wyższe częstotliwości. Częstotliwość fali wyrażana jest w hercach (Hz) i określa liczbę cykli fali w ciągu sekundy.

Metody modulacji

Przesyłanie mowy, muzyki i innych dźwięków za pomocą fal radiowych polega na zmianie (czyli **modulacji**) sygnału prądu przemiennego tzw. nośnej sygnału. Każdy rodzaj bezprzewodowej sieci transmisji danych działa w określonym paśmie częstotliwości radiowych (2,4 GHz, 5 GHz).

W sieciach bezprzewodowych wykorzystuje się trzy rodzaje modulacji:

1. **DSSS** (ang. *Direct Sequence Spread Spectrum*) – technologia rozszerzonego widma z bezpośrednim szeregowaniem bitów. Strumienie danych są tu rozdzielane przy transmitowaniu z wykorzystaniem specjalnych bitów (zwanych **bitami szumów**), a odbiornik musi dysponować układem deszyfrującym (który wykorzystuje tzw. *chipping code*, interpretując w odpowiedni sposób poszczególne strumienie danych). Cały proces

polega na rozbiciu informacji na wiele „podbitów", dzięki czemu pakiety są transmitowane przy użyciu dużo szerszego pasma przenoszenia danych niż w przypadku normalnej transmisji.

2. **FHSS** (ang. *Frequency Hopping Spread Spectrum*) – strumienie danych są przełączane z jednej częstotliwości na drugą (a każda częstotliwość to oddzielny kanał komunikacyjny), pozostając na każdej z nich nie dłużej niż 100 ms.
3. **OFDM** (ang. *Orthogonal Frequency Division Multiplexing*) – została tak zoptymalizowana, aby interfejs bezprzewodowy mógł transmitować dane w środowiskach pełnych zakłóceń, takich jak zatłoczone obszary miejskie.

Standardy sieci bezprzewodowych

Tabela 1.
Standardy sieci bezprzewodowych

Nazwa standardu	Częstotliwość radiowa	Zasięg sygnału	Maksymalna szybkość transmisji
802.11b	2.4 GHz	30 metrów	11 Mb/s
802.11a	5 GHz	30 metrów	54 Mb/s
802.11g	2.4 GHz	30 metrów	54 Mb/s
802.11n (proponowany)	2.4 GHz	50 metrów	540 Mb/s
802.15.1 Bluetooth	2.4 GHz	10 metrów	2 Mb/s

Sieci bezprzewodowe opierają się przede wszystkim na standardach z grupy IEEE 802. IEEE. W tej rodzinie, sieci bezprzewodowych dotyczy grupa standardów IEEE 802.11. Rodzina 802.11 obejmuje trzy zupełnie niezależne protokoły skupiające się na kodowaniu (a, b, g). Pierwszym powszechnie zaakceptowanym standardem był 802.11b, potem weszły 802.11a oraz 802.11g. Standard 802.11n nie jest jeszcze oficjalnie zatwierdzony, ale coraz więcej sprzętu sieciowego jest kompatybilna z tą technologią.

Pierwszym standardem sieci radiowej był opublikowany w 1997 roku IEEE standard **802.11.** Umożliwiał on transmisję z przepustowością 1 oraz 2 Mb/s przy użyciu podczerwieni bądź też pasma radiowego 2,4 GHz. Urządzenia tego typu są już praktycznie niestosowane.

Standard **802.11b** został zatwierdzony w 1999 roku. Pracuje w paśmie o częstotliwości 2,4 GHz. Umożliwia maksymalną teoretyczną szybkość transmisji danych do 11 Mb/s. Jego zasięg ograniczony jest do 30 metrów w pomieszczeniu i do 100 metrów w otwartej przestrzeni.

Standard **802.11a** został zatwierdzony w 1999 roku. Pracuje w paśmie częstotliwości 5 GHz. Jego maksymalna teoretyczna przepływność sięga 54 Mb/s.

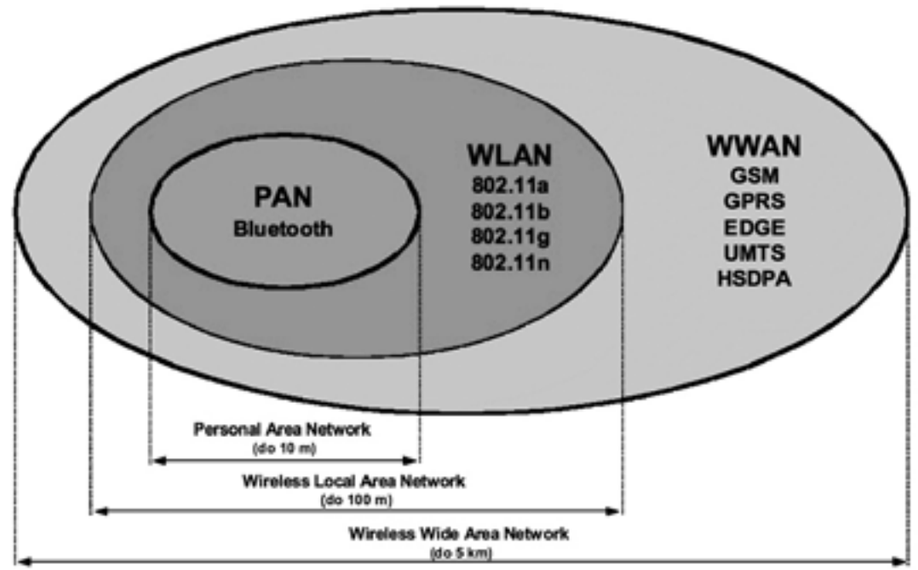
Standard **802.11g** oficjalnie został zatwierdzony w 2003 roku. Pracuje podobnie jak standard 802.11g w paśmie o częstotliwości 2,4 GHz. Umożliwia osiągnięcie maksymalnej teoretycznej szybkości transmisji danych do 54 Mb/s. Zasięg jego działania w budynku ograniczony jest do 30 metrów natomiast w przestrzeni otwartej dochodzi do 100 metrów.

Najnowszy standard **802.11n** został zatwierdzony we wrześniu 2009 roku. Może on pracować na częstotliwości 2,4 Ghz oraz 5 Ghz. Pozwala osiągnąć maksymalną teoretyczną szybkość transmisji danych do 600 Mb/s. Jego zasięg działania został wydłużony do 50 metrów w pomieszczeniu i ponad 100 metrów w otwartej przestrzeni.

Podział zasięgu sieci bezprzewodowych

Pod względem zasięgu działania (patrz rys. 2) sieci bezprzewodowe możemy podzielić na trzy kategorie:

1. **Sieci PAN** (ang. *Personal Area Network*) – działają na odległości do 10 metrów. Jako przykład tej sieci można podać standard Bluetooth.
2. **Sieci WLAN** (ang. *Wireless Local Area Network*) – działają w zakresie do 100 metrów w otwartej przestrzeni. Przykłady tych sieci to standardy IEEE 802.11a/b/g/n.
3. **Sieci WWAN** (ang. *Wireless Wide Area Network*) – działają na odległości nawet do 5 kilometrów. To przede wszystkim systemy sieci telefonii komórkowej (GSM, GPRS, EDGE, UMTS, HSDPA).



Rysunek 2.
Podział zasięgu sieci bezprzewodowych

2. TECHNOLOGIA WI-FI

Rysunek 3.
Przykłady urządzeń wykorzystujących technologię Wi-Fi

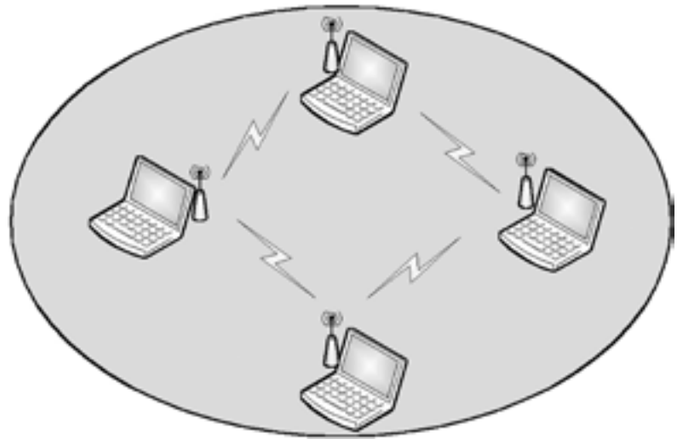
Technologia Wi-Fi polega na bezprzewodowej łączności w dwóch zakresach częstotliwości: 2,4 GHz oraz 5 GHz.

Kanały transmisyjne

Rysunek 4.
Kanały transmisyjne

Dokładna częstotliwość stosowana w określonej sieci bezprzewodowej zależy od wykorzystywanego kanału transmisyjnego. Na przykład w USA używa się 11 kanałów, w Polsce 13, w Japonii 14, a we Francji tylko 4. Aby zachować światowy standard, na całym świecie używa się tej samej numeracji kanałów, czyli kanał nr 6 w Warszawie odpowiada tej samej częstotliwości co w Tokio czy Los Angeles. W przypadku wyjazdu za granicę może być konieczne przestawienie karty sieciowej na inny kanał, aczkolwiek robią one to automatycznie. Jeśli nie mamy pewności, z jakich kanałów można korzystać w danym kraju, wystarczy sprawdzić to w lokalnym urzędzie regulacyjnym. Niezależnie od tego można skorzystać z kanałów o numerach 10 i 11, które są dostępne na całym świecie (poza Izraelem).

Technologia sieci ad-hoc – IBSS

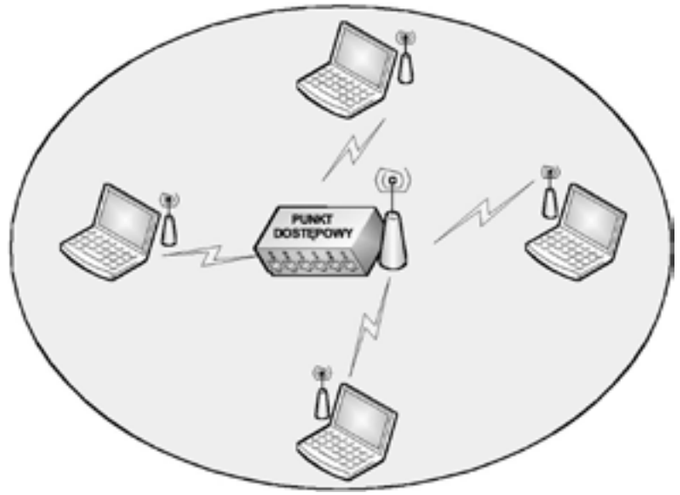


Rysunek 5.
Technologia sieci *ad-hoc*

Sieci Wi-Fi mogą działać w dwóch trybach pracy: *ad hoc* (równorzędnym) i infrastrukturalnym.

Sieć w **technologii *ad-hoc***, określana mianem **IBSS** (ang. *Independent Basic Service Set*) – rysunek 5 – może być wykorzystana do wymiany danych między kilkoma komputerami bez użycia punktu dostępowego, ale i bez dostępu do istniejącej struktury sieciowej.

Technologia sieci infrastrukturalnej – BSS

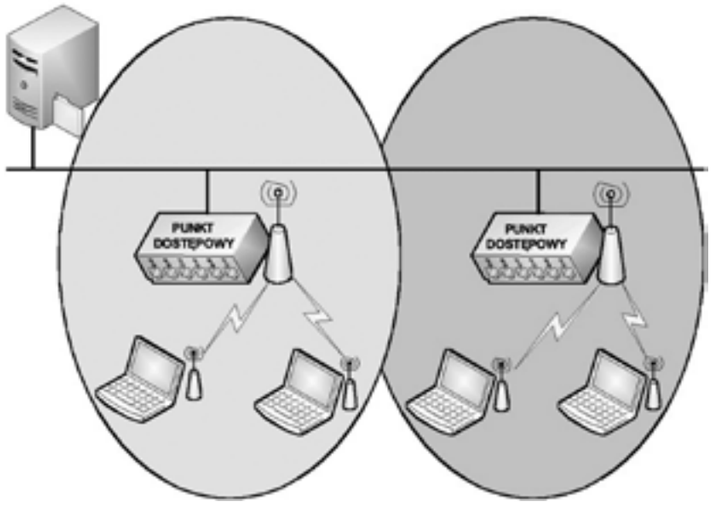


Rysunek 6.
Technologia sieci infrastrukturalnej BSS

W skład **sieci infrastrukturalnej** (rys. 6) wchodzi zwykle jeden lub więcej punktów dostępowych, które przyłączone są przeważnie do istniejącej przewodowej lokalnej sieci komputerowej. Każda stacja bezprzewodowa wymienia komunikaty i dane z punktem dostępowym, które są przekazywane dalej do innych węzłów sieci LAN (ang. *Local Area Network)* lub WLAN.

Sieć infrastrukturalna, zawierająca tylko jedną stację bazową (punkt dostępowy, router), jest określana mianem **BSS** (ang. *Basic Service Set*).

Technologia sieci infrastrukturalnej – ESS



Rysunek 7.
Technologia sieci infrastrukturalnej ESS

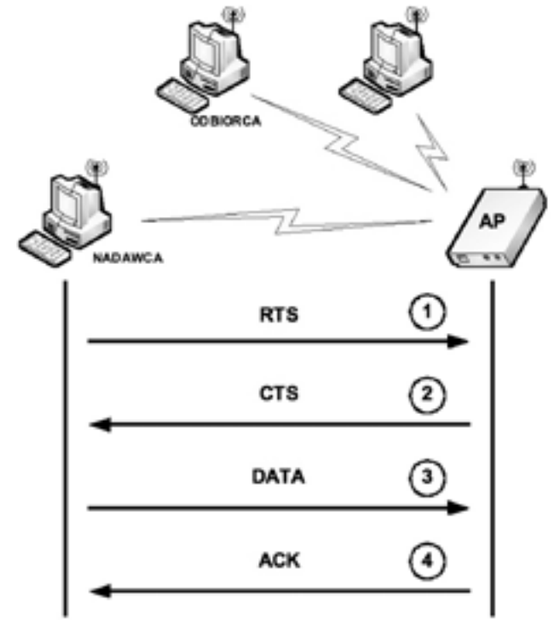
Jeśli infrastrukturalna sieć bezprzewodowa korzysta z kilku punktów dostępowych, określa się ją mianem **ESS** (ang. *Extended Service Set*) (rysunek 7).

Metoda dostępu CSMA/CA

Metoda **dostępu CSMA/CA** (ang. *Carrier Sense with Multiple Access/Collision Avoidance*), stosowana w sieciach bezprzewodowych, polega na unikaniu kolizji.

W sieciach WLAN nie jest możliwe stosowanie używanego w sieciach LAN mechanizmu CSMA/CD (ang. *CSMA/Collision Detection*). Stacja próbująca nadawać nie może bowiem jednocześnie nasłuchiwać kanału, gdyż jej własny sygnał zagłuszałby wszystkie inne. Stacja chcąca nadawać prowadzi nasłuch pasma: jeśli przez określony czas nie wykryje transmisji, przełącza się w tryb gotowości do nadawania i czeka określony czas. Następnie, jeśli nadal nikt nie prowadzi nadawania, stacja rozpoczyna transmisję. Mechanizm ten jest określany skrótem CCA (ang. *Clear Channel Assessment*). Dodatkowo, dla każdej przesłanej ramki, do nadawcy musi dotrzeć potwierdzenie poprawności otrzymania danych, wysłane przez odbiorcę ACK (ang. *Acknowledge*).

Ponieważ stacje mogą być oddalone od siebie na odległość większą od swojego zasięgu nadawania, mechanizm CCA nie spełnia swoich zadań. W tym przypadku stacja nadawcza najpierw wysyła ramkę RTS (1) (ang. *Request To Send*), będącą informacją dla pozostałych stacji w jego zasięgu o zamiarze nadawania. Następnie Punkt dostępowy (AP) wysyła ramkę CTS (2) (ang. *Clear To Send*), informującą o gotowości do odbioru. Sygnał CTS dotrze do wszystkich stacji w zasięgu (wiadomość typu rozgłoszenie), czyli dotrze również do stacji odbiorczej, która dzięki temu zostanie powiadomiona o rozpoczynającej się transmisji. Po wymianie ramek RTS i CTS rozpoczyna się właściwa transmisja ramki (DATA) (3), której otrzymanie odbiorca potwierdza ramką ACK (4). Jeśli nadawca nie dostanie potwierdzenia ACK, musi ponowić transmisję danych.



Rysunek 8.
Schemat działania metody CSMA/CA

Rozmieszczenie punktów dostępu

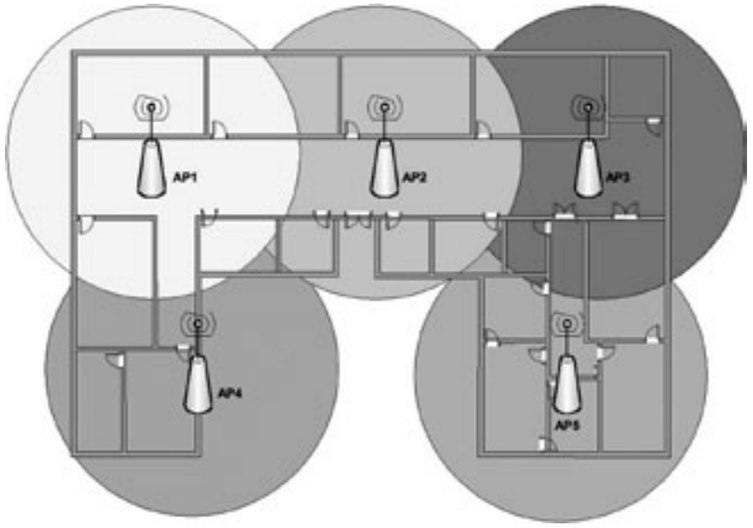
Jeden **punkt dostępu** (**AP**) może być całkowicie wystarczający do obsługi bezprzewodowej sieci lokalnej w domku jednorodzinnym lub w małej firmie. Jeśli jednak sieć ma obejmować większy obszar (o średnicy ponad 30 metrów), to są potrzebne dodatkowe punkty dostępu. Specyfikacja Wi-Fi zawiera funkcję ***roamingu***, która automatycznie przestawia połączenie sieciowe z jednego punku dostępu do innego, gdy jakość sygnału udostępnianego przez nowy punkt jest lepsza niż jakość sygnału obsługującego oryginalne połączenie.

Punkty dostępu powinny być tak rozmieszczone, aby ich obszary oddziaływania zachodziły na siebie, ale jednocześnie działały na kanałach o innych numerach. Aby maksymalnie zmniejszyć zakłócenia pomiędzy nimi, każda para sąsiadujących ze sobą punktów dostępu powinna mieć przydzielone kanały odległe o co najmniej pięć numerów.

W większości przypadków, jeśli korzysta się z wielu punktów dostępu, powinny być one rozmieszczone w taki sposób, aby obszary oddziaływania sąsiednich punktów nakładały się na siebie w około 30% (patrz rysunek 9).

Bezpieczeństwo sieci Wi-Fi

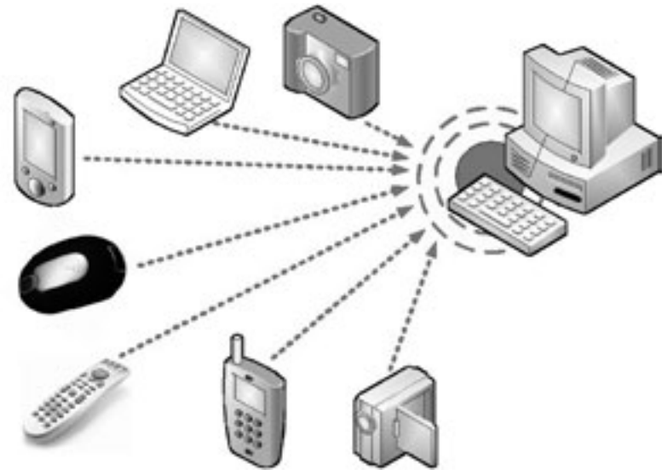
Sieci bezprzewodowe są bardzo narażone na zagrożenia sieciowe. Narzędzia bezpieczeństwa w specyfikacji Wi-Fi nie są doskonałe, ale mogą w miarę skutecznie je zabezpieczyć. Najważniejsze mechanizmy bezpieczeństwa w sieciach to:



Rysunek 9.
Przykładowe rozmieszczenie punktów dostępu

1. **Identyfikator SSID** (ang. *Service Set ID*) – wszystkie punkty dostępu oraz wszyscy klienci znajdujący się w sieci muszą mieć ustawiony taki sam SSID. Identyfikator ten zapewnia pewną bardzo ograniczoną formę kontroli dostępu, ponieważ trzeba go podać w trakcie nawiązywania połączenia do sieci Wi-Fi i jest on wartością tekstową, którą można dowolnie określić. Większość punktów dostępu rozsyła zwykle sygnał kontrolny, który rozgłasza identyfikator SSID danej sieci. Gdy karta sieciowa przeprowadza skanowanie sygnałów radiowych, wykrywa je i wyświetla listę znalezionych identyfikatorów SSID w swoim programie kontrolnym (można również tę funkcję wyłączyć).
2. **Szyfrowanie WEP** (ang. *Wired Equivalent Privacy*) – jest dostępne w każdym systemie działającym w standardzie Wi-Fi. Szyfrowanie to bazuje na współdzielonym kluczu szyfrującym o długości 40 lub 104 bitów oraz 24-bitowym wektorze inicjującym.
3. **Standard 802.1x** – scentralizowanie identyfikacji użytkowników, uwierzytelnianie, dynamiczne zarządzanie kluczami. Wszystkie te środki zapewniają dużo większe bezpieczeństwo w sieci niż kontrola dostępu wbudowana w protokół 802.11.
4. **Szyfrowanie WPA** (ang. *Wi-Fi Protected Access*) – znacznie bezpieczniejsze szyfrowanie niż WEP, ponieważ używa protokołu TKIP (ang. *Temporal Key Integrity Protocol*), w celu automatycznej zmiany klucza szyfrującego po upływie określonego czasu lub gdy nastąpi wymiana określonej liczby pakietów. Na szyfrowanie WPA składają się poniższe składniki:
   WPA = 802.1x + EAP + TKIP + MIC
   EAP (ang. *Extensible Authentication Protocol*)
   TKIP (ang. *Temporal Key Integrity Protocol*)
   MIC (ang. *Message Integrity Check*)
5. **Standard 802.11i** – zatwierdzony w lipcu 2004 roku, znany pod nazwą *Robust Security Networking*.

3 TECHNOLOGIA IRDA



Rysunek 10.
Przykłady urządzeń wykorzystujących technologię IrDA

W **technologii IrDA** (ang. *Infrared Data Association*) – rysunek 10 – wykorzystywana jest silnie skupiona wiązka światła w paśmie podczerwieni (850–900 nm). Koniecznym warunkiem zastosowania tej technologii jest bezpośrednia widoczność nadajnika i odbiornika.

Właściwości technologii IrDA

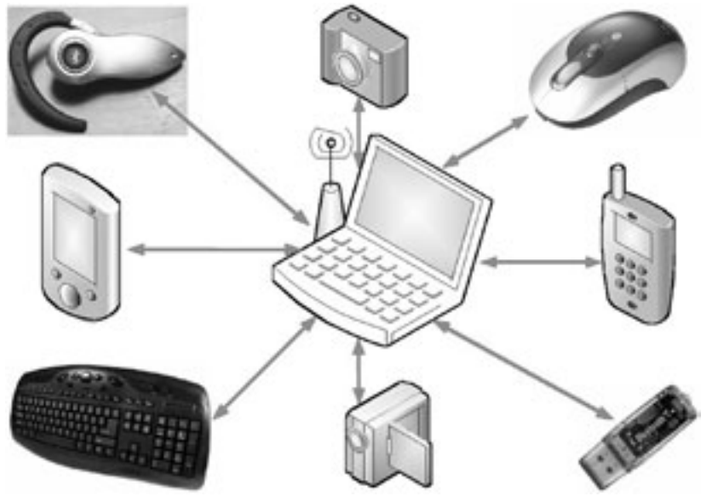
Tabela 2.
Wybrane parametry technologii IrDA

Tryb transmisji	Szybkość transmisji
Serial InfraRed SIR	2.4 - 115.2 kbps
Medium InfraRed MIR	0.576 - 1.15 Mbps
Fast InfraRed FIR	1.15 - 4 Mbps
Very Fast InfraRed VFIR	16 Mbps

Podstawowe właściwości technologii IrDA to:

- prosta i tania implementacja;
- mały pobór mocy;
- połączenie typu punkt-punkt;
- długość fali świetlnej: 850–900 nm;
- zasięg: do 10 metrów;
- kąt wiązki transmisji: 30°.

4 TECHNOLOGIA BLUETOOTH



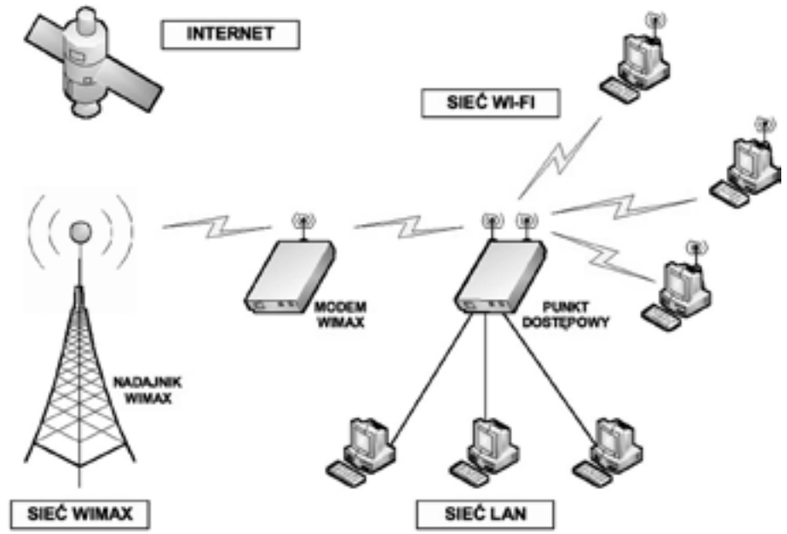
Rysunek 11.
Przykłady urządzeń wykorzystujących technologię Bluetooth

Technologia Bluetooth (rys. 11) jest globalną inicjatywą bezprzewodowego dostępu radiowego grupy producentów: Ericsson, IBM, Intel, Nokia i Toshiba. Standard Bluetooth powstał w 1994 roku w Szwecji. Jego nazwa pochodzi od przydomka żyjącego w X wieku duńskiego króla Haralda I – *Blaatand* (czyli Sinozęby).

Technologia Bluetooth jest standardem połączeń radiowych o ograniczonym zasięgu między telefonami komórkowymi, komputerami przenośnymi, urządzeniami peryferyjnymi (klawiatury, myszy, monitory, drukarki), a także audiowizualnymi (piloty, odbiorniki TV i radiowe). W Bluetooth stosuje się bezkierunkowe łącze radiowe o niewielkim zasięgu (do 10 m), o częstotliwościach pracy w paśmie 2,402–2,480 GHz. Możliwa jest komunikacja między różnymi urządzeniami przenośnymi (maks. 256) z przepływnością do 1 Mb/s.

5 TECHNOLOGIA WIMAX

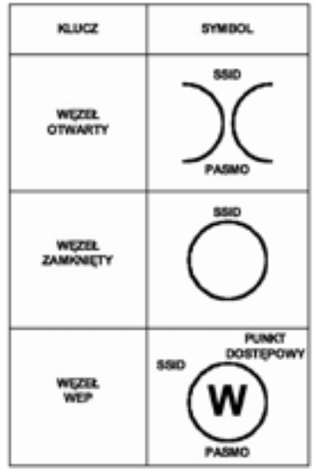
Technologia WiMAX (ang. *Worldwide Interoperability for Microwave Access*) – rysunek 12 – to bezprzewodowa metoda szerokopasmowej transmisji danych na dużych obszarach geograficznych. Jest to bezprzewodowa sieć miejska, w której zazwyczaj stosuje się jedną lub więcej stacji bazowych, z których każda dystrybuuje sygnał drogą radiową w promieniu do 50 km. Oficjalnie technologia WiMAX jest opisana w specyfikacji IEEE 802.16d. Każdy dostawca usługi WiMAX korzysta z koncesjonowanych częstotliwości z zakresu pomiędzy 2 GHz a 11 GHz. Łącze WiMAX może teoretycznie przesyłać dane z przepływnością do 70 Mb/s.



Rysunek 12.
Schemat działania technologii WiMAX

6 WARCHALKING, WARDRIVING

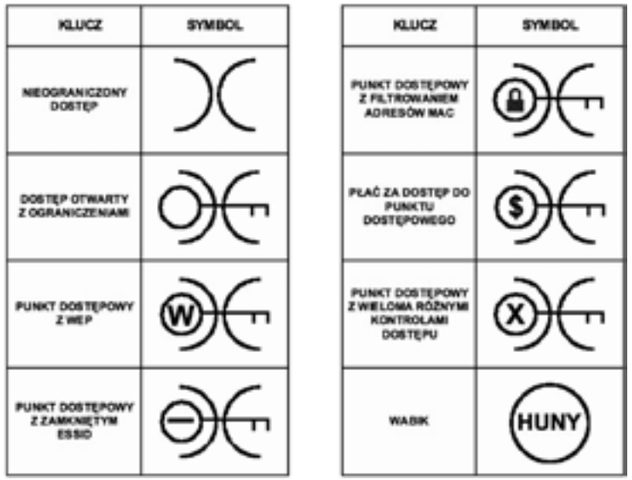
Znaki naznaczonego dostępu – oryginały



Rysunek 13.
Oryginalne znaki naznaczonego dostępu

Anglik Matt Jones zaczął w czerwcu 2002 roku rysować w Londynie kredą na chodnikach i ścianach domów symbole identyfikujące miejsca, do których dochodzą sygnały sieci bezprzewodowych IEEE 802.11b i jest możliwe uzyskanie „bezpłatnego" dostępu do Internetu. Na rysunku 13 przedstawiono trzy oryginalne znaki naznaczonego dostępu.

Znaki naznaczonego dostępu – propozycja



Rysunek 14.
Propozycja znaków naznaczonego dostępu

Na rysunku 14 przedstawiono propozycję nowych znaków naznaczonego dostępu do sieci Internet. Znaki te są malowane za pomocą kredy, by osoba oznaczająca punkt dostępu nie została posądzona o wandalizm, jak to bywa w przypadku graffiti wykonanego sprayem (kredę łatwo da się zmyć).

Wardriving

Rysunek 15.
Przykład wardrivingu [źródło: http://techteachtoo.com/internet-security/wardriving-secure-wifi/]

Termin wardriving określa przede wszystkim techniki namierzania sieci bezprzewodowych, najczęściej wykorzystując do tego celu samochód. Jako narzędzi używa się laptopa wyposażonego w kartę Wi-Fi, antenę wzmacniającą oraz odpowiednie oprogramowanie. Metoda ta wiąże się nierozerwalnie z techniką warchalkingu, czyli oznaczania ścian budynków lub chodników w miejscach, gdzie rozpoznany został otwarty punkt dostępowy. Istnieje także odmiana wardrivingu, czyli warstrolling, polegająca na pieszym przemieszczaniu się po mieście w celu odkrycia sieci Wi-Fi.

Najbardziej popularnym programem używanym przez amatorów rozpoczynających swą przygodę z wardrivingiem jest Netstumbler wykorzystujący metody wykrywania za pomocą skanowania aktywnego. Polega ona na oczekiwaniu odpowiedzi w postaci ramek Probe Response, na uprzednio wysłane ramki Probe Request. Dzięki nim uzyskuje się takie informacje, jak identyfikator ESSID, numer kanału oraz dotyczące mechanizmów WEP, natężenia ruchu i prędkości. Ta metoda jest możliwa jedynie do zastosowania w sieciach otwartych, sieci zamknięte bowiem nie odpowiadają na takie zapytania, a jej funkcjonalność może zostać znacznie ograniczona przez administratora sieci, który stosuje filtrowanie ramek niosących identyfikator ESSID. Oprócz tego użycie narzędzi tego typu jest ograniczone z powodu wymogu fizycznej obecności w obrębie zasięgu nadawania karty. Programy, takie jak Netstumbler nie analizują również ruchu sieciowego, rejestrują jedynie ruch ramek odpowiedzi, przez co osobę go stosującą można bardzo szybko wykryć. Znacznie częściej są używane programy działające w trybie monitorowania sieci, jak Kismet, AirTraf, WifiScanner czy Wellenreiter.

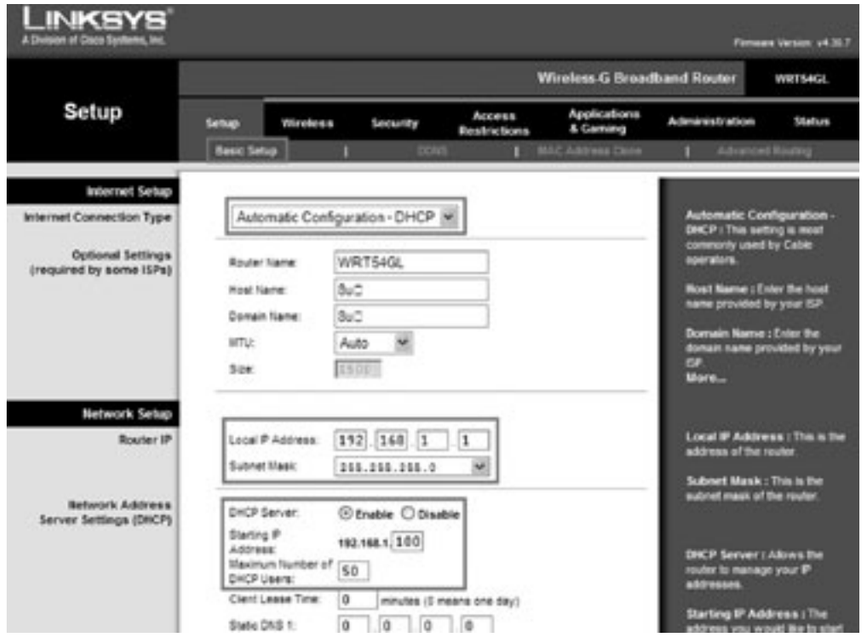
7 KONFIGURACJA SIECI BEZPRZEWODOWEJ

Konfiguracja punktu dostępu

Rysunek 16.
Logowanie do konfiguracji punktu dostępu

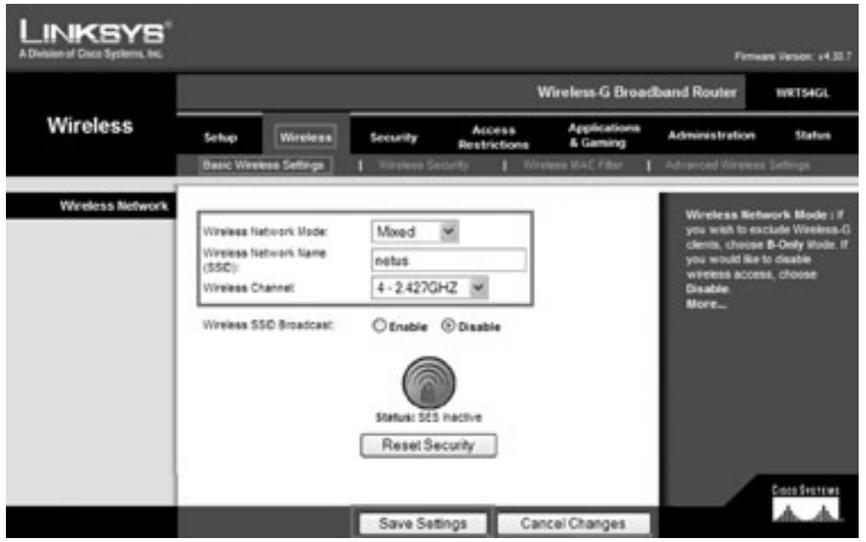
Aby móc korzystać z usług sieciowych w trybie bezprzewodowym, należy właściwie skonfigurować zarówno punkt dostępu, jak i kartę sieciową Wi-Fi. Pokazujemy to na kolejnych ilustracjach. Większość współczesnych punktów dostępu (ang. *routerów*) można skonfigurować poprzez przeglądarkę internetową. W jej górnym pasku (rys. 16) należy wpisać adres IP punktu dostępu (w naszym przypadku 192.168.1.1). Pojawi się wówczas zakładka logowania (oczywiście wcześniej należy utworzyć konto użytkownika i przypisać mu hasło). Po wpisaniu nazwy użytkownika i podaniu hasła klikamy na przycisku OK.

Po poprawnym zalogowaniu zgłasza się panel konfiguracyjny punktu dostępu (rys. 17). W kategorii Setup wybieramy zakładkę Basic Setup, w której możemy określić między innymi: typ konfiguracji serwera DHCP, lokalny adres IP routera (punktu dostępu) i jego maskę podsieci, a także zaznaczyć, czy serwer DHCP jest dostępny czy nie, a jeśli tak, to przeznaczyć dla niego odpowiednią pulę adresów IP poczynając od wybranej wartości.



Rysunek 17.
Podstawowa konfiguracja punktu dostępu

W kategorii Wireless (rys. 18) wybieramy zakładkę Basic Wireless Settings, w której określamy właściwy tryb sieci bezprzewodowej, jej nazwę (SSID) oraz numer kanału transmisyjnego i częstotliwość jego pracy. Po dokonaniu zmian, klikamy na przycisku Save Settings, aby zapisać ustawienia, albo na przycisku Cancel Changes, aby anulować zmiany.



Rysunek 18.
Konfiguracja trybu sieci bezprzewodowej, jej nazwy oraz kanału transmisyjnego

Rysunek 19.
Konfiguracja zabezpieczeń sieci bezprzewodowej

Natomiast w zakładce Wireless Security (rys. 19) konfigurujemy zabezpieczenia sieci bezprzewodowej. W naszym przypadku jako tryb zabezpieczeń został wybrany bardzo silny protokół szyfrowania WPA2 wykorzystujący algorytm AES (ang. *Advanced Encryption Standard*). W polu WPA Shared Key można wpisać klucz współdzielony, a w polu Group Key Renewal – podać częstotliwość odnawiania tego klucza, czyli jak często router (punkt dostępu) powinien zmieniać klucze szyfrujące. Podobnie, albo zatwierdzamy zmiany (przycisk Save Settings), albo anulujemy (Cancel Changes).

Z kolei w zakładce Wireless MAC Filter (rys. 20) możemy zaznaczyć opcję filtrowania adresów MAC i pozwolić korzystać z sieci bezprzewodowej tylko tym hostom, których adresy MAC na ich kartach sieciowych znajdują się na odpowiedniej liście.

W kolejnej zakładce Advanced Wireless Settings w kategorii Wireless (rys. 21) możemy bardzo szczegółowo zdefiniować poszczególne elementy zabezpieczeń sieci bezprzewodowej.

W kategorii Security wybieramy zakładkę Firewall (rys. 22), w której możemy uaktywnić ochronę sieci bezprzewodowej poprzez konfigurację ściany ogniowej.

W kategorii Access Restrictions (rys. 23) wybieramy zakładkę Internet Access, w której możemy określić politykę dostępu do sieci Internet. Dostępne opcje dają mnóstwo możliwości precyzyjnego określenia, które stacje robocze, kiedy i w jakim czasie mogą korzystać z Internetu bądź też nie.

Rysunek 20.
Konfiguracja filtrowania adresów MAC

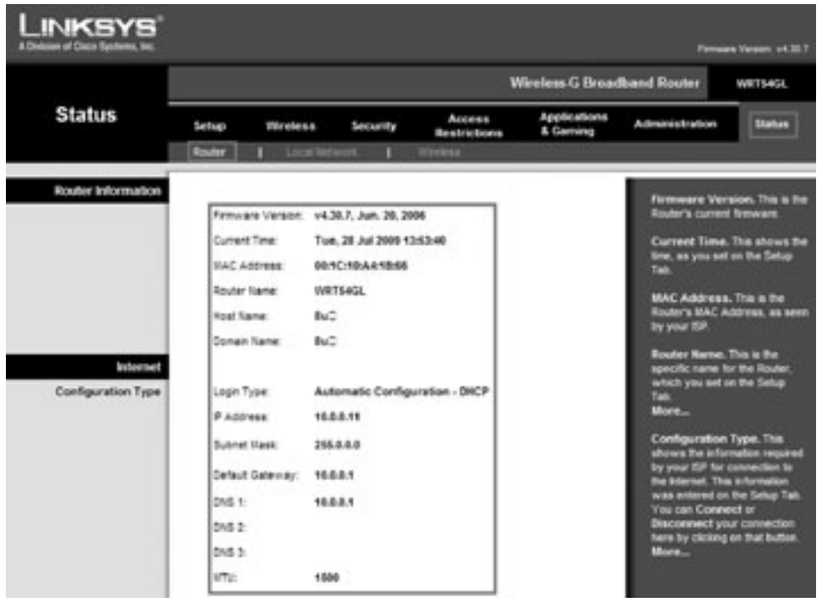
Rysunek 21.
Konfiguracja zaawansowanych zabezpieczeń sieci bezprzewodowej

Rysunek 22.
Konfiguracja ściany ogniowej

Rysunek 23.
Konfiguracja dostępu do sieci Internet

Rysunek 24.
Konfiguracja hasła dostępu do routera (punktu dostępu)

Z kolei w zakładce Management w kategorii Administration (rys. 24) określamy hasło dostępu do routera (punktu dostępu), czyli to hasło, które podaliśmy logując się do konfiguracji punktu dostępu. Możemy również zaznaczyć, czy chcemy zarządzać punktem dostępu zdalnie czy tylko lokalnie.



Rysunek 25.
Podgląd podstawowych informacji o konfiguracji punktu dostępu

W zakładce Router w kategorii Status (rys. 25) możemy odczytać informacje o routerze (punkcie dostępu), takie jak: wersja oprogramowania sprzętowego, bieżący czas, adres sprzętowy (MAC) routera i jego nazwę. Ponadto możemy zapoznać się z konfiguracją adresów IP.

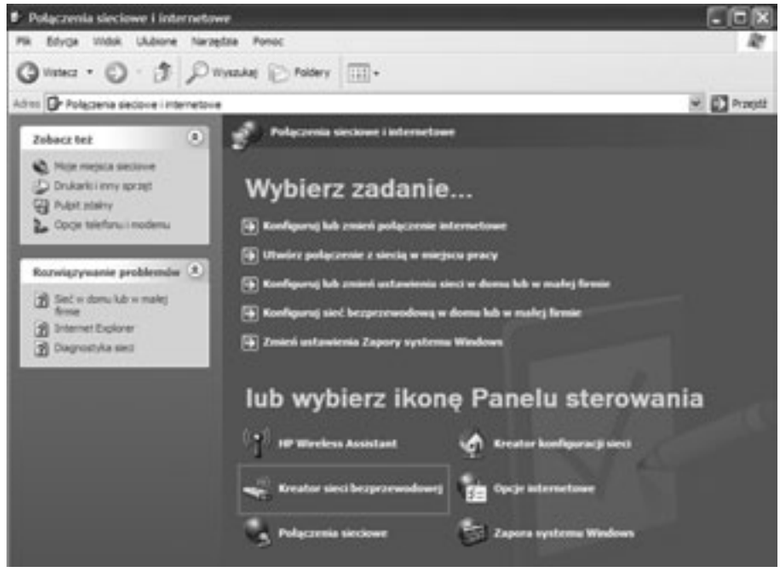
Rysunek 26.
Zbiorcze podsumowanie konfiguracji sieci bezprzewodowej

I wreszcie w zakładce Wireless w kategorii Status (rys. 26) otrzymujemy zbiorcze podsumowanie najważniejszych ustawień konfiguracyjnych naszej sieci bezprzewodowej – adres MAC punktu dostępu, tryb jego pracy, jego nazwę, dostępność serwera DHCP, numer kanału transmisyjnego oraz czy zostały uaktywnione funkcje szyfrujące przepływ danych.

Konfiguracja karty sieciowej Wi-Fi

Rysunek 27.
Wybór kategorii – połączenia sieciowe i internetowe

Po skonfigurowaniu punktu dostępu należy skonfigurować kartę sieciową Wi-Fi. W Panelu sterowania dwukrotnie klikamy na opcji Połączenia sieciowe i internetowe (rys. 27), następnie dwukrotnie klikamy na ikonie Kreator sieci bezprzewodowej (rys. 28), co prowadzi nas do okna Kreatora sieci bezprzewodowej, w którym klikamy na przycisku Dalej (rys. 29).



Rysunek 28.
Wybór kreatora sieci bezprzewodowej

Rysunek 29.
Zgłoszenie się kreatora sieci bezprzewodowej

Rysunek 30.
Wybór konfiguracji nowej sieci bezprzewodowej

W następnym kroku (rys. 30) musimy wybrać jedno z dwóch dostępnych zadań – Skonfiguruj nową sieć bezprzewodową lub Dodaj nowe komputery lub urządzenia do sieci Darek (w naszym przypadku). Wybieramy opcję pierwszą, a następnie klikamy na przycisku Dalej.

Rysunek 31.
Tworzenie nazwy dla sieci bezprzewodowej

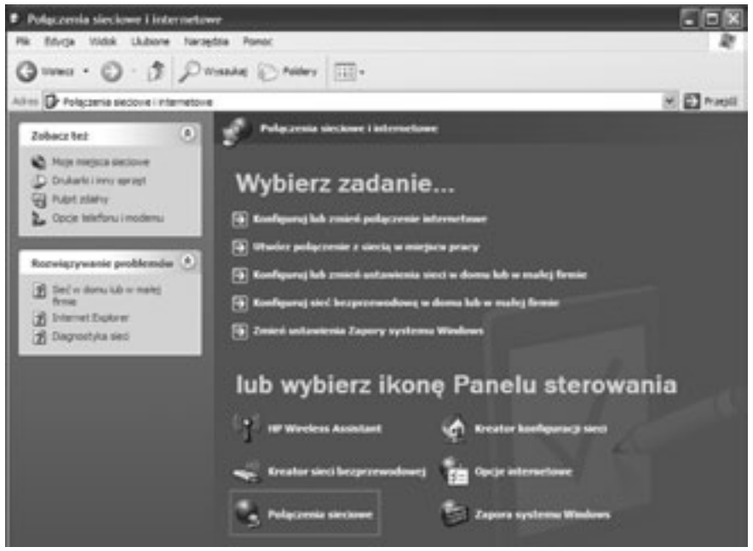
W procesie tworzenia nowej sieci bezprzewodowej (rys. 31) musimy najpierw podać dla niej nazwę (maksymalnie 32 znaki), a następnie zaznaczyć, jaką formę klucza sieciowego chcemy ustawić (automatycznie czy ręcznie). Wybieramy pierwszą opcję, a ponadto zaznaczamy Użyj szyfrowania WPA zamiast WEP (WPA jest silniejsze niż WEP, ale nie wszystkie urządzenia są zgodne z WPA). Następnie klikamy na przycisku Dalej.

Rysunek 32.
Wybór metody konfiguracji sieci bezprzewodowej

Na dalszym etapie (rys. 32) możemy określić sposób skonfigurowania sieci bezprzewodowej. Mamy do wyboru albo użycie dysku flash, albo skonfigurowanie ręczne. Z uwagi na to, że konfigurujemy małą sieć komputerową, wybieramy opcję drugą, a następnie klikamy na przycisku Dalej.

Rysunek 33.
Koniec pracy kreatora sieci bezprzewodowej

Zwieńczenie dzieła jest pokazane na rysunku 33 – oznajmienie, że praca kreatora została pomyślnie ukończona. Możemy skorzystać z dostępnej opcji i wydrukować ustawienia sieci. Aby zamknąć kreatora sieci bezprzewodowej, klikamy na przycisku Zakończ.



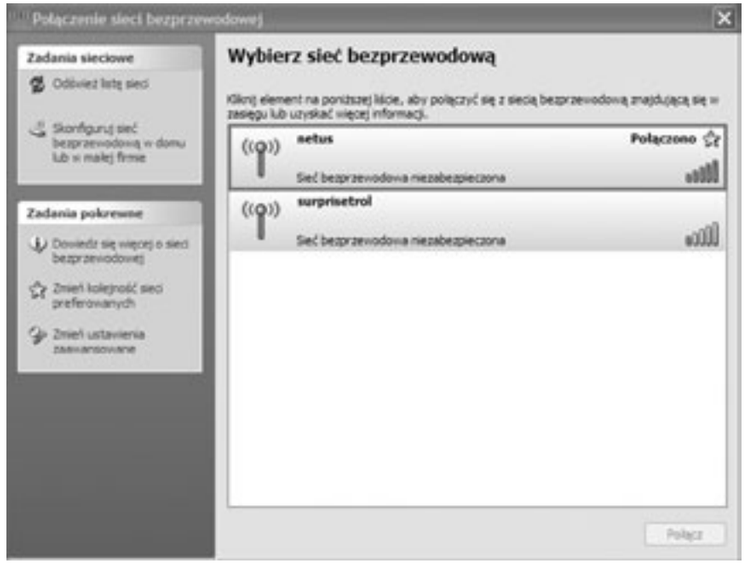
Rysunek 34.
Pierwszy krok do konfiguracji karty sieciowej Wi-Fi

W kolejnym kroku konfigurowania karty sieciowej Wi-Fi wybieramy Połączenia sieciowe z kategorii Połączenia sieciowe i internetowe (rys. 34) i klikamy dwukrotnie na ikonie Połączenia sieci bezprzewodowej (rys. 35).

Rysunek 35.
Wybór zakładki – połączenie sieci bezprzewodowej

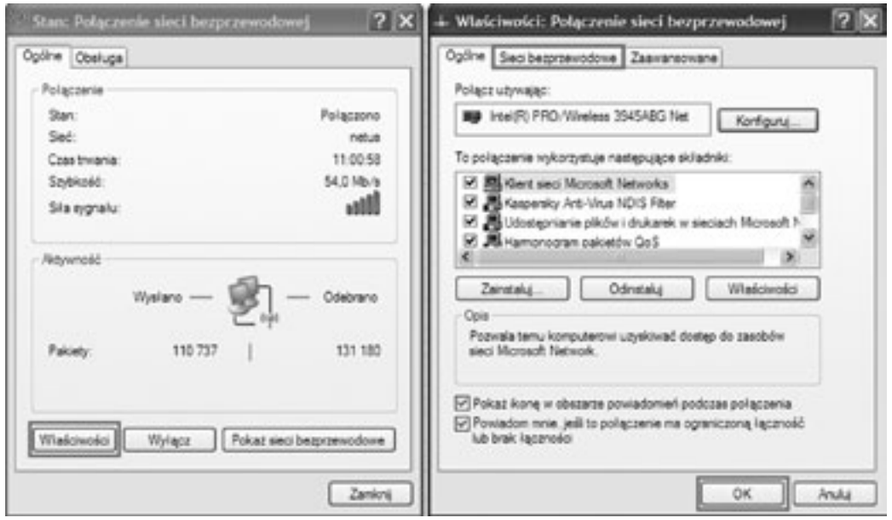
Rysunek 36.
Podgląd stanu połączenia sieci bezprzewodowej

Ukazuje się podgląd stanu połączenia sieci bezprzewodowej (rys. 36), widać jej nazwę, czas trwania połączenia, szybkość transmisji danych oraz siłę sygnału radiowego. Możemy także odczytać, ile pakietów zostało wysłanych i ile zostało odebranych przez naszą sieć Wi-Fi.



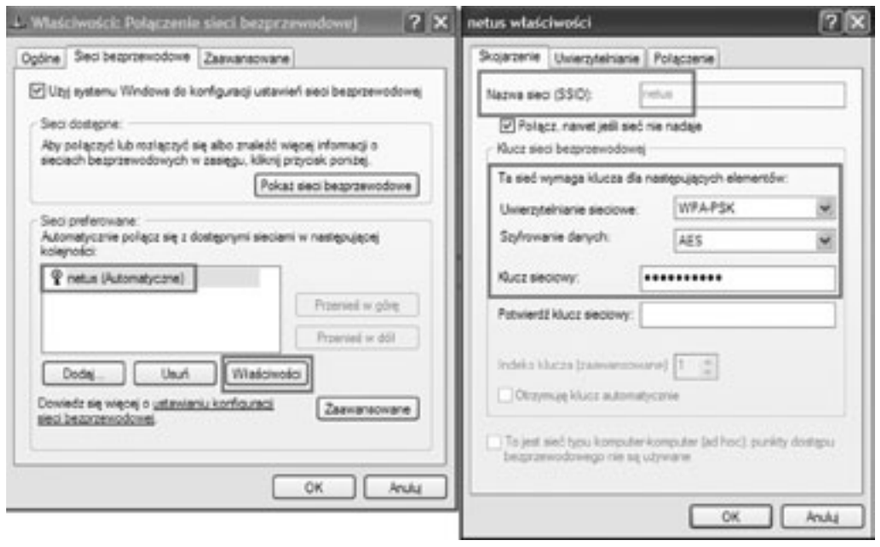
Rysunek 37.
Lista wykrytych sieci bezprzewodowych

Możemy również podejrzeć (rys. 37) wykryte przez kartę sieciową sieci bezprzewodowe i wybrać tę, którą wcześniej skonfigurowaliśmy (patrz rysunek 38 – efekt wyboru).



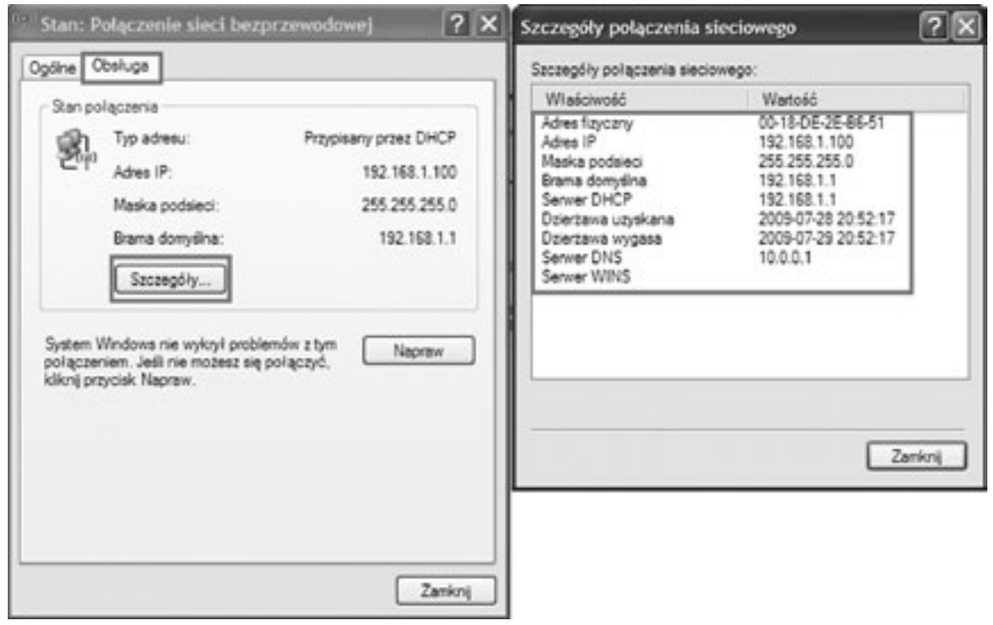
Rysunek 38.
Szczegółowe informacje o wykrytej i wybranej sieci bezprzewodowej

Po wybraniu przycisku Właściwości (rys. 38, lewa strona) możemy poznać szczegółowe informacje dotyczące wykrytych sieci bezprzewodowych, klikamy więc na przycisku OK z prawej strony.



Rysunek 39.
Podgląd podstawowej konfiguracji sieci bezprzewodowej

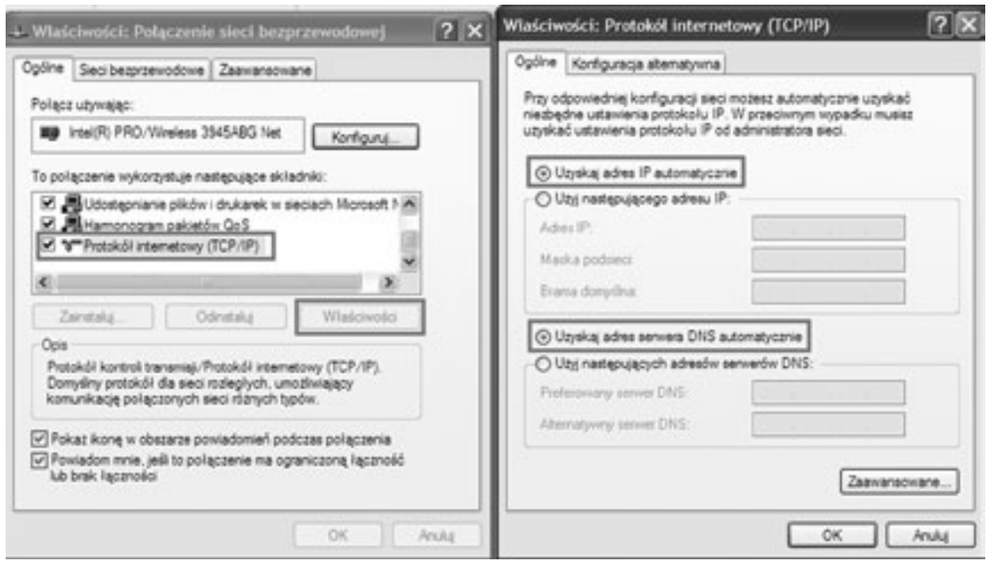
Na rysunku 39, z lewej strony widać sieci preferowane i metodę połączenia – automatyczną lub ręczną. Po zaznaczeniu wybranej sieci klikamy na przycisku Właściwości – ukazuje się podgląd ważnych informacji: nazwa sieci (SSID), protokoły służące uwierzytelnianiu sieciowemu, protokół szyfrowania danych, a także klucz sieciowy.



Rysunek 40.
Podgląd szczegółowych informacji o konfiguracji sieci bezprzewodowej

Jeśli natomiast wybierzemy zakładkę Obsługa z kategorii Ogólne (rys. 40), to otrzymamy następujące informacje: typ adresu (przypisany ręcznie czy przez usługę DHCP), adres IP, maskę podsieci oraz adres IP bramy domyślnej. Gdy klikniemy na przycisku Szczegóły, to ukażą się dodatkowe informacje (z prawej strony na rys. 40): adres fizyczny karty sieciowej (adres MAC), adres IP serwera DHCP, data uzyskania dzierżawy adresu IP i data jej wygaśnięcia, a także adres IP serwera DNS.

W zakładce Ogólne możemy również podejrzeć ustawienie protokołu TCP/IP. Wybieramy więc ten protokół, a następnie klikamy na przycisku Właściwości. Na ekranie, widocznym z prawej strony na rys. 41, możemy zdecydować, czy adres IP hosta, maskę podsieci i bramę domyślną przypiszemy ręcznie, czy zrobi to za nas usługa DHCP.



Rysunek 41.
Podgląd ustawień protokołu TCP/IP

LITERATURA

1. Engst A., Fleishman G., *Sieci bezprzewodowe. Praktyczny podręcznik*, Helion, Gliwice 2005
2. Krysiak K., *Sieci komputerowe. Kompendium*, Helion, Gliwice 2005
3. Mucha M., *Sieci komputerowe. Budowa i działanie*, Helion, Gliwice 2003
4. Ross J., *Sieci bezprzewodowe. Przewodnik po sieciach Wi-Fi i szerokopasmowych sieciach bezprzewodowych*, Wydanie II, Helion, Gliwice 2009

Podstawy działania wybranych usług sieciowych

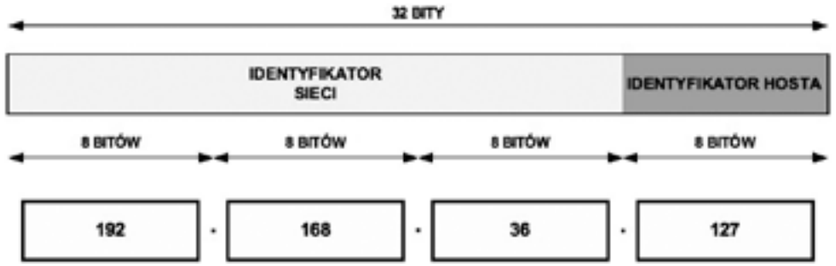
Dariusz Chaładyniak
Warszawska Wyższa Szkoła Informatyki
dchalad@wwsi.edu.pl

Streszczenie
Istnieje wiele dostępnych usług sieciowych, z których możemy korzystać, gdy mamy komputer wpięty do sieci komputerowej. Wykład omawia trzy wybrane usługi sieciowe, których zrozumienie opiera się na podstawowej wiedzy związanej z adresowaniem IP. Aby móc korzystać z dowolnych zasobów WWW musimy mieć publiczny adres IP, który może być współdzielony przez wiele komputerów z zastosowaniem translacji NAT (statycznej lub dynamicznej) lub translacji z przeciążeniem PAT. Adres IP dla naszego komputera może być przypisany ręcznie lub przydzielony dynamicznie poprzez usługę DHCP. Aby przeglądarka internetowa właściwie zinterpretowała adres domenowy, musi być dostępna usługa odwzorowująca ten adres na adres IP zrozumiały dla oprogramowania sieciowego.

Spis treści

1 PODSTAWY ADRESOWANIA IPV4

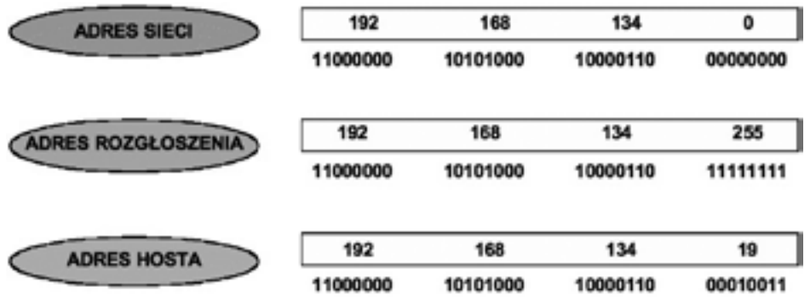
Format adresu IPv4



Rysunek 1.
Format adresu IP w wersji 4

Adres IPv4 jest 32-bitową liczbą binarną konwertowaną do notacji kropkowo-dziesiętnej. Składa się z identyfikatora sieci przydzielonego przez odpowiedni RIR (ang. *Regional Internet Registry*) oraz identyfikatora hosta (zarządzanego przez administratora sieciowego).

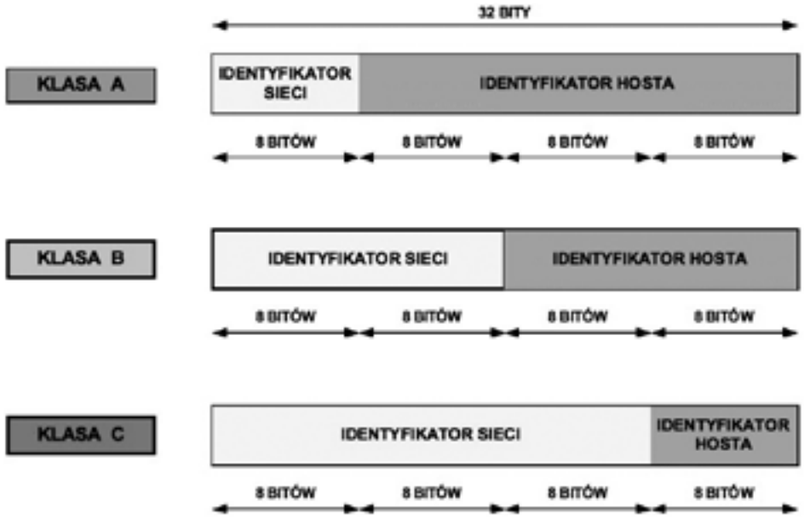
Rodzaje adresów IPv4



Rysunek 2.
Rodzaje adresów IP w wersji 4

Adres sieci charakteryzuje się tym, że w części hostowej są same zera. **Adres rozgłoszenia** jest rozpoznawalny po tym, że ma same jedynki w części hostowej. **Adres hosta** jest zakresem pomiędzy adresem sieci i adresem rozgłoszenia.

Klasy adresów IPv4
W adresowaniu klasowym wyróżniono pięć klas adresowych – A, B, C, D i E. Trzy pierwsze klasy – A, B i C – wykorzystuje się do adresacji hostów w sieciach komputerowych, natomiast klasy D i E są przeznaczone dla specyficznych zastosowań.



Rysunek 3.
Klasy adresów IP w wersji 4

Adresowanie klasowe

Klasa A

klasa A – pierwszy bit adresu jest równy 0, a następne 7 bitów określa sieć. Kolejne 24 bity wskazują komputer w tych sieciach. Adres rozpoczyna się liczbą między 1 i 127. Można zaadresować 126 sieci (adres 127.x.y-.z został zarezerwowany dla celów diagnostycznych jako adres loopback) po 16 777 214 (2^24 – 2) komputerów.

Rysunek 4.
Klasa A

Klasa B

klasa B – dwa pierwsze bity adresu to 1 i 0, a następne 14 bitów określa sieć. Kolejne 16 bitów identyfikuje komputer. Adres rozpoczyna się liczbą między 128 i 191. Można zaadresować 16 384 (2^14) sieci po 65 534 (2^16 – 2) komputery.

Rysunek 5.
Klasa B

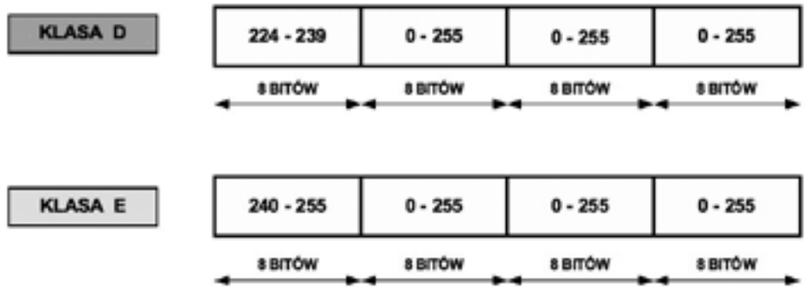
Klasa C

klasa C – trzy pierwsze bity adresu to 1, 1 i 0, a następnych 21 bitów identyfikuje adresy sieci. Ostatnie 8 bitów służy do określenia numeru komputerów w tych sieciach. Adres rozpoczyna się liczbą między 192 i 223. Może zaadresować 2 097 152 (2^21) sieci po 254 (2^8 – 2) komputery.

Rysunek 6.
Klasa C

Klasy D i E

klasa D – cztery pierwsze bity adresu to 1110. Adres rozpoczyna się liczbą między 224 i 239. Adresy tej klasy są stosowane do wysyłania rozgłoszeń typu multicast.



Rysunek 7.
Klasy D i E

klasa E – cztery pierwsze bity adresu to 1111. Adres rozpoczyna się liczbą między 240 i 255 (adres 255.255.255.255 został zarezerwowany dla celów rozgłoszeniowych). Adresy tej klasy są zarezerwowane dla przyszłych zastosowań.

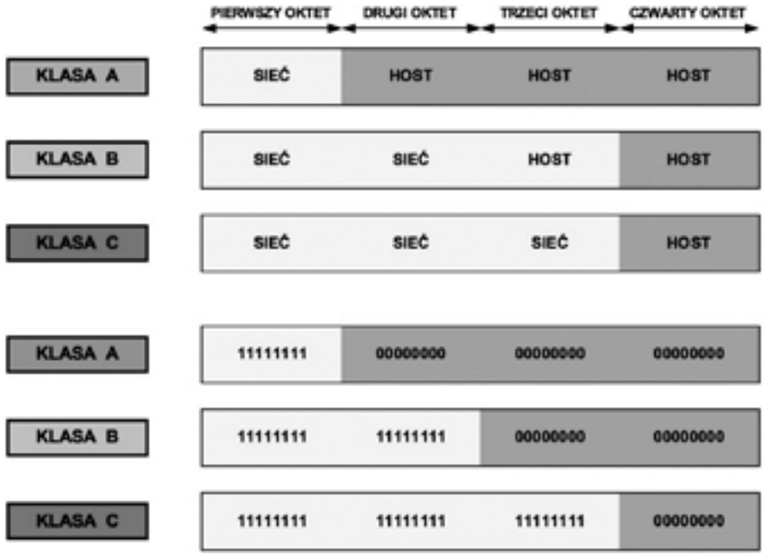
Wprowadzenie do adresowania bezklasowego

Podział adresów na klasy A, B i C, przy gwałtownym wzroście zapotrzebowania na nie, okazał się bardzo nieekonomiczny. Dlatego obecnie powszechnie jest stosowany model adresowania bezklasowego, opartego na tzw. **maskach podsieci**. W tym rozwiązaniu dla każdej podsieci definiuje się tzw. **maskę**, mającą podobnie jak adres IPv4 postać 32-bitowej liczby, ale o dosyć szczególnej budowie.

Na początku maski podsieci występuje ciąg jedynek binarnych, po których następuje ciąg samych zer binarnych. Część maski podsieci z samymi jedynkami określa sieć, natomiast część maski z zerami określa liczbę możliwych do zaadresowania hostów. Maskę podsieci zapisujemy podobnie jak adres IPv4 w notacji kropkowo-dziesiętnej.

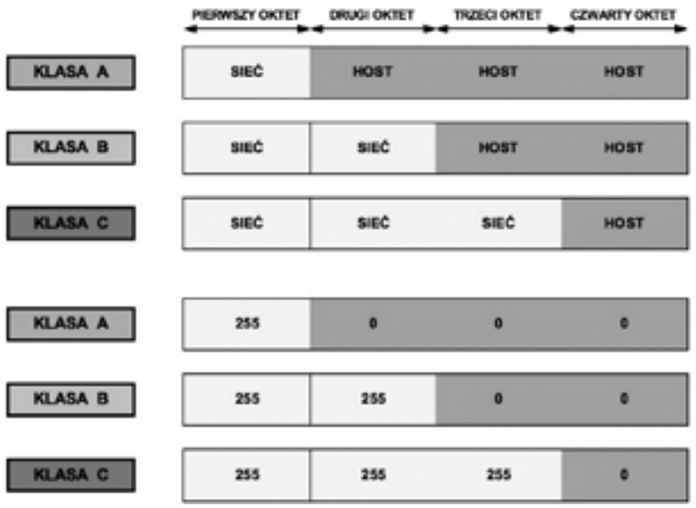
Maski podsieci można zapisywać w notacji binarnej lub dziesiętnej. W przypadku zapisu binarnego, w części identyfikatora sieci występują same jedynki, natomiast w części identyfikatora hosta znajdują się same zera.

Standardowe maski podsieci w postaci binarnej



Rysunek 8.
Standardowe maski podsieci w zapisie binarnym

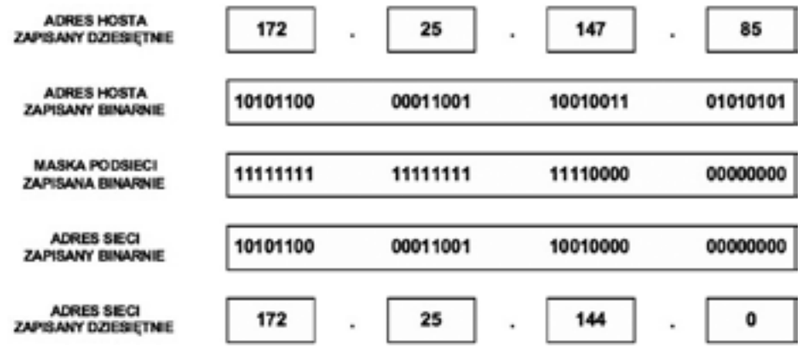
Standardowe maski podsieci w notacji dziesiętnej



Rysunek 9.
Standardowe maski podsieci w zapisie dziesiętnym

W przypadku notacji dziesiętnej, maski podsieci w części identyfikatora sieci mają wartość 255 natomiast w części identyfikatora hosta wartość 0. Na przykład standardowa maska podsieci w klasie A to 255.0.0.0, w klasie B to 255.255.0.0, a w klasie C to 255.255.255.0.

Określanie identyfikatora sieci



Rysunek 10.
Określanie identyfikatora sieci

Identyfikator sieci jest wykorzystywany do określenia, czy host docelowy znajduje się w sieci lokalnej czy rozległej.

Aby określić sieć, do której należy dowolny adres IPv4, najpierw zamieniamy zapis dziesiętny na binarny, zarówno adresu IP hosta, jak i jego maski podsieci. Następnie używając operacji logicznej koniunkcji AND porównujemy odpowiadające sobie bity IP hosta i maski podsieci. Wynik jest równy 1, gdy oba porównywane bity są równe 1. W przeciwnym wypadku wynik jest równy 0.

Na przykład, jaki jest identyfikator sieci dla hosta o adresie 172.25.147.85 z maską podsieci 255.255.240.0? Odpowiedź: należy zamienić obie liczby na ich binarne odpowiedniki i zapisać jeden pod drugim. Następnie wykonać operację AND dla każdego bitu i zapisać wynik. Otrzymany identyfikator sieci jest równy 172.25.144.0.

2 USŁUGA NAT I PAT

Adresy prywatne

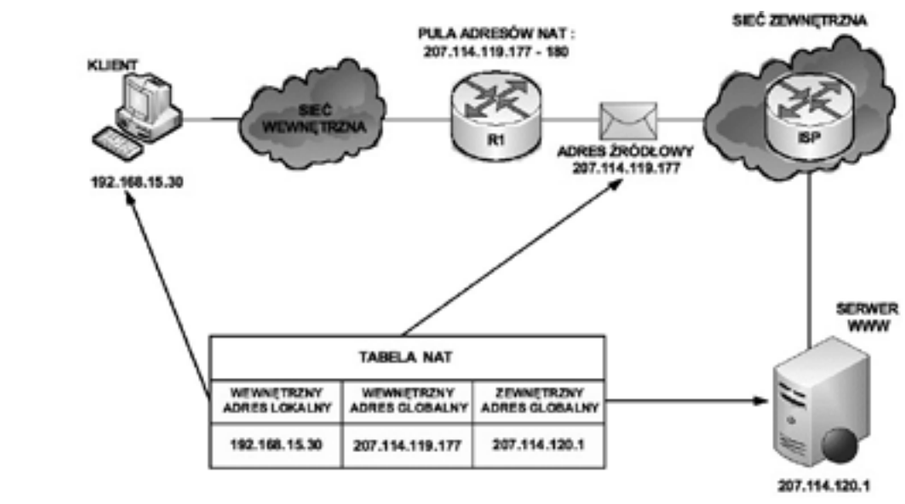
W dokumencie RFC 1918 wyróżniono trzy pule adresów IP przeznaczonych tylko do użytku prywatnego. Adresy te mogą być stosowane tylko i wyłącznie w sieci wewnętrznej. W zależności od tego, jak dużą sieć zamierzamy skonfigurować, wybieramy jedną z klas adresów (A, B lub C). Pakiety z takimi adresami nie są routowane przez Internet.

Tabela 1.
Dostępne zakresy prywatnych adresów IP

KLASA	ZAKRES ADRESÓW PRYWATNYCH RFC 1918	STANDARDOWA MASKA PODSIECI	ILOŚĆ SIECI	ILOŚĆ HOSTÓW NA SIEĆ	CAŁKOWITA ILOŚĆ HOSTÓW
A	10.0.0.0 – 10.255.255.255	255.0.0.0	1	16 777 214	16 777 214
B	172.16.0.0 – 172.31.255.255	255.255.0.0	16	65 534	1 048 544
C	192.168.0.0 – 192.168.255.255	255.255.255.0	256	254	65 024

Prywatne adresy IP są zarezerwowane i mogą zostać wykorzystane przez dowolnego użytkownika. Oznacza to, że ten sam adres prywatny może zostać wykorzystany w wielu różnych sieciach prywatnych. Router nie powinien nigdy routować adresów wymienionych w dokumencie RFC 1918. Dostawcy usług internetowych zazwyczaj konfigurują routery brzegowe tak, aby zapobiec przekazywaniu ruchu przeznaczonego dla adresów prywatnych. Zastosowanie mechanizmu NAT zapewnia wiele korzyści dla poszczególnych przedsiębiorstw i dla całego Internetu. Zanim opracowano technologię NAT, host z adresem prywatnym nie mógł uzyskać dostępu do Internetu. Wykorzystując mechanizm NAT, poszczególne przedsiębiorstwa mogą określić adresy prywatne dla niektórych lub wszystkich swoich hostów i zapewnić im dostęp do Internetu.

Działanie translacji NAT



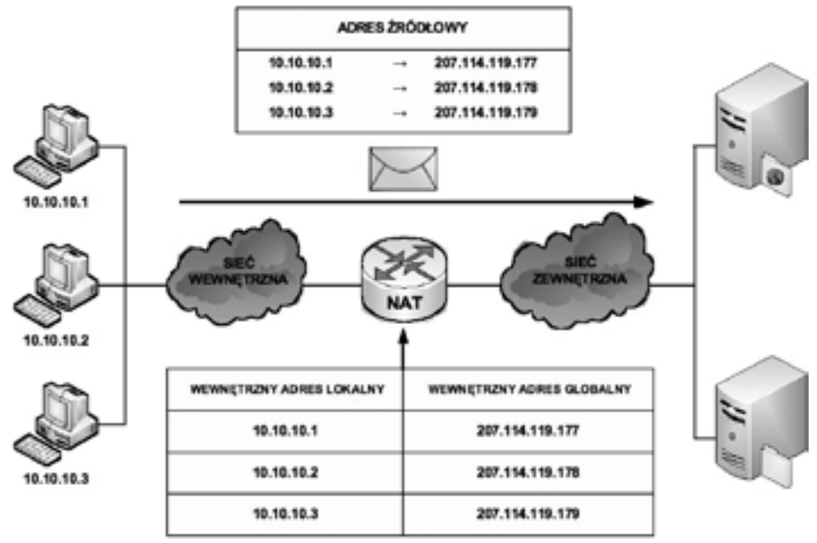
Rysunek 11.
Działanie translacji NAT

Na rysunku 11 wyjaśnione jest działanie usługi NAT (ang. *Network Address Translation*):

- Klient o adresie prywatnym 192.168.15.30 (wewnętrzny adres lokalny) zamierza otworzyć stronę WWW przechowywaną na serwerze o adresie publicznym 207.114.120.1 (zewnętrzny adres globalny).

- Komputer kliencki otrzymuje z puli adresów przechowywanych na routerze R1 publiczny adres IP (wewnętrzny adres globalny) 207.114.119.177.
- Następnie router ten wysyła pakiet o zmienionym adresie źródłowym do sieci zewnętrznej (router ISP), z której trafia do serwera WWW.
- Kiedy serwer WWW odpowiada na przypisany przez usługę NAT adres IP 207.114.119.177, pakiet powraca do routera R1, który na podstawie wpisów w tabeli NAT ustala, że jest to uprzednio przekształcony adres IP.
- Następuje translacja wewnętrznego adresu globalnego 207.114.119.177 na wewnętrzny adres lokalny 192.168.15.30, a pakiet przekazywany jest do stacji klienckiej.

Statyczna translacja NAT



Rysunek 12.
Statyczna translacja NAT

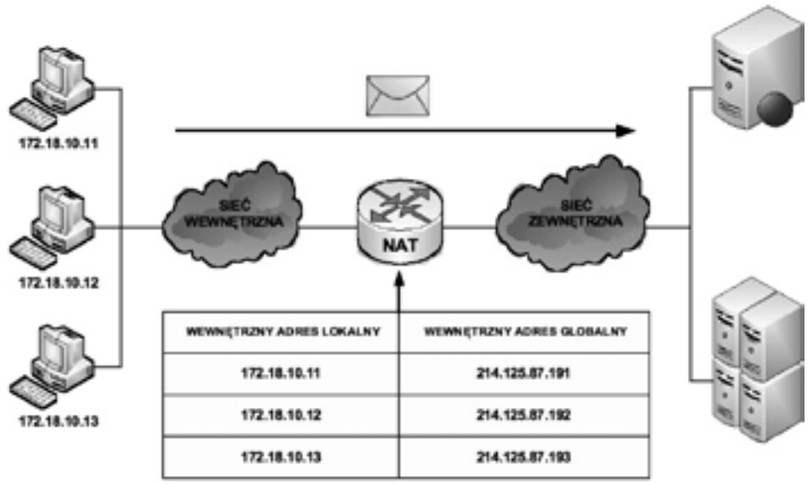
Statyczna translacja NAT (ang. *static NAT*) umożliwia utworzenie odwzorowania typu jeden-do-jednego pomiędzy adresami lokalnymi i globalnymi pomiędzy sieciami wewnętrzną i zewnętrzną. Jest to szczególnie przydatne w przypadku hostów, które muszą mieć stały adres dostępny z Internetu. Takimi wewnętrznymi hostami mogą być serwery lub urządzenia sieciowe w przedsiębiorstwie. W tym rozwiązaniu administrator ręcznie konfiguruje predefiniowane skojarzenia adresów IP. Ten typ translacji tak naprawdę nie ma nic wspólnego z oszczędzaniem przestrzeni adresowej IP, gdyż każdemu prywatnemu adresowi w sieci wewnętrznej trzeba przypisać adres publiczny w sieci zewnętrznej. Jednakże takie odwzorowanie daje gwarancję, że żaden przesyłany pakiet nie zostanie odrzucony z powodu braku dostępnej przestrzeni adresowej.

Na rysunku 12 widzimy, że trzem adresom prywatnym (10.10.10.1, 10.10.10.2, 10.10.10.3) zamapowano trzy adresy publiczne (odpowiednio 207.114.119.177, 207.114.119.178, 207.114.119.179).

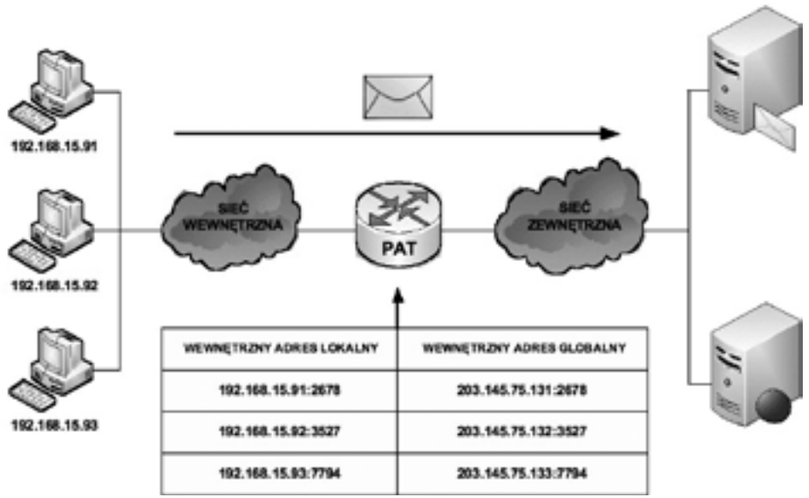
Dynamiczna translacja NAT

Dynamiczna translacja NAT (ang. *dynamic NAT*) (patrz rysunek 13) służy do odwzorowania prywatnego adresu IP na dowolny adres publiczny (z uprzednio zdefiniowanej puli). W translacji dynamicznej unikamy

stosowania dokładnie takiej samej puli adresów publicznych co prywatnych. Oznacza to, że z jednej strony możemy zaoszczędzić dostępną przestrzeń adresową, ale istnieje ryzyko braku gwarancji zamiany adresów w przypadku wyczerpania się puli adresów routowalnych. Z tego powodu na administratorze sieci spoczywa obowiązek zadbania o odpowiedni zakres puli adresów publicznych, aby możliwa była obsługa wszystkich możliwych translacji. Ponieważ nie wszyscy użytkownicy sieci komputerowej potrzebują jednoczesnego dostępu do zasobów zewnętrznych, można skonfigurować pulę adresów publicznych mniejszą od liczby adresów prywatnych. Dlatego w tym przypadku unikamy przypisywania wszystkim użytkownikom adresów routowalnych, jak w usłudze translacji statycznej NAT.



Rysunek 13.
Dynamiczna translacja NAT



Rysunek 14.
Translacja PAT

Translacja PAT

Translacja PAT (ang. *Port Address Translation*) (patrz rysunek 14) służy do odwzorowania wielu prywatnych adresów IP na jeden publiczny adres IP. Istnieje możliwość odwzorowania wielu adresów na jeden adres IP, ponieważ z każdym adresem prywatnym związany jest inny numer portu. W technologii PAT tłumaczone adresy są rozróżniane przy użyciu unikatowych numerów portów źródłowych powiązanych z globalnym adresem IP. Numer portu zakodowany jest na 16 bitach. Całkowita liczba adresów wewnętrznych, które mogą być przetłumaczone na jeden adres zewnętrzny, może teoretycznie wynosić nawet 65 536. W rzeczywistości do jednego adresu IP może zostać przypisanych około 4000 portów. W mechanizmie PAT podejmowana jest zawsze próba zachowania pierwotnego portu źródłowego. Jeśli określony port źródłowy jest już używany, funkcja PAT przypisuje pierwszy dostępny numer portu, licząc od początku zbioru numerów odpowiedniej grupy portów (0–511, 512–1023 lub 1024–65535). Gdy zabraknie dostępnych portów, a skonfigurowanych jest wiele zewnętrznych adresów IP, mechanizm PAT przechodzi do następnego adresu IP w celu podjęcia kolejnej próby przydzielenia pierwotnego portu źródłowego. Ten proces jest kontynuowany aż do wyczerpania wszystkich dostępnych numerów portów i zewnętrznych adresów IP.

Zalety translacji NAT i PAT

Do głównych zalet translacji adresów prywatnych na publiczne należą:

1. Eliminacja konieczności ponownego przypisania adresów IP do każdego hosta po zmianie dostawcy usług internetowych (ISP). Użycie mechanizmu NAT umożliwia uniknięcie zmiany adresów wszystkich hostów, dla których wymagany jest dostęp zewnętrzny, a to wiąże się z oszczędnościami czasowymi i finansowymi.
2. Zmniejszenie liczby adresów przy użyciu dostępnej w aplikacji funkcji multipleksowania na poziomie portów. Gdy wykorzystywany jest mechanizm PAT, hosty wewnętrzne mogą współużytkować pojedynczy publiczny adres IP podczas realizacji wszystkich operacji wymagających komunikacji zewnętrznej. W takiej konfiguracji do obsługi wielu hostów wewnętrznych wymagana jest bardzo niewielka liczba adresów zewnętrznych. Prowadzi to do oszczędności adresó IP.
3. Zwiększenie poziomu bezpieczeństwa w sieci. Ponieważ w przypadku sieci prywatnej nie są rozgłaszane wewnętrzne adresy ani informacje o wewnętrznej topologii, sieć taka pozostaje wystarczająco zabezpieczona, gdy dostęp zewnętrzny odbywa się z wykorzystaniem translacji NAT.

3 USŁUGA DHCP

Podstawy działania DHCP

Usługa DHCP (ang. *Dynamic Host Configuration Protocol*) działa w trybie klient-serwer i została opisana w dokumencie RFC 2131. Umożliwia ona klientom DHCP w sieciach IP uzyskiwanie informacji o ich konfiguracji z serwera DHCP. Użycie usługi DHCP zmniejsza nakład pracy wymagany przy zarządzaniu siecią IP. Najważniejszym elementem konfiguracji odbieranym przez klienta od serwera jest adres IP klienta. Klient DHCP wchodzi w skład większości nowoczesnych systemów operacyjnych, takich jak systemy Windows, Sun Solaris, Linux i MAC OS. Klient żąda uzyskania danych adresowych z sieciowego serwera DHCP, który zarządza przydzielaniem adresów IP i odpowiada na żądania konfiguracyjne klientów.

Serwer DHCP może odpowiadać na żądania pochodzące z wielu podsieci. Protokół DHCP działa jako proces serwera służący do przydzielania danych adresowych IP dla klientów. Klienci dzierżawią informacje pobrane z serwera na czas ustalony przez administratora. Gdy okres ten dobiega końca, klient musi zażądać nowego adresu. Zazwyczaj klient uzyskuje ten sam adres.

Administratorzy na ogół preferują serwery sieciowe z usługą DHCP, ponieważ takie rozwiązanie jest skalowalne i łatwo nim zarządzać. Konfigurują oni serwery DHCP tak, aby przydzielane były adresy ze zdefiniowanych pul adresów. Na serwerach DHCP mogą być dostępne także inne informacje: adresy serwerów DNS, adresy serwerów WINS i nazwy domen. W większości serwerów DHCP administratorzy mogą także zdefiniować adresy MAC obsługiwanych klientów i automatycznie przypisywać tym klientom zawsze te same adresy IP.

Rysunek 15.
Działanie usługi dynamicznego przydzielania adresów IP

Protokołem transportowym wykorzystywanym przez protokół DHCP jest **UDP** (ang. *User Datagram Protocol*). Klient wysyła komunikaty do serwera na port 67. Serwer wysyła komunikaty do klienta na port 68.

Sposoby przydzielania adresów IP

Istnieją trzy mechanizmy przydzielania adresów IP klientom:

1. **Alokacja automatyczna** – serwer DHCP przypisuje klientowi stały adres IP.
2. **Alokacja ręczna** – adres IP jest przydzielany klientowi przez administratora. Serwer DHCP przesyła adres do klienta.
3. **Alokacja dynamiczna** – serwer DHCP dzierżawi klientowi adres IP na pewien ograniczony czas.

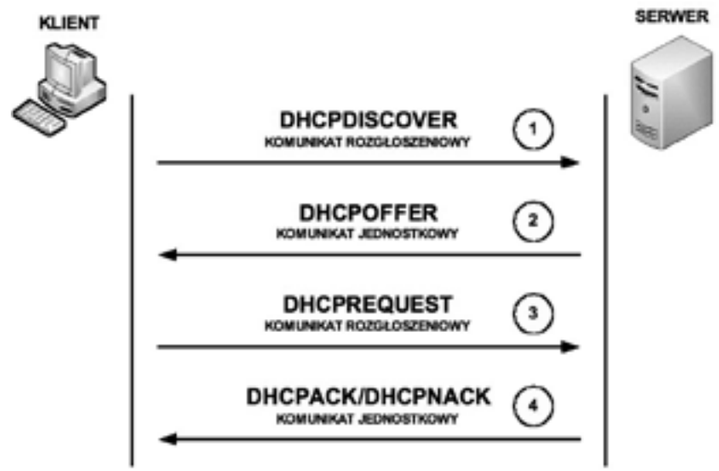
Serwer DHCP tworzy pule adresów IP i skojarzonych z nimi parametrów. Pule przeznaczone są dla poszczególnych logicznych podsieci IP. Dzięki temu jeden klient IP może uzyskiwać adresy od wielu serwerów DHCP i może być przenoszony. Jeśli klient uzyska odpowiedź od wielu serwerów, może wybrać tylko jedną z ofert.

Wymiana komunikatów protokołu DHCP

W procesie konfiguracji klienta DHCP wykonywane są następujące działania:

1. Na kliencie, który uzyskuje dostęp do sieci, musi być skonfigurowany protokół DHCP. Klient wysyła do serwera żądanie uzyskania konfiguracji IP. Czasami klient może zaproponować adres IP, na przykład wówczas, gdy żądanie dotyczy przedłużenia okresu dzierżawy adresu uzyskanego od serwera DHCP wcześniej. Klient wyszukuje serwer DHCP, wysyłając komunikat rozgłoszeniowy DHCPDISCOVER.
2. Po odebraniu tego komunikatu serwer określa, czy może obsłużyć określone żądanie przy użyciu własnej bazy danych. Jeśli żądanie nie może zostać obsłużone, serwer może przekazać odebrane żądanie dalej, do innego serwera DHCP. Jeśli serwer DHCP może obsłużyć żądanie, do klienta jest wysyłana oferta z konfiguracją IP w postaci komunikatu transmisji pojedynczej (unicast) DHCPOFFER. Komunikat DHCPOFFER zawiera propozycję konfiguracji, która może obejmować adres IP, adres serwera DNS i okres dzierżawy.

3. Jeśli określona oferta jest odpowiednia dla klienta, wysyła on inny komunikat rozgłoszeniowy, DHCPREQUEST, z żądaniem uzyskania tych konkretnych parametrów IP. Wykorzystywany jest komunikat rozgłoszeniowy, ponieważ pierwszy komunikat DHCPDISCOVER mógł zostać odebrany przez wiele serwerów DHCP. Jeśli wiele serwerów wyśle do klienta swoje oferty, dzięki komunikatowi rozgłoszeniowemu DHCPREQUEST serwery te będą mogły poznać ofertę, która została zaakceptowana. Zazwyczaj akceptowana jest pierwsza odebrana oferta.



Rysunek 16.
Wymiana komunikatów protokołu DHCP

4. Serwer, który odbierze sygnał DHCPREQUEST, publikuje określoną konfigurację, wysyłając potwierdzenie w postaci komunikatu transmisji pojedynczej DHCPACK. Istnieje możliwość (choć jest to bardzo mało prawdopodobne), że serwer nie wyśle komunikatu DHCPACK. Taka sytuacja może wystąpić wówczas, gdy serwer wydzierżawi w międzyczasie określoną konfigurację innemu klientowi. Odebranie komunikatu DHCPACK upoważnia klienta do natychmiastowego użycia przypisanego adresu.

   Jeśli klient wykryje, że określony adres jest już używany w lokalnym segmencie, wysyła komunikat DHCPDECLINE i cały proces zaczyna się od początku. Jeśli po wysłaniu komunikatu DHCPREQUEST klient otrzyma od serwera komunikat DHCPNACK, proces rozpocznie się od początku.

   Gdy klient nie potrzebuje już adresu IP, wysyła do serwera komunikat DHCPRELEASE.

   Zależnie od reguł obowiązujących w przedsiębiorstwie, użytkownik końcowy lub administrator może przypisać dla hosta statyczny adres IP dostępny w puli adresów na serwerze DHCP.

Automatyczna konfiguracja adresów IP

Aby automatycznie skonfigurować adresy IP (adres hosta, maska podsieci, brama domyślna, główny serwer DNS, zapasowy serwer DNS) w systemie Windows XP należy wykonać kolejne kroki:

- Klikamy przycisk Start, a następnie wybieramy zakładkę Panel sterowania. W oknie, które się pojawi (rys. 17), klikamy w kategorię Połączenia sieciowe i internetowe.

Rysunek 17.
Początek automatycznego konfigurowania adresów IP

- Z kategorii Połączenia sieciowe i internetowe wybieramy Połączenia sieciowe (patrz rys. 18).

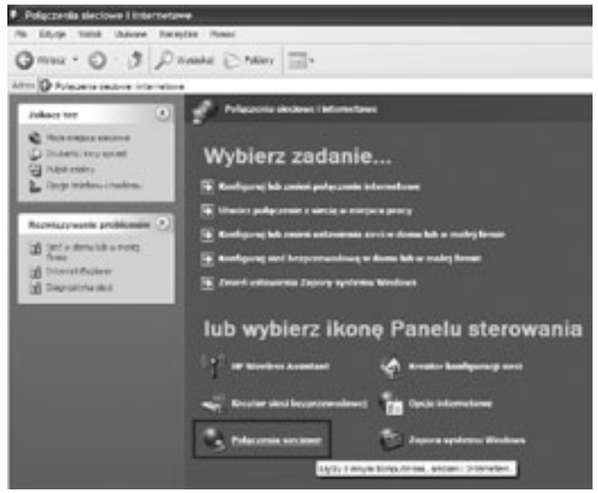
- W kategorii Połączenia sieciowe wybieramy Połączenie lokalne (patrz rys. 19).

- W oknie na rysunku 20 jest ukazany podgląd stanu Połączenia lokalnego, z którego możemy odczytać: stan połączenia, czas trwania połączenia, szybkość połączenia, a także jego aktywność (liczbę pakietów wysłanych i odebranych). W oknie tym klikamy na zakładkę Właściwości.

- Po wybraniu zakładki Właściwości ukazuje nam się kolejne okno (rys. 21), w którym wybieramy składnik Protokół internetowy (TCP/IP), a następnie klikamy w zakładkę Właściwości.

- Po wybraniu składnika Protokół internetowy (TCP/IP) i kliknięciu w zakładkę Właściwości otwiera się okno (rys. 22), w którym wybieramy następujące opcje: Uzyskaj adres IP automatycznie oraz Uzyskaj adres serwera DNS automatycznie. Po wybraniu tych opcji zostaną nadane automatycznie następujące adresy IP: adres IP hosta, jego maska podsieci, adres IP bramy domyślnej, adres IP preferowanego serwera DNS oraz adres IP alternatywnego serwera DNS.

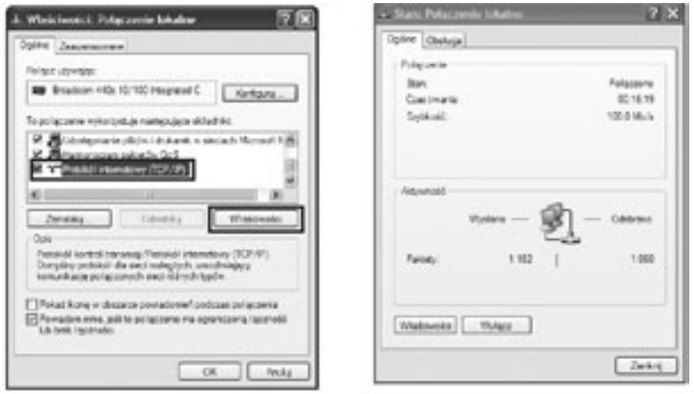
- Po kliknięciu w zakładkę Zaawansowane w oknie Właściwości: Protokół internetowy (TCP/IP) otrzymujemy podgląd w zaawansowane ustawienia stosu protokołów TCP/IP, w którym możemy zauważyć, że jest włączony serwer DHCP (patrz rys. 23).



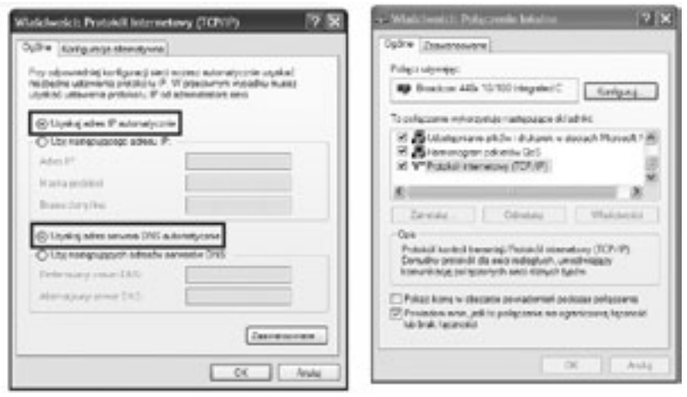
Rysunek 18.
Wybór wśród połączeń sieciowych i internetowych

Rysunek 19.
Wybór połączenia lokalnego wśród połączeń sieciowych

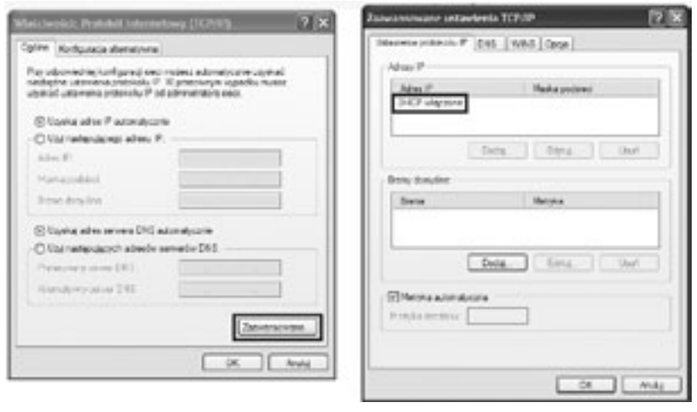
Rysunek 20.
Okno ukazujące stan połączenia lokalnego



Rysunek 21.
Okno z właściwościami połączenia lokalnego

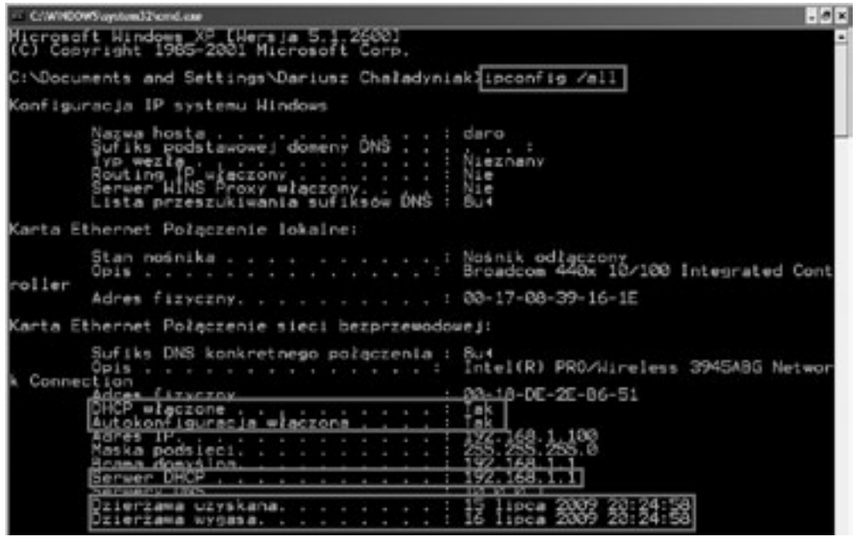


Rysunek 22.
Odznaczenie automatycznych wyborów adresów IP



Rysunek 23.
Efekt wybrania zakładki Zaawansowane w oknie Właściwości Protokołu internetowego TCP/IP

Testowanie konfiguracji usługi DHCP



Rysunek 24.
Testowanie konfiguracji usługi DHCP

Aby przetestować konfigurację usługi DHCP wydajemy polecenie ipconfig z opcją all. W wyniku jego wykonania otrzymujemy informację, czy usługa DHCP jest włączona i czy włączona jest jej autokonfiguracja. Ponadto otrzymujemy informację o adresie IP serwera DHCP (w tym przypadku – 192.168.1.1) oraz daty: uzyskania dzierżawy usługi DHCP i jej wygaśnięcia (rys. 24).

4 USŁUGA DNS

Adresy domenowe

Posługiwanie się adresami IP jest bardzo niewygodne dla człowieka, ale niestety oprogramowanie sieciowe wykorzystuje je do przesyłania pakietów z danymi. Aby ułatwić użytkownikom sieci komputerowych korzystanie z usług sieciowych, obok adresów IP wprowadzono tzw. **adresy domenowe** (symboliczne). Nie każdy komputer musi mieć taki adres. Są one z reguły przypisywane tylko komputerom udostępniającym w Internecie jakieś usługi. Umożliwia to użytkownikom chcącym z nich skorzystać łatwiejsze wskazanie konkretnego serwera. Adres symboliczny zapisywany jest w postaci ciągu nazw, tzw. **domen**, które są rozdzielone kropkami, podobnie jak w przypadku adresu IP. Części adresu domenowego nie mają jednak żadnego związku z poszczególnymi fragmentami adresu IP – chociażby ze względu na fakt, że o ile adres IP składa się zawsze z czterech części, o tyle adres domenowy może ich mieć różną liczbę – od dwóch do siedmiu lub jeszcze więcej. Kilka przykładowych adresów domenowych:
http://www.wwsi.edu.pl
http://www.onet.pl
http://www.microsoft.com
ftp://public.wwsi.edu.pl

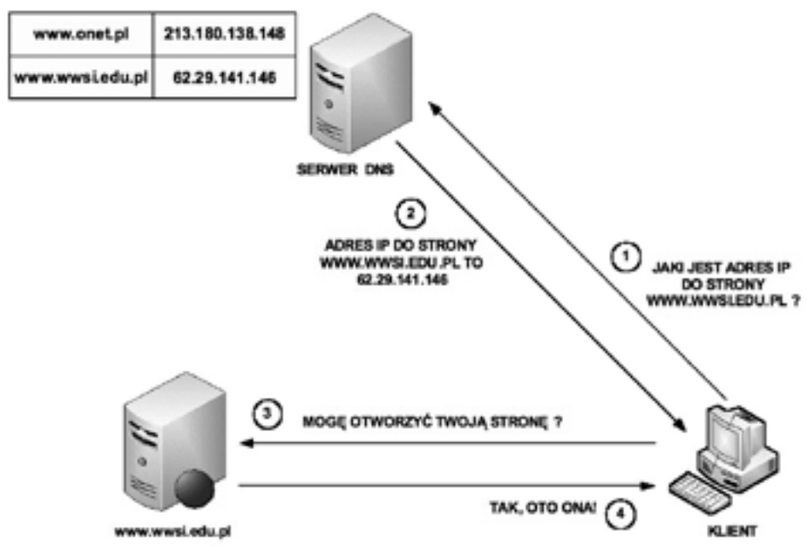
http://www.nask.pl
http://www.mf.gov.pl/

Domeny

Odwrotnie niż adres IP, adres domenowy czyta się od tyłu. Ostatni jego fragment, tzw. domena najwyższego poziomu (ang. *top-level domain*), jest z reguły dwuliterowym oznaczeniem kraju (np. .pl, .de). Jedynie w USA dopuszcza się istnienie adresów bez oznaczenia kraju na końcu. W tym przypadku domena najwyższego poziomu opisuje „branżową" przynależność instytucji, do której należy dany komputer. Może to być:

com/co – firmy komercyjne (np. Microsoft, IBM, Intel);
edu/ac – instytucje naukowe i edukacyjne (np. uczelnie);
gov – instytucje rządowe (np. Biały Dom, Biblioteka Kongresu, NASA, Sejm RP);
mil – instytucje wojskowe (np. MON);
org – wszelkie organizacje społeczne i inne instytucje typu *non-profit*;
int – organizacje międzynarodowe niedające się zlokalizować w konkretnym państwie (np. NATO);
net – firmy i organizacje zajmujące się administrowaniem i utrzymywaniem sieci komputerowych (np. EARN);
biz – biznes;
info – informacje;
name – nazwy indywidualne;
pro – zawody.

Działanie usługi DNS



Rysunek 25.
Przykład działania usługi DNS

Działanie usługi DNS sprowadza się do następujących kolejnych czynności (patrz rys. 25):

1. Klient z przeglądarką internetową pragnie otworzyć stronę www.wwsi.edu.pl przechowywaną na serwerze WWW. Z uwagi, że oprogramowanie sieciowe wymaga adresu IP, klient wysyła zapytanie do serwera DNS o adres IP dla żądanej strony WWW.
2. Serwer DNS na podstawie odpowiednich wpisów w swojej tablicy DNS odsyła klientowi odpowiedź, że stronie www.wwsi.edu.pl odpowiada adres IP o wartości 62.29.141.146.
3. Klient po otrzymaniu właściwego adresu IP wysyła do serwera WWW zapytanie o możliwość otwarcia strony www.wwsi.edu.pl.
4. Serwer WWW po zweryfikowaniu właściwego skojarzenia strony WWW z adresem IP odsyła klientowi zgodę na otwarcie żądanej strony internetowej.

LITERATURA

1. Dye M.A., McDonald R., Rufi A.W., *Akademia sieci Cisco. CCNA Exploration. Semestr 1*, Mikom, Warszawa 2008
2. Graziani R., Vachon B., *Akademia sieci Cisco. CCNA Exploration. Semestr 4*, Mikom, Warszawa 2009
3. Komar B., *TCP/IP dla każdego*, Helion, Gliwice 2002
4. Krysiak K., *Sieci komputerowe. Kompendium*, Helion, Gliwice 2005
5. Mucha M., *Sieci komputerowe. Budowa i działanie*, Helion, Gliwice 2003
6. Odom W., Knot T., *CCNA semestr 1. Podstawy działania sieci*, Mikom, Warszawa 2007

Podstawy bezpieczeństwa sieciowego

Dariusz Chaładyniak
Warszawska Wyższa Szkoła Informatyki
dchalad@wwsi.edu.pl

Streszczenie

Bezpieczeństwo danych przesyłanych w sieciach komputerowych jest jednym z najważniejszych zadań współczesnej teleinformatyki. Wykład przedstawia podstawowe rodzaje złośliwego oprogramowania (wirusy, trojany, robaki) oraz wybrane programy antywirusowe (skanery, monitory, szczepionki). Opisano także najczęściej spotykane metody ataków na systemy i sieci komputerowe (zewnętrzne, wewnętrzne, tradycyjne, rozproszone) oraz ich rodzaje (DoS, DDoS, phishing, spam). Przedstawione będą ponadto wybrane narzędzia i aplikacje do zabezpieczania danych, działanie systemów wykrywania włamań oraz metody przeciwdziałania atakom sieciowym z wykorzystaniem zapór ogniowych (sprzętowych i programowych).

Spis treści

1 ZŁOŚLIWE OPROGRAMOWANIE

1.1 RODZAJE ZŁOŚLIWEGO OPROGRAMOWANIA

Złośliwe oprogramowanie (ang. *malicious software*) to programy, które muszą zostać wprowadzone do komputera użytkownika. Mogą one uszkodzić system, zniszczyć dane, a także uniemożliwić dostęp do sieci, systemów lub usług. Mogą one też wykraść dane lub informacje osobiste ze stacji użytkownika i przesłać je samoczynnie do przestępców. W większości przypadków umieją same się replikować i rozprzestrzeniać na inne hosty dołączone do sieci. Czasem techniki te są używane w połączeniu z socjotechniką, aby oszukać nieostrożnego użytkownika, by ten nieświadomie uruchomił taki atak. Przykładami złośliwego oprogramowania są wirusy, robaki oraz konie trojańskie.

Wirus jest programem, który działa i rozprzestrzenia się przez modyfikowanie innych programów lub plików. Wirus nie może uruchomić się sam, musi zostać uaktywniony. Po uaktywnieniu, może nie robić nic poza replikacją i rozprzestrzenianiem się. Nawet prosty typ wirusa jest niebezpieczny, gdyż może szybko zużyć całą dostępną pamięć komputera i doprowadzić system do zatrzymania. Groźniejszy wirus, przed rozprzestrzenieniem się, może usunąć lub uszkodzić pliki. Wirusy mogą być przenoszone przez załączniki poczty elektronicznej, pobierane pliki, komunikatory, a także dyskietki, płyty CD/DVD lub urządzenia USB.

Rodzaje wirusów komputerowych:

1. **Pasożytnicze** – wykorzystują swoje ofiary do transportu;
2. **Polimorficzne** – mogą zmieniać swój kod;
3. **Wirusy plików wsadowych** – wykorzystują do transportu pliki z rozszerzeniem .bat.

Najbardziej znane wirusy to: Chernobyl (CIH), Christmas Tree.

Robak (ang. *worm*) jest podobny do wirusa, lecz w odróżnieniu od niego nie musi dołączać się do istniejącego programu. Robak używa sieci do rozsyłania swych kopii do podłączonych hostów. Robaki mogą działać samodzielnie i szybko się rozprzestrzeniać. Nie wymagają aktywacji czy ludzkiej interwencji. Samorozprzestrzeniające się robaki sieciowe są o wiele groźniejsze niż pojedynczy wirus, gdyż mogą szybko zainfekować duże obszary Internetu. Najbardziej znane robaki to: I Love You, Melissa, Mydoom, Netsky.

Koń trojański (ang. *trojan horse*), zwany również **trojanem**, jest programem, który nie replikuje się samodzielnie. Wygląda jak zwykły program, lecz w rzeczywistości jest narzędziem ataku. Idea działania konia trojańskiego polega na zmyleniu użytkownika, by ten uruchomił jego kod myśląc, że uruchamia bezpieczny program. Koń trojański jest zwykle mało szkodliwy, ale może zupełnie zniszczyć zawartość twardego dysku. Trojany często tworzą furtkę dla hakerów – pełny dostęp do zasobów komputera. Najbardziej znane trojany to: Connect4, Flatley Trojan, Poison Ivy.

Bomba logiczna (ang. *logical bomb*), w odróżnieniu od konia trojańskiego, nie uruchamia ukrytego złośliwego oprogramowania od razu tylko w odpowiednim czasie (np. po zajściu określonego zdarzenia lub po kilkukrotnym uruchomieniu wybranej aplikacji).

Exploit jest programem wykorzystującym błędy programistyczne i przejmującym kontrolę nad działaniem procesu.

Keylogger jest oprogramowaniem, mającym na celu wykradanie haseł poprzez przejęcie kontroli nad obsługą klawiatury.

Ransomware (ang. *ransom* – okup) jest aplikacją wnikającą do atakowanego komputera, a następnie szyfrującą dane jego właściciela. Perfidia tego złośliwego oprogramowania polega na zostawieniu odpowiedniej notatki z instrukcją, co musi zrobić właściciel zainfekowanego komputera, aby odzyskać dane.

Rootkit jest programem ułatwiającym włamanie do systemu komputerowego poprzez ukrycie niebezpiecznych plików i procesów mających kontrolę nad systemem. Wykrycie takiego programu w zainfekowanym komputerze jest bardzo trudne, gdyż jest on w stanie kontrolować pracę specjalistycznych narzędzi do jego wykrywania. Najbardziej znane to: Hacker Defender, CD Sony Rootkit.

Spyware to złośliwe oprogramowanie mające na celu szpiegowanie działań użytkownika komputera. Zadaniem spyware jest gromadzenie informacji o użytkowniku (adresy stron internetowych odwiedzanych przez użytkownika, dane osobowe, numery kart kredytowych i płatniczych, hasła, adresy e-mail). Najbardziej znane spyware to: Gator, Cydoor, Save Now.

Stealware jest oprogramowaniem okradającym nieświadomego użytkownika poprzez śledzenie jego działań. Instalacja takiego programu odbywa się bez wiedzy i zgody użytkownika za pomocą odpowiednio spreparowanych wirusów komputerowych, robaków lub stron WWW wykorzystujących błędy i luki w przeglądarkach internetowych. Stealware w przypadku stwierdzenia próby płatności przez Internet podmienia numer konta, na które zostaną wpłacone pieniądze.

1.2 RODZAJE PROGRAMÓW ANTYWIRUSOWYCH

Poniżej przedstawiamy wybrane rodzaje programów antywirusowych.

Skaner (ang. *scanner*) należy do najstarszych i najprostszych sposobów ochrony przed wirusami komputerowymi. Zasada działania skanera polega na wyszukiwaniu pewnej sekwencji bajtów w zadanym ciągu danych. Skaner jest tym skuteczniejszy, im wirus zawiera w sobie bardziej charakterystyczny napis lub ciąg bajtów.

Monitor (ang. *resident monitor*) to oprogramowanie antywirusowe zainstalowane w systemie operacyjnym jako program rezydentny. Skuteczność monitora zależy od tego, czy przejął on kontrolę nad systemem przed działaniem wirusa, czy po jego działaniu oraz od tego, jak głęboko wnika on w system operacyjny.

Szczepionka (ang. *disinfector*) jest oprogramowaniem antywirusowym, działającym przeciwko konkretnym infekcjom. Po wykryciu wirusa i poddaniu odpowiedniej analizie jego kodu można zdefiniować pewne właściwości umożliwiające przygotowanie właściwej szczepionki.

Program zliczający sumy kontrolne (ang. *integrity checker*) przy pierwszym uruchomieniu dokonuje odpowiednich obliczeń dla plików zgromadzonych na dysku, a następnie wykorzystuje te dane, aby porównać z bieżąco wyliczoną sumą kontrolną i na tej podstawie stwierdzić ewentualną obecność wirusa.

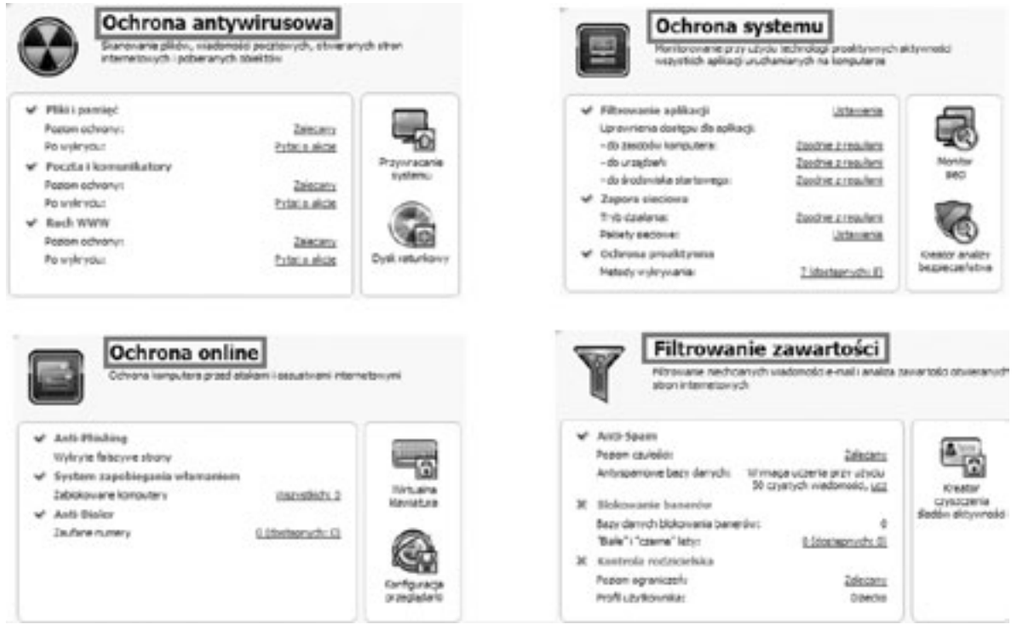
1.3 PROFILAKTYKA ANTYWIRUSOWA

Jedną z najlepszych metod zabezpieczenia się przed wirusami komputerowymi jest posiadanie najnowszego oprogramowania do ich zwalczania. Na rysunku 1 pokazano zakładkę Centrum zabezpieczeń w Panelu sterowania z włączoną funkcją ochrony przed wirusami.

Rysunek 1.
Zakładka Centrum zabezpieczeń w systemie Windows XP z informacją o włączonej ochronie przed wirusami

Jednym z najpopularniejszych i najbardziej skutecznych programów antywirusowych jest Kaspersky Internet Security 2009 (rys. 2). Aplikacja ta kompleksowo chroni komputer przed złośliwym oprogramowaniem, oszustwami internetowymi oraz nieautoryzowanym dostępem. Prowadzi ochronę antywirusową (skanowanie plików, wiadomości pocztowych, otwieranych stron internetowych i pobieranych obiektów), ochronę systemu (filtrowanie aplikacji, zapora sieciowa, ochrona proaktywna), ochronę *on-line* (ochrona komputera przed atakami i oszustwami internetowymi), a także filtrowanie zawartości (filtrowanie niechcianych wiadomości e-mail, analiza zawartości otwieranych stron internetowych) – patrz rysunek 3.

Rysunek 2.
Program antywirusowy Kaspersky Internet Security 2009



Rysunek 3.
Funkcje programu Kaspersky Internet Security 2009

2 WYBRANE ATAKI NA SIECI TELEINFORMATYCZNE

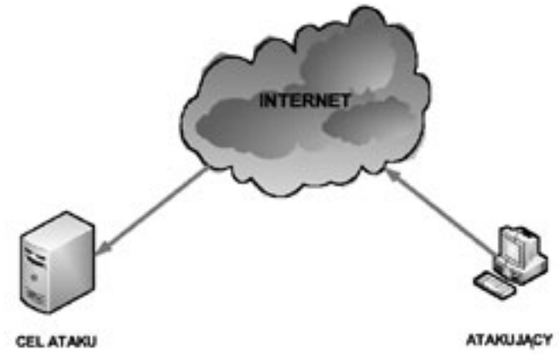
2.1 SPOSOBY ATAKOWANIA SIECI

Sieć można atakować na wiele sposobów:

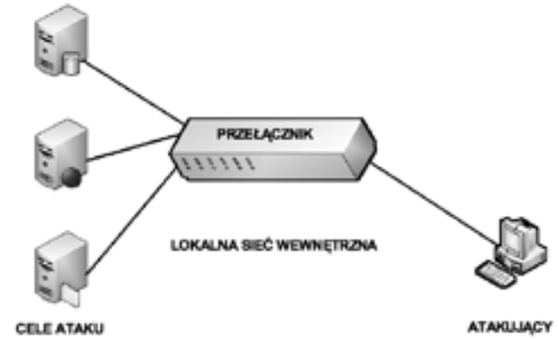
- atak zewnętrzny;
- atak wewnętrzny;
- atak tradycyjny;
- atak przy pomocy węzłów pośredniczących;
- atak rozproszony.

Atak zewnętrzny (rys. 4) jest powodowany przez osoby, które nie pracują w danej organizacji. Atakujący z zewnątrz toruje sobie drogę do sieci głównie przez Internet, łącza bezprzewodowe lub usługi wdzwaniane.

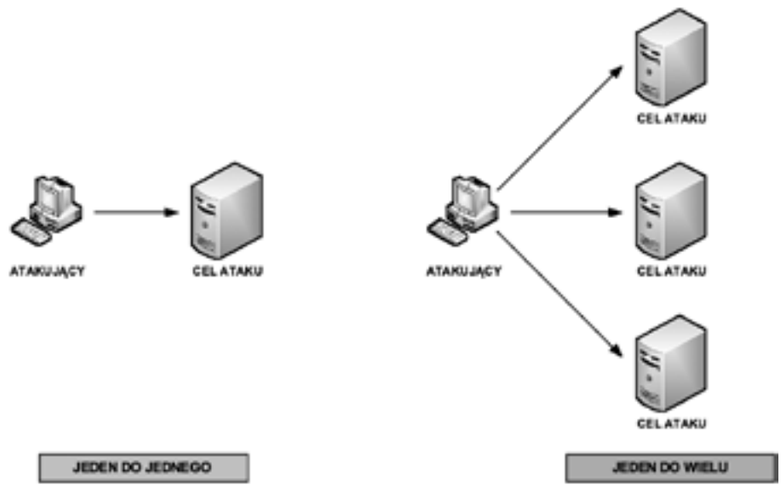
Atak wewnętrzny (rys. 5) może przeprowadzić ktoś, kto ma dostęp do sieci, czyli posiada konto lub ma dostęp fizyczny. Atakujący przeważnie zna ludzi oraz politykę wewnętrzną firmy. Nie wszystkie wewnętrzne ataki są celowe. W niektórych przypadkach zagrożenie wewnętrzne może powodować niefrasobliwy pracownik, który ściągnie i uruchomi wirusa, a następnie nieświadomie wprowadzi go do wnętrza sieci. Większość firm wydaje znaczące sumy na ochronę przed zewnętrznymi atakami, mimo iż gros zagrożeń pochodzi ze źródeł wewnętrznych. Jak podają statystyki, dostęp z wewnątrz i nadużycie systemów komputerowych stanowi ok. 70% zgłoszonych naruszeń bezpieczeństwa.



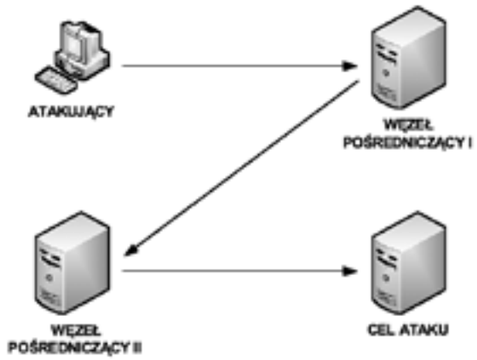
Rysunek 4.
Przykład ataku z sieci zewnętrznej



Rysunek 5.
Przykład ataku z sieci wewnętrznej



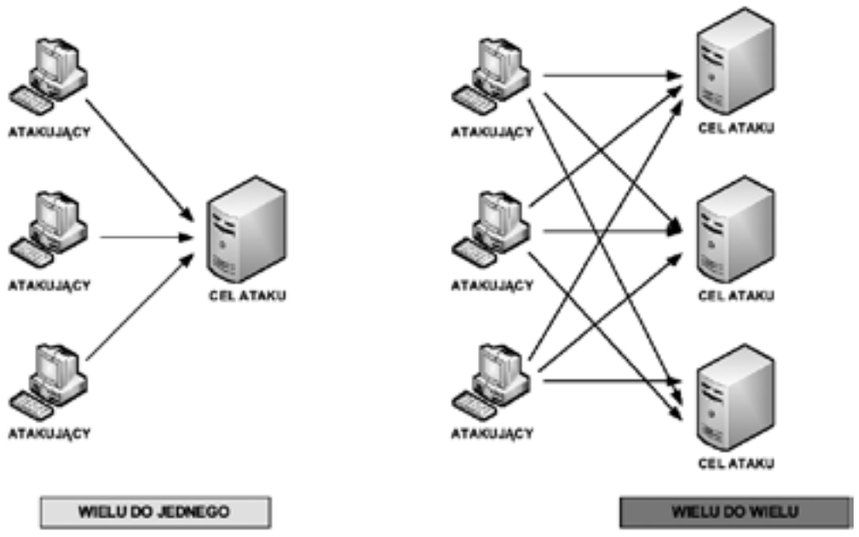
Rysunek 6.
Przykłady ataków tradycyjnych



Rysunek 7.
Przykład ataku przy udziale węzłów pośredniczących

Atak tradycyjny (rys. 6) polega na atakowaniu z jednego komputera jednego lub wielu hostów sieciowych.

Często zdarza się, że włamywacze nie atakują bezpośrednio, a korzystają z komputerów ofiar dla ukrycia prawdziwego źródła ataku oraz utrudnienia ich odnalezienia. Jak widać na rysunku 7, intruz korzysta z kilku węzłów pośredniczących tak, aby atakowany obiekt zinterpretował je jako źródła ataków.



Rysunek 8.
Przykłady ataków rozproszonych

Atak rozproszony (rys. 8) polega na zainicjowaniu przez atakującego wielu jednoczesnych ataków na jeden lub wiele celów. Zwykle następuje on w dwóch fazach. Początkowo atakujący musi przygotować węzły, z których atak taki mógłby być przeprowadzony. Polega to na ich znalezieniu i zainstalowaniu oprogramowania, które będzie realizowało właściwą fazę ataku rozproszonego. Cechą charakterystyczną drugiej fazy jest wysyłanie pakietów przez atakującego z węzłów pośredniczących, a nie z hosta atakującego. Ataki rozproszone

przynoszą atakującemu korzyści w postaci utajenia źródła ataku, zmasowanej siły ataku, poszerzenia bazy wiedzy na temat atakowanego celu i wreszcie trudności w jego zatrzymaniu.

2.2 RODZAJE WŁAMAŃ SIECIOWYCH

Po uzyskaniu dostępu do sieci haker może powodować następujące zagrożenia (rys. 9):

1. **Kradzież informacji** – włamanie do komputera w celu uzyskania poufnych informacji. Skradzione informacje mogą zostać użyte do różnych celów lub sprzedane.
2. **Kradzież tożsamości** – forma kradzieży, w której przedmiotem kradzieży stają się informacje osobiste, mająca na celu przejęcie czyjejś tożsamości. Używając takich informacji, włamywacz może uzyskać dokumenty, wyłudzić kredyt lub dokonać zakupów w sieci. Jest to coraz powszechniejsza forma włamania sieciowego powodująca miliardowe straty.
3. **Utrata i zmiana danych** – włamanie do komputera, w celu zniszczenia lub dokonania manipulacji danych. Przykłady utraty danych to: wysłanie wirusa formatującego dysk twardy ofiary lub dokonanie zmiany np. ceny danego towaru.
4. **Blokada usług** – uniemożliwienie świadczenia usług sieciowych.

Rysunek 9.
Wybrane rodzaje włamań do sieci komputerowych

2.3 RODZAJE ATAKÓW SIECIOWYCH

Spam

Niechciane masowe przesyłki e-mail to kolejny dokuczliwy produkt wykorzystujący naszą potrzebę elektronicznej komunikacji. Niektórzy handlowcy nie tracą czasu na ukierunkowanie reklamy. Chcą wysyłać reklamy do jak największej liczby użytkowników w nadziei, że ktoś będzie zainteresowany ich produktem lub usługą. Takie szeroko dystrybuowane podejście do marketingu w Internecie określane jest mianem **spamu**.

Spam stanowi poważne zagrożenie, które może przeciążyć sieci dostawców usług sieciowych, serwery pocztowe oraz komputery użytkowników. Osoba lub organizacja odpowiedzialna za wysyłanie spamu jest nazywana **spamerem**. Spamerzy zwykle wykorzystują niezabezpieczone serwery pocztowe do rozsyłania poczty. Mogą też użyć technik hakerskich, takich jak: wirusy, robaki i konie trojańskie do przejęcia kontroli nad domowymi komputerami. Komputery te są wówczas używane do wysyłania spamu bez wiedzy właściciela. Spam może być rozsyłany przez pocztę elektroniczną lub, przez komunikatory sieciowe.

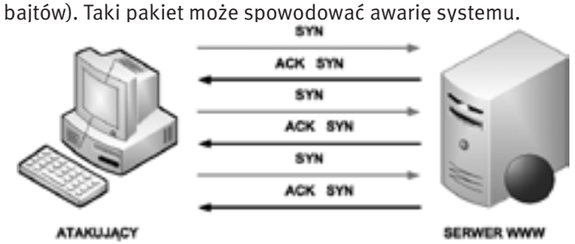
Atak DoS

Ataki DoS (ang. *Denial of Service*) są prowadzone na pojedyncze komputery lub grupy komputerów i mają na celu uniemożliwienie korzystania z usług. Celem ataku DoS mogą być systemy operacyjne, serwery, routery i łącza sieciowe. Główne cele ataków DoS to:

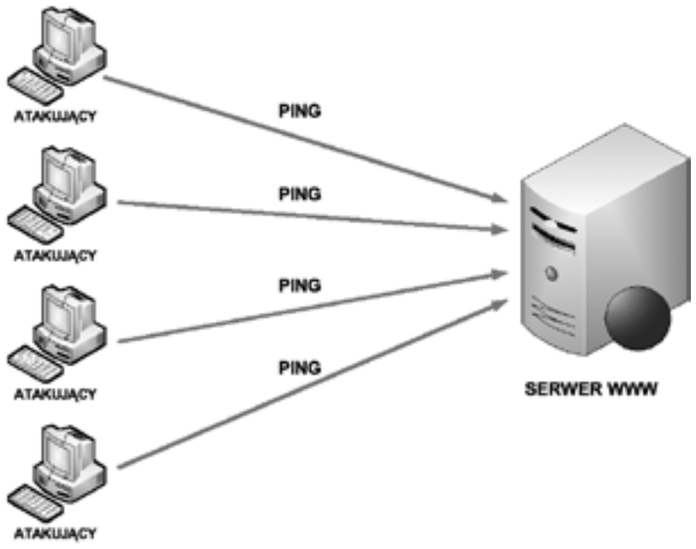
- Zalanie systemu (lub sieci) ruchem, aby zablokować ruch pochodzący od użytkowników.
- Uszkodzenie połączenia pomiędzy klientem i serwerem, aby uniemożliwić dostęp do usługi.

Istnieje kilka typów ataków DoS. Administratorzy odpowiedzialni za bezpieczeństwo muszą być świadomi ich istnienia i wiedzieć, jak się przed nimi uchronić. Dwa podstawowe przykłady ataków DoS to:

1. **Zalewanie SYN** (synchroniczne) – zalewanie serwera pakietami rozpoczynającymi nawiązanie połączenia. Pakiety te zawierają nieprawidłowy źródłowy adres IP. Serwer nie odpowiada na żądania użytkowników, ponieważ jest zajęty generowaniem odpowiedzi na fałszywe zapytania (rys. 10).
2. **Ping śmierci** (ang. *Ping of death*) – do urządzenia sieciowego wysyłany jest pakiet o rozmiarze większym niż maksymalny (65 535 bajtów). Taki pakiet może spowodować awarię systemu.



Rysunek 10.
Przykład ataku typu DoS



Rysunek 11.
Przykład ataku typu DDoS

Atak DDoS

Atak DDoS (ang. *Distributed Denial od Service*) jest odmianą ataku DoS, ale o wiele bardziej wyrafinowaną i potencjalnie bardziej szkodliwą. Został stworzony, aby nasycić sieć bezużytecznymi danymi. DDoS działa na znacznie większą skalę niż ataki DoS. Zwykle atakuje setki lub tysiące miejsc jednocześnie. Tymi miejscami mogą być komputery zainfekowane wcześniej kodem DDoS. Służą do tego najczęściej komputery, nad którymi przejęto kontrolę przy użyciu specjalnego złośliwego oprogramowania. Na dany sygnał komputery zaczynają jednocześnie atakować system ofiary, zasypując go fałszywymi próbami skorzystania z usług, jakie oferuje. Dla każdego takiego wywołania atakowany komputer musi przydzielić pewne zasoby (pamięć, czas procesora, pasmo sieciowe), co przy bardzo dużej ilości żądań prowadzi do wyczerpania dostępnych zasobów, a w efekcie do przerwy w działaniu lub nawet zawieszenia systemu (rys. 11).

Phishing

Phishing jest techniką wyłudzania poufnych informacji poprzez podszywanie się pod osobę pracującą w atakowanej organizacji, np. w banku. Atakujący zwykle kontaktuje się za pomocą poczty elektronicznej. Może poprosić o weryfikację informacji (np. hasła, nazwy użytkownika), by rzekomo zabezpieczyć ofiarę przed groźnymi konsekwencjami.

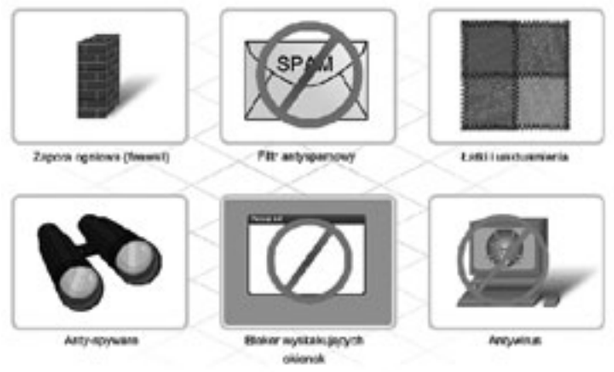
3 WYBRANE METODY BEZPIECZEŃSTWA SIECIOWEGO

3.1 NARZĘDZIA I APLIKACJE DO ZABEZPIECZANIA SIECI

Polityka bezpieczeństwa powinna być centralnym punktem procesów zabezpieczania, monitorowania, testowania i ulepszania sieci. Tę politykę realizują procedury bezpieczeństwa, które określają procesy konfiguracji, logowania, audytu oraz obsługi hostów i urządzeń sieciowych. Mogą definiować kroki prewencyjne zmniejszające ryzyko jednocześnie informując, jak radzić sobie po stwierdzeniu naruszenia zasad bezpieczeństwa. Procedury te mogą zawierać proste zadania, takie jak zarządzanie i aktualizacja oprogramowania, ale też złożone implementacje zapór ogniowych i systemów wykrywania włamań.

Przykłady narzędzi i aplikacji używanych do zabezpieczania sieci (rys. 12):

1. **Zapora ogniowa** (ang. *firewall*) – sprzętowe lub programowe narzędzie bezpieczeństwa, które kontroluje ruch do i z sieci.
2. **Bloker spamu** – oprogramowanie zainstalowane na serwerze lub komputerze użytkownika, identyfikujące i usuwające niechciane wiadomości.
3. **Łatki i aktualizacje** – oprogramowanie dodane do systemu lub aplikacji naprawiające luki w bezpieczeństwie lub dodające użyteczną funkcjonalność.
4. **Ochrona przed spyware** – oprogramowanie zainstalowane na stacji użytkownika do wykrywania i usuwania spyware i adware.
5. **Blokery wyskakujących okienek** – oprogramowanie zainstalowane na komputerze użytkownika do zabezpieczania przed wyskakiwaniem okienek z reklamami.
6. **Ochrona przed wirusami** – oprogramowanie zainstalowane na komputerze użytkownika lub serwerze, wykrywające i usuwające wirusy, robaki oraz konie trojańskie z plików i wiadomości e-mail.

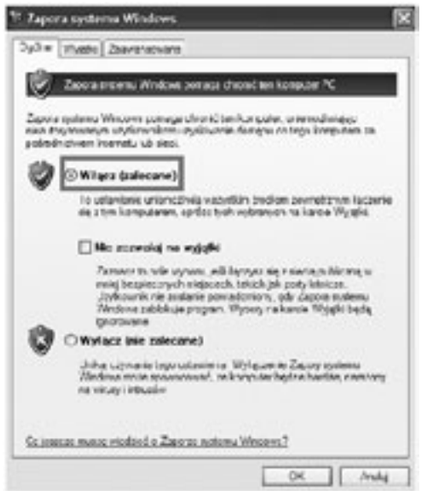


Rysunek 12.
Wybrane narzędzia i aplikacje do zabezpieczania sieci komputerowych

3.2 INSTALACJA ZAPORY OGNIOWEJ

Jedną ze skuteczniejszych metod zabezpieczenia sieci komputerowej przed atakiem jest włączenie i skonfigurowanie zapory ogniowej. Na rysunku 13 pokazano zakładkę Centrum zabezpieczeń w Panelu sterowania z włączoną funkcją zapory ogniowej. Aby skonfigurować zaporę ogniową w systemie operacyjnym, należy kliknąć na opcji Zapora systemu Windows.

Rysunek 13.
Zakładka Centrum zabezpieczeń w systemie Windows XP z informacją o włączonej zaporze ogniowej

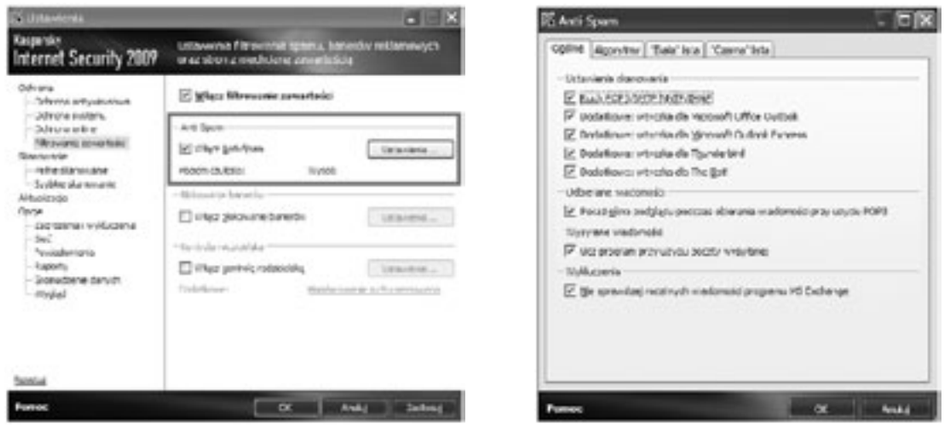


Rysunek 14.
Ustawienia zapory ogniowej

Zapora ogniowa chroni komputer przed nieautoryzowanym dostępem z sieci. Włączenie zapory (rys. 14) uniemożliwia połączenie się z tym komputerem z zewnątrz poza wybranymi wyjątkami (rys. 15).

Na rysunku 16 przedstawiono ustawienie filtrowania spamu w programie Kaspersky Internet Security 2009.

Rysunek 15.
Zaznaczenie programów i usług nieblokowanych przez zaporę ogniową



Rysunek 16.
Ustawienia filtrowania spamu w programie Kaspersky Internet Security 2009

Do głównych narzędzi polepszających zabezpieczenia sieci komputerowych jest przeprowadzanie automatycznych aktualizacji. Rysunek 17 pokazuje zakładkę Centrum zabezpieczeń w Panelu sterowania z włączoną funkcją Aktualizacje automatyczne. Po kliknięciu na zaznaczonej opcji ukazuje się obraz pokazany na rysunku 18.

Rysunek 17.
Zakładka Centrum zabezpieczeń w systemie Windows XP z informacją o włączonych automatycznych aktualizacjach

Rysunek 18.
Ustawienia automatycznych aktualizacji

Rysunek 19.
Kreator kopii zapasowej

Jedną z głównych gwarancji bezpieczeństwa danych jest ich regularne archiwizowanie, które można wykonywać na trzy sposoby:

1. **Kopia pełna** (ang. *full backup*) – polega na skopiowaniu wszystkich wybranych plików i oznaczeniu każdego z nich jako zarchiwizowany. Kopie pełne są najłatwiejsze w użyciu podczas odzyskiwania plików, ponieważ wymagają jedynie posiadania najświeższego pliku. Wykonywanie kopii pełnych zajmuje jednak najwięcej przestrzeni na nośnikach (i zazwyczaj czasu), ponieważ kopiowany jest każdy plik, niezależnie od tego, czy został zmieniony od czasu tworzenia ostatniej kopii zapasowej.
2. **Kopia przyrostowa** (ang. *incremental backup*) – polega na kopiowaniu jedynie tych plików, które zostały utworzone lub zmienione od czasu utworzenia ostatniej kopii przyrostowej lub pełnej oraz na oznaczeniu ich jako zarchiwizowane. Przed utworzeniem pierwszej kopii przyrostowej powinno się utworzyć pełną kopię systemu. Jeżeli korzysta się z kombinacji kopii pełnych oraz przyrostowych, to do odtworzenia danych niezbędne są, w chronologicznym porządku: ostatnio utworzona kopia pełna oraz wszystkie kolejne kopie przyrostowe.
3. **Kopia różnicowa** (ang. *differential backup*) – polega na kopiowaniu jedynie tych plików, które zostały utworzone lub zmienione od czasu utworzenia ostatniej kopii pełnej. Podczas wykonywania kopii różnicowej kopiowane pliki nie są oznaczane jako zarchiwizowane. Przed utworzeniem pierwszej kopii różnicowej zalecane jest wykonanie pełnej kopii. Jeżeli korzysta się z kombinacji kopii pełnych oraz różnicowych, to do odtworzenia danych niezbędne są: ostatnia kopia pełna oraz ostatnia kopia różnicowa.

4 SYSTEMY WYKRYWANIA INTRUZÓW (WŁAMAŃ)

4.1 SYSTEMY IDS

Zadaniem **systemu wykrywania intruzów** (ang. *Intrusion Detection System,* IDS) jest identyfikacja zagrożenia w sieci komputerowej. Podstawą wykrywania włamań jest monitorowanie ruchu w sieci. Systemy wykrywania włamań działają w oparciu o informacje odnoszące się do aktywności chronionego systemu – współczesne systemy IDS analizują w czasie rzeczywistym aktywność w sieci.

Włamanie do systemu najczęściej przebiega w dwóch etapach:

- Etap pierwszy – próba penetracji systemu będącego celem ataku. Intruz usiłuje znaleźć lukę w systemie (na przykład próbuje skanować porty), umożliwiającą wtargnięcie do systemu poprzez ominięcie systemów zabezpieczających.
- Etap drugi – wtargnięcie do systemu. Jednocześnie odbywa się próba zamaskowania obecności intruza poprzez odpowiednie zmiany w logach systemowych. Włamywacz podejmuje również próby modyfikacji narzędzi systemowych tak, by uniemożliwić swoje wykrycie.

Systemy IDS analizują procesy zachodzące w newralgicznych obszarach sieci objętej ochroną. Umożliwiają więc wykrycie niepożądanych zajść podczas próby włamania oraz po udanym włamaniu – jest to bardzo ważne ze względów bezpieczeństwa, ponieważ IDS działa dwufazowo – nawet jeżeli intruz zdoła włamać się do systemu, nadal może zostać wykryty i unieszkodliwiony, mimo usilnego zacierania śladów swojej działalności.

Systemy IDS korzystają z czterech podstawowych metod, dzięki którym możliwe jest zidentyfikowanie intruza wewnątrz chronionej sieci:

1. **Dopasowywanie wzorców** – jest to najprostsza metoda detekcji intruza; pojedynczy pakiety porównywany jest z listą reguł. Jeśli któryś z warunków jest spełniony, to jest uruchamiany alarm.
2. **Kontekstowe dopasowywanie wzorców** – w kontekstowym dopasowywaniu pakietu, system bierze pod uwagę kontekst każdego pakietu. Śledzi połączenia, dokonuje łączenia fragmentowanych pakietów.

3. **Analiza heurystyczna** – wykorzystuje algorytmy do identyfikacji niepożądanego działania. Algorytmy te są zwykle statystyczną oceną normalnego ruchu sieciowego. Przykładowo, algorytm stwierdzający skanowanie portów wykazuje, że takie wydarzenie miało miejsce, jeżeli z jednego adresu w krótkim czasie nastąpi próba połączeń z wieloma portami.
4. **Analiza anomalii** – sygnatury anomalii starają się wykryć ruch sieciowy, który odbiega od normy. Największym problemem jest określenie stanu uważanego za normalny.

Mimo ciągłego rozwoju systemów IDS napotykają one na liczne przeszkody, które zniekształcają prawidłowe działanie oprogramowania:

1. Mnogość aplikacji – w przypadku ataku na konkretną aplikację, polegającym na podawaniu jej nietypowych danych, system musi „rozumieć" protokół, którego dana aplikacja używa. Protokołów sieciowych jest bardzo wiele i system IDS na ogół nie zna ich wszystkich, a tylko pewien ich podzbiór. Jest to wykorzystywane przy próbach ataku na sieć chronioną przez IDS.
2. Defragmentacja pakietów – wykrycie ataku rozłożonego na kilka pakietów wymaga monitorowania przebiegu sesji. Takie działanie pochłania jednak część zasobów komputerowych: pamięć i czas.
3. Fałszywe alarmy.
4. Ograniczenia zasobów – zajęcie wszystkich zasobów sensora jest wykorzystywane do ataków na sieci chronione przez IDS.

Istnieją cztery główne rodzaje ataków, które systemy klasy IDS są w stanie rozpoznać:

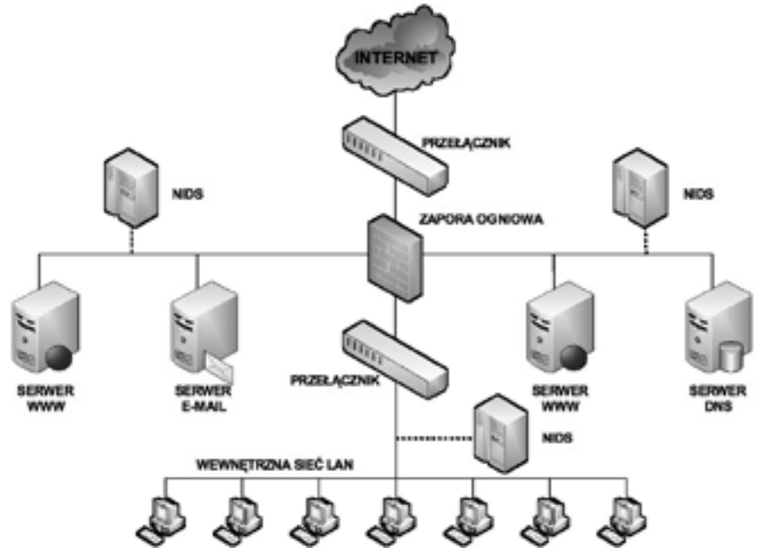
1. Nieautoryzowany dostęp do zasobów – najbardziej liczna grupa ataków zawierająca w sobie między innymi łamanie haseł dostępowych, używanie koni trojańskich oraz podszywanie się.
2. Nieuprawniona modyfikacja zasobów – to nieuprawnione modyfikacje, kasowanie danych oraz generowanie nieuprawnionych transmisji danych.
3. Blokowanie usług – przede wszystkim ataki typu DoS/DDoS.
4. Ataki zorientowane na aplikacje – ataki wykorzystujące błędy oraz luki zawarte w aplikacjach.

4.2 RODZAJE SYSTEMÓW IDS

Działanie pierwszych systemów do wykrywania włamań polegało przede wszystkim na szczegółowej analizie wystąpienia niebezpiecznego zdarzenia. Współczesne aplikacje IDS wykonują dodatkowo monitorowanie sieci oraz wykrywanie i reagowanie w czasie rzeczywistym na nieautoryzowane działania w sieci. Wyróżnia się trzy główne rodzaje systemów IDS:

1. **NIDS** (ang. *Network Intrusion Detection System,* sieciowy system wykrywania intruzów) – rozwiązania sprzętowe lub programowe śledzące sieć.
2. **HIDS** (ang. *Host Intrusion Detection System,* hostowy system wykrywania intruzów) – aplikacje instalowane na chronionych serwerach usług sieciowych.
3. **NNIDS** (ang. *Network Node Intrusion Detection System,* hybrydowy system wykrywania intruzów) – rozwiązania hybrydowe.

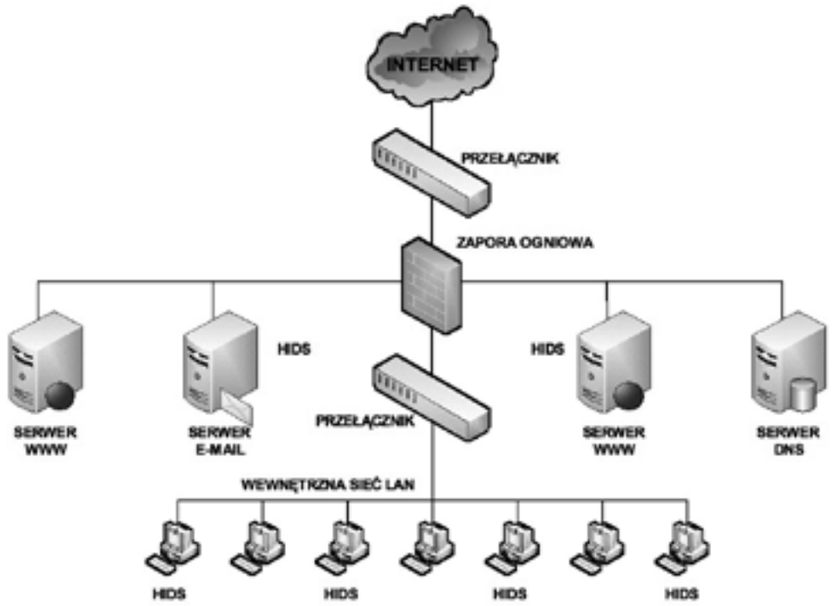
Na rysunku 20 pokazany jest schemat sieciowego systemu wykrywania intruzów (NIDS). Takie rozwiązanie umożliwia skuteczne monitorowanie wydzielonego segmentu sieci. System NIDS może podsłuchiwać wszelką komunikację prowadzoną w tej sieci. To rozwiązanie jest nastawione na ochronę publicznie dostępnych serwerów zlokalizowanych w podsieciach stref zdemilitaryzowanych (patrz p. 5.2).



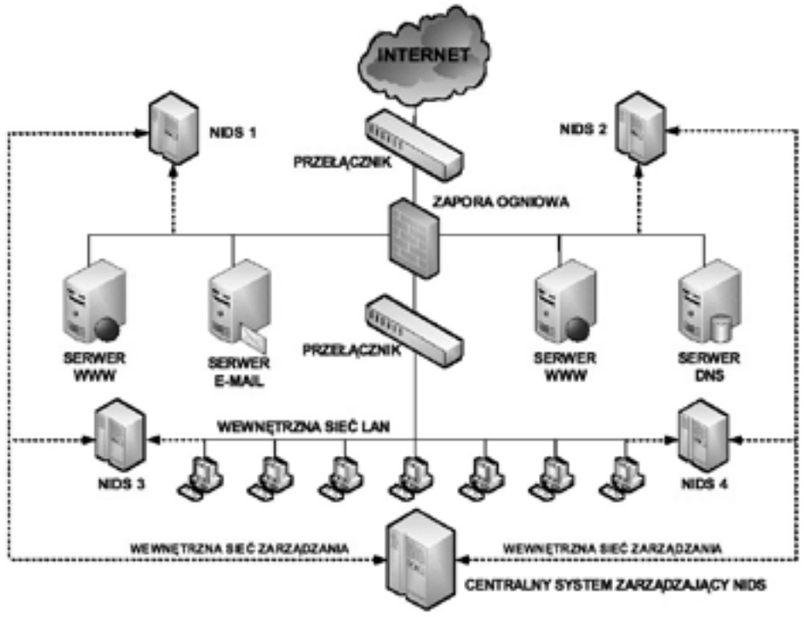
Rysunek 20.
Schemat systemu wykrywania włamań typu NIDS

Schemat hostowego systemu wykrywania intruzów (HIDS) jest przedstawiony na rysunku 21. Podstawowa różnica między systemami HIDS a NIDS polega na tym, że w pierwszym przypadku chroniony jest tylko komputer, na którym system rezyduje. Ponadto system HIDS można uruchamiać na zaporach ogniowych, zabezpieczając je w ten sposób.

Rysunek 22 pokazuje hybrydowy system wykrywania intruzów (NNIDS), składający się z czterech sensorów i centralnego systemu zarządzającego. Standardowo systemy NNIDS funkcjonują w ramach architektury przeznaczonej do obsługi zarządzania i badania sieci. Sensory wykrywania włamań systemów NIDS są zlokalizowane zdalnie w odpowiednich miejscach i składają raporty do centralnego systemu zarządzania NIDS. Dzienniki ataków są co jakiś czas dostarczane do systemu zarządzającego i mogą być tam przechowywane w centralnej bazie danych. Z kolei nowe sygnatury ataków mogą być ładowane do systemów-sensorów. Zgodnie z przedstawionym schematem, sensory NIDS1 i NIDS2 operują w cichym trybie odbierania i chronią serwery dostępu publicznego. Natomiast sensory NIDS3 i NIDS4 chronią systemy hostów znajdujących się wewnątrz sieci zaufanej.



Rysunek 21.
Schemat systemu wykrywania włamań typu HIDS



Rysunek 22.
Schemat systemu wykrywania włamań typu NNIDS

5 DZIAŁANIE ZAPÓR OGNIOWYCH

5.1 PODSTAWOWE FUNKCJE ZAPÓR OGNIOWYCH

Zapora ogniowa jest jednym z najefektywniejszych narzędzi, służących do zabezpieczania wewnętrznych użytkowników sieci przed zagrożeniami zewnętrznymi. Zapora ogniowa stoi na granicy dwóch lub więcej sieci i kontroluje ruch pomiędzy nimi oraz pomaga zapobiec nieupoważnionemu dostępowi. Zapory ogniowe używają różnych technik w celu określenia, jaki dostęp do sieci ma zostać przepuszczony, a jaki zablokowany.

Ochrona systemów informatycznych określona w polityce bezpieczeństwa zakłada wykorzystywanie zapór ogniowych jako blokady przesyłania nieautoryzowanych danych między sieciami wewnętrzną i zewnętrzną. Podstawowe funkcje tych urządzeń to:

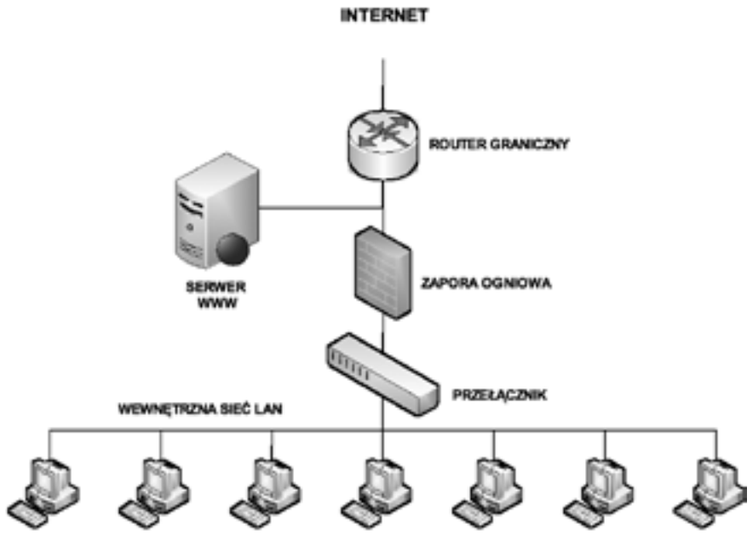
- ochrona adresów IP i przesyłanie komunikacji – dzięki tej funkcji możliwe jest tworzenie dodatkowych podsieci; mając do dyspozycji pojedynczy adres IP można utworzyć sieć lokalną LAN, a nawet rozległą WAN;
- oddzielenie sieci – zapora jest przede wszystkim narzędziem służącym do tworzenia granic między sieciami, nie musi ona jednak być umieszczona między siecią publiczną a prywatną; zapory ogniowe umieszcza się również wewnątrz sieci firmowych;
- ochrona przed atakami i skanowaniem – za pomocą zapór ogniowych można ograniczyć dowolny typ komunikacji sieciowej;
- filtrowanie adresów IP – funkcja ta umożliwia zarządzanie połączeniami w zależności od adresu IP oraz portu;
- filtrowanie zawartości – serwery pośredniczące proxy są jedynym typem zapór ogniowych, które są w stanie analizować komunikację badając adresy URL oraz zawartość stron WWW;
- przekierowywanie pakietów – funkcja ta polega na kierowaniu komunikacji na zupełnie inny port lub host niż ten, do którego został wysłany;
- uwierzytelnienie i szyfrowanie – zapora ogniowa umożliwia uwierzytelnienie użytkowników i szyfrowanie transmisji wykonywanych między nią a zaporą innej sieci;
- rejestrowanie komunikacji w dziennikach – zapora ogniowa umożliwia przegląd szczegółowych informacji na temat pakietów sieciowych przechodzących przez nią.

5.2 PRZYPADKI UŻYCIA ZAPORY OGNIOWEJ

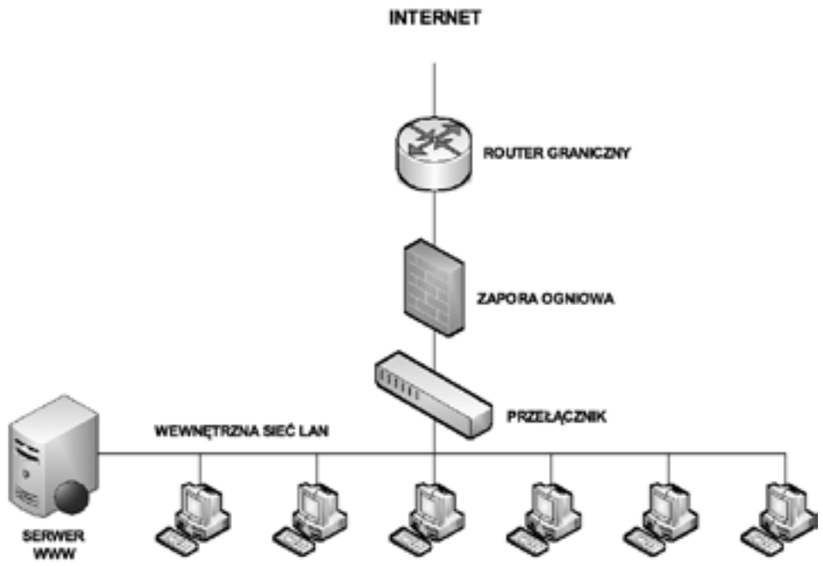
W sieci z rysunku 23 zastosowano zaporę ogniową do ochrony wewnętrznych zasobów sieci komputerowej. Natomiast serwer WWW dedykowany klientom z Internetu jest całkowicie dostępny na ataki, a jego działanie uzależnione jest od zastosowanej platformy serwerowej i poprawności konfiguracji.

W przypadku przedstawionym na rysunku 24, pomimo że serwer WWW jest umieszczony za zaporą ogniową, nie jest to rozwiązanie jeszcze idealne. Konfiguracja umożliwiająca przepuszczanie ruchu na porcie 80 (protokół http) i 443 (protokół https), niezbędna dla zapewnienia właściwej obsługi ruchu przychodzącego, daje włamywaczowi możliwość przeprowadzenia ataku na wewnętrzną sieć LAN.

Rysunek 25 przedstawia zastosowanie **strefy zdemilitaryzowanej** (ang. *Demilitarized Zone,* DMZ). Określenie to zostało zapożyczone z terminologii wojskowej, gdzie DMZ jest obszarem pomiędzy wrogimi siłami, w którym aktywność militarna jest zakazana. W sieciach komputerowych DMZ jest obszarem sieci, który jest dostępny zarówno dla wewnętrznych, jak i zewnętrznych użytkowników. Jest bardziej bezpieczny od zewnętrznej sieci, lecz mniej bezpieczny od wewnętrznej. Obszar ten jest tworzony przez jedną lub kilka zapór ogniowych i ma za zadanie odseparowanie sieci zewnętrznej i wewnętrznej od siebie. Serwery WWW przeznaczone do publicznego dostępu często umieszcza się właśnie w DMZ.



Rysunek 23.
Zapora ogniowa chroniąca wewnętrzną sieć LAN



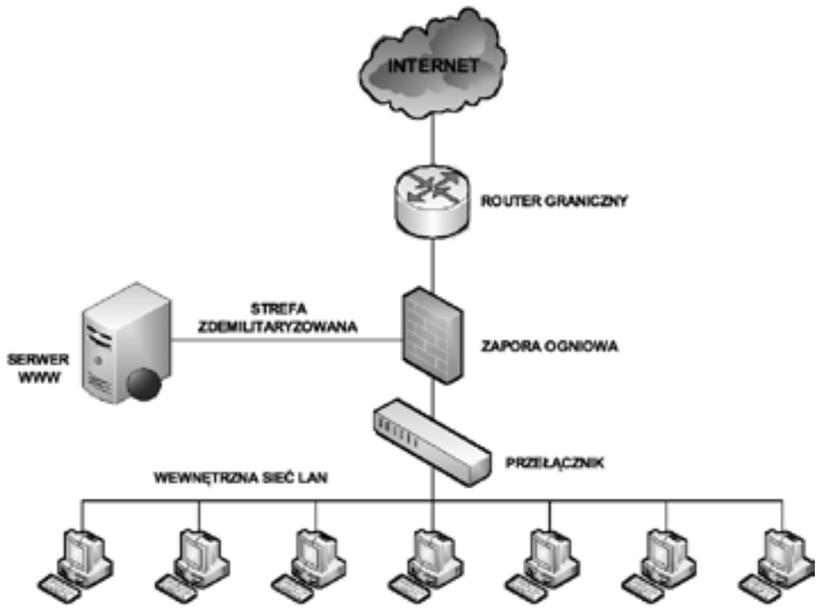
Rysunek 24.
Zapora ogniowa chroniąca wewnętrzną sieć LAN oraz serwer WWW

W wariancie z rysunku 26 zastosowano dwie zapory ogniowe ze strefą DMZ umieszczoną pomiędzy nimi. Zewnętrzna zapora ogniowa jest mniej restrykcyjna i zezwala użytkownikom z Internetu na dostęp do usług w DMZ, jednocześnie przepuszczając ruch zainicjowany przez użytkowników wewnętrznych. Wewnętrzna zapora ogniowa jest bardziej restrykcyjna – chroni wewnętrzną sieć przed nieupoważnionym dostępem.

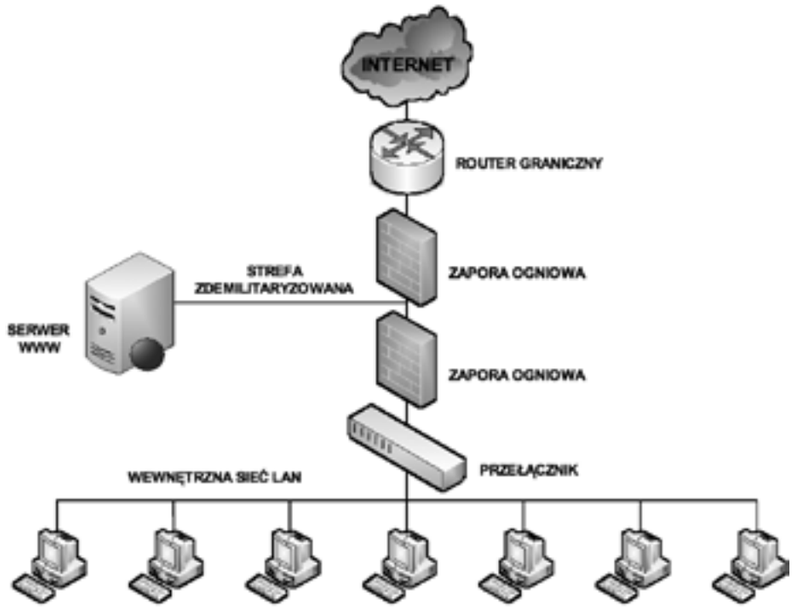
Konfiguracja z jedną zaporą ogniową jest zalecana w mniejszych sieciach. Taka konfiguracja stanowi pojedynczy punkt awarii i jednocześnie sama zapora może zostać przeciążona. Konfiguracja z dwiema zaporami ogniowymi jest polecana dla większych i bardziej rozbudowanych sieci, gdzie natężenie ruchu jest znacznie większe.

Planowanie bezpieczeństwa sieciowego wymaga oceny ryzyka związanego z utratą danych, uzyskaniem nieautoryzowanego dostępu. Plan musi również uwzględniać czynnik kosztów, stopień wyszkolenia personelu, platformy i sprzęt wykorzystywany w sieci.

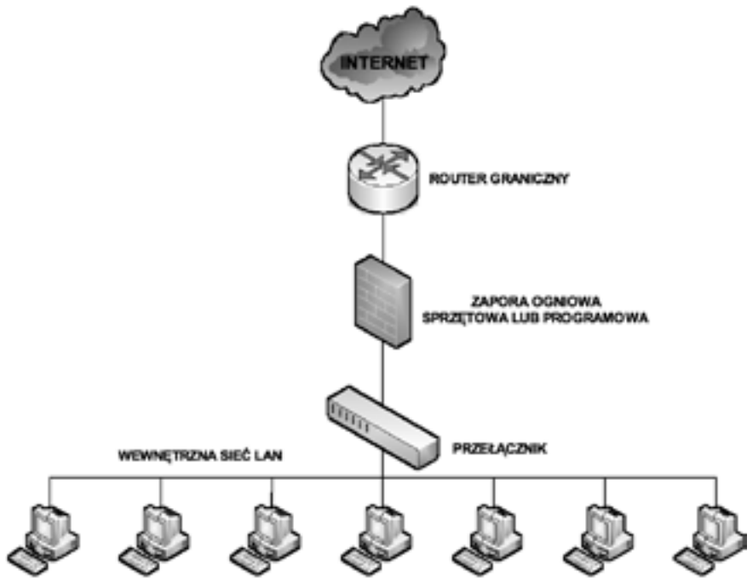
Zapory ogniowe, a co za tym idzie strefy DMZ, zapewniają wielowarstwowy model bezpieczeństwa. W przeszłości zapory ogniowe były wykorzystywane jedynie do podziału sieci na dwie części, sieć wewnętrzną i zewnętrzną (publiczną). Obecnie, ze względu na dostępność narzędzi służących włamaniom i łatwość z jaką mogą być przeprowadzane ataki, włącznie z atakami wykorzystującymi fałszowanie adresów, niezbędne jest dokładniejsze izolowanie sieci i lepsza ochrona przechowywanych w niej sieci. Służy temu stosowanie stref zdemilitaryzowanych.



Rysunek 25.
Zapora ogniowa oddzielająca wewnętrzną sieć LAN od strefy zdemilitaryzowanej



Rysunek 26.
Zastosowanie dwóch zapór ogniowych



Rysunek 27.
Przykład budowy prostej strefy DMZ

LITERATURA

1. Dye M.A., McDonald R., Rufi A.W., *Akademia sieci Cisco. CCNA Exploration. Semestr 1*, WN PWN, Warszawa 2008
2. Krysiak K., *Sieci komputerowe. Kompendium*, Helion, Gliwice 2005
3. Mucha M., *Sieci komputerowe. Budowa i działanie*, Helion, Gliwice 2003
4. Szmit M., Tomaszewski M., Lisiak D., Politowska I., *13 najpopularniejszych sieciowych ataków na Twój komputer. Wykrywanie, usuwanie skutków i zapobieganie*, Helion, Gliwice 2008
5. Szmit M., Gusta M., Tomaszewski M., *101 zabezpieczeń przed atakami w sieci komputerowej*, Helion, Gliwice 2005

NOTA O WYDAWCY

Warszawska Wyższa Szkoła Informatyki jest wyższą uczelnią niepubliczną utworzoną na podstawie decyzji Ministra Nauki i Szkolnictwa Wyższego z dnia 19 lipca 2000 roku.

Uczelnia prowadzi studia wyższe na poziomie inżynierskim, magisterskim i podyplomowym wyłącznie na kierunku informatyka.

Decyzją Ministra Nauki i Szkolnictwa Wyższego z dnia 7 lutego 2008 roku Warszawska Wyższa Szkoła Informatyki otrzymała uprawnienia do prowadzenia studiów I stopnia na kierunku informatyka – na czas nieokreślony.

Decyzją Ministra Spraw Nauki i Szkolnictwa Wyższego z dnia 24 września 2008 roku Warszawska Wyższa Szkoła Informatyki uzyskała uprawnienia do prowadzenia studiów II stopnia – magisterskich.

W Warszawskiej Wyższej Szkole Informatyki aktualnie (2010/2011) kształci się blisko tysiąc trzystu polskich i zagranicznych studentów zdobywających kompetencje zawodowe z zakresu informatyki. Kompetencje te są potwierdzane dyplomami państwowymi – dyplomem inżyniera, magistra oraz dyplomem studiów podyplomowych na kierunku informatyka. Ponadto Uczelnia oferuje możliwość uzyskania cenionych na ryku pracy certyfikatów branżowych ICT (PRINCE2® Foundation, PMI®, CCNA, Audytora wewnętrznego systemu zarządzania bezpieczeństwem informacji wg ISO 27001, Projektanta zabezpieczeń sieci teleinformatycznych wg ISO 27001, Menedżera zarządzania bezpieczeństwem informacji, MCTS oraz MCITP).

Uczelnia prowadzi w szerokim zakresie otwarte studia informatyczne (kursy, szkolenia, warsztaty) dla uczniów szkół średnich – między innymi w ramach projektu Informatyka+, dla absolwentów szkół wyższych oraz dla wszystkich osób zainteresowanych pogłębieniem swojej wiedzy i kwalifikacji informatycznych, niezależnie od wieku oraz stopnia znajomości informatyki.

Warszawska Wyższa Szkoła Informatyki prowadzi działalność o charakterze dydaktycznym i naukowo-badawczym, ze szczególnym uwzględnieniem problematyki edukacyjnej i kształcenia w zakresie informatyki, jako dziedziny o potencjalnie największych perspektywach na rynku pracy. Podstawowym zadaniem prowadzonych w Uczelni studiów oraz innych form kształcenia jest wyposażenie studentów w wiedzę teoretyczną oraz w umiejętności praktyczne z dziedziny informatyki, które pozwolą na uzyskanie zatrudnienia zgodnego z posiadanymi kwalifikacjami, oczekiwaniami i predyspozycjami. W dłuższym horyzoncie czasowym zdobyta w Warszawskiej Wyższej Szkole Informatyki wiedza i umiejętności powinny również sprzyjać dalszemu doskonaleniu własnego warsztatu pracy poprzez kontynuację nauki w różnych formach kształcenia ustawicznego lub poprzez samokształcenie.

Więcej informacji o Uczelni na stronie www.wwsi.edu.pl