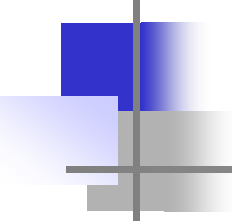


Multimedialne bazy danych

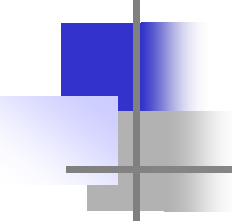
Marek Wojciechowski

Instytut Informatyki
Politechnika Poznańska



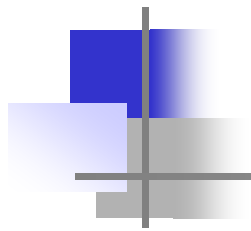
Plan przedmiotu

- Wprowadzenie, podstawowe definicje
- Duże obiekty w bazach danych
- Składowanie i prezentacja multimedialnych
- Charakterystyka danych multimedialnych:
 - Obrazy, dane audio, dane wideo
- Przetwarzanie zapytań w multimedialnych BD
- Standardy SQL/MM, MPEG-7 i MPEG-21
- Multimedia w komercyjnych SZBD
 - Oracle Multimedia

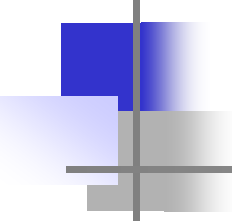


Literatura

- V.S. Subrahmanian, **Principles of Multimedia Database Systems**, Morgan K.
- B. Thuraisingham, **Managing and Mining Multimedia Databases**, CRC Press
- H. Kosch, **Distributed Multimedia Database Technologies Supported by MPEG-7 and MPEG-21**, CRC Press
- Artykuły, prace dyplomowe, tutoriale, wykłady, m.in.:
 - O. Kao, Parallel and Distributed Multimedia Database Systems
 - C. Böhm, S. Berchtold, D.A. Keim, Searching in High-dimensional Spaces - Index Structures for Improving the Performance of Multimedia Databases
 - M. Menth, Multimedia Databases, Seminar Thesis, University of Wurzburg
 - S. Christodoulakis, Multimedia Databases, VLDB97 Tutorial
 - I. Schmitt, Vorlesung Multimedia-Datenbanken
 - W. Skarbek, MPEG-7, IX konf. PLOUG, 2003
 - José M. Martínez, MPEG-7 Overview
 - J. Bormans, K. Hill, MPEG-21 Overview
 - Ł. Matuszczak, Obsługa danych multimedialnych w systemach baz danych (praca mgr)
 - H. Kosch, Multimedia- and Distributed DBMS, wykłady
- Specyfikacje standardów ISO: SQL/MM, MPEG-7
- Dokumentacja Oracle, dokumentacja IBM DB2
- Zasoby Internetu (Wikipedia, ...)

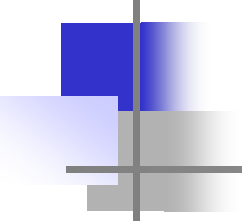


Wprowadzenie



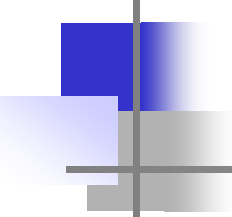
Geneza systemów multimedialnych

- Rzeczywistość postrzegana przez człowieka ma charakter multimedialny
- Na obraz rzeczywistości składają się informacje przekazane za pomocą wielu mediów (obraz, dźwięk, itd.)
- Imitacja reprezentacji otaczającej człowieka rzeczywistości, sposobu jej postrzegania, symulacji postrzegania stanowią motywację dla rozwoju multimedialnych systemów informatycznych



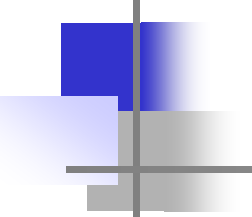
Możliwe zastosowania systemów multimedialnych

- Video on demand
- Telewizja cyfrowa (w tym interaktywna)
- Muzyczne bazy danych
- Bazy danych z obrazami
 - Nauka
 - Sztuka
 - Medycyna
- Zdalne nauczanie:
 - Kursy
 - Porady i instrukcje
- Bazy danych wykorzystywane przez policję



Przyczyny gwałtownego rozwoju systemów multimedialnych

- Dostępność stacji roboczych zdolnych do przetwarzania danych multimedialnych (wydajne CPU + sprzętowe kodery/dekodery)
- Dostępność urządzeń do składowania danych o dużej pojemności
- Szybkie sieci komputerowe
 - Wzrost prędkości transmisji
 - Rozwój infrastruktury
 - Spadek cen nośników (światłowodów)
 - Szybkie technologie transmisji danych (ATM)
- Rozwój technologii mobilnych
 - Telefony, PDA, coraz powszechniejszy UMTS



Klasyfikacja mediów wg zmysłów

- **Wizualne**

- Teksty, piktogramy
- Obrazy (fotografie), grafika

- **Akustyczne**

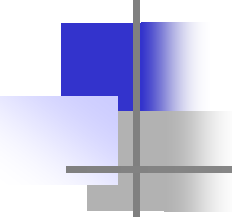
- Mowa
- Muzyka

- **Dotykowe**

- **Smakowe**

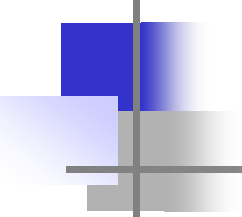
- **Zapachowe**

(!) Obecnie systemy multimedialne typowo ograniczają się do tekstu, obrazu i dźwięku



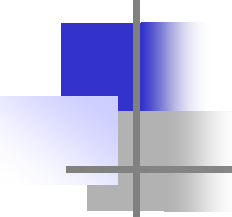
Klasyfikacja mediów wg zmienności w czasie

- Media **niezmienne w czasie**
(dyskretne, statyczne)
 - Tekst
 - Obrazy
 - Grafika
- Media **zmienne w czasie**
(ciągłe, dynamiczne)
 - Filmy wideo
 - Sekwencje audio
 - Animacje



Multimedia – podstawowe definicje

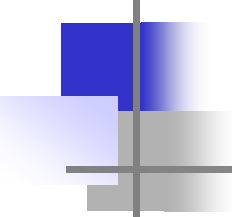
- **Obiekt medialny** – informacja zrozumiała dla komputera, zakodowana w jednym medium
- **Obiekt multimedialny** - informacja zrozumiała dla komputera, zakodowana w jednym lub więcej mediów (co najmniej jedno medium nie-alfanumeryczne)
- **Dane medialne/multimedialne** – zbiór obiektów medialnych/multimedialnych



System multimedialny

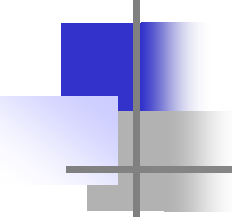
- **System multimedialny** – System komputerowy wspierający wymianę informacji z użytkownikiem za pomocą kilku różnych mediów [Meyer-Wegener]
- **System multimedialny** – System komputerowy umożliwiający w sposób zintegrowany tworzenie, przetwarzanie, prezentację i przesyłanie informacji zakodowanych w co najmniej jednym medium ciągłym i jednym dyskretnym

(!) Brak jednoznacznej, „ostrej” definicji



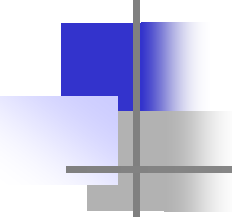
Systemy multimedialnych baz danych

- **System multimedialnej bazy danych (SMBD)** składa się bazy danych o dużej pojemności i wysoce wydajnego systemu zarządzania bazą danych (SZBD), który wspiera i obsługuje obok alfanumerycznych typów danych również dane multimedialne w zakresie ich składowania, wyszukiwania i przetwarzania zapytań [Khoshafian, Baker]



Systemy zarządzania multimedialną bazą danych

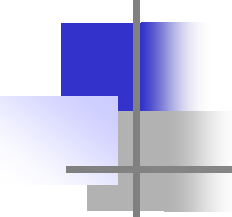
- **System zarządzania multimedialną bazą danych (SZMBD) =**
 - SZBD
 - Architektury do efektywnego składowania dużych wolumenów danych multimedialnych
 - Techniki information retrieval (IR) – wyszukiwanie wg zawartości
- Systemy zarządzania multimedialnymi bazami danych integrują wiele technologii
- Podzbiorem ich funkcjonalności jest funkcjonalność tradycyjnych SZBD



Funkcjonalność SZBD

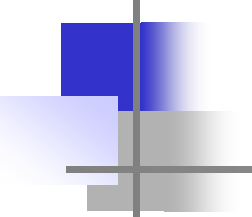
wymagana w SZMBD (1/2)

- Przetwarzanie transakcyjne – własności ACID:
 - **Atomowość** (ang. atomicity) - zbiór operacji składowych transakcji musi być wykonany w całości lub wcale
 - **Spójność** (ang. consistency) - transakcja przeprowadza bazę danych z jednego stanu spójnego do innego stanu spójnego
 - **Izolacja** (ang. isolation) - polega na logicznym odseparowaniu od siebie transakcji współbieżnie wykonywanych w systemie (możliwe różne poziomy izolacji)
 - **Trwałość** (ang. durability) - odporność wyników zatwierdzonej transakcji na awarię systemu



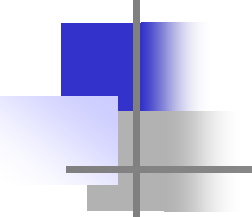
Funkcjonalność SZBD wymagana w SZMBD (2/2)

- Zarządzanie współbieżnością
 - Synchronizacja działań wielu użytkowników współbieżnie ubiegających się o te same zasoby
- Odtwarzanie po awarii
 - Powrót do spójnego stanu sprzed awarii
- Wersjonowanie
 - Możliwość przechowywania kilku wersji danego obiektu, dostęp do wersji aktualnej i poprzednich
- Bezpieczeństwo
 - Ochrona zawartości bazy danych przed nieuprawnionymi użytkownikami i niedozwolonymi operacjami



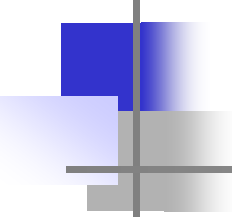
Specyfika SMBD (1/2)

- Duży rozmiar danych
 - Składowane w bazie danych (jako BLOB) lub poza bazą danych (system plików serwera, serwery WWW, serwery audio/video)
 - Składowanie danych na zewnątrz, z dostępem przez referencje składowane w bazie danych utrudnia zapewnienie ochrony danych i przetwarzania transakcyjnego dla danych multimedialnych
- Złożone struktury danych
 - Dane binarne uzupełnione opisem formatu i zawartości
 - Obiektowy i obiektowo-relacyjny model danych bardziej odpowiedni dla multimedii niż "czysty" model relacyjny



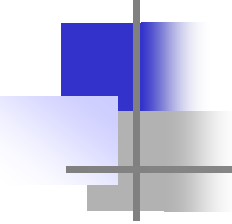
Specyfika SMBD (2/2)

- Zaawansowane przetwarzanie danych
 - np. konwersja formatu, skalowanie obrazów
- Zapytania różniące się od tradycyjnych
 - Wyszukiwanie w oparciu o zawartość (w oparciu o metadane i/lub automatycznie ekstrahowane właściwości)
 - Wagi przypisywane kryteriom selekcji
 - Tolerowanie przybliżonych/niekompletnych wyników
- Problemy prezentacji danych
 - Ograniczona przepustowość serwera i kanałów transmisyjnych
 - Ograniczony bufor danych klienta
 - Dane wrażliwe na opóźnienia (audio, wideo)

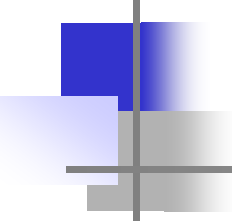


SMBD a bazy danych z multimedialną zawartością

- Systemy oparte o bazę danych z multimedialną zawartością, nierealizujące funkcjonalności SZBD i wyszukiwania wg zawartości nie stanowią jeszcze systemów multimedialnych baz danych
- Przykłady baz danych z multimedialną zawartością:
 - Płyty CD-ROM z multimediami
 - Katalogi obrazków z ikonkami prowadzącymi do obrazów w większej rozdzielczości
 - Systemy video on demand z wyszukiwaniem wg typowych słów kluczowych (tytuł, obsada, itp.)
 - Systemy zarządzania dużymi zbiorami złożonych dokumentów



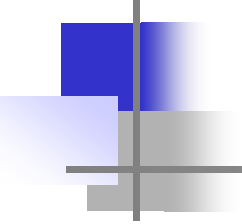
Duże obiekty (LOB) w bazach danych



Rodzaje danych składowanych w bazach danych

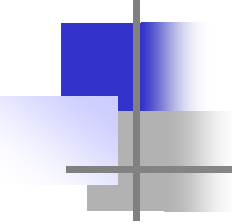
- Dane strukturalne o prostej strukturze
 - klasyczne dane relacyjne
- Dane strukturalne o złożonej strukturze
 - wykorzystujące mechanizmy obiektowo-relacyjne
- Dane semistrukuralne
 - Dane posiadające strukturę logiczną, która typowo nie jest interpretowana przez SZBD, a raczej przez zewnętrzne aplikacje
 - Przykłady: dokumenty XML, formaty dokumentów tekstowych
- Dane niestrukuralne
 - Dane nieposiadające wewnętrznej struktury logicznej, typowe nieinterpretowane ani przez SZBD ani przez aplikację biznesową
 - Przykłady: cyfrowo zapisane obrazy, dźwięki

LOBs



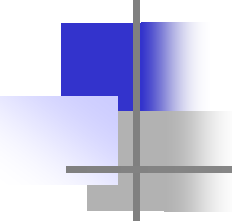
Duże obiekty wg standardu SQL

- Uwzględnione w standardzie SQL:1999
- Typy dużych obiektów (LOB) uwzględnione w standardzie:
 - BINARY LARGE OBJECT (BLOB)
 - CHARACTER LARGE OBJECT (CLOB)
 - NATIONAL CHARACTER LARGE OBJECT (NCLOB)
- Restrykcje dotyczące LOBs:
 - nie mogą być użyte w więzach PRIMARY KEY, UNIQUE, FOREIGN KEY
 - nie mogą brać udziału w innych porównaniach niż =, <>
 - nie mogą wystąpić w klauzulach GROUP BY i ORDER BY
- Inne założenia:
 - CLOB, NCLOB zachowują się jak inne typy znakowe (z uwzględnieniem powyższych restrykcji)
 - Aplikacje działają na wartościach LOB poprzez tzw. LOB locator



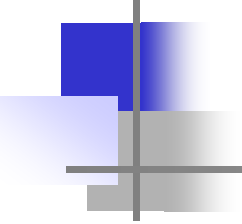
Duże obiekty w Oracle

- Internal LOBs
 - wewnętrzne, czyli składowane w bazie danych
 - typy danych: BLOB, CLOB, NCLOB
 - trwałe (persistent) lub tymczasowe (temporary)
 - trwałe: zapisane w wierszach tabel w bazie danych, podlegają przetwarzaniu transakcyjnemu i odtwarzaniu po awarii (ACID),
 - tymczasowe: gdy tworzone tylko w zasięgu aplikacji, mogą stać się trwałymi gdy zostaną zapisane w tabeli b.d.
- External LOBs
 - zewnętrzne, tj. składowane w systemie plików serwera (poza przestrzeniami tabel)
 - typ danych BFILE
 - nie podlegają przetwarzaniu transakcyjnemu (integralność i trwałość musi być zapewniona na poziomie systemu operacyjnego)
 - tylko do odczytu z poziomu bazy danych



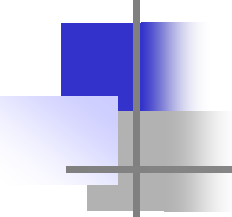
Porównanie wewnętrznych i zewnętrznych LOBs w Oracle

Internal LOBs	External LOBs
Różne typy danych dla danych tekstowych i binarnych	Jeden wspólny typ danych dla wszystkich rodzajów danych (plik binarny - brak informacji o kodowaniu znaków)
Składowane w wierszach tabel bazy danych	Składowane w plikach systemu operacyjnego
Modyfikowalne, podlegają przetwarzaniu transakcyjnemu	Tylko do odczytu z poziomu b.d., nie biorą udziału w transakcjach
Semantyka kopiowania przy operacjach INSERT i UPDATE	Semantyka referencji przy operacjach INSERT i UPDATE
Każda komórka tabeli zawierająca wewnętrzny LOB posiada własny LOB locator i odrębną kopię wartości LOB	Każda komórka tabeli zawierająca BFILE posiada własny BFILE locator, ale mogą one wskazywać na ten sam plik
Mogą być tymczasowe	Nie mogą być tymczasowe



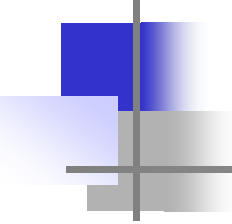
Typy CLOB i NCLOB w Oracle

- Typy analogiczne do:
 - CHAR i NCHAR
 - VARCHAR2 i NVARCHAR2
- Baza danych Oracle wykorzystuje 2 zestawy znaków (określane przy tworzeniu bazy danych)
 - DATABASE CHARACTER SET
 - jedno- lub wielobajtowy zestaw znaków (może być UTF8)
 - wykorzystywany przez typy danych CHAR, VARCHAR2 and CLOB
 - wykorzystywany w nazwach obiektów i źródłach SQL i PL/SQL
 - NATIONAL CHARACTER SET
 - Unicode (UTF-8 lub UTF16)
 - wykorzystywany przez typy danych NCHAR, NVARCHAR2 and NCLOB
- Przetwarzanie za pomocą:
 - funkcji znakowych: LENGTH, UPPER, SUBSTR, INSTR, itd.
 - ze względów wydajnościowych zalecane tylko dla LOB-ów do 100KB
 - pakietu DBMS_LOB



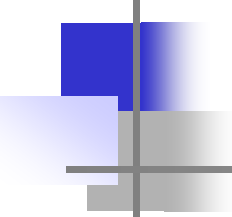
Lokalizatory i wartości LOB w Oracle

- Z każdym LOB związany jest lokalizator (LOB locator)
 - wskaźnik do wartości LOB
 - składowany w kolumnie tabeli (wartość razem z nim lub poza tabelą)
 - przekazywany do interfejsów programistycznych do obsługi LOB, np. pakiet DBMS_LOB dla PL/SQL, OCI dla C, JDBC dla Javy
- Składowanie wartości LOB
 - in-line (wraz z lokalizatorem w tabeli)
 - domyślnie, ale tylko gdy rozmiar (wliczając locator) < 4000 bajtów
 - out of line (poza tabelą)
 - w odrębnym segmencie (jeden LOB SEGMENT dla każdej kolumny LOB)
 - lokalizator pozostaje w kolumnie tabeli
 - domyślnie segment dla LOBs w tej samej przestrzeni tabel co tabela



Parametry składowania LOB w Oracle

```
create table pracownicy (  
    id_prac      number(12),  
    nazwisko     varchar2(80),  
    ...,  
    zdjecie      blob)  
tablespace app_data  
lob (zdjecie)  
store as (  
    tablespace app_lobs  
    disable storage in row  
    chunk 4096  
    pctversion 20  
    nocache  
    nologging)
```



Inicjalizacja kolumn LOB w Oracle

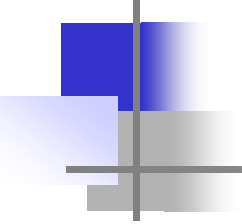
- Poprzez przypisanie NULL
 - brak lokalizatora LOB w komórce
- Poprzez przypisanie `EMPTY_BLOB()` lub `EMPTY_CLOB()`
 - istnieje lokalizator LOB w komórce, ale brak wartości LOB

```
UPDATE songs  
SET audio = EMPTY_BLOB() ;
```

- Dla BFILE inicjalizacja funkcją `BFILENAME`:
 - **BFILENAME** ('KATALOG' , 'NAZWA_PLIKU')

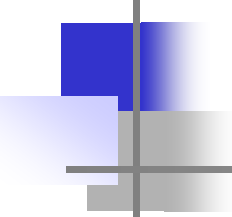
```
CREATE DIRECTORY katalog  
as '/var/movies';
```





Pakiet DBMS_LOB w Oracle

- Zawiera podprogramy do działania na obiektach BLOB, CLOB, NCLOB (w tym obsługę obiektów tymczasowych) oraz BFILE
 - podprogramy umożliwiają dostęp do całych LOB i ich fragmentów
 - podprogramy operują na lokalizatorach LOB
- Przykłady podprogramów zawartych w DBMS_LOB:
 - OPEN, CLOSE (FILEOPEN, FILECLOSE dla BFILE)
 - APPEND, COMPARE, COPY, ERASE, TRIM
 - CREATETEMPORARY, FREETEMPORARY
 - GETLENGTH (dla znakowych i binarnych LOB)
 - INSTR, SUBSTR (dla znakowych i binarnych LOB)
 - LOADFROMFILE , LOADBLOBFROMFILE , LOADCLOBFROMFILE
 - READ, WRITE, WRITEAPPEND



Otwieranie i zamykanie LOB w Oracle

- BFILE musi być jawnie otwarty przed odczytem danych, a następnie zamknięty
- BLOB, CLOB, NCLOB nie muszą być jawnie otwierane (OPEN)
 - jeśli są jawnie otwarte, to muszą zostać jawnie zamknięte
 - w przypadku jawnego otwarcia, indeksy funkcyjne i domenowe są aktualizowane po zamknięciu dużego obiektu
 - gdy LOB nie był jawnie otwarty, indeksy funkcyjne i domenowe są aktualizowane na bieżąco po każdej modyfikacji dużego obiektu

```
DECLARE
    fil BFILE;
BEGIN
    dbms_lob.fileopen(fil);
    -- operacje odczytu pliku
    dbms_lob.fileclose(fil);
END;
/
```

Tymczasowe obiekty LOB w Oracle

- Charakterystyka:
 - tworzone w tymczasowej przestrzeni tabel
 - czas życia nie przekracza sesji
 - nie podlegają przetwarzaniu transakcyjnemu ani mechanizmom zapewnienia spójności odczytu
 - używane do transformacji dużych obiektów
 - są tworzone puste
(nie wspierają funkcji EMPTY_BLOB()/EMPTY_CLOB())

```
DECLARE
  a blob;
BEGIN
  dbms_lob.createtemporary(a, TRUE);
  -- operacje na LOB
  dbms_lob.freetemporary(a);
END;
/
```

umieszczenie
w buffer cache



Ładowanie danych z BFILE do BLOB

```
DECLARE
  lobd blob;
  fils BFILE := BFILENAME('IMG_DIR','k-pax.jpg');
BEGIN
  INSERT INTO covers(movie_id, image)
  VALUES (99, EMPTY_BLOB());

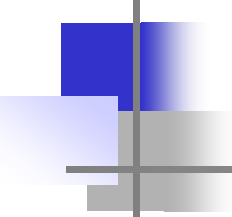
  SELECT image INTO lobd
  FROM covers
  where movie_id=99
  FOR UPDATE;

  DBMS_LOB.fileopen(fils, DBMS_LOB.file_readonly);
  DBMS_LOB.LOADFROMFILE(lobd,fils,DBMS_LOB.GETLENGTH(fils));
  DBMS_LOB.FILECLOSE(fils);

  COMMIT;

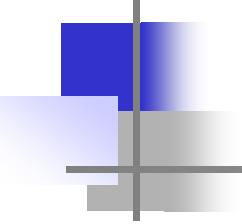
  SELECT DBMS_LOB.GETLENGTH(image)
  FROM covers WHERE movie_id=99;
END;
/
```

blokada wiersza



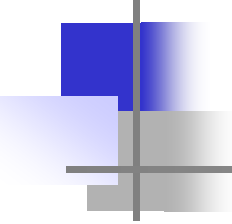
Ładowanie danych z BFILE do CLOB/NCLOB (1/2)

- Procedura DBMS_LOB.LOADFROMFILE nie obsługuje translacji między zestawami znaków
- Do ładowania danych znakowych do CLOB/NCLOB z pliku poprzez BFILE należy użyć DBMS_LOB.LOADCLOBFROMFILE
- Parametry DBMS_LOB.LOADCLOBFROMFILE:
 - dest_lob, src_bfile, amount
 - dest_offset , src_offset – przesunięcie (origin: 1)
 - bfile_csid – id zestawu znaków źródła
 - można go uzyskać funkcją NLS_CHARSET_ID, np. NLS_CHARSET_ID('EE8MSWIN1250') = 170
 - Wartość 0 oznacza przyjęcie database character set dla CLOB, a national character set dla NCLOB
 - lang_context – informacje kontekstowe przekazywane między kolejnymi operacjami ładowania danych z tego samego pliku
 - warning – informacja o problemie w trakcie ładowania
 - aktualnie tylko z powodu napotkania niekonwertowalnego znaku



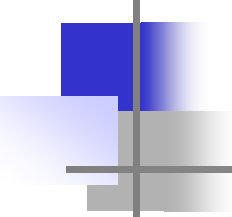
Ładowanie danych z BFILE do CLOB/NCLOB (2/2)

```
DECLARE
    lobd clob;
    fils BFILE := BFILENAME('DOC_DIR', 'dokument.txt');
    doffset integer := 1;
    soffset integer := 1;
    langctx integer := 0;
    warn integer := null;
BEGIN
    SELECT dokument INTO lobd FROM dokumenty
    WHERE id=2 FOR UPDATE;
    DBMS_LOB.fileopen(fils, DBMS_LOB.file_readonly);
    DBMS_LOB.LOADCLOBFROMFILE(lobd, fils, DBMS_LOB.LOBMAXSIZE,
        doffset, soffset, 873, langctx, warn); -- 873 to utf-8
    DBMS_LOB.FILECLOSE(fils);
    COMMIT;
    DBMS_OUTPUT.PUT_LINE('Status operacji: '||warn);
END;
/
```



SecureFiles w Oracle 11g

- Nowa architektura składowania dużych obiektów
- Interfejs kompatybilny wstecz
- Najlepsze rozwiązanie dla nieustrukturalizowanej zawartości
- Oferuje funkcjonalność charakterystyczną dla zaawansowanych systemów plików
 - wysoka wydajność
 - deduplikacja
 - kompresja
 - szyfrowanie
 - zaawansowane mechanizmy utrzymywania logu



SecureFiles w Oracle 11g – Przykład

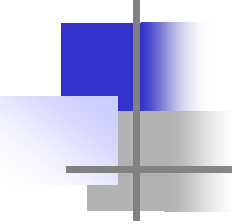
- Utworzenie tabeli ze wskazaniem architektury składowania dużego obiektu

- domyślnie:
STORE AS BASICFILE
(zależnie od param.
DB_SECUREFILE)

```
create table pracownicy (  
    id_prac      number(12),  
    nazwisko     varchar2(80),  
    ...,  
    zdjecie      blob)  
tablespace app_data  
lob (zdjecie)  
store as securefile (  
    tablespace app_lobs  
    enable storage in row  
    chunk 4096  
    pctversion 20  
    nocache  
    nologging)
```



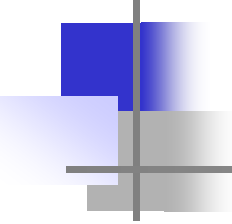
Metadane i modele danych dla multimedialnych baz danych



Modele danych dla multimedialnych baz danych (1/3)

- **Model relacyjny**

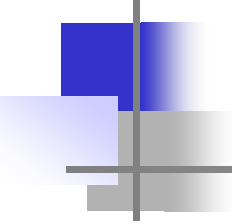
- zaproponowany przez E.F. Codd w 1970 r
- dane w postaci zbioru wzajemnie ze sobą powiązanych relacji (tabel); związki oparte o wartości (PK, FK)
- obecnie obok alfanumerycznych typów danych systemy relacyjne obsługują również typ BLOB (Binary Large Object)
 - BLOB umożliwia składowanie dużych obiektów (np. multimedialnych) w komórce tabeli
 - Istnienie typu BLOB jest odejściem od „czystego” modelu relacyjnego
 - BLOB jest łańcuchem bajtów, nie posiada struktury i dlatego nie wspiera operacji przeszukiwania danych multimedialnych



Modele danych dla multimedialnych baz danych (2/3)

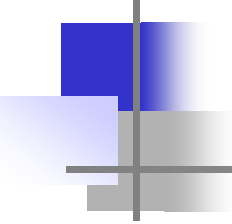
- **Model obiektowy**

- Rozwijany przez grupę ODMG
- Stanowi odpowiedź na ubóstwo modelu relacyjnego
- Intuicyjny dla człowieka
- Umożliwia modelowanie złożonych struktur i obiektów
- Wiąże dane z operacjami na nich
- Atrakcyjny dla multimediiów: powiązanie zawartości binarnych ze złożoną reprezentacją semantycznej zawartości obiektu
- Nie odniósł dużego sukcesu gdyż mimo 20 lat rozwoju ciągle nie zaproponowano satysfakcjonujących rozwiązań problemów: współbieżnego dostępu, optymalizacji zapytań, odtwarzania po awarii, itp.



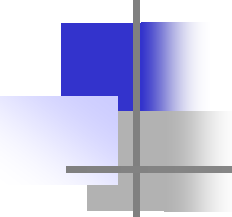
Modele danych dla multimedialnych baz danych (3/3)

- **Model obiektowo-relacyjny**
 - Odpowiedź „świata relacyjnego” na model obiektowy
 - Standard SQL99 (SQL3)
 - Elementy obiektowości uzupełniające model relacyjny: złożone typy danych, dziedziczenie, kolekcje, referencje do obiektów, ...
 - Stanowi bazę dla standardu SQL/MM dotyczącego obsługi multimediiów w bazach danych
 - Specyfikacja standardowych typów obiektowych do przechowywania obiektów multimedialnych (zawartość binarna + proste metadane)



Model danych a możliwości wyszukiwania danych

- Model danych w decydujący sposób wpływa na możliwości przeszukiwania kolekcji danych multimedialnych
- Powszechnie przyjmuje się że oprócz cyfrowej postaci medium (sekwencja bajtów, BLOB) w bazie danych utrzymywane będą metadane
- Metadane: dane o danych
 - W multimedialnych bazach danych stanowią dodatkową informację ułatwiającą interpretację i przeszukiwanie zawartości multimedialnej



Klasyfikacja metadanych o obiektach multimedialnych

- **Metadane sygnałowe**

- Wyodrębnione właściwości sygnału stanowiącego zapis zawartości multimedialnej
- Mogą być stosunkowo łatwo wygenerowane automatycznie
- Np. histogram kolorów, tekstura, dynamika sekwencji wideo, melodia

- **Metadane semantyczne**

- Informacje o obiektach, postaciach, zdarzeniach, itp. reprezentowanych przez treści multimedialne
- Trudne bądź niemożliwe do w pełni automatycznego uzyskania
- Powstają w wyniku tworzenia adnotacji przez człowieka

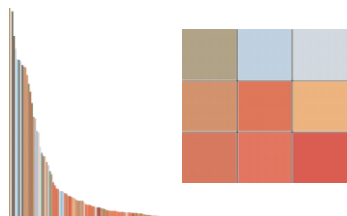
- **Metadane zewnętrzne**

- Informacje nieodnoszące się bezpośrednio do zawartości medium, tylko do sposobu jego tworzenia, przechowywania, udostępniania, klasyfikacji
- Np. reżyser filmu, nazwa pliku źródłowego, typ kompresji pliku, tytuł zdjęcia, czas i miejsce wykonania zdjęcia

Typy metadanych - Przykład

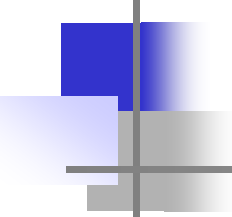


metadane sygnałowe



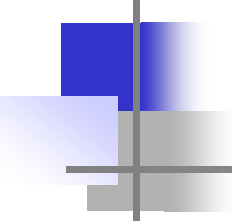
metadane semantyczne
łąka pełna maków
na dole po prawej kobieta
z parasolką i dziecko
...

metadane zewnętrzne
Autor: Claude Monet
Tytuł: Maki w Argenteuil
Styl: impresjonizm
Format: jpeg
Rozmiar: 640x480



Model danych i zapytania w świetle klasyfikacji metadanych (1/2)

- Metadane zewnętrzne są na ogół proste do zamodelowania, np. z użyciem modelu relacyjnego
 - Przykład relacji: informacje o pracownikach jako zbiór atrybutów alfanumerycznych i fotografia jako BLOB
 - Przykład zapytania: znajdź zdjęcia wszystkich Kowalskich
- Zapytania o metadane sygnałowe wymagają specjalistycznych algorytmów ekstrakcji i porównywania właściwości sygnałowych
 - Opcjonalnie również złożonych modeli reprezentacji właściwości
 - Często warunki selekcji oparte o metadane sygnałowe występują w połączeniu z warunkami dotyczącymi innych typów metadanych
 - Przykład zapytania: znajdź w Galerii Drezdeńskiej obrazy podobne pod względem kolorystyki do „Damy z łąsiczką” Leonarda Da Vinci



Model danych i zapytania w świetle klasyfikacji metadanych (2/2)

- Zapytania o metadane semantyczne wymagają złożonych modeli do reprezentacji cech obiektów, zdarzeń i zależności między nimi
 - Możliwość oparcia modelu multimedialnej bazy danych o model obiektowy
 - Stworzenie ogólnego modelu dla metadanych semantycznych jest zadaniem trudnym
 - Przykład zapytania: znajdź wszystkie sekwencje wideo, na których Jerzy Dudek broni rzut karny

Relacyjny model danych dla MBD – Przykład

relacja SEKWENCJE_WIDEO

ID	ID TEMATU	SEKWENCJA (BLOB)	DLUGOSC W KLATKACH
10	1		1000
20	2		3450

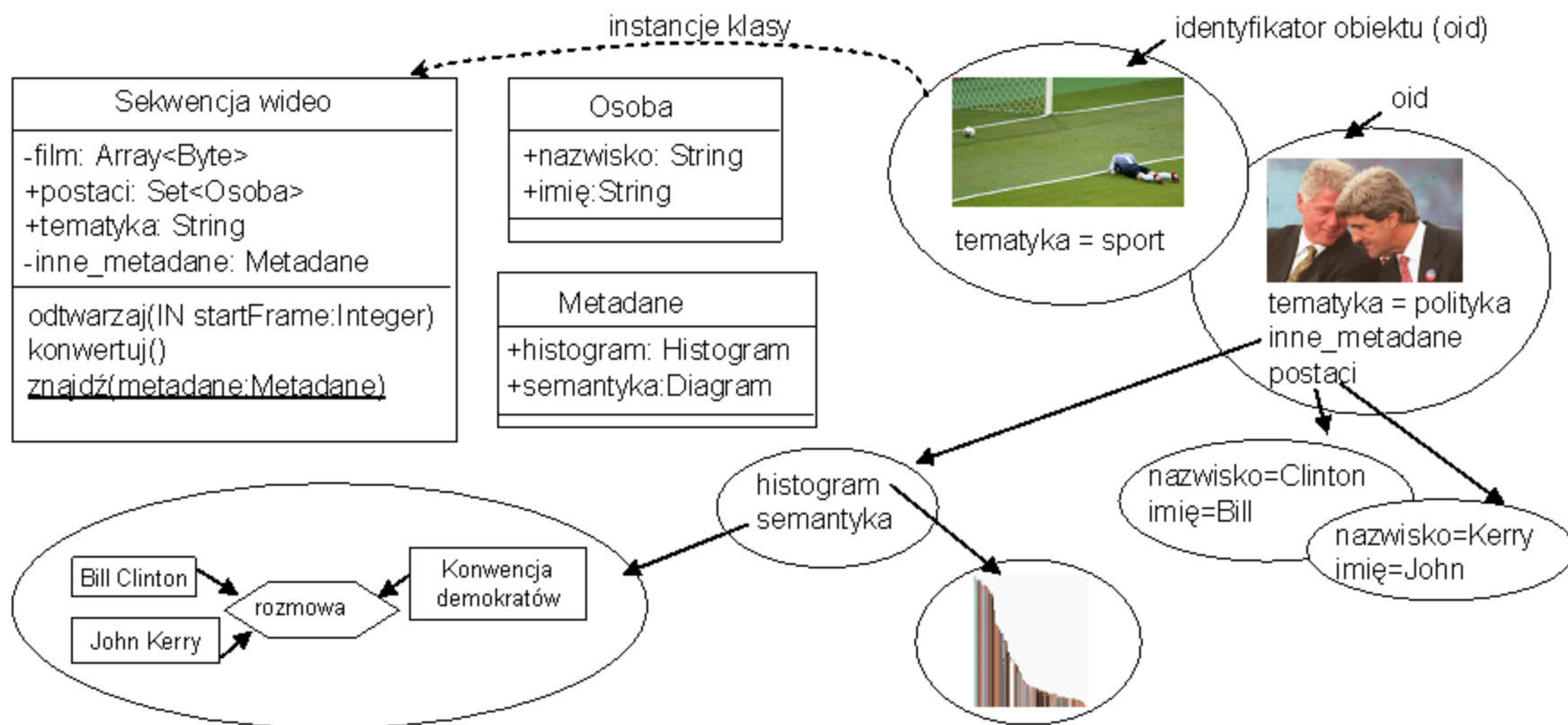
relacja TEMATY

ID	NAZWA	OPIS
1	POLITYKA	
2	SPORT	RELACJE SPORTOWE

relacja POSTACI

ID	NAZWISKO	ID_SEKW	START_FRAME	STOP_FRAME
1	CLINTON	10	0	800
2	KERRY	10	500	1000
...				

Obiektowy model danych dla MBD – Przykład



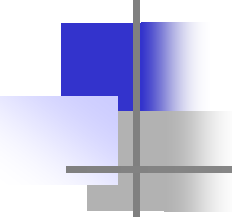
Obiektowo-relacyjny model danych + XML dla MBD – Przykład

relacja SEKWENCJE_VIDEO

ID	SEKWENCJA	METADANE
10	Sekwencja_wideo	XMLType
		
	długość_w_klatkach = 1000 odtwarzaj(IN startFrame:Integer) konwertuj(nowyFormat Varchar)	
20	Sekwencja_wideo	
		
	długość w klatkach = 3450	

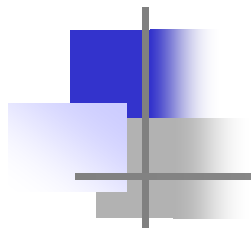
```

<Mpeg7>
...
<VideoSegment>
  <MediaTime>
    <MediaTimePoint> T0:0:0 </MediaTimePoint>
    <MediaIncrDuration mediaTimeUnit="PT1N25F"> 1000
  </MediaIncrDuration>
  </MediaTime>
  <TextAnnotation>
    <StructuredAnnotation>
      <Who><Name> Bill Clinton</Name></Who>
      <Who><Name> John Kerry</Name></Who>
      <WhatAction><Name>talk</Name></WhatAction>
    </StructuredAnnotation>
  </TextAnnotation>
  <GofGopColorHistogram HistogramTypeInfo = "Average">
    ....
  </GofGopColorHistogram>
</VideoSegment>
...
</Mpeg7>
  
```

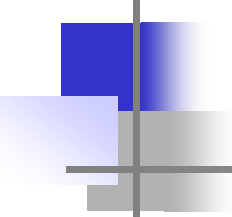


Modele danych dla multimedialnych baz danych – stan aktualny

- Wiele rozwiązań prototypowych do reprezentacji metadanych semantycznych:
 - Typowo dedykowane dla jednego obszaru zastosowań
 - Brak wpływu na standardy i systemy komercyjne
- Systemy komercyjne (Oracle, IBM DB2):
 - Wykorzystanie rozszerzeń obiektowo-relacyjnych
 - Ekstrakcja i składowanie metadanych o formacie
 - Ekstrakcja i składowanie metadanych sygnałowych dla obrazów (aktualnie odwrót od tej funkcjonalności...)
 - W zasadzie brak wsparcia dla metadanych semantycznych
- Standardy:
 - SQL/MM - brak wsparcia dla metadanych semantycznych
 - MPEG-7 – reprezentacja wszystkich typów metadanych w postaci XML; na razie słabe wsparcie w bazach danych

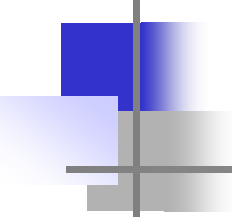


Składowanie i prezentacja danych multimedialnych



Problemy prezentacji danych multimedialnych

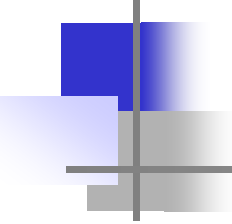
- Zawartość bazy danych składowana jest w pamięci masowej
- Bazy danych dostępne są zazwyczaj poprzez sieć komputerową
- Dostęp do pamięci masowej i transmisja sieciowa wnoszą opóźnienia
- W przypadku danych alfanumerycznych i statycznych multimedialnych (obrazów) opóźnienia te nie wpływają negatywnie na postrzeganie zawartości przez użytkownika
- W przypadku dynamicznych danych multimedialnych (audio, wideo) prezentowanych w czasie rzeczywistym opóźnienia mogą zakłócić odbiór przekazu przez użytkownika – architektury serwerów multimedialnych muszą brać ten problem pod uwagę



Źródła opóźnień przy dostępie do danych

- Odczyt danych z pamięci masowej (czas dostępu, prędkość transmisji)
 - Neutralizacja poprzez: (1) duplikowanie dysków lub partycjonowanie filmów na kilka dysków w celu zrównoleglenia odczytu; (2) cache
- Inne elementy serwera: CPU, magistrala
- Przepustowość sieci komputerowej
 - Neutralizacja przez buforowanie po stronie klienta

(!) Problem występowania opóźnień uwidacznia się w przypadku gdy wielu użytkowników korzysta z zasobów serwera



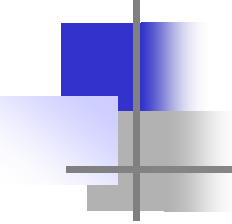
Architektury serwowania dynamicznych danych multimedialnych

- **Client Pull**

- Klient odpowiada za żądania kolejnych porcji wymaganych danych
- Typowe podejście w dzisiejszych systemach (np. serwery WWW)

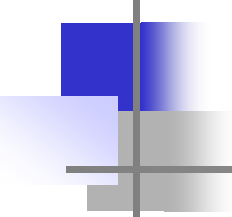
- **Server Push**

- Klient subskrybuje dany film video
- Serwer pamięta stan dla każdego klienta i sam przesyła klientowi kolejne porcje danych
- Ułatwia zapewnienie odpowiedniej jakości usług przez kontrolę obciążenia serwera



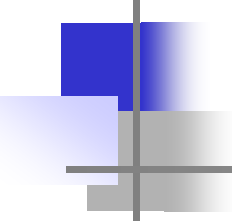
Prezentacja danych wrażliwych na opóźnienia (1/2)

- Dla pojedynczego strumienia danych ilość produkowanych danych musi w każdej chwili przewyższać ilość danych konsumowanych
- Problemy:
 - Poziom konsumpcji danych może być zmienny w czasie ze względu na kompresję danych
 - Poziom produkcji danych może być zmienny ze względu na konieczność zmiany ścieżek, zmienne prędkości transmisji danych dla różnych regionów nośnika, itp.



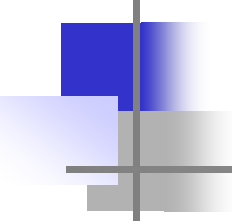
Prezentacja danych wrażliwych na opóźnienia (2/2)

- W przypadku obsługi wielu strumieni współbieżnie, są one obsługiwane w tzw. rundach – w każdej rundzie do każdego strumienia przekazywana jest odpowiednia ilość danych (zgodnie z poziomem konsumpcji danych)
- Możliwych jest wiele algorytmów szeregowania obsługi kolejnych strumieni np.:
 - Round Robin (ustalona kolejność)
 - SCAN (w danej rundzie głowica dysku przesuwa się w jedną stronę)
 - wymagany bufor na dwie rundy
 - Grouped sweeping (sets of streams: RR, within each set: SCAN)
 - EDF (Earliest Deadline First)
 - SCAN-EDF (EDF, a gdy kilka równy deadline to dla nich SCAN)
 - możliwość wymuszania identycznych deadline'ów



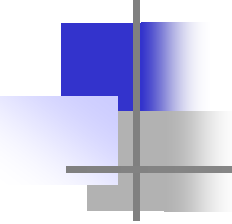
Gwarancje jakości usług

- Jednoczesna obsługa zbyt wielu klientów w zakresie prezentacji wideo/audio w czasie rzeczywistym doprowadzi do wyczerpania zasobów i pogorszenia jakości odbioru
- Oferowanie klientowi usług powinno mieć miejsce tylko wtedy gdy system jest w stanie te usługi zapewnić
- Gwarancje odpowiedniej jakości usług oznaczają jednocześnie ograniczenie liczby obsługiwanych żądań



Rodzaje gwarancji

- **Deterministyczne**
 - Najbardziej restrykcyjne, oparte na analizie najgorszego przypadku w oparciu o parametry systemu
- **Statystyczne**
 - Oparte o rozkłady statystyczne, gwarantują np. że w 99,9% przypadków dane zostaną przesłane na czas
- **Oparte na predykcji**
 - Wyznaczone na bazie doświadczeń z dotychczasowej eksploatacji systemu
- **Best effort**
 - Brak gwarancji, stan obecnie powszechny (np. serwery WWW), nieakceptowalne

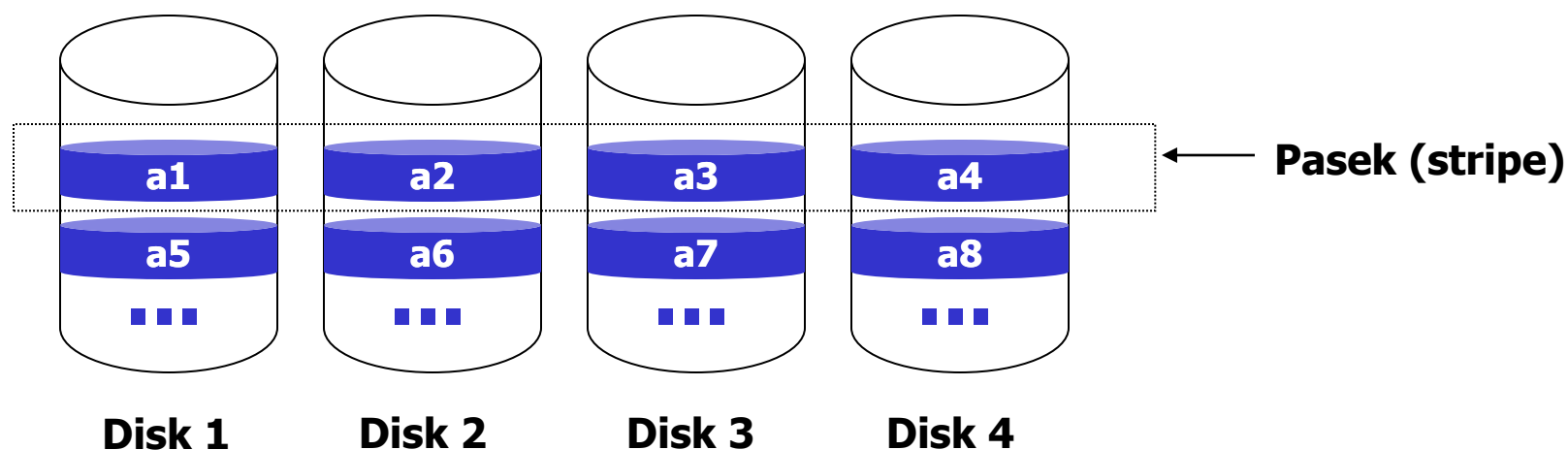


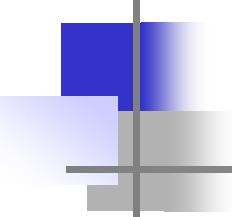
Techniki usprawniające prezentację danych wrażliwych na opóźnienia

- Sposoby składowania danych:
 - Striping
 - Replikacja
- Techniki obsługi transmisji strumieniowej:
 - Batching
 - Bridging
 - Piggybacking
 - Buforowanie (cache)

Składowanie danych – Striping (1/2)

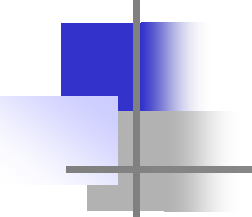
- Polega na alokacji kolejnych fragmentów pliku cyklicznie na kolejnych dyskach





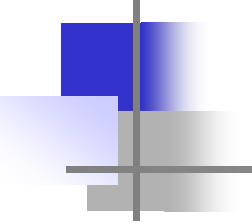
Składowanie danych – Striping (2/2)

- Dane z jednego paska mogą być odczytane równolegle
- Prędkość transmisji dla macierzy k dysków
= $k * \text{prędkość transmisji pojedynczego dysku}$
(zwiększenie przepustowości pamięci dyskowej serwera)
- Technika odpowiednia dla:
 - Filmów wymagających dużej przepustowości
 - Pamięci masowych (dysków) o małej prędkości transmisji danych, taśm



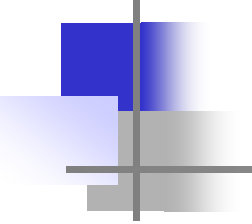
Składowanie danych - Replikacja

- Replikacja danych na kilku dyskach zwiększa przepustowość dostępu do pamięci dyskowej, ale wiąże się z marnotrawstwem miejsca
- Stopniowe zwiększanie przepustowości jest łatwiejsze niż gdy stosowany jest striping
- Replikacja zwiększa odporność na awarię
- Istnieje możliwość replikacji części plików i dystrybucji pozostałych w oparciu o striping



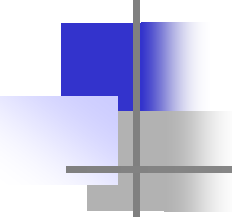
Batching, Bridging, Piggybacking

- Techniki ograniczające liczbę strumieni dla których trzeba pobrać dane z pamięci wtórnej
- **Batching** – polega na odsunięciu w czasie rozpoczęcia transmisji – być może nadejdą identyczne żądania i grupę żądań obsłuży jeden strumień
- **Bridging** – dla strumieni dotyczących tego samego filmu, nieznacznie przesuniętych w czasie, polega na przechowywaniu danych z okna czasowego między strumieniami w pamięci cache
- **Piggybacking** – dla strumieni dotyczących tego samego filmu, nieznacznie przesuniętych w czasie, polega na przyspieszeniu drugiego filmu, tak aby oba żądania były od pewnego momentu obsługiwane przez jeden strumień



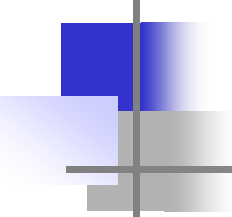
Batching – problemy, rozwiązania

- Problemy:
 - długi okres oczekiwania może sprawić, że użytkownik zrezygnuje z usługi i opuści system
 - który strumień spośród wstrzymanych uruchomić jako następny?
- Możliwe rozwiązania:
 - kolejność FIFO
 - w pierwszej kolejności strumień, na który oczekuje największa liczba użytkowników
 - uwzględnienie częstotliwości żądań dotyczących poszczególnych filmów
 - najpopularniejsze uruchamiane okresowo
 - pozostałe na zasadzie FCFS



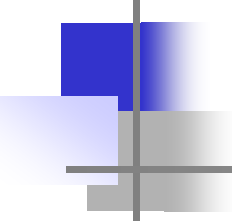
Wsparcie dla operacji VCR (1/2)

- Funkcjonalność swobodnej nawigacji stanowi duże wyzwanie dla systemu strumieniowej dystrybucji wideo i wymaga dedykowanego wsparcia
- Podejścia:
 - Pobieranie wszystkich klatek i ich pomijanie po stronie klienta
 - niepotrzebne obciążenie systemu
 - Pobieranie tylko klatek typu I
 - wymaga ich odrębnego składowania
 - Wykorzystanie indeksu do dostępu do dowolnych klatek w czasie rzeczywistym
 - wymaga częściowego dekodowania zapisu wideo (np. MPEG)



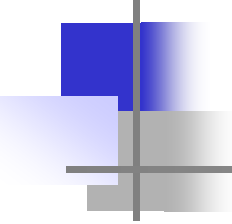
Wsparcie dla operacji VCR (2/2)

- Kwestie równoważenia obciążenia w przypadku architektur składowania opartych o striping lub replikację
 - łatwiejsze dla stripingu
 - przy replikacji wybór kopii musi uwzględniać aktualne obciążenie
- Batching w świetle operacji VCR
 - operacje VCR mogą spowodować konieczność podziału strumienia na dwa strumienie
 - w pierwszej fazie po operacji system może wykorzystać bridging
 - docelowo system musi uwzględniać konieczność podziału na niezależne strumienie
 - system wspierający operacje VCR powinien rezerwować kilka kanałów na potrzeby ewentualnych podziałów strumieni



Buforowanie transmisji strumieniowej

- Buforowanie jest powszechną techniką zwiększenia wydajności dostępu do danych
- Buforowanie polega na przechowywaniu części danych w szybszej pamięci (cache), aby były łatwiej dostępne
- Buforowanie może być realizowane poprzez:
 - Odczyt danych z wyprzedzeniem
 - Pozostawienie w buforze wykorzystanych danych, z myślą o kolejnych żądaniach
- Po wypełnieniu pamięci cache, aby sprowadzić nowe dane należy pewne dane usunąć z cache:
 - Algorytm LRU używany powszechnie np. w systemach operacyjnych nie jest odpowiedni dla zbioru strumieni
 - Dobrym rozwiązaniem jest **Interval Caching**

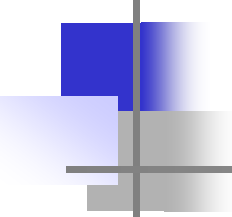


Algorytm Interval Caching

- Daje podobny efekt do techniki bridging
- Gdy istnieje konieczność zwolnienia miejsca w pamięci cache:
 - Usuwane są dane, które nie będą potrzebne aktualnie obsługiwanym użytkownikom (jeśli nie zmieniają żądania)
 - Pozostają w buforze dane, które będą w przyszłości wymagane przez aktywne strumienie
 - Spośród danych, które będą w przyszłości wymagane przez aktywne strumienie, preferowane są dane, dla których przesunięcie w czasie między strumieniami jest najmniejsze

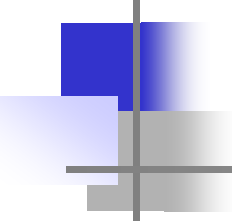


Przetwarzanie zapytań w multimedialnych bazach danych



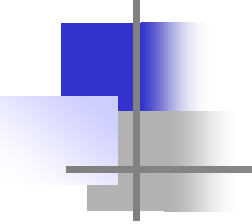
Charakterystyka zapytań do multimedialnych baz danych

- Zapytania o metadane, ale również o zawartość (content-based retrieval)
- Języki tekstowe, ale również duża rola narzędzi wizualnych
 - np. zapytanie poprzez narysowanie przykładowego obrazu
- Przy zapytaniach o zawartość podejście Information Retrieval (wyszukiwanie informacji):
 - Wyszukiwanie ma charakter procesu iteracyjnego z dostrajaniem parametrów selekcji
 - Możliwość ustalania wag dla poszczególnych kryteriów
 - Dopuszczalne pominięcie pewnych obiektów w wynikach zapytania, subiektywność relacji podobieństwa



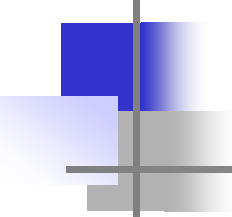
Information Retrieval (IR)

- Nauka o wyszukiwaniu w oparciu o informacje zawarte w obiektach i metadanych o nich
 - początkowo dotyczyła dokumentów
 - zakres zastosowań rozszerzony o relacyjne bazy danych, World Wide Web, multimedia
- Integruje wiele różnych dziedzin naukowych:
 - informatyka, matematyka, statystyka, nauka o informacji, fizyka
 - bibliotekarstwo, językoznawstwo, psychologia poznawcza
- Ukierunkowana na rozwiązanie problemu nadmiaru informacji
- Najpopularniejsze obecnie aplikacje IR to wyszukiwarki internetowe



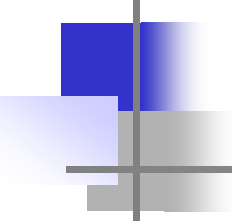
Proces wyszukiwania informacji

- 1) Użytkownik wysyła zapytanie do systemu
 - zapytanie jest formalnym zapisem potrzeb informacyjnych
 - wiele obiektów może pasować do zapytania, potencjalnie w różnym stopniu
- 2) Warunki zapytania są dopasowywane do obiektów z bazy danych
 - baza danych może zawierać obiekty lub jedynie reprezentację ich charakterystyk
- 3) Obiekty najlepiej pasujące do warunków zapytania są zwracane użytkownikowi
 - znajdujące się na początku utworzonego rankingu dopasowania
- 4) Użytkownik może skorygować zapytanie
 - jeśli nie jest usatysfakcjonowany wynikami
 - proces może mieć charakter iteracyjny



Miary oceny działania systemów IR

- Precision = dokładność
- Recall = kompletność
- F-score (F-measure) = miara F

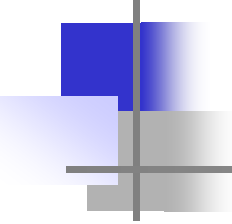


Precision

- Odsetek dokumentów odpowiadających potrzebom użytkownika wśród dokumentów zwróconych przez zapytanie

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

- Może być również wyznaczana tylko dla top-n dokumentów zwróconych przez zapytanie (precision at n)

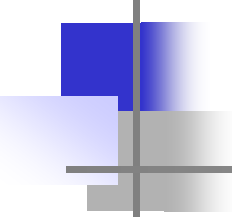


Recall

- Odsetek dokumentów odpowiadających potrzebom użytkownika, które zostały zwrócone przez zapytanie

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

- Może być interpretowana jako prawdopodobieństwo, że interesujący użytkownika dokument zostanie zwrócony przez zapytanie
- Ponieważ 100% kompletności można osiągnąć zwracając wszystkie obiekty z bazy danych, recall w praktyce jest używana w połączeniu z inną miarą – typowo: precision



F-score

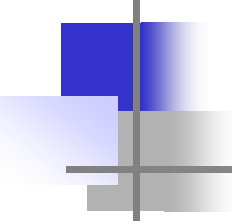
- Miara uwzględniająca precision i recall
- Tradycyjna postać (zwana miarą F1) to średnia harmoniczna precision i recall

$$F = \frac{2 \cdot \text{precision} \cdot \text{recall}}{(\text{precision} + \text{recall})}$$

- Ogólna formuła
 - β – ile razy bardziej istotna recall niż precision ($\beta \geq 0$)

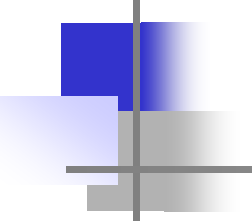
$$F_{\beta} = \frac{(1 + \beta^2) \cdot (\text{precision} \cdot \text{recall})}{(\beta^2 \cdot \text{precision} + \text{recall})}$$

- popularne warianty: $F_{0.5}$ i F_2



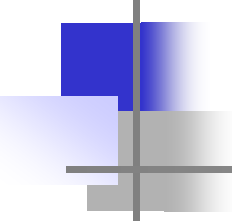
Content-based Retrieval

- Technologia najlepiej dopracowana dla obrazów
- Wyszukiwanie w oparciu o opisy obiektów wygenerowane automatycznie:
 - technikami ekstrakcji własności wizualnych
 - technikami rozpoznawania obiektów
- Alternatywa lub uzupełnienie (!) dla wyszukiwania w oparciu o alfanumeryczne opisy wprowadzone „ręcznie”



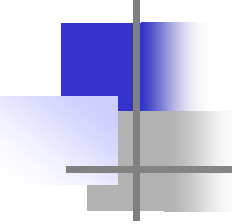
Content-based Retrieval - Zalety

- Tworzenie alfanumerycznych opisów zawartości obrazów jest czasochłonne, a nie może być wykonane bez udziału człowieka
- Manualny opis zawartości jest niepraktyczny
 - dla dużych, uaktualnianych kolekcji obrazów
- Nie wszystkie własności obrazu można w sposób naturalny opisać słownie, np. tekstury, kształty
- Wyszukiwanie w oparciu o zawartość może uzupełniać tradycyjne wyszukiwanie w oparciu o opis tekstowy, słowa kluczowe, itp.



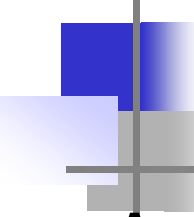
Content-based Retrieval – Praktyczne zastosowania

- Ochrona praw autorskich, wykrywanie plagiatów:
 - Utwory muzyczne, znaki zastrzeżone, loga, ...
- Praktyczne zastosowania w kontekście obrazów:
 - przeszukiwanie galerii obrazów
 - przeszukiwanie katalogów:
 - ubrań, tkanin, materiałów
 - tapet i innych materiałów do wykańczania i dekoracji wnętrz



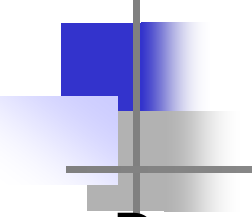
Przykłady zapytań o metadane (metadane zewnętrzne)

- Audio
 - Znajdź wszystkie utwory wykonawcy X
 - Znajdź wszystkie utwory o zadanym czasie trwania
- Obrazy
 - Znajdź wszystkie fotografie wykonane przez X, dostępne w formacie PNG
- Wideo
 - Znajdź wszystkie komedie o czasie trwania od 1,5h do 2h
 - Znajdź filmy reżysera Y, dostępne w rozdzielczości 640x480 i co najmniej 30 klatek na sekundę



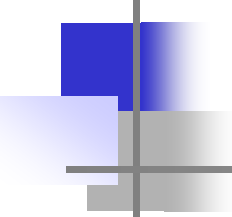
Przykłady zapytań o zawartość (metadane sygnałowe i semantyczne)

- Audio
 - Znajdź wszystkie utwory zawierające motyw zanucony przez użytkownika do mikrofonu (query by humming)
 - Znajdź wszystkie przemówienia dotyczące problemu bezrobocia
- Obrazy
 - Znajdź obrazy zawierające dużo czerwonego koloru
 - Znajdź obrazy podobne do danego
 - Znajdź obrazy przedstawiające prezydenta z premierem
- Wideo
 - Znajdź filmy (sceny), w których ...
 - Znajdź wszystkie filmy z aktorem przedstawionym na danym zdjęciu



Klasyfikacja interfejsów do zapytań multimedialnych

- Przeglądanie (browsing)
 - Bez ustalonych kryteriów selekcji, począwszy od predefiniowanego zbioru obiektów
- Wyszukiwanie wg słów kluczowych
 - Wymaga obecności w bazie danych opisu słownego zawartości multimedialnej
 - Opis słowny nie zawsze łatwy np. dla tekstur
- Wyszukiwanie podobieństw
 - W oparciu o cechy wyznaczone przez specjalne algorytmy
 - Wykorzystywane cechy obiektów mają charakter abstrakcyjny, niezwiązany z treścią, abstrakcyjny dla zwykłych użytkowników
 - Zapytania przez podanie przykładu (QBE)

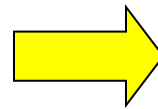
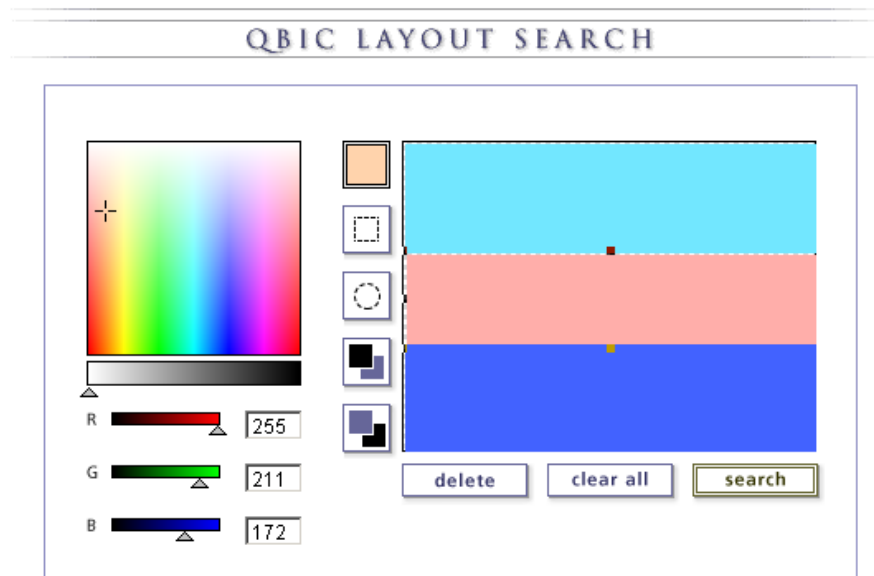


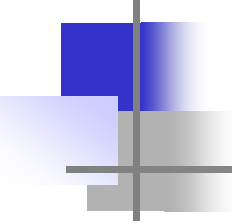
Interfejsy do zapytań przez przykład

- Query by Pictorial Example
 - Zapytanie w postaci pełnego obrazka
- Query by Painting, Sketch Retrieval
 - Użytkownik rysuje wzorcowy obrazek za pomocą prostego interfejsu
- Selection from Standards
 - np. wybór kształtów i tekstur z listy
- Image Montage
 - Składanie wzorcowego obrazka z elementów
- Visual Concepts and Dictionaries
 - Początkowy wzorzec dla wyszukiwania wybierany z katalogu

Przykład interfejsu: Query by Sketch

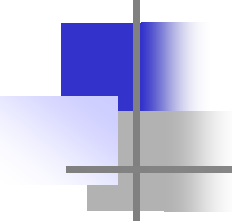
- www.hermitagemuseum.org
 - Przeszukiwanie galerii obrazów na podstawie zadanego rozkładu przestrzennego kolorów





Obsługa zapytań o zawartość – Obrazy (1/2)

- W zakresie zapytań o zawartość większość rozwiązań zaproponowano w kontekście przeszukiwania baz obrazów
- Zapytania o zawartość obrazów doczekały się standaryzacji (SQL/MM) i implementacji przez komercyjne systemy
 - Oracle
 - wcześniej również IBM DB2 i Informix
- Obsługa zapytań o zawartość w kontekście danych audio i wideo jest przedmiotem ciągłych badań (prototypy badawcze)



Obsługa zapytań o zawartość – Obrazy (2/2)

- Typowe właściwości obrazów wykorzystywane w zapytaniach:
 - Średni kolor
 - Histogram kolorów
 - Kontury segmentów obrazu (kształty)
 - Tekstury
 - Lokalizacja kolorów, kształtów i tekstur na obrazie
- (!) Wykorzystywane miary podobieństwa umożliwiają użytkownikowi podanie wag dla poszczególnych właściwości

Właściwości obrazów: kolory

- **Średni kolor**

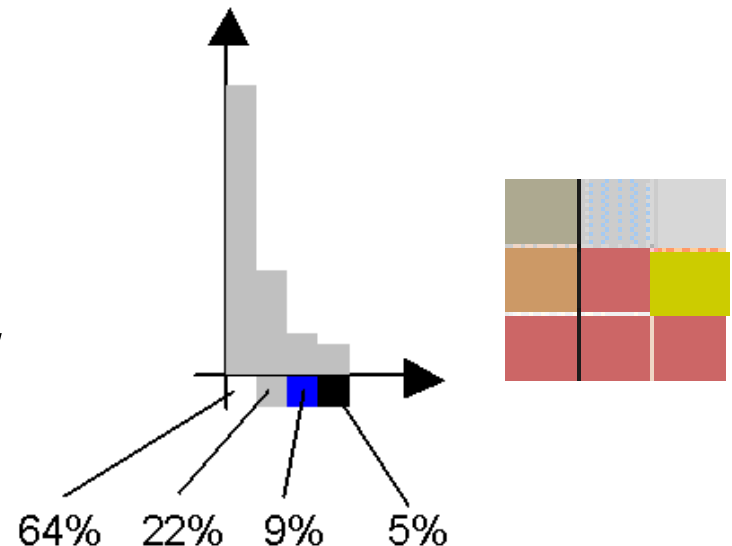
- Obraz jest dzielony na n próbek
- Komponenty RGB koloru poszczególnych próbek są niezależnie od siebie sumowane i dzielone przez liczbę próbek

- **Histogram kolorów**

- Udział kolorów w obrazie
- Często z redukcją liczby kolorów do np. 256

- **Lokalizacja kolorów**

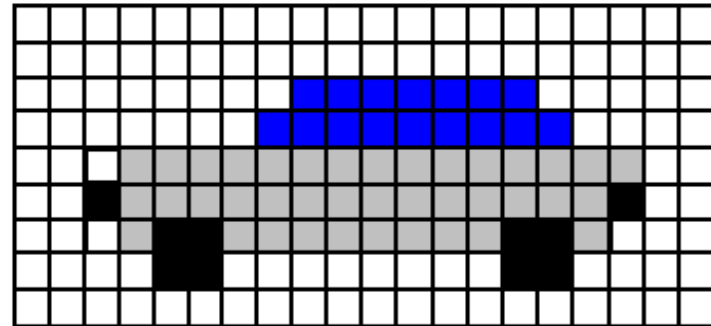
- obraz jest dzielony na $m \times n$ prostokątów i dla każdego z prostokątów wyznaczany jest dominujący kolor



Właściwości obrazów: kształty i tekstury

- **Kształty**

- Odkrywane w procesie segmentacji obrazu
- Segmenty wyznaczane jako spójne regiony o tym samym kolorze

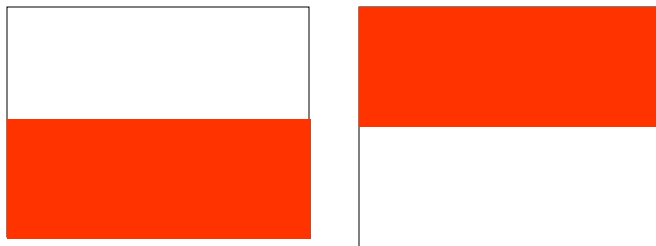


- **Tekstury**

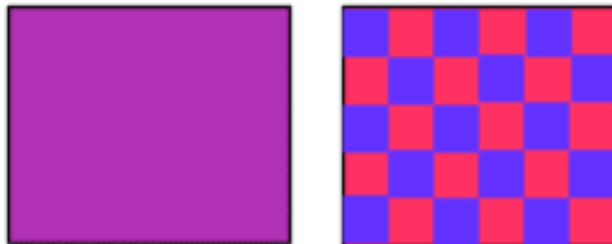
- Opisują chropowatość, kontrast, kierunkowość wzorców wypełniających kształty i tło obrazu
- Reprezentowane w postaci wektora współczynników liczbowych, macierzy kowariancji

Podobieństwo obrazów – Przykłady (1/3)

- Obrazy maksymalnie podobne pod względem koloru (kolor średni, udział kolorów)
- Obrazy niepodobne pod względem lokalizacji kolorów

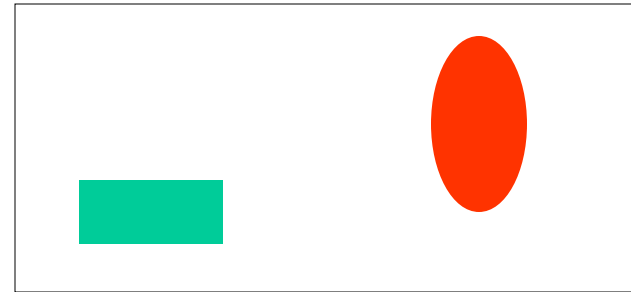
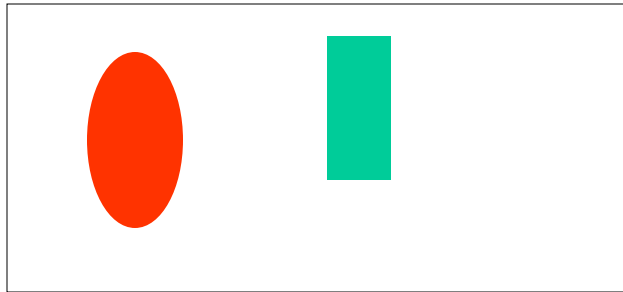


- Obrazy identyczne pod względem średniego koloru

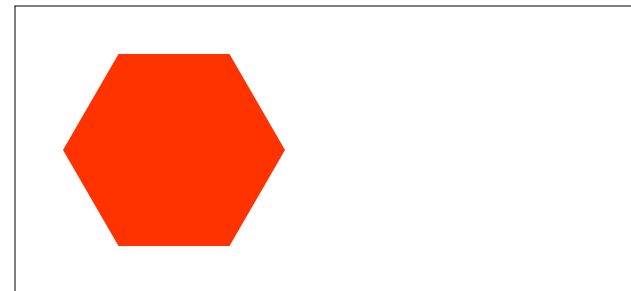
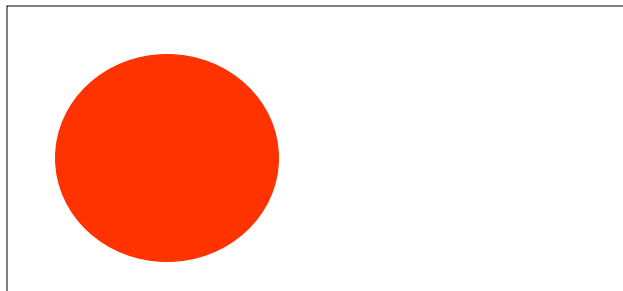


Podobieństwo obrazów – Przykłady (2/3)

- Obrazy bardzo podobne pod względem koloru

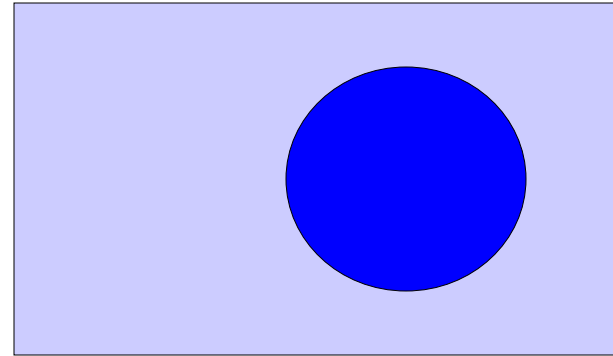
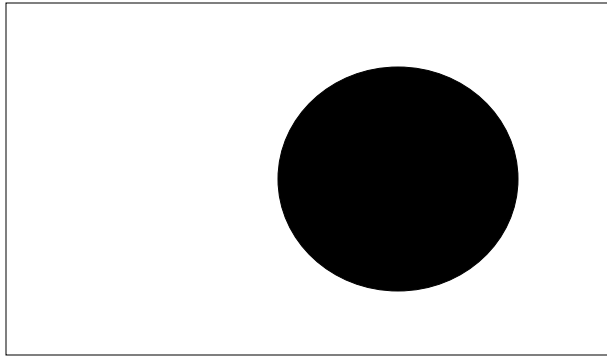


- Obrazy podobne pod względem koloru i lokalizacji kolorów

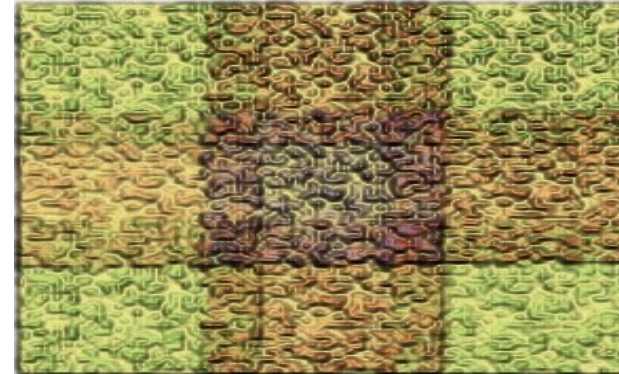
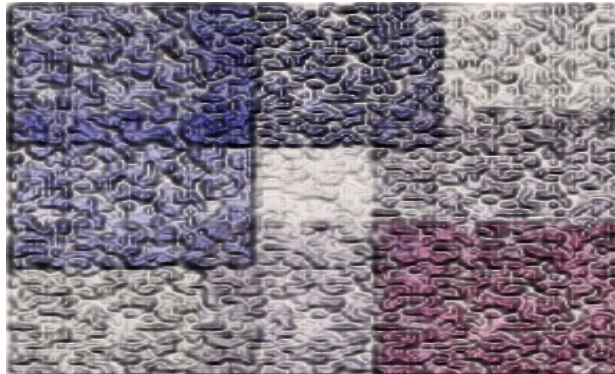


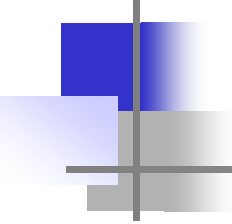
Podobieństwo obrazów – Przykłady (3/3)

- Obrazy bardzo podobne pod względem kształtu



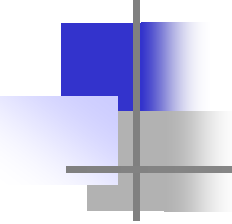
- Obrazy o podobnej teksturze





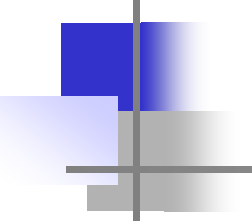
Indeksowanie multimedialnych baz danych

- Rozumiane jako przyporządkowanie lub ekstrakcja właściwości, które będą stanowiły bazę dla zapytań
 - często powiązane z segmentacją – podziałem na jednostki wyszukiwania (retrieval units)
 - właściwości mogą być umieszczone w strukturach indeksowych
- Dwa podejścia do indeksowania multimediiów
 - manualne - nazywanie obiektów i związków między nimi za pomocą języka naturalnego lub predefiniowanych fraz
 - automatyczne – identyfikacja matematycznych (sygnałowych) charakterystyk zawartości
 - z możliwością ręcznej korekty
 - różne algorytmy dla różnych rodzajów multimediiów



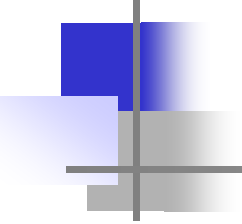
Automatyczne indeksowanie obrazów

- Segmentacja w oparciu o kryterium homogeniczności
 - np. segment jako spójny podzbiór pikseli o zbliżonych wartościach (zbliżonym kolorze)
- Indeksowanie poprzez ekstrakcję obiektów i niskopoziomowych właściwości
 - kształty, tekstury, histogram kolorów, lokalizacja



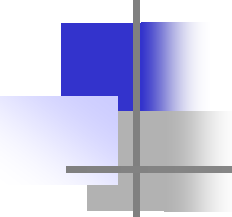
Automatyczne indeksowanie audio

- Segmentacja jako podział na sekwencje stanowiące elementarne jednostki indeksowania i wyszukiwania
 - często manualna
- Indeksowanie
 - rozpoznawanie mowy i indeksowanie wynikowych tekstów
 - analiza akustyczna – ekstrakcja melodii, zapisu nutowego
 - reprezentacja w formie łańcuchów symboli



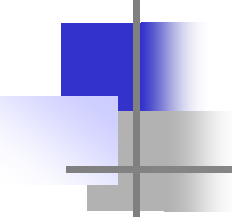
Automatyczne indeksowanie wideo

- Segmentacja
 - poprzez detekcję początków i końców ujęć kamery
 - w oparciu o granice w towarzyszącej ścieżce audio
 - w oparciu o segmentację scenariusza (jeśli dostępny)
 - w oparciu o reguły heurystyczne wynikające ze znajomości natury danego typu zawartości (news, sport)
- Indeksowanie
 - identyfikacja obiektów i aktywności



Podejścia do automatycznej ekstrakcji właściwości obiektów

- Ekstrakcja a priori
 - W zapytaniach dopuszczalne są tylko właściwości z predefiniowanego zestawu
 - Przy wstawianiu obiektów do bazy danych, dla każdego z nich wyznaczany jest dany zestaw właściwości
 - Właściwości obiektów z bazy danych mogą być umieszczone w indeksie
- Ekstrakcja dynamiczna
 - Występuje np. przy zapytaniach o obrazy zawierające obiekt podobny do fragmentu obrazu przykładowego
 - Wszystkie obiekty z bazy danych są przetwarzane
- Kombinacja ekstrakcji a priori i dynamicznej
 - Niektóre właściwości wyznaczone a priori i poindeksowane

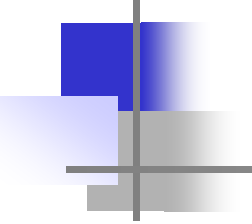


Zapytania w MBD w świetle struktur indeksowych

- Typy zapytań w tradycyjnych bazach danych:
 - Point queries
 - Range queries
 - Partial match queries
 - Partial range queries
- Specyfika multimedialnych baz danych:
 - Wyszukiwanie obiektów **podobnych** do podanego
 - Kluczowe jest pojęcie (miara) podobieństwa
 - Brak ogólnej definicji gdyż zależy od aplikacji
 - Ogólna postać:

$$\delta: Obj \times Obj \rightarrow \mathbb{R}_0^+$$

- Podobieństwo jako „odległość”: 0 dla obiektów identycznych



Rodzaje zapytań względem miary podobieństwa

- **ε -similarity queries**

- Wyszukiwanie obiektów, dla których wartość miary podobieństwa do zadanego obiektu jest poniżej progu ε
- Dwa obiekty obj_1 i obj_2 są określane jako **ε -podobne** (ε -similar) iff:
$$\delta(obj_2, obj_1) < \varepsilon$$
- Dla $\varepsilon=0$ obiekty są określane jako **identyczne**

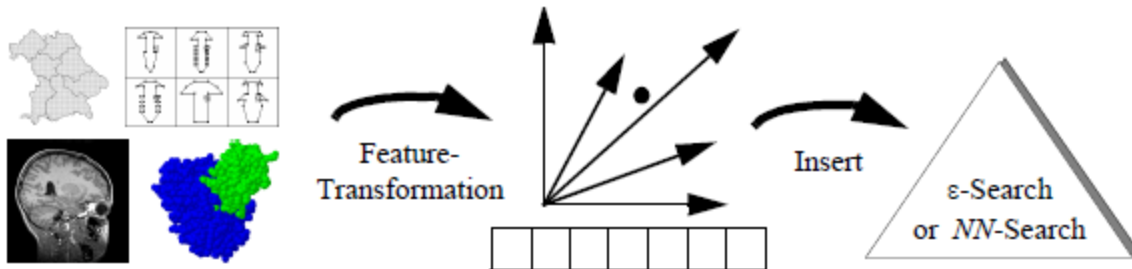
- **NN-similarity queries**

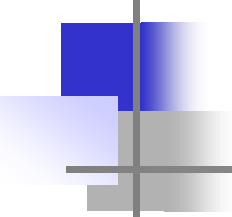
- Wyszukiwanie obiektów najbardziej podobnych do zadanego
- Dwa obiekty obj_1 i obj_2 są określane jako **NN-podobne** (NN-similar) w kontekście bazy danych DB iff:

$$\forall obj \in DB, obj \neq obj_1: \delta(obj_2, obj_1) \leq \delta(obj_2, obj)$$

Wyszukiwanie podobieństw w oparciu o właściwości

- Podobieństwo obiektów jest typowo wyznaczane w oparciu o ich właściwości (np. automatycznie ekstrahowane)
 - „feature-based similarity search”
- Właściwości obiektów są reprezentowane w formie wielowymiarowych wektorów właściwości (feature vectors)
 - „feature transformation”: $F: Obj \rightarrow \mathbb{R}^d$
- Szukanie podobnych obiektów sprowadza się do szukania podobnych wektorów właściwości
 - Miara podobieństwa obiektów: $\delta(obj_1, obj_2) = \delta_{Euclid}(F(obj_1), F(obj_2))$

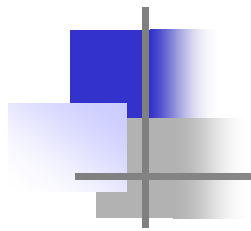




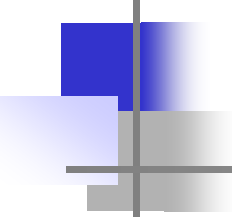
Wymagania względem struktur indeksowych w MBD

- Duże multimedialne bazy danych wymagają odpowiednich struktur indeksów wielowymiarowych
 - Obsługa typów zapytań charakterystycznych dla MBD
 - Akceptowalna wydajność dla bardzo wielu wymiarów
 - Problem „dimensionality curse”
 - Ze wzrostem liczby wymiarów spada wydajność (nawet wykładniczo)
 - Ze wzrostem liczby wymiarów maleje skuteczność realizacji niektórych operacji
 - W bazach danych przestrzennych 2-4 wymiarów
 - W bazach danych multimedialnych nawet 256-1000 wymiarów
- Wsparcie dla zapytań:
 - Punktowych (point) – dla zapytań o identyczne
 - Przedziałowych (range) – dla zapytań o ε -podobne
 - Wariant: window query
 - Najbliższy sąsiad (nearest neighbor) – dla zapytań o NN-podobne
 - Warianty: k-NN, approximate NN, approximate k-NN

Również all-pairs!
(spatial join
względem predykatu)

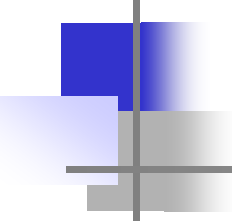


Standard SQL/MM



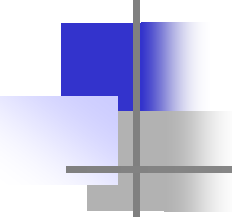
Standard SQL/MM

- SQL/MM: SQL Multimedia and Application Packages
- Opracowywany i publikowany przez ISO
- Oparty o SQL i jego typy definiowane przez użytkownika (typy obiektowe SQL99)
- Standard obejmujący wiele części:
 - podobnie jak SQL, ale części SQL/MM są mniej ściśle powiązane ze sobą
 - części poświęcone szeroko pojmowanym multimediom
 - dane przestrzenne i tekstowe uznawane za multimedialne
 - części poświęcone specjalistycznym zastosowaniom



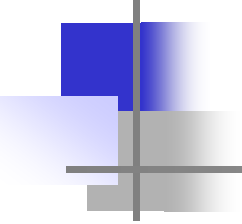
Geneza standardu SQL/MM

- Wnioski z prac nad rozszerzeniami SQL dla danych tekstowych, obrazów, danych przestrzennych, itp.
 - Te same operatory – inne znaczenie w zależności od kontekstu np. CONTAINS
 - Przewidywane trudności w implementacji wszystkich rozszerzeń
- Dostępność obiektowych typów danych od SQL99
 - Zamiast rozszerzeń SQL – biblioteki typów obiektowych dla poszczególnych typów złożonych danych -> SQL/MM



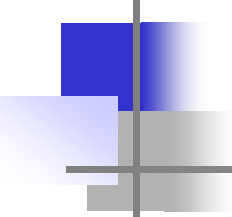
Części standardu SQL/MM (1/3)

- **Part 1: Framework**
 - informacje o zakresie standardu
 - koncepcje wspólne dla części poświęconych poszczególnym typom danych
 - opis sposobu korzystania z mechanizmu obiektowych typów danych przez pozostałe części standardu
- **Part 2: Full-Text** (tekstowe bazy danych)
 - typy danych dla dokumentów tekstowych oraz wzorców wyszukiwania
 - procedury i algorytmy dopasowywania dokumentów tekstowych do wzorca
 - konwersje do/z typów znakowych



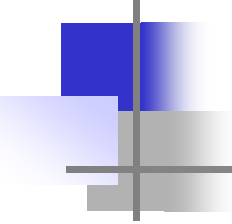
Części standardu SQL/MM (2/3)

- Part 3: **Spatial** (przestrzenne bazy danych)
 - typy danych dla danych przestrzennych – punkty, krzywe, powierzchnie i ich kolekcje
 - procedury i algorytmy tworzenia, porównywania, porządkowania i obliczania własności geometrycznych obiektów przestrzennych
 - konwersje do/z typów znakowych i binarnych
- Part 5: **Still Image** (obrazy)
 - typy danych dla obrazów i ich właściwości
 - procedury i algorytmy do tworzenia, modyfikacji i przeszukiwania obrazów statycznych
 - konwersje do/z typów binarnych



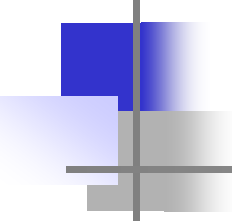
Części standardu SQL/MM (3/3)

- **Part 6: Data Mining** (eksploracja danych)
 - biblioteka typów i ich metod do eksploracji danych (odkrywania wcześniej nieznananych, interesujących informacji z dużych zbiorów danych)
 - ma umożliwiać wymianę modeli eksploracji danych między systemami
- **Part 7: History** (dane historyczne)
 - Obsługa historii zmian dla tabel bazy danych
 - Składowanie, zarządzanie i udostępnianie danych historycznych
 - Historia obejmuje operacje INSERT, UPDATE i DELETE na tabeli
 - Proces standaryzacji jeszcze nie zakończony (stan na listopad 2009)



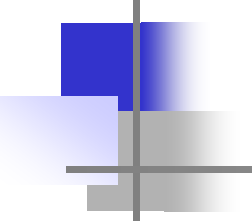
Rozwój standardu SQL/MM

- Zarzucono prace nad Part 4: General Purpose Facilities dotyczącą ogólnych operacji matematycznych
- Istniejące specyfikacje są ciągle rozwijane
 - aktualnie (11.2009) obowiązują drugie lub trzecie edycje
 - trwają prace nad czwartą edycją SQL/MM Spatial
- Rozpoczęto prace nad Part 8: Metadata Registries (MDR)
 - SQL-owy interfejs do dowolnych rejestrów metadanych
 - standaryzacja dostępu aplikacji do MDR
- Czy pojawią się części dotyczące audio i wideo?
 - założenie przyjęte podczas prac nad SQL/MM Still Image: nowe specyfikacje dla multimediiów w zależności od "odzewu" na SQL/MM Still Image
 - mimo upływu 10 lat ciągle nawet nie rozpoczęto prac



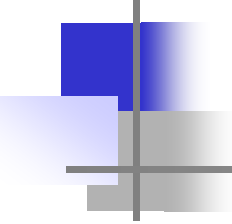
SQL/MM Full Text

- Dotyczy danych tekstowych różniących się od tradycyjnych danych znakowych:
 - Rozmiarem
 - Strukturą (zdania, akapity)
 - Typami operacji
- SQL/MM przewiduje:
 - Wyszukiwanie tekstów zawierających dane słowa i frazy
 - z możliwością uwzględnienia odmiany
 - z możliwością uwzględnienia wyrazów bliskoznacznych
 - Wyszukiwanie tekstów zawierających słowa o zbliżonym brzmieniu
- SQL/MM zakłada obsługę wielu języków
 - Lepsze wsparcie języków zachodnich niż np. japońskiego



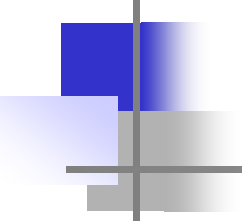
Typy danych SQL/MM Full Text

- Podstawowe znaczenie ma typ danych **FullText**
 - Reprezentuje dokumenty tekstowe
- Metody typu *FullText*:
 - **Contains** – testuje (0/1) czy tekst pasuje do wzorca
 - **Score** – zwraca wartość miary zgodności dokumentu ze wzorcem
- Wzorce reprezentowane są jako
 - Wystąpienia typu **FT_Pattern**
 - Łańcuchy znaków (CHARACTER VARYING)



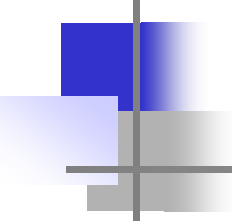
Klasy wzorców (1/3)

- Wystąpienie konkretnego słowa frazy
 - `tresc.Contains(' "standard" ')`
 - `tresc.Contains(' "standard języka" ')`
- Wzorce ze znakami `_` i `%`
 - `tresc.Contains(' "sport_" ')`
 - `tresc.Contains(' "standard%" ')`
- Wystąpienie co najmniej jednego słowa/frazy z listy
 - `tresc.Contains(' "standard", "język%" ')`
- Wystąpienie formy danego słowa/frazy wg reguł odmiany danego języka (domyślny jest angielski)
 - `tresc.Contains('STEMMED FORM OF POLISH "standard języka" ')`



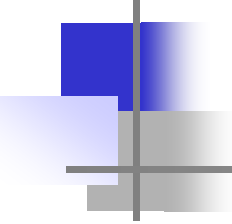
Klasy wzorców (2/3)

- Wystąpienie słów/fraz w danym kontekście
 - `tresc.Contains(' "standard" NEAR ("SQL", "MPEG") WITHIN 2 PARAGRAPHS ')`
 - `tresc.Contains(' "standard" IN SAME SENTENCE AS "SQL" ')`
- Wzorce złożone za pomocą operatorów logicznych *AND, OR* i *NOT*
 - `tresc.Contains(' "SQL/MM" & ("Text" | "Spatial") & NOT "Image" ')`
- Wzorce odwołujące się do tematyki tekstu
 - `tresc.Contains('IS ABOUT "standard języka zapytań" ')`



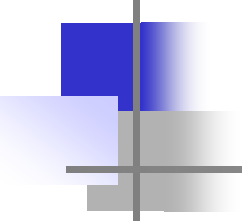
Klasy wzorców (3/3)

- Wzorce odwołujące się do brzmienia w danym języku (domyślny jest angielski)
 - `tresc.Contains(' SOUNDS LIKE "sequel" ')`
- Dopasowanie przybliżone (odporne na „literówki”)
 - `tresc.Contains(' FUZZY FORM OF "teacher" ')`
- Wzorce dopuszczające synonimy
 - `tresc.Contains(THESAURUS "computer science" EXPAND SYNONYM TERM OF "list" ' )`
 - Warianty:
 - EXPAND SYNONYM TERM OF
 - EXPAND BROADER TERM OF
 - EXPAND NARROWER TERM OF
 - EXPAND TOP TERM OF
 - EXPAND PREFERRED TERM OF
 - EXPAND RELATED TERM OF



Metoda Score

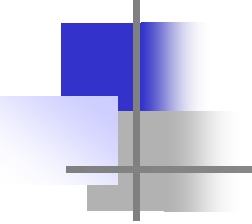
- Przyjmuje takie same parametry jak *Contains*
- Służy np. do sporządzania rankingu dokumentów wg dopasowania do zadanego wzorca
- Zwraca liczbę zmiennoprzecinkową
 - Większa wartość oznacza lepsze dopasowanie



SQL/MM Full Text – Przykład

```
CREATE TABLE information  
(  
    docno INTEGER,  
    document FULLTEXT  
)
```

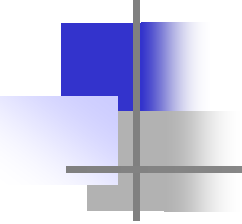
```
SELECT docno  
FROM information  
WHERE document.CONTAINS('STEMMED FORM OF "standard"  
    IN SAME PARAGRAPH AS SOUNDS LIKE "sequel"') = 1
```



Funkcje SQL SQL/MM Full Text

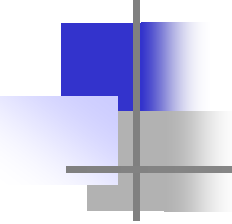
- Dla metod *Contains* i *Score* typu *FullText*, standard przewiduje istnienie ich odpowiedników w postaci funkcji SQL o tych samych nazwach
 - Pierwszym argumentem jest dokument
 - Drugim argumentem jest wzorzec

```
SELECT docno
FROM information
WHERE CONTAINS(document, 'STEMMED FORM OF "standard"
IN SAME PARAGRAPH AS SOUNDS LIKE "sequel"') = 1
```



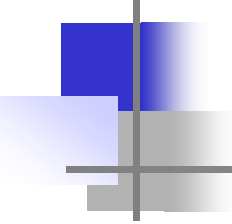
SQL/MM Spatial

- Dotyczy danych przestrzennych
 - Geometria
 - Położenie
 - Topologia
- Przede wszystkim zastosowania geograficzne, przetwarzanie informacji o obiektach na powierzchni Ziemi
- Rozwijany równolegle z 2 innymi standardami dla danych przestrzennych opracowywanymi przez:
 - ISO Technical Committee TC 211
 - Open GIS Consortium



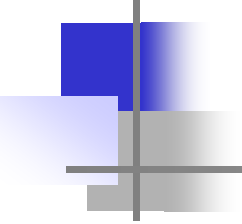
Możliwości i rozwój SQL/MM Spatial (1/2)

- Do drugiej edycji włącznie standard wspierał dane:
 - 0-wymiarowe (punkty)
 - 1-wymiarowe (linie)
 - 2-wymiarowe (figury płaskie)
- Wsparcie dla testów przecinania, pokrywania, itp.
- Wsparcie dla różnych przestrzennych układów odniesienia (ang. spatial reference system)
 - przede wszystkim opisujących geografię naszej planety i jej regionów (krzywizna Ziemi)
 - wybór układu odniesienia istotny dla wyznaczania odległości, powierzchni i kierunków dla dużych obszarów



Możliwości i rozwój SQL/MM Spatial (2/2)

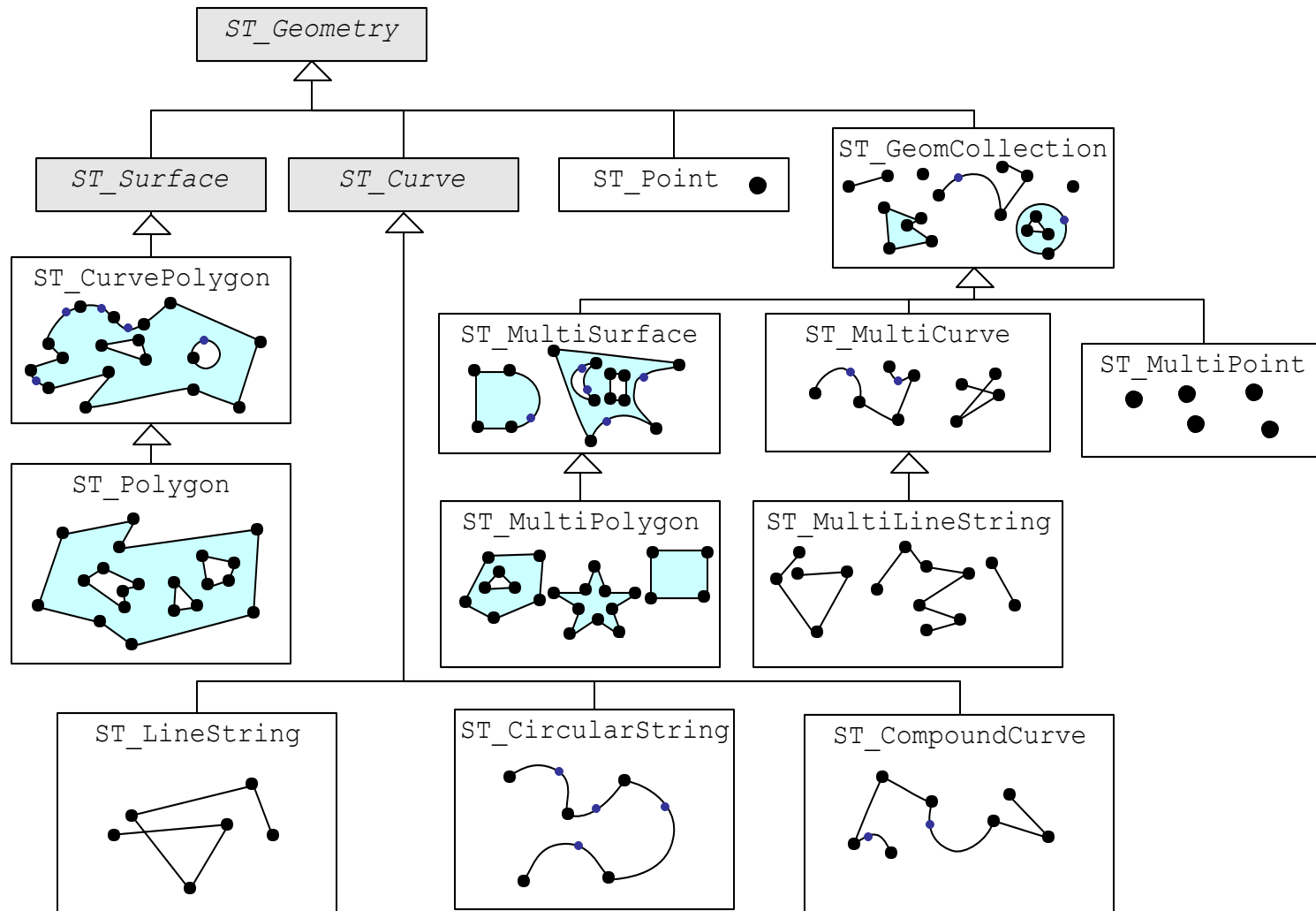
- W trzeciej edycji standardu możliwe jest definiowanie geometrii w przestrzeni współrzędnych 3- i 4-wymiarowej
 - same geometrie nadal będą posiadać co najwyżej 2 wymiary
 - każdy punkt definiujący taką geometrię (np. początkowy i końcowy punkt linii prostej) może posiadać cztery współrzędne
 - trzecia współrzędna z interpretowana jest na ogół jako wysokość
 - czwarta współrzędna m to dowolna inna miara
 - ma stanowić podstawę dla zastosowań takich jak modelowanie sieci telekomunikacyjnych, dróg, gazociągów, itp.
- Trwają prace nad czwartą edycją standardu, która ma rozszerzyć jego możliwości o wsparcie dla geometrii 3D

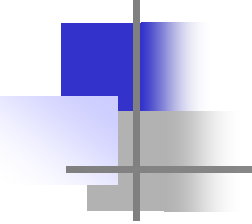


Typy danych SQL/MM Spatial

- Typy opisujące kształty geometryczne:
abstrakcyjny **ST_Geometry** i jego podtypy:
 - Abstrakcyjne: ST_Geometry, ST_Curve, ST_Surface
 - Nieabstrakcyjne np. ST_Point, ST_LineString, ...
- Typ **ST_SpatialRefSystem** do reprezentacji przestrzennych układów odniesienia
- Typy **ST_Angle** i **ST_Direction** reprezentujące odpowiednio kąty i kierunki

SQL/MM Spatial - Hierarchia typów danych dla geometrii



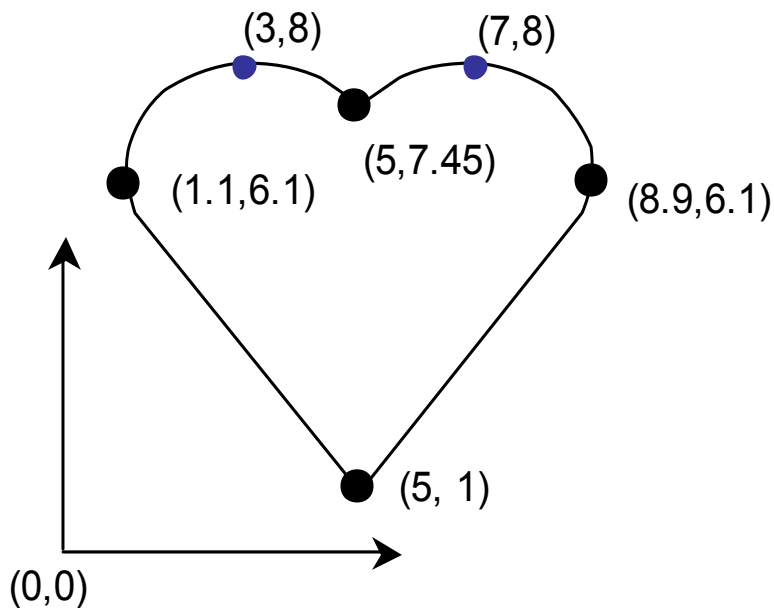


Metody typów SQL/MM Spatial

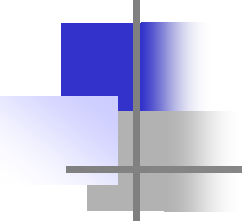
- Metody do konwersji z/do zewnętrznych formatów danych (tekstowych, binarnych i formatu GML – Geography Markup Language)
 - np. *ST_MPointFromGML*, *ST_LineFromGML*, *ST_AsGML*
- Metody do odczytu właściwości i miar dla obiektów geometrycznych
 - np. *ST_Boundary*, *ST_Length*, *ST_Area*
- Metody do porównywania obiektów geometrycznych
 - *ST_Equals*, *ST_Disjoint*, *ST_Intersects*, *ST_Crosses*, *ST_Overlaps*, *ST_Touches*, *ST_Within*, *ST_Contains* i *ST_Distance*
- Metody do tworzenia nowych obiektów geometrycznych
 - *ST_Difference*, *ST_Intersection* i *ST_Union*

SQL/MM Spatial – Przykład 1

```
CREATE TABLE figury (  
  id_figury    INTEGER,  
  nazwa        VARCHAR(30),  
  kolor        VARCHAR(20),  
  ksztalt      ST_GEOMETRY )
```



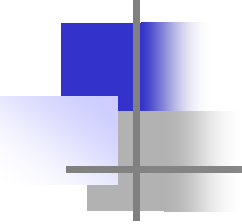
```
INSERT INTO figury VALUES  
  ( 1, 'serce', 'czerwony',  
    NEW ST_CurvePolygon(  
      NEW ST_CompoundCurve ( ARRAY [  
        NEW ST_CircularString (ARRAY [  
          NEW ST_Point (8.9,6.1),  
          NEW ST_Point (7,8),  
          NEW ST_Point (5,7.45),  
          NEW ST_Point (3,8),  
          NEW ST_Point (1.1,6.1)  
        ]), NEW ST_LineString (ARRAY [  
          NEW ST_Point (1.1,6.1),  
          NEW ST_Point (5,1),  
          NEW ST_Point (8.9,6.1),  
        ])  
      ])  
    )  
  )
```



SQL/MM Spatial – Przykład 2

```
CREATE TABLE CITY (  
    NAME          VARCHAR(30) ,  
    POPULATION    INTEGER,  
    CITY_PARKS    VARCHAR(30) ARRAY[10] ,  
    LOCATION      ST_GEOMETRY )
```

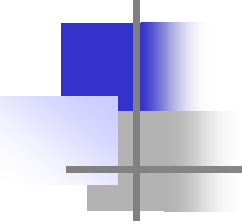
```
SELECT location.area  
FROM CITY  
WHERE name = 'San Francisco'
```



SQL/MM Spatial – Przykład 3

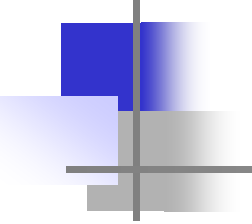
```
CREATE TABLE kraje (  
    nazwa_kraju VARCHAR(30),  
    ...,  
    lokalizacja ST_GEOMETRY )
```

```
SELECT nazwa_kraju FROM kraje  
WHERE lokalizacja.ST_Touches(SELECT lokalizacja  
                                FROM kraje  
                                WHERE nazwa_kraju =  
                                    'POLSKA'  
                                ) ;
```



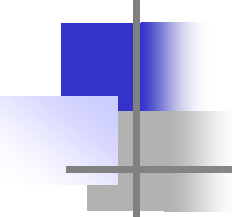
SQL/MM Still Image

- Dotyczy składowania, przetwarzania i wyszukiwania obrazów (np. fotografii)
- Implementacje standardu mogą wspierać wiele różnych formatów: JPEG, GIF, TIFF, ...
 - Zakłada się, że obraz jest dwuwymiarową tablicą pikseli (obraz bitmapowy)
- Przetwarzanie obejmuje m.in. skalowanie i inne transformacje, wycinanie
- Wsparcie dla zapytań dotyczących zawartości:
 - „Średni” kolor
 - Histogramy kolorów (udział kolorów w obrazie)
 - Lokalizacja kolorów
 - Miara podobieństwa obrazów
- Specyfikacja zaimplementowana w Oracle10g !!!



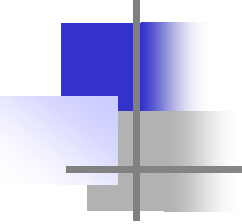
Typy SQL/MM Still Image

- **SI_StillImage** – reprezentuje obraz
- **SI_Color** – reprezentuje kolor
- **SI_AverageColor** – reprezentuje średni kolor obrazu
- **SI_ColorHistogram** – reprezentuje histogram kolorów
- **SI_PositionalColor** – reprezentuje lokalizację kolorów obrazu
- **SI_Texture** – reprezentuje teksturę obrazu
- **SI_FeatureList** – reprezentuje listę właściwości wizualnych obrazu



Typ SI_StillImage: Atrybuty

- **Standardowe:**
 - SI_content (typu BLOB), SI_contentLength, SI_reference (typu DATALINK – do reprezentacji zewnętrznych źródeł poprzez URL)
 - SI_format, SI_height, SI_width
- **Rozszerzenia/niezgodności w Oracle:**
 - mimeType_ora, contentFormat_ora, compressionFormat_ora + atrybuty do zapamiętania i automatycznej pielęgnacji właściwości wizualnych (retainFeatures_SI, averageColorSpec_ora, colorsList_ora, frequenciesList_ora, colorPositions_ora, textureEncoding_ora)
 - zmiana nazw atrybutów standardowych (np. SI_content -> content_SI)
 - content_SI typu ORDSOURCE
 - Obejmuje BLOB dla składowania w bazie i linki do lokalizacji zewnętrznych
 - Oracle10g R1 zgodny z pierwszą wersją SQL/MM SI, typ DATALINK dostępny od drugiej wersji standardu



Typ SI_StillImage: Metody

- **Konstruktory:**
 - SI_StillImage(BLOB), SI_StillImage(BLOB, ...)
- **Metody do odczytu atrybutów:**
 - np. SI_Height, SI_Width, SI_Format, ...
- Związane z **przetwarzaniem obrazów:**
 - SI_SetContent, SI_ChangeFormat
 - SI_Thumbnail, SI_Scale, SI_Resize
 - SI_Rotate (jest w standardzie, brak w Oracle 10g/11g)
- Związane z **pielęgnacją właściwości wizualnych** (Oracle 10g/11g):
 - SI_InitFeatures
 - Wyznacza i zapamiętuje wartości własności wizualnych
 - Wartości te będą automatycznie pielęgnowane
 - SI_ClearFeatures

Metody SI_StillImage - Przykład



metoda **SI_Thumbnail**



metoda **SI_Scale**

parametr width = nowy SI_width



metoda **SI_Resize**

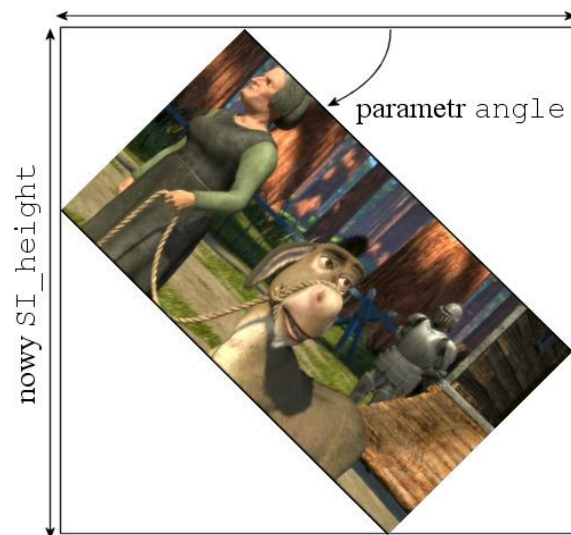
parametr width = nowy SI_width

parametr height =
nowy SI_height

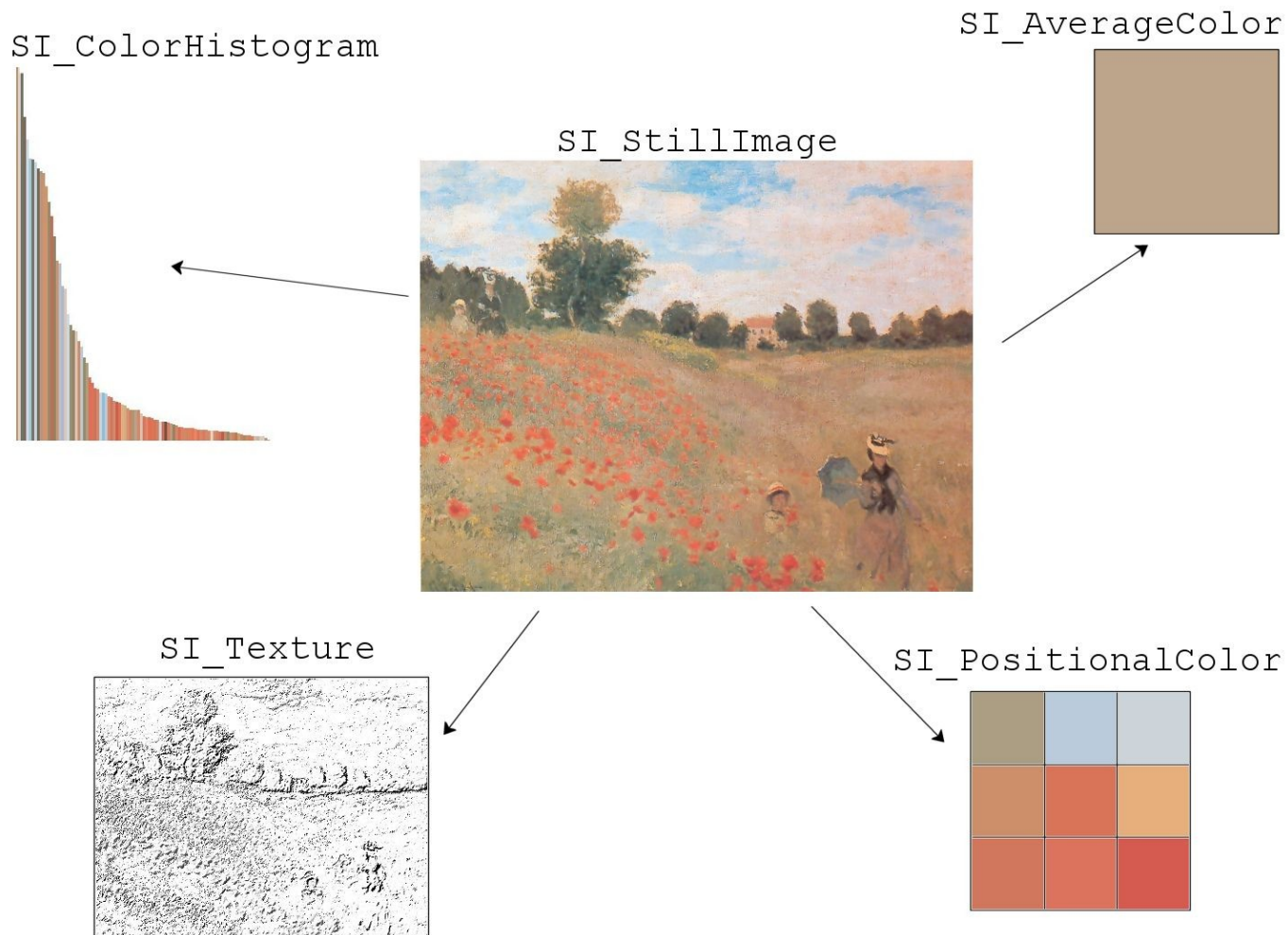


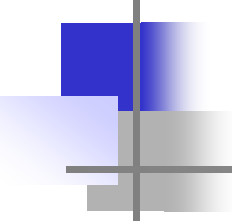
metoda **SI_Rotate**

nowy SI_width



SQL/MM Still Image: Typy danych reprezentujące własności sygnałowe obrazu





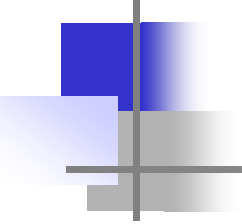
Typ danych **SI_Color**

- Typ pomocniczy, wykorzystywany przez `SI_AverageColor`, `SI_ColorHistogram` i `SI_PositionalColor`
- Atrybuty:
 - `redValue`, `greenValue`, `blueValue` (wartości od 0 do 255)
- Metody:
 - brak specyficznych konstruktorów – dostępny jedynie domyślny systemowy (nieweryfikujący poprawności danych!)
 - inicjalizacja koloru: `SI_RGBColor (redValue, greenValue, blueValue)`

Typ danych SI_AverageColor

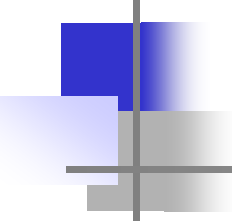
- Atrybuty:
 - SI_AverageColorSpec (typu SI_Color)
- Metody:
 - **konstruktor**: SI_AverageColor(SI_StillImage), SI_AverageColor(SI_Color)
 - SI_Score(SI_StillImage) – wyznaczająca "odległość" obrazu od wzorcowego średniego koloru (wartość od 0.0 do 100.0)
- Algorytm wyznaczania średniego koloru dla obrazu:
 - obraz jest dzielony na n próbek
 - komponenty koloru R, G i B z poszczególnych próbek są niezależnie od siebie sumowane i dzielone przez liczbę próbek

$$(\bar{R}, \bar{G}, \bar{B}) = \left(\frac{\sum_{i=1}^{SI_width} \sum_{j=1}^{SI_height} R(i, j)}{SI_width * SI_height}, \frac{\sum_{i=1}^{SI_width} \sum_{j=1}^{SI_height} G(i, j)}{SI_width * SI_height}, \frac{\sum_{i=1}^{SI_width} \sum_{j=1}^{SI_height} B(i, j)}{SI_width * SI_height} \right)$$



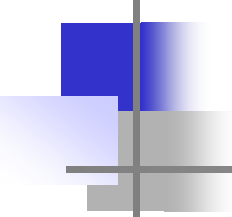
Typ danych SI_ColorHistogram

- Atrybuty:
 - SI_ColorsList (typu SI_Color ARRAY),
SI_FrequenciesList (typu DOUBLE ARRAY)
 - Tablice mają taką samą liczbę elementów (maksymalny rozmiar SI_MaxHistogramLength zależny od implementacji)
 - Wartości częstotliwości ograniczone zakresem od 0 do 100
 - W Oracle10g zamiast ARRAY - "pomocnicze" typy colorsList i colorFrequenciesList, zdefiniowane jako kolekcje VARRAY
- Metody:
 - **konstruktor**: SI_ColorHistogram(SI_StillImage),
SI_ColorHistogram(SI_Color, DOUBLE PRECISION),
SI_ColorHistogram(ARRAY, ARRAY)
 - SI_Append(SI_Color, DOUBLE PRECISION) – dodająca kolor i jego częstość do histogramu
 - SI_Score(SI_StillImage) – wyznaczająca "odległość" obrazu od wzorcowego histogramu kolorów (wartość od 0.0 do 100.0)



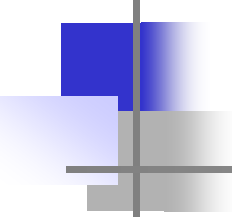
SI_ColorHistogram – algorytm generacji histogramu

- Realizowana poprzez wywołanie konstruktora `SI_ColorHistogram(SI_StillImage)`
- Zasada działania algorytmu:
 - przestrzeń kolorów dzielona jest na pewną liczbę zakresów
 - Zależną od implementacji
 - Ograniczoną przez `SI_MaxHistogramLength`
 - każdy obszar przestrzeni kolorów obejmuje pewien zbiór kolorów i jest reprezentowany przez jeden kolor C_i
 - dla każdego obszaru kolorów określana jest jego częstotliwość F_i w ramach obrazu poprzez iterację po wszystkich pikselach
 - każdy piksel zwiększa wartość F_i o 1 dla tego zakresu i do którego należy jego kolor
 - wartości F_i są normalizowane, aby zawierały się w zakresie od 0 do 100



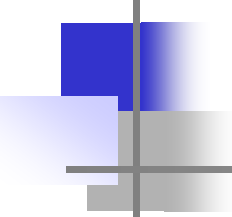
Typ danych SI_PositionalColor

- Atrybuty:
 - SI_ColorPositions (typu SI_Color ARRAY)
 - Maksymalny rozmiar tablicy: SI_NumberSections
 - W Oracle10g zamiast ARRAY - "pomocniczy" typ colorPositions
- Metody:
 - **konstruktor**: SI_PositionalColor(SI_StillImage)
 - SI_Score(SI_StillImage) – wyznaczająca „odległość” obrazu od wzorcowej lokalizacji kolorów (wartość od 0.0 do 100.0)
- Algorytm wyznaczania lokalizacji kolorów dla obrazu:
 - obraz jest dzielony na „siatkę” $m \times n$ prostokątów
 - dla każdego z prostokątów wyznaczany jest dominujący kolor



Typ danych SI_Texture

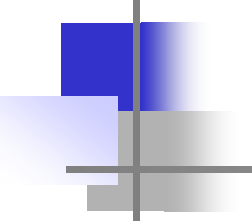
- Atrybuty:
 - SI_TextureEncoding
 - W Oracle10g typu textureEncoding – kolekcja VARRAY zawierająca współczynniki opisujące takie cechy tekstury jak: "chropowatość", kontrast, kierunek ułożenia wzoru
- Metody:
 - **konstruktor**: SI_Texture(SI_StillImage)
 - SI_Score(SI_StillImage) – wyznaczająca "odległość" obrazu od wzorcowej tekstury (wartość od 0.0 do 100.0)
- Sposób opisu tekstury, algorytm wyznaczania współczynników i określania podobieństwa – zależne od implementacji



Typ danych SI_FeatureList

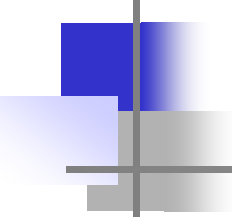
- Atrybuty:
 - **właściwości i ich wagi:** SI_AvgClrFtr (typu SI_AverageColor), SI_AvgClrFtrWght, SI_ClrHstgrFtr (typu SI_ColorHistogram), SI_ClrHstgrFtrWght, SI_PstnlClrFtr (typu SI_PositionalColor), SI_PstnlClrFtrWght, SI_TextureFtr (typu SI_Texture), SI_TextureFtrWght
- Metody:
 - **konstruktor:** SI_FeatureList(SI_AverageColor, DOUBLE PRECISION, SI_ColorHistogram, DOUBLE PRECISION, SI_PositionalColor, DOUBLE PRECISION, SI_Texture, DOUBLE PRECISION)
 - metody do **ustawiania poszczególnych właściwości i ich wag:** SI_SetFeature(SI_AverageColor, DOUBLE PRECISION), SI_SetFeature(SI_ColorHistogram, DOUBLE PRECISION), SI_SetFeature(SI_PositionalColor, DOUBLE PRECISION), SI_SetFeature(SI_Texture, DOUBLE PRECISION),
 - metody do **odczytu atrybutów**, np. SI_AvgClrFtr, SI_AvgClrFtrWght, ...
 - SI_Score(SI_StillImage) – wyznaczająca "odległość" obrazu od wzorcowej listy właściwości (wartość od 0.0 do 100.0 – średnia ważona odległości dla poszczególnych właściwości)

$$\frac{\sum_{i=1}^N F_i \cdot SI_Score(image) \cdot W_i}{\sum_{i=1}^N W_i}$$



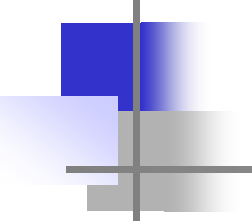
SQL/MM Still Image – Funkcje SQL

- Dla każdego konstruktora i metody typów SQL/MM Still Image istnieje równoważna funkcja lub procedura SQL
- Charakterystyka funkcji SQL dla Still Image:
 - uruchamiane z prawami wywołującego
 - posiadające publiczne synonimy
 - dostępne dla wszystkich użytkowników bazy danych
 - W Oracle10g utworzone w schemacie ORDSYS (podobnie jak wszystkie typy danych interMedia)
- Przykłady:
 - SI_MkStillImage1(BLOB) dla SI_StillImage.SI_StillImage(BLOB)
 - SI_MkRGBClr dla SI_Color.SI_RGBColor()
 - SI_GetHeight(StillImage) dla SI_StillImage.SI_Height()



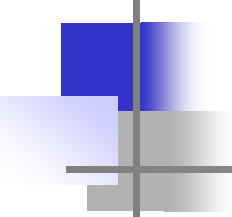
SQL/MM Still Image – Perspektywy informacyjne (1/2)

- Standard SQL/MM udostępnia szereg perspektyw informacyjnych, które mogą być wykorzystywane przez użytkownika aby ocenić np. wspierane przez standard formaty graficzne i dostępne dla nich funkcje
- Perspektywy te są również wykorzystywane przez metody zdefiniowanych typów UDT w celu ustalenia, czy dana operacja może być wykonana dla danego formatu i ewentualnego zwrócenia wyjątku



SQL/MM Still Image – Perspektywy informacyjne (2/2)

- **SI_FEATURES** – zawiera nazwy dostępnych cech wizualnych w danej implementacji (np. 'SI_AverageColor')
- **SI_IMAGE_FORMATS** – nazwa formatu graficznego i informacja o wspieraniu operacji skalowania i obrotów
- **SI_IMAGE_FORMAT_CONVERSIONS** – określa konwersje z jakiego formatu na jaki są dostępne w danej implementacji
- **SI_IMAGE_FORMAT_FEATURES** – określa jakie właściwości wizualne mogą być automatycznie wyprowadzone dla poszczególnych formatów
- **SI_THUMBNAİL_FORMATS** – określa, dla których formatów można tworzyć miniatury
- **SI_VALUES** – wartości stałych wykorzystywanych w implementacji, np. **SI_MaxHistogramLength**



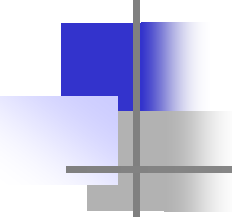
SQL/MM Still Image w Oracle10g – Przykład (1/4)

- Utworzenie tabeli do składowania obrazów

```
CREATE TABLE si_wzory  
(nazwa VARCHAR2(40),  
obraz ORDSYS.SI_StillImage);
```

- Wstawienie obrazu do tabeli za pośrednictwem BFILE

```
DECLARE  
  lobd blob;  
  fils BFILE := BFILENAME('TKANINY','kratka.gif');  
BEGIN  
  DBMS_LOB.CREATETEMPORARY(lobd, TRUE);  
  DBMS_LOB.fileopen(fils, DBMS_LOB.file_readonly);  
  DBMS_LOB.LOADFROMFILE(lobd, fils, DBMS_LOB.GETLENGTH(fils));  
  DBMS_LOB.FILECLOSE(fils);  
  INSERT INTO SI_WZORY (nazwa, obraz)  
    VALUES('kratka', new ORDSYS.SI_StillImage(lobd));  
  DBMS_LOB.FREETEMPORARY(lobd);  
  COMMIT;  
END;  
/
```

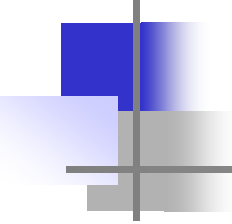


SQL/MM Still Image w Oracle10g – Przykład (2/4)

- Wyznaczenie sygnatury dla obrazu

```
DECLARE
  tempimage SI_StillImage;
BEGIN
  SELECT obraz INTO tempimage FROM si_wzory
    WHERE nazwa = 'kratka' FOR UPDATE;
  tempimage.SI_InitFeatures;
  UPDATE si_wzory
    SET obraz = tempimage where nazwa = 'kratka';
  COMMIT;
END;
/
```

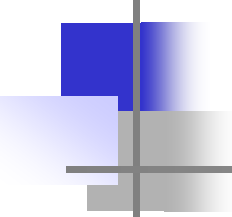
- UWAGA: w rzeczywistości należałoby przejść kursorem po wszystkich wierszach tabeli i modyfikować bieżący wiersz w kursorze (WHERE CURRENT OF)



SQL/MM Still Image w Oracle10g – Przykład (3/4)

- Wyszukanie obrazów podobnych do danego w sensie tekstury

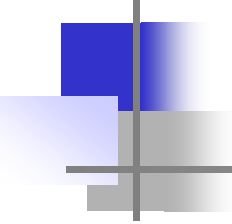
```
SELECT w.nazwa FROM si_wzory w
WHERE SI_FindTexture(
    (SELECT v.obraz FROM si_wzory v
    WHERE v.nazwa
    = 'kratka')) .SI_Score(w.obraz) < 30;
```



SQL/MM Still Image w Oracle10g – Przykład (4/4)

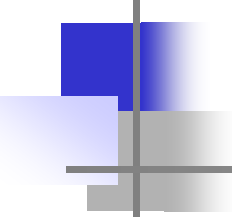
- Wyszukanie obrazów podobnych do danego w sensie koloru i tekstury

```
DECLARE
    tempimage          SI_StillImage;
    tempAvgColor       SI_AverageColor;
    tempTexture        SI_Texture;
    myFeatureList       SI_FeatureList;
    score              DOUBLE PRECISION;
BEGIN
    SELECT obraz INTO tempimage FROM si_wzory
    WHERE nazwa = 'kratka';
    tempAvgColor := NEW SI_AverageColor(tempimage);
    tempTexture := NEW SI_Texture(tempimage);
    myFeatureList := NEW SI_FeatureList(
        tempAvgColor, 0.7, NULL, NULL, NULL, NULL, tempTexture, 0.3);
    FOR c IN (SELECT nazwa, obraz FROM si_wzory) LOOP
        score := myFeatureList.SI_Score(c.obraz);
        IF score < 10 THEN
            dbms_output.put_line(c.nazwa || ' score: ' || score);
        END IF;
    END LOOP;
END;
/
```



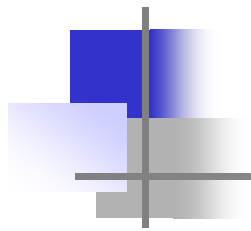
SQL/MM Data Mining

- Przykład pakietu aplikacyjnego (Application Package) w ramach standardu SQL/MM
- Stanowi próbę dostarczenia standardowych interfejsów do algorytmów eksploracyjnych na gruncie obiektowo-relacyjnych baz danych
 - nie określa algorytmów implementujących poszczególne modele
 - nie określa formatu przechowywanych danych dla eksploracji danych
- Obecnie wspiera cztery metody eksploracji danych:
 - odkrywanie reguł asocjacyjnych
 - klasyfikację
 - analizę skupień
 - regresję
- Jeden z wielu standardów dotyczących DM
 - w przyszłości możliwa zgodność z Java Data Mining API

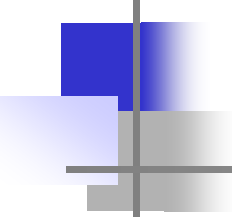


SQL/MM - Podsumowanie

- Komercyjny sukcesie SQL/MM z pewnością niepełny
 - SQL/MM Still Image zaimplementowany w Oracle 10g/11g
 - w DB2 przed wersją 9 była dostępna podobna funkcjonalność, ale inne nazwy typów i składnia (w wersji 9.x brak wsparcia)
 - SQL/MM Spatial zaimplementowany w Oracle 11g i IBM DB2
 - Wsparcie ze strony komercyjnych systemów w zakresie funkcjonalności SQL/MM Full-Text, ale składnia inna niż w standardzie (np. Oracle i IBM DB2)
- Pojawienie się w bliskiej przyszłości części poświęconych innym typom danych multimedialnych (np. wideo) wydaje się wątpliwe
- SQL/MM dla obrazów nie standaryzuje metod opisu zawartości (nie stanowi w tym zakresie konkurencji dla standardu MPEG-7)

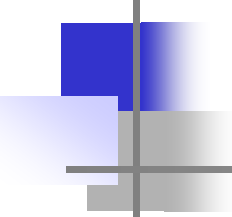


Standard MPEG-7



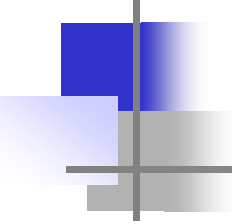
MPEG-7

- Standardy MPEG-1, MPEG-2, MPEG-4 dotyczą **reprezentacji** zawartości obiektów multimedialnych
- MPEG-7 (Multimedia Content Description Interface) normuje **opisy** zawartości multimedialnej
 - Format opisów oparty o XML
 - Opisy dla różnych typów danych multimedialnych
 - Zestaw cech uwzględnianych w opisie może zależeć od przewidywanego zastosowania opisu (możliwe różne opisy tej samej zawartości)
- Adresatami opisu mają być ludzie lub programy komputerowe
- Baza dla aplikacji wymagających efektywnych operacji wyszukiwania, przeglądania i filtracji danych multimedialnych



Założenia standardu MPEG-7

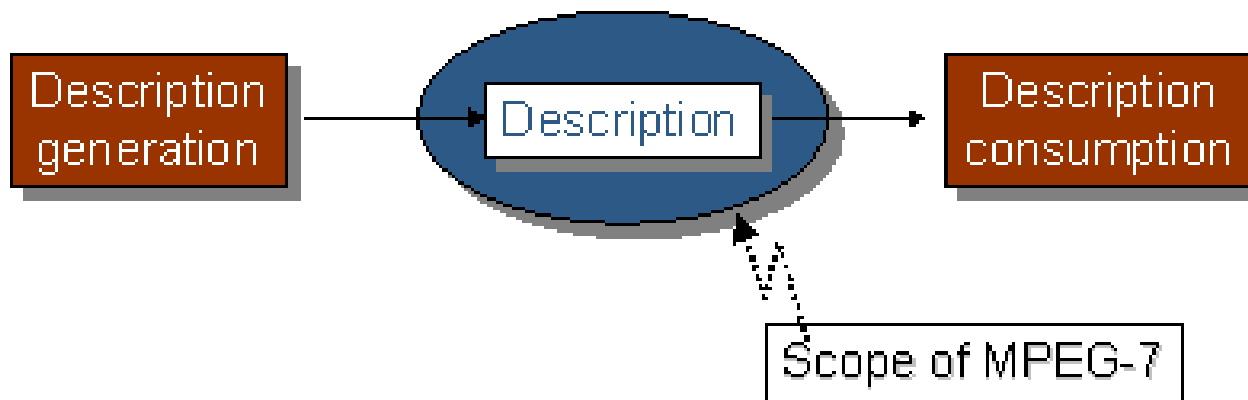
- **Standaryzacja opisu metadanych**
 - Standard nie obejmuje ekstrakcji i wykorzystywania metadanych
- **Niezależność od sposobu kodowania medium**
 - Medium przechowywane w dowolny sposób i w dowolnym miejscu
- **Elastyczność**
 - Standard opisu dla wielu dziedzin; efekt analizy szerokiego spektrum potencjalnych zastosowań
- **Rozszerzalność**
 - Możliwość rozszerzenia składni języka metadanych o elementy przydatne w specyficznych zastosowaniach
- **Oparcie o język XML**
 - Model semistukturalny: elastyczność opisu z zachowaniem pewnej struktury
 - Popularność języka XML
 - XMLSchema jako podstawa dla języka definicji opisów



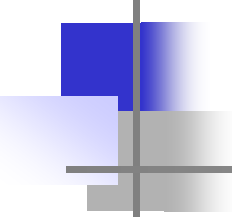
Typy zawartości opisywanych przez MPEG-7

- Obrazy, grafika, modele 3D
- Audio, mowa
- Wideo
- Kompozycje powyższych elementów tworzące multimedialne prezentacje
- Charakterystyka ludzkich twarzy jako wyróżniony, ważny obszar zastosowań

Zakres standardu MPEG-7 (1/2)

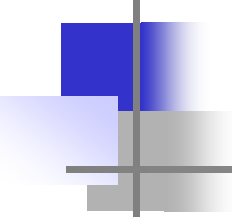


- Standard dotyczy tylko formatu **opisu zawartości**
- Techniki **generacji opisów, ekstrakcji właściwości** oraz **wykorzystywania opisów** (np. przetwarzania zapytań) nie wchodzą w zakres standardu MPEG-7



Zakres standardu MPEG-7 (2/2)

- Przyczyny wyłączenia generacji opisów, ekstrakcji właściwości oraz wykorzystywania opisów z zakresu standardu:
 - Mechanizmy te mają stanowić obszar swobodnej konkurencji, sprzyjającej rozwojowi technologicznemu
 - Standaryzacja algorytmów nie jest konieczna dla możliwości współdziałania różnych systemów
 - Charakter i funkcjonalność wyszukiwarek zwykle zależy od zastosowania, a format MPEG-7 ma z założenia być niezależny od zastosowań

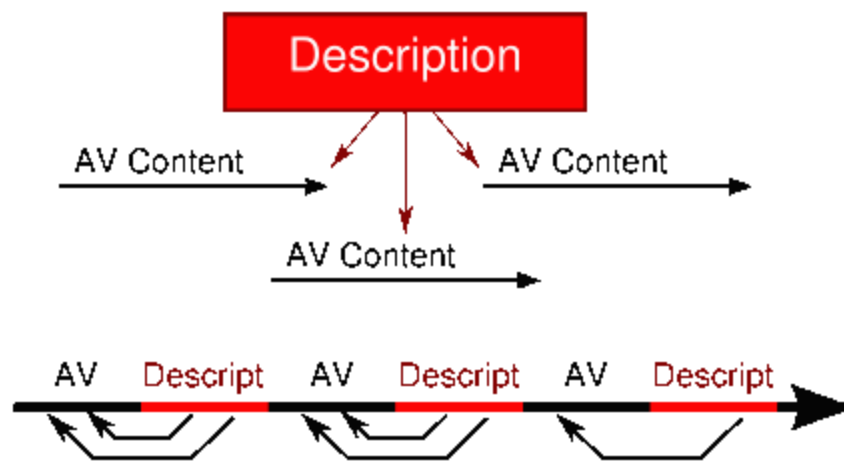


Format opisów MPEG-7

- Reprezentacja tekstowa
 - Opis w postaci dokumentu XML (TeM)
 - DDL (Description Definition Language) – język definicji składni opisu oparty o XMLSchema
 - XMLSchema + rozszerzenia np. typ danych dla tablic wielowymiarowych
- Reprezentacja binarna
 - Binary format for MPEG-7 data (BiM)
 - Kompresja, odporność na błędy, bezpośredni dostęp do składowych opisu
- (!) Opis MPEG-7 może towarzyszyć opisywanej zawartości lub być składowany niezależnie

Niezależność opisu i zawartości

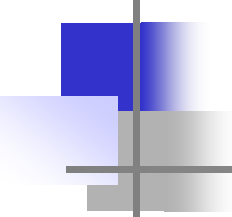
- Wymaganiem architektonicznym MPEG-7 jest rozdział opisu od zawartości audiowizualnej
- Mimo tego musi istnieć powiązanie opisu z zawartością
- Rozwiązanie: opis jest multipleksowany z zawartością



Przykład opisu MPEG-7

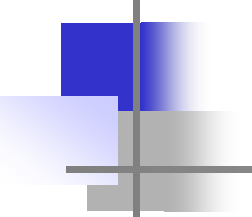
```
<Mpeg7 ...>
  <Description xsi:type="ContentEntityType">
    <MultimediaContent xsi:type="ImageType">
      <Image id="a102">
        <MediaLocator><MediaUri>kuszcza.jpg</MediaUri></MediaLocator>
        <TextAnnotation><FreeTextAnnotation>
          Polska - Kolumbia (1:2). Polski bramkarz Tomasz
          Kuszcza wpuszcza gola na 0:2 po strzale bramkarza Kolumbii
        </FreeTextAnnotation></TextAnnotation>
        <SpatialDecomposition>
          <StillRegion>
            <TextAnnotation><FreeTextAnnotation>
              Polski bramkarz Tomasz Kuszcza w wyskoku
              myśli że piłka przechodzi nad poprzeczką
            </FreeTextAnnotation></TextAnnotation>
            <SpatialLocator>
              <Polygon><Coords dim="2 4"> 45 20 0 -20 25 0 60 0
              </Coords></Polygon>
            </SpatialLocator>
          </StillRegion>
        </SpatialDecomposition>
      </Image>
    </MultimediaContent>
  </Description>
  ...
</Mpeg7>
```





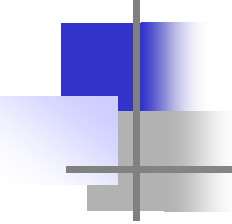
Poziomy abstrakcji opisów MPEG-7

- Opis na wysokim poziomie abstrakcji:
 - Niesie informację semantyczną, np.
"Scena przedstawia czekającego dużego czarnego psa po prawej stronie i uciekającego listonosza po lewej. W tle słychać śmiech sąsiadów."
- Opis na niskim poziomie abstrakcji:
 - Kształt, rozmiar, tekstura, kolor, lokalizacja, trajektoria
 - Właściwości audio np. tempo, barwa, linia melodyczna
- Możliwe poziomy pośrednie:
 - Narzędzia do rozpoznawania twarzy
 - Narzędzia do rozpoznawania mowy



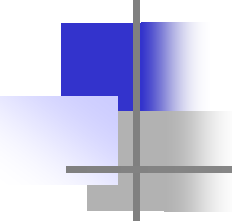
MPEG-7: Informacje uzupełniające opis zawartości

- Forma
 - Sposób kodowania (JPEG, MPEG-2, ...)
 - Rozmiar
- Dostępność materiału
 - Cena
 - Ochrona praw autorskich
- Klasyfikacja
 - Kategorie tematyczne
 - Kategorie wiekowe odbiorców
- Kontekst
 - Czas i miejsce nagrania, wydarzenie (np. igrzyska olimpijskie)



Podstawowe elementy standardu MPEG-7

- Narzędzia opisu (Description Tools):
 - Deskryptory (D) – definiujące składnię i semantykę poszczególnych cech (elementów metadanych)
 - Schematy opisu (DS) – specyfikujące strukturę i semantykę związków między elementami opisu (deskryptorami i schematami opisu)
- Description Definition Language (DDL) – do definiowania składni narzędzi opisu (DS i D) w celu tworzenia nowych DS/D lub rozszerzania albo modyfikacji istniejących
- Narzędzia systemowe wspierające:
 - Binarne kodowanie opisów (dla efektywnego składowania i transmisji)
 - Mechanizmy transmisji (dla postaci tekstowej i binarnej)
 - Synchronizacja opisu z zawartością
 - Ochrona własności intelektualnej opisów

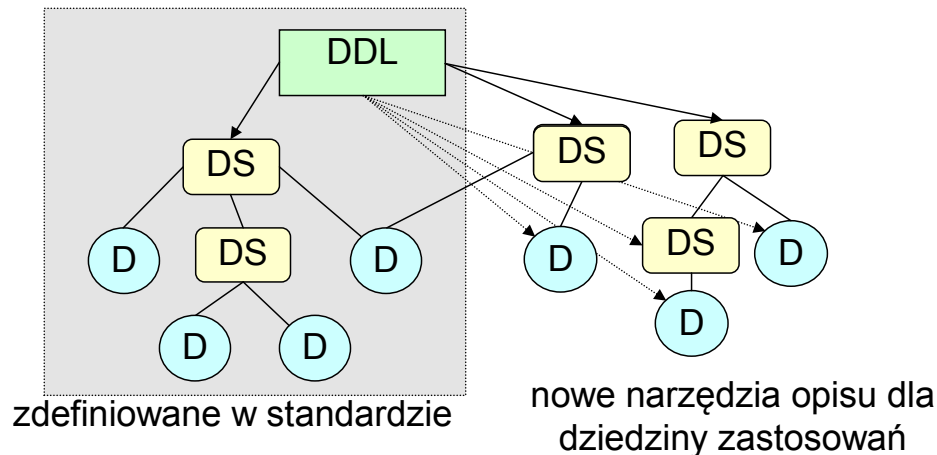


Deskryptory jako reprezentacje cech

- Cecha charakteryzuje dane multimedialne pod jednym kątem
 - Przykłady cech:
 - kolor obrazu statycznego, ruch kamery w sekwencji wideo, itd.
 - Cechy nie mogą być porównywane między sobą bez konkretnej reprezentacji (deskryptora)
- Deskryptor stanowi reprezentację cechy
 - Dla reprezentacji cechy definiuje się jej składnię (sposób zapisu w formacie XML) oraz semantykę
 - Deskryptory są atomowe
 - zawierają jedynie proste typy danych
 - nie odwołują się do innych deskryptorów (D) ani schematów opisu (DS)
 - Przykłady deskryptorów: dla cechy "kolor" deskryptorem może być:
 - histogram kolorów (ScalableColor D)
 - kolor dominujący (DominantColor D)

Rozszerzalność standardu MPEG 7 o nowe deskryptory (D) i schematy opisu (DS)

- W języku DDL definiuje się elementy DS oraz D
 - DS mogą zawierać w sobie elementy D oraz kolejne elementy DS
- Standard MPEG 7 umożliwia tworzenie za pomocą języka DDL własnych deskryptorów (D) oraz schematów opisu (DS)
 - np. dla konkretnej dziedziny zastosowań takiej jak zdjęcia medyczne, nadzór wideo, itd.
- Zdefiniowane nowe schematy opisu mogą korzystać z predefiniowanych w standardzie elementów D i DS



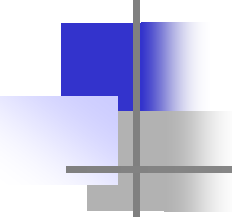
MPEG-7 DDL - Język definicji opisów

- W języku MPEG-7 DDL opisana jest składnia wszystkich predefiniowanych schematów opisu oraz deskryptorów
 - może on również służyć do tworzenia nowych narzędzi opisu
- MPEG-7 DDL jest oparty na języku XMLSchema
 - nadrzędnym elementem każdego dokumentu MPEG 7 powinien być znacznik **<Mpeg7>** opatrzony odpowiednimi atrybutami dotyczącymi przestrzeni nazw:

```
<?xml version="1.0" encoding="iso-8859-2"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xsi:schemaLocation="urn:mpeg:mpeg7:schema:2001 Mpeg7-2001.xsd">
```

```
<! - Zawartość opisu w formacie MPEG 7 - >
```

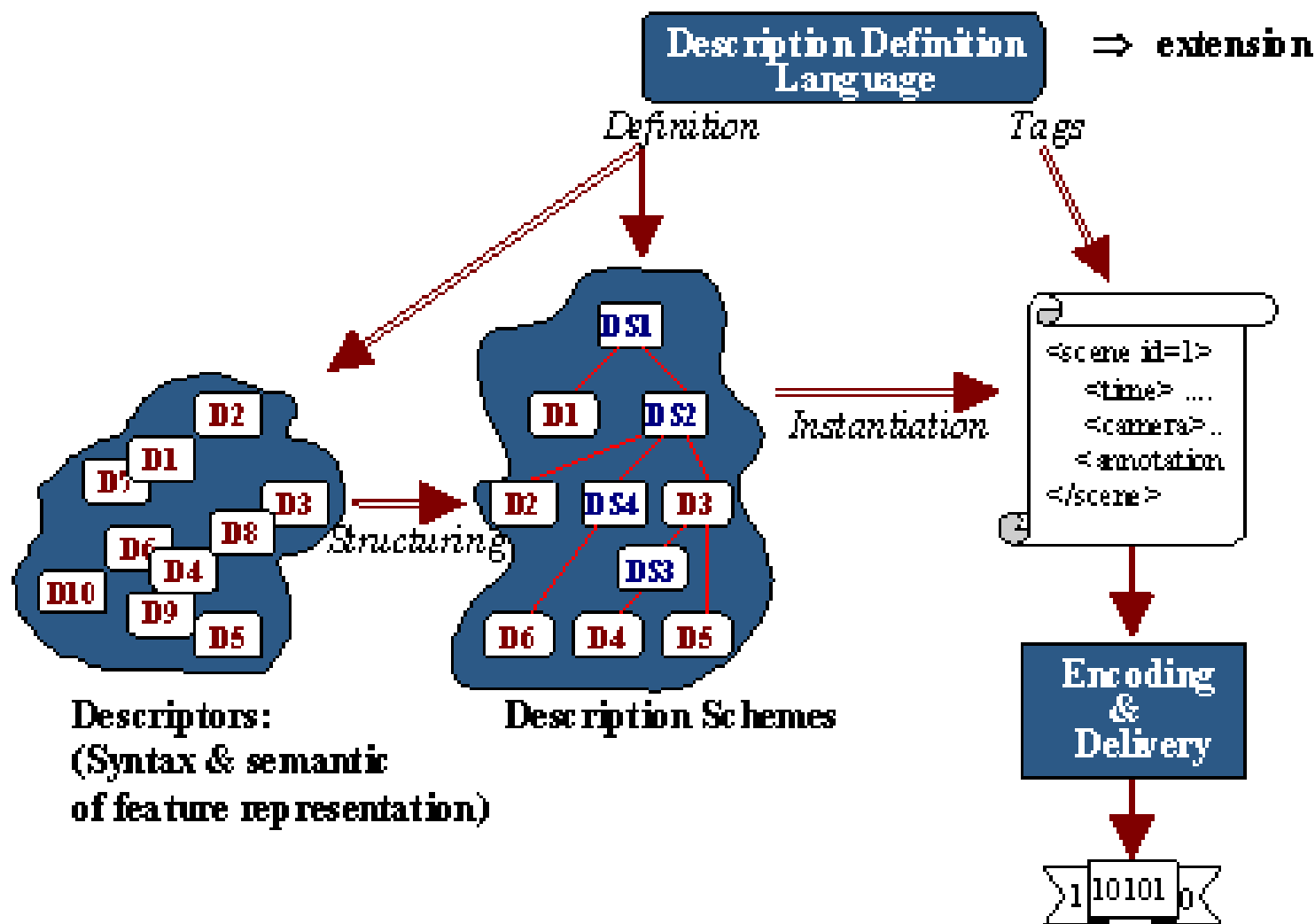
```
</Mpeg7>
```



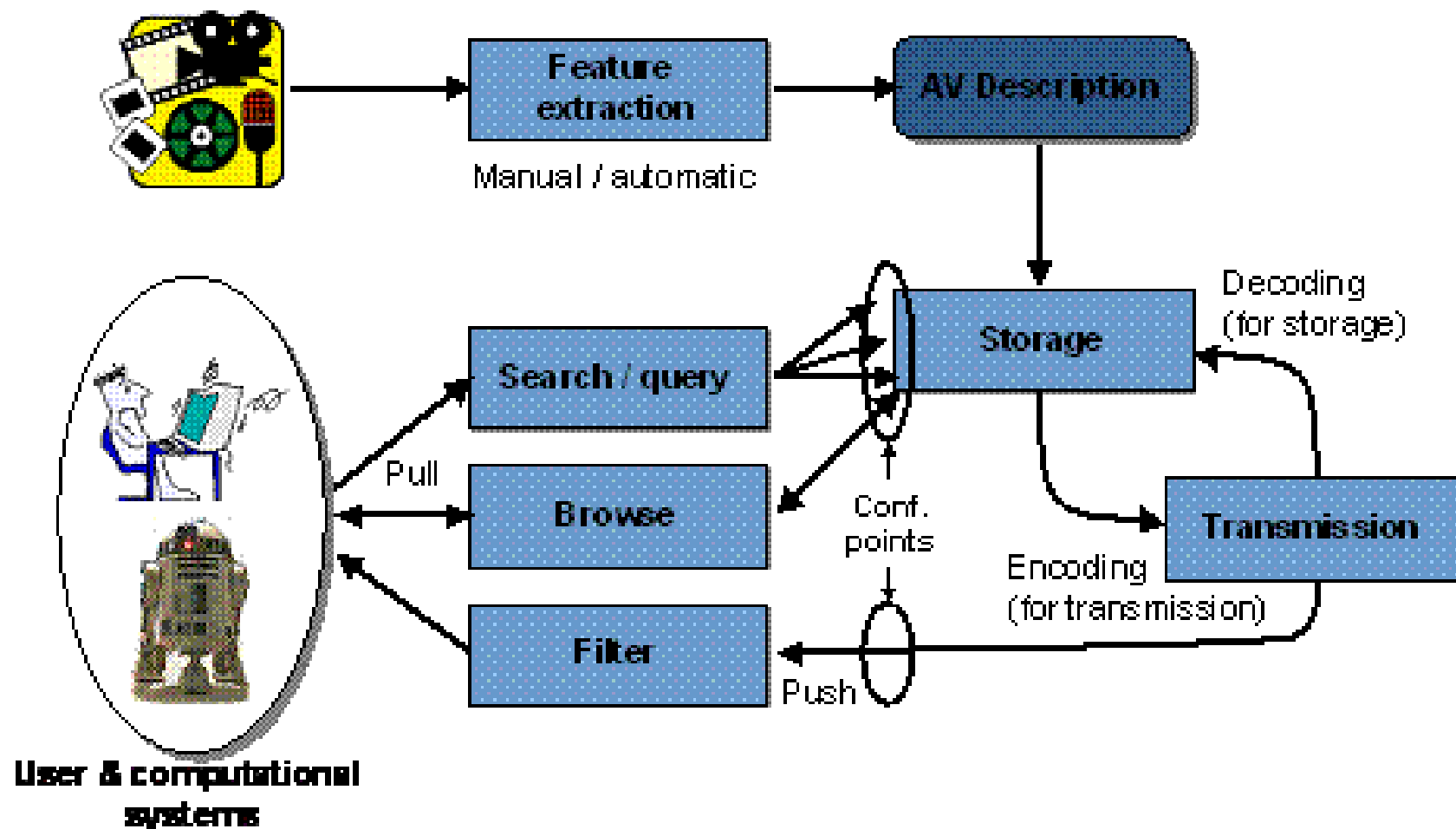
Typy metadanych uwzględniane przez MPEG-7

- Informacje dotyczące procesu tworzenia zawartości
 - Np. tytuł, reżyser
- Informacje dotyczące wykorzystania zawartości
 - Np. prawa autorskie, historia/plan emisji
- Informacje o sposobie zapisu zawartości
 - Format, kodowanie
- Informacje strukturalne o przestrzennych, temporalny i przestrzenno-temporalnych składnikach zawartości
 - Sceny, segmenty, śledzenie ruchu dla regionów
- Niskopoziomowe właściwości zawartości
 - Kolory, tekstury, sygnałowy opis dźwięku, melodia
- Informacje konceptualne o przedstawianej rzeczywistości
 - Obiekty, zdarzenia, interakcje między obiektami, kolekcje obiektów
- Informacje wspierające przeszukiwanie zawartości
 - Np. streszczenia scen/ujęć
- Informacje związane z interakcją z użytkownikiem
 - Ustawienia preferencji, historia

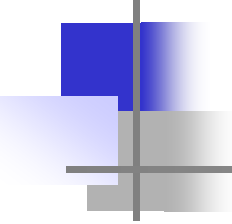
Elementy MPEG-7 – Podsumowanie*



MPEG-7 w praktyce*



* José M. Martínez, UAM-GTI, ES

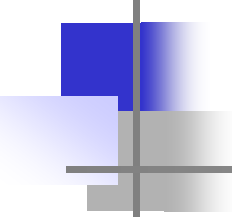


MPEG-7 a systemy multimedialnych baz danych

- MPEG-7 dotyczy opisów zawartości – zagadnienia nieobjętego standardem SQL/MM
 - Brak współpracy między standardami
- W porównaniu z prototypowymi modelami semantycznymi dla MMBD – "płaskie" opisy
- Obsługa zapytań do opisów MPEG-7 poza zakresem standardu
- Obecne wsparcie dla analizy opisów MPEG-7 w SZBD na poziomie ogólnego wsparcia dla XML
 - Składowanie: CLOB, XMLType, XPath, XQuery, wsparcie dla XMLSchema

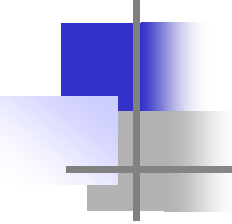


Obsługa danych multimedialnych w SZBD ogólnego przeznaczenia



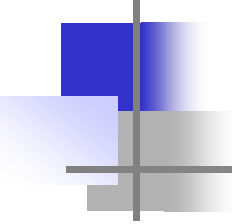
Oracle 11g

- Zaawansowana funkcjonalność serwera bazy danych Oracle:
 - Features (dostępność zależy od wersji serwera: EE/SE/PE)
 - Options (podlegają oddzielnej licencji): np. Oracle Spatial
- Za obsługę danych multimedialnych odpowiada Oracle Multimedia (dawniej interMedia)
 - feature, dostępna w edycjach serwera EE/SE
 - w Oracle 11g obsługa zdjęć medycznych DICOM jako ważny obszar funkcjonalny
- Elementy serwera Oracle odpowiedzialne za przetwarzanie innych zaawansowanych typów danych:
 - Oracle XML DB
 - Oracle Text
 - Oracle Spatial (dodatkowo płatna opcja)



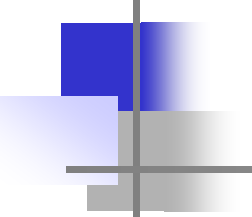
Oracle Multimedia

- Pozwala na obsługę danych multimedialnych składowanych jako: BLOB (limit 128 TB), BFILE, External URL
- Obsługa oparta o mechanizm typów obiektowych
- Typy danych: ORDAudio, ORDDoc, ORDImage, ORDVideo, ORDSource, typy SQL/MM Still Image (od wersji 10g)
- W przypadku wewnętrznego składowania, ochrona danych i bezpieczeństwo jak dla danych tradycyjnych
- Zapytania o metadane
 - Dla obrazów również o zawartość (CBIR) - cechy reprezentowane jako obiekty ORDImageSignature (zdeprecjonowane w 11g) lub obiekty typów SQL/MM SI
 - CBIR w ramach ORDImage (ORDImageSignature) zdeprecjonowany w 11g
- Automatyczna ekstrakcja i pielęgnacja metadanych z danych multimedialnych w standardowych formatach
- Obsługa typowych formatów danych
 - Możliwość rozszerzeń
 - Możliwość współpracy z zewnętrznymi źródłami danych



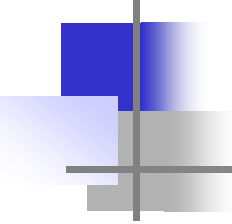
IBM DB2 9.7

- Składowanie danych multimedialnych
 - BLOB (limit 2GB)
 - dla dużych danych tekstowych CLOB, DBCLOB (DB = double-byte)
- DB2 V8 do obsługi multimediiów posiadała AIV Extenders obejmujące:
 - DB2 Audio Extender
 - DB2 Image Extender
 - DB2 Video Extender
- Funkcjonalność AIV Extenders podobna do Oracle
 - w DB2 Image Extender: QBIC
- AIV Extenders niedostępne w DB2 Version 9



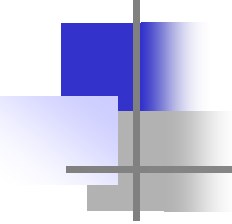
Informix Dynamic Server 11.50

- SZBD Informix został zakupiony w 2001 przez firmę IBM i obecnie jest rozwijany równolegle do DB2
- Składowanie danych multimedialnych:
 - podział typów na "Simple-Large-Object Data Types" (BYTE/TEXT) i "Smart-Large-Object Data Types" (BLOB/CLOB)
 - Dla danych binarnych: BYTE (do 2 GB), BLOB (do 4 TB)
 - Dla dużych danych tekstowych odpowiednio: TEXT i CLOB
- Zaawansowana funkcjonalność serwera dostępna poprzez moduły rozszerzające dla bazy Informix (tzw. DataBlades)
- DataBlades do obsługi multimediiów np.
 - IBM Informix Image Foundation DataBlade
 - IBM Informix Video Foundation DataBlade
 - Excalibur Image DataBlade 1.2 (dostępny dla wersji 9.x)
 - Content-Based Image Retrieval (color content, shape content, texture content, brightness structure, color structure, aspect ratio)



Microsoft SQL Server 2008

- Obsługa multimediiów ogranicza się do składowania
 - Składowanie poprzez typ varbinary(max) (limit 2GB)
 - Duże dane tekstowe jako varchar(max) lub nvarchar(max)
 - Dawniej typy image, text, ntext (obecnie niezalecane)
 - Nowość w 2008: varbinary(max) FILESTREAM
- Atrybut FILESTREAM
 - Składowanie BLOB-ów w systemie plików NTFS
 - Zalecane gdy rozmiar > 1MB i ważny szybki odczyt strumieniowy
 - Z możliwością modyfikacji i transakcyjnością
 - Wymagana konfiguracja obsługi funkcjonalności FILESTREAM na poziomie instancji serwera
 - Dostęp z poziomu T-SQL lub T-SQL i Win32 API (z użyciem tokenu transakcyjnego)
 - Dla bazy danych ustawienie wskazujące katalog na dysku:
FILEGROUP <nazwa> CONTAINS FILESTREAM



Multimedia w innych SZBD ogólnego przeznaczenia

- Sybase (Adaptive Server Enterprise 15)
 - Składowanie poprzez typ IMAGE (limit 2GB)
 - Duże dane tekstowe jako TEXT lub UNITEXT
- MySQL (5.4)
 - Składowanie poprzez typy TINYBLOB, BLOB, MEDIUMBLOB, LONGBLOB (limit 4GB)
 - Dla dużych danych tekstowych typy TINYTEXT, TEXT, MEDIUMTEXT, LONGTEXT
- PostgreSQL (8.2)
 - Składowanie poprzez typ bytea (limit 1GB)
 - Dla dużych danych tekstowych typ TEXT