



# Bazy Danych

---

## Systemy Zarządzania Bazami Danych

Krzysztof Regulski  
WIMiP, KISiM,  
regulski@agh.edu.pl  
B5, pok. 408

---

- **Dane to fakty**

Dana, jako jednostka danych, jest to jeden lub kilka symboli, użytych do reprezentowania czegoś

- **Informacja to zinterpretowane dane**

Informacje to dane umieszczone w znaczącym kontekście

- **Wiedza jest otrzymywana z informacji**

przez jej zintegrowanie z istniejącą wiedzą.

- **Informacja ma charakter subiektywny.**

Informacja musi być zawsze rozpatrywana w kontekście jej odbiorcy. Te same dane mogą być różnie interpretowane przez różnych ludzi w zależności od posiadanej wiedzy i kontekstu.

# System informacyjny (IS)

- **System informacyjny** to system, który dostarcza informacje do pewnego przedsiębiorstwa lub jego części.
- System może być zdefiniowany jako **spójny zbiór niezależnych składowych**, które istnieją w jakimś celu, mają pewną stabilność i mogą być przydatne przy ich łącznym rozpatrywaniu.
- Systemy są zwykle przedstawiane za pomocą modelu **wejście-proces-wyjście** istniejącego w danym środowisku.
- **Środowisko systemu** może być zdefiniowane jako wszystko to, co znajduje się poza systemem, co ma wpływ na sposób działania systemu.
- **Wejściami** do systemu są zasoby, które system pozyskuje ze swojego środowiska lub z innych systemów.
- **Wyjściami** z systemu jest to, co system dostarcza do środowiska lub innych systemów.
- **Proces** - to działanie, które przekształca wejścia systemu w jego wyjścia.

- Pojęcie systemu znalazło zastosowanie w wielu dziedzinach, (fizyka, biologia, elektronika, itd.).
- Systemy informacyjne obsługują systemy związane z działalnością człowieka.
- Systemy takie mają dodatkowy składnik dodany do opisanego wyżej modelu wejście-proces-wyjście: człowieka.
- Systemy związane z działalnością człowieka składają się z
  - » ludzi,
  - » konwencji i
  - » wytworów pracy przeznaczonych do zaspokajania potrzeb ludzkich.
- Każdy system związany z działalnością człowieka zawiera jeden lub kilka systemów informacyjnych. Celem tych systemów jest **zarządzanie systemem związanym z działalnością człowieka.**

# Technologia informacyjna

---

- Technologia informacyjna zapewnia środki służące do konstruowania nowoczesnych systemów informacyjnych.
- **Technologia informacyjna** (*Information Technology; IT*) obejmuje komputery (sprzęt i oprogramowanie) oraz **komunikację**.
- Komputery i sieci komunikacyjne są głównie używane do obsługi różnych aspektów systemów informacyjnych przedsiębiorstwa.
- Systemy informacyjne **istniały przed wynalezieniem** technologii informacyjnej, a więc systemy informacyjne nie potrzebują istnienia tej technologii.
- W nowoczesnych, złożonych przedsiębiorstwach większość systemów informacyjnych opiera się w większym lub mniejszym stopniu na technologii informacyjnej.

- **Podsystem interfejsu**

Jest odpowiedzialny za utrzymywanie wszystkich interakcji z użytkownikiem. Często nazywa się interfejsem użytkownika.

- **Podsystem reguł**

Zajmuje się logiką aplikacji z punktu widzenia zdefiniowanego modelu reguł działania.

- **Podsystem transakcji**

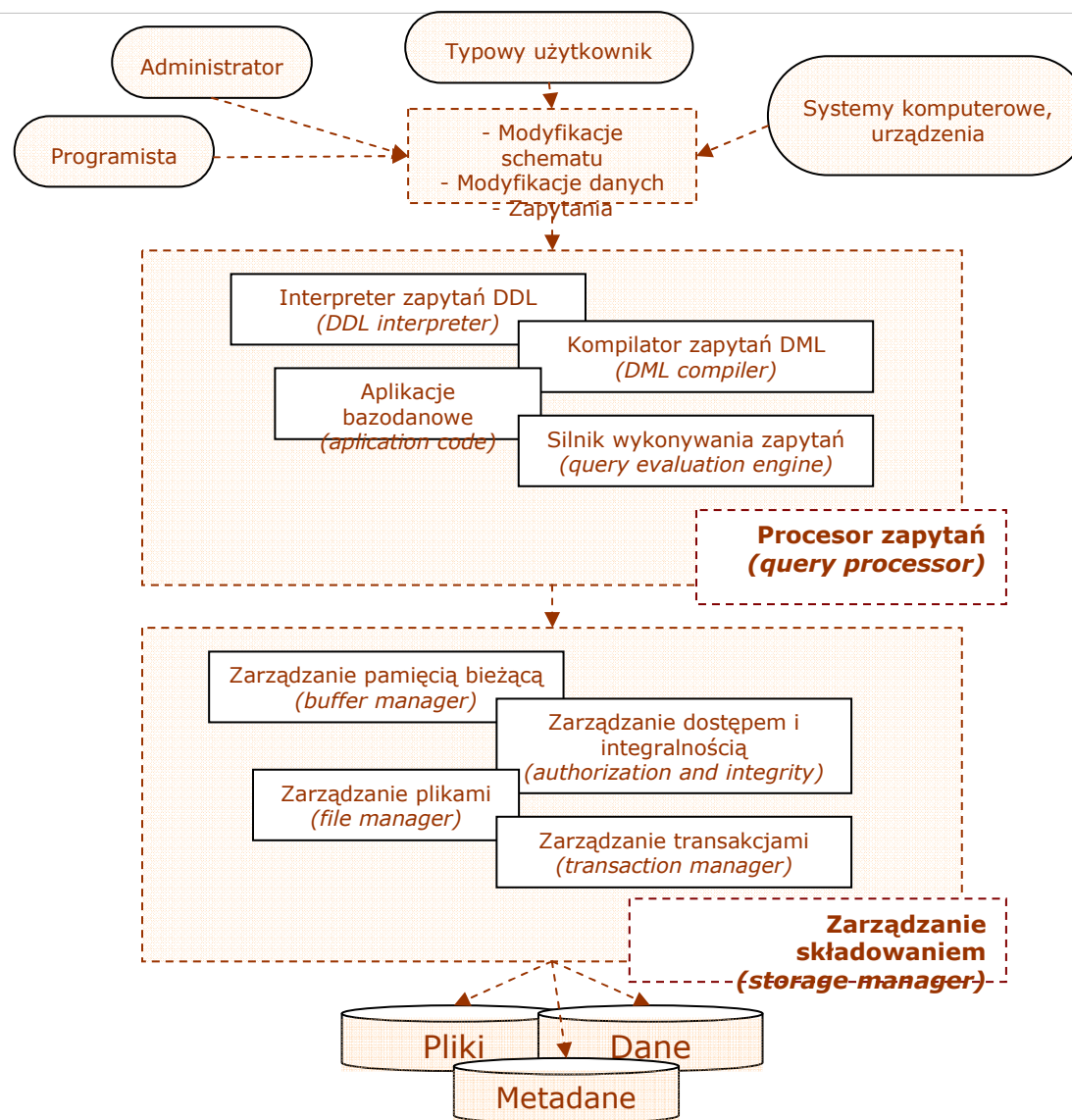
Działa jako połączenie między podsystemem danych a podsystemem interfejsu. Tutaj są wywoływane działania wyszukiwania, wstawiania i aktualizacji, sprawdzane przez podsystem reguł i łączone w jednostki (transakcje), które będą inicjować odpowiedzi lub zmiany w podsystemie danych.

- **Podsystem danych**

Jest odpowiedzialny za zarządzanie danymi potrzebnymi do aplikacji.

- Narzędzia **graficznego interfejsu** użytkownika przeznaczone do tworzenia zaawansowanych interfejsów użytkownika.
- **Języki czwartej generacji** przeznaczone do kodowania reguł działania i logiki aplikacji.
- **Systemy przetwarzania transakcji** przeznaczone do otrzymywania dużej przepustowości transakcji.
- **Systemy zarządzania bazą danych** przeznaczone głównie do zarządzania dostępem wielu użytkowników do danych.

# Ogólna struktura DBMS





- **Korporacyjny model danych**

Określa wymagania dotyczące danych dla całego przedsiębiorstwa: struktura, nazewnictwo i znaczenie danych w przedsiębiorstwie.

- **Model danych obszaru biznesowego**

Określają wymagania dotyczące danych dla poszczególnych obszarów biznesowych – model danych obszaru działania.

- **Model danych aplikacji**

Określa wymagania dotyczące danych dla poszczególnych zastosowań systemów informacyjnych. Zorientowany na poszczególne aplikacje.

- **System zarządzania bazą danych** może być traktowany jako bufor między programami użytkowymi, użytkownikami końcowymi i bazą danych zaprojektowaną z zachowaniem cechy niezależności danych.
- W 1975 r. **American National Standards Institute - Standards Planning and Requirements Committee (ANSI-SPARC)** zaproponował trypoziomową architekturę dla tego bufora.
- Architektura ta wyróżnia trzy poziomy abstrakcji. Poziomy te są nazywane czasem schematami lub widokami

- **Poziom zewnętrzny lub użytkownika**

Poziom ten opisuje sposób w jaki widzą dane użytkownicy lub programy użytkowe. Ten sam widok może być wykorzystywany przez kilka programów lub użytkowników.

- **Poziom koncepcyjny lub pojęciowy**

Ten poziom opisuje widok wszystkich danych w bazie danych postrzegany przez organizacje, związki między danymi i więzy odnoszące się do bazy danych. Poziom ten opisuje logiczny widok bazy danych - widok bez szczegółów dotyczących realizacji.

- **Poziom wewnętrzny lub fizyczny**

Ten poziom opisuje sposób przechowywania danych oraz sposób dostępu do danych. Jest to fizyczny widok bazy danych.

## Architektura ANSI/SPARC

– Każdy poziom jest opisany jako schemat – odwzorowanie bazy danych. Trzy poziomowa architektura służy do realizacji niezależności za pośrednictwem dwóch poziomów odwzorowania:

- » między schematem zewnętrznym a koncepcyjnym, oraz
- » między schematem koncepcyjnym a fizycznym.

– Niezależność danych ma dwie formy:

» **Logiczna niezależność danych.**

Oznacza ona niewrażliwość schematów zewnętrznych na zmiany w schemacie koncepcyjnym. Na przykład można dodać nową daną do schematu koncepcyjnego bez wpływania na poziom zewnętrzny.

» **Fizyczna niezależność danych.**

Oznacza ona niewrażliwość schematu koncepcyjnego na zmiany w schemacie fizycznym. Na przykład można zmienić strukturę przechowywania danych w bazie danych bez wpływania na schemat koncepcyjny.

## Architektura ANSI/SPARC

---

- **Jądro DBMS** oznacza centralną maszynę, która realizuje podstawowe funkcje zarządzania danymi
- Zestaw narzędzi DBMS oznacza szeroki wybór narzędzi, które obecnie mogą stanowić część DBMS lub mogą pochodzić od innych dostawców. Na przykład arkusze kalkulacyjne, języki czwartej generacji, programy monitorujące wydajność itp
- Aby powiązać jądro z zestawem narzędzi, musimy mieć zdefiniowany interfejs. Jest to standardowy język, który łączy narzędzie, takie jak język czwartej generacji, z funkcjami jądra

## 4GL (4th Generation Language)

język czwartej generacji; język programowania, pozwalający przy użyciu krótkich instrukcji stworzyć program, którego napisanie w językach niższej (np. trzeciej, 3GL) generacji wymaga użycia setek lub tysięcy razy większej liczby wierszy programu źródłowego; 4GL często dopuszcza pisanie fragmentów kodu w kilku językach 3GL jednocześnie. Podstawowym wyróżnikiem języka 4GL jest jego specjalizacja, tworząca z języka efektywne narzędzie w ramach ściśle określonego obszaru zastosowań. (*wikipedia*)

Języki takie mają wbudowane polecenia realizujące potężny zasób algorytmów statystyki i analizy danych, język zapytań, język generowania raportów, język projektowania formatek ekranowych i ograniczony zestaw niezbędnych konstrukcji

### – **Funkcje CRUD**

DBMS musi umożliwić użytkownikowi tworzenie struktur danych, zmiany danych w strukturach oraz usuwanie danych ze struktur. Funkcje te są określane łącznie jako operacje CRUD

- » Create (utwórz),
- » Read (czytaj),
- » Update (aktualizuj),
- » Delete (usuń).

### – **Słownik danych**

DBMS musi obsługiwać repozytorium metadanych - danych o danych. Repozytorium to nazywa się słownikiem danych lub katalogiem systemu. Zwykle słownik danych przechowuje dane o strukturze danych, związkach między danymi, więzach integralności nałożonych na dane, nazwach i uprawnieniach użytkowników. Słownik danych realizuje w praktyce trypoziomową architekturę DBMS.

## – Zarządzanie transakcjami

DBMS musi obsługiwać pojęcie transakcji i kierować sytuacją, gdy wiele transakcji działa na bazę danych. Transakcja to seria działań, które mają dostęp lub powodują zmiany w bazie danych

## – Sterowanie współbieżnością

DBMS musi umożliwiać wielu użytkownikom wspólne korzystanie z danych w bazie - współbieżny dostęp do danych. DBMS musi zapewniać, że w wypadku gdy dwie transakcje mają dostęp do tych samych danych, baza danych nie znajdzie się w stanie niespójnym.



## – Odtwarzanie

DBMS musi zapewnić, że baza danych może zostać odtworzona po awarii sprzętu lub oprogramowania powodującej uszkodzenie bazy.

## – Uprawnienia

DBMS musi mieć narzędzia zapewniające bezpieczeństwo bazy danych. Ogólnie mówiąc, DBMS musi obsługiwać pojęcie uprawnionego użytkownika bazy danych oraz umożliwiać powiązanie **uprawnień** każdego użytkownika z dostępem do danych w bazie i/lub narzędzi DBMS.

## – Komunikacja danych

DBMS musi mieć możliwość integracji z oprogramowaniem komunikacyjnym działającym w kontekście systemu technologii informacyjnej. Szczególnie ważne jest zapewnienie połączenia oprogramowania narzędziowego z jądrem DBMS.

**ODBC** (ang. *Open DataBase Connectivity*) – 1992r.

## – Integralność danych

Integralność danych jest właściwością bazy danych zapewniającą, że pozostaje ona wiernym odzwierciedleniem obszaru analizy. Aby to umożliwić, DBMS musi obsługiwać **więzy integralności**. DBMS musi mieć możliwość wymuszania więzów w kontekście operacji CRUD.

## – Narzędzia administracji

DBMS powinien zapewnić odpowiednie narzędzia do administrowania bazą danych:

- » narzędzia **importowania** danych do bazy z innych źródeł danych;
- » narzędzia **eksportowania** danych z bazy danych do innych źródeł danych;
- » narzędzia **monitorowania użycia** i operacji na bazie danych;
- » narzędzia **monitorowania wydajności** bazy danych i zwiększania tej wydajności.

Interfejs DBMS składa się z subjęzyka bazy danych. Jest to język programowania przeznaczony specjalnie do inicjowania funkcji DBMS. Składa się on z jednej lub kilku wymienionych niżej części.

– **Język definiowania danych (data definition language, DDL)**

jest stosowany do tworzenia i usuwania struktur danych oraz do uzupełniania istniejących struktur. DDL aktualizuje metadane przechowywane w słowniku danych.

– **Język operowania danymi (data manipulation language, DML)**

jest używany do określania poleceń, które realizują działania CRUD na bazie danych. DML jest podstawowym mechanizmem stosowanym przy określaniu transakcji wykonywanych na bazie danych.

- **Język integralności danych {data integrity language, DIL)**

jest używany do określania więzów integralności. Więzy określone w DIL są zapisywane w słowniku danych.

- **Język kontroli danych {data control language, DCL)**

jest częścią subjęzyka bazy danych przeznaczoną do wykorzystania przez administratora bazy. Jest on stosowany zwłaszcza do definiowania użytkowników bazy danych oraz przyznanych im uprawnień. Głównym obecnie przykładem takiego subjęzyka jest strukturalny język zapytań SQL. Ten subjęzyk jest często stosowany w powiązaniu z innymi narzędziami tworzenia aplikacji, takimi jak język czwartej generacji (4GL) do realizacji systemu technologii informacyjnej

Większość DBMS stanowią fragmenty oprogramowania umiejscowione na szczycie rdzennego systemu operacyjnego. Z tego powodu DBMS musi współdziałać z systemem operacyjnym, zwłaszcza przy realizacji dostępu do bazy danych i katalogów systemowych przechowywanych zwykle na dyskach. W tym współdziałaniu zasadnicze znaczenie mają trzy elementy systemu operacyjnego:

- **Menedżer plików**

Menedżer plików systemu operacyjnego jest tłumaczem między strukturami danych, na których działa DBMS, a plikami na dysku. W celu wykonania tego zadania tworzy on zapis fizycznych struktur na dysku

- **Mechanizmy dostępu**

Menedżer plików nie zarządza bezpośrednio fizycznym wejściem i wyjściem danych. Współdziała on z odpowiednimi mechanizmami dostępu określonymi dla różnych struktur fizycznych

- **Bufory systemu**

Odczyt i zapis danych jest zwykle prowadzony przez bufory systemu operacyjnego. Są to tymczasowe struktury służące przechowywaniu danych i wykonywaniu na nich operacji.

- **Baza danych i katalog systemowy**

przechowują dane i metadane aplikacji.

- **System operacyjny**

Dostęp do urządzeń dyskowych jest kontrolowany głównie przez system operacyjny gospodarza, który szereguje dyskowe operacje we-wy.

- **Menedżer bazy danych**

kontroluje dostęp do informacji DBMS przechowywanej na dysku. Wszystkie operacje dostępu do informacji w bazie danych są wykonywane pod jego kontrolą. Menedżer danych używa funkcji zarządzania danymi niższego poziomu, dostarczonymi przez system operacyjny gospodarza w jego systemie zarządzania plikami lub używa swoich własnych podprogramów. Takie funkcje będą realizować przesyłanie danych między pamięcią główną a dyskiem.

- **Kompilator DDL**

Kompilator języka definicji danych DDL bierze instrukcje wyrażone w DDL i uaktualnia katalog systemowy. Wszystkie moduły DBMS, które potrzebują informacji na temat obiektów bazy danych, muszą mieć dostęp do katalogu.

- **Procesor czasu wykonywania**

obsługuje operacje wyszukiwania i modyfikowania, wyrażone względem bazy danych. Dostęp do dysku jest realizowany przez menedżera danych,

- **Procesor zapytań**

obsługuje interaktywne zapytania, wyrażone w języku operowania danymi (DML) takim jak SQL. Dokonuje on analizy składniowej i semantycznej zapytania, po czym przystępuje do generowania wywołań procesora czasu wykonywania.

- **Prekompilator, kompilator DML, kompilator języka gospodarza**

Prekompilator wycina polecenia DML z programu użytkowego napisanego w języku gospodarza. Polecenia te są następnie przekazywane do kompilatora DML, który tworzy kod wynikowy realizujący dostęp do bazy danych. Pozostałe części programu użytkowego są wysyłane do konwencjonalnego kompilatora i następnie te dwie części są wiązane razem, tworząc transakcję, która może być wykonana na bazie danych

# Menedżer bazy danych

---

Moduł DBMS zajmujący się zarządzaniem bazą danych wykonuje następujące funkcje:

- **Kontrola uprawnień**

Menedżer bazy danych musi kontrolować uprawnienia wszystkich prób dostępu do bazy.

- **Przetwarzanie poleceń**

Menedżer bazy danych musi wykonywać polecenia subjęzyka bazy danych.

- **Kontrola spójności**

Menedżer bazy danych musi sprawdzić, czy każda operacja próbująca wykonać zmianę w bazie danych nie łamie zdefiniowanych więzów integralności.

- **Optymalizacja zapytań**

Ta funkcja wyznacza optymalną strategię wykonywania zapytań do bazy danych.



- **Zarządzanie transakcjami**

Menedżer bazy danych musi wykonywać transakcje na bazie danych.

- **Planowanie transakcji**

Ta funkcja zapewnia współbieżny dostęp transakcji do bazy danych.

- **Zarządzanie odtwarzaniem**

Ta funkcja odpowiada za zatwierdzanie i anulowanie transakcji wykonywanych na bazie danych. Zarządzanie buforami. Ta funkcja odpowiada za transfer danych do i z pamięci głównej i pomocniczej

Podstawową funkcją większości systemów technologii informacyjnych jest ich **współdziałanie z użytkownikami**.

Nawet system wsadowy może mieć pewien mechanizm pozwalający użytkownikom uruchomić i kończyć proces.

Bez względu na to, jak poprawnie zbudowany jest system informacyjny, jego efektywność jest w znacznym stopniu zależna od interfejsu użytkownika. Tak więc funkcjonalność (*utility*) systemu informacyjnego jest wyznaczona przez jego „używalność” (użyteczność, *usability*).

## – Menu

Składa się z wyświetlanej listy wyborów. Użytkownik wybiera pozycję z listy, naciskając kombinację klawiszy lub przesuwając kursor na odpowiednią pozycję i wybierając ją przez kliknięcie myszką. Menu są często stosowane w systemach baz danych jako sposób uruchamiania przygotowanych zapytań.

## – Formularze

Te formy interfejsu są stosowane w celu wprowadzania lub wyszukiwania danych. Formularz jest prostym zestawem pól rozmieszczonych na ekranie lub, co obecnie jest bardziej czytelne, w kontekście okna. Pole do wprowadzania danych ma zwykle obszar nagłówka i obszar do wyświetlania komunikatów błędów lub odpowiedzi. Wiele współczesnych narzędzi budowy aplikacji automatycznie tworzy domyślne ekrany do wprowadzania danych na podstawie schematów bazy danych.

## – Język poleceń

Przy takim interfejsie użytkownik wprowadza polecenia wyrażone w pewnym języku formalnym w celu wykonania funkcji. Interfejsy na poziomie poleceń były historycznie **pierwszym typem interfejsów ekranowych**. Tego typu interfejsy mają systemy operacyjne takie jak MS DOS i Unix. Również wiele DBMS oferuje interfejs do SQL na poziomie poleceń.

## – Język naturalny

Tak zwane interfejsy języka naturalnego nie są dokładnie językami naturalnymi w tym sensie, że przyjmują wyrażenia języka codziennego jako wejściowy ciąg znaków. Dokładniej są to interfejsy ograniczonego języka, w którym zdanie „Podaj listę wynagrodzeń pracowników” może być dopuszczalne, natomiast zdanie „Podaj wynagrodzenia moich pracowników” – nie. Takie interfejsy osiągnęły pewien sukces jako dostęp do baz danych, zwłaszcza w połączeniu z techniką rozpoznawania mowy.

# Typy interfejsów

---

- **Bezpośrednie działanie**

Ten typ interfejsu jest zwykle powiązany ze środowiskiem ikon. Nazywa się on bezpośrednim działaniem, ponieważ użytkownik wywołuje zdarzenia, operując obiektami graficznymi za pomocą myszki. Takie interfejsy są dominujące obecnie w większości systemów informacyjnych

- **Interfejs multimedialny**

Jest to rozszerzenie interfejsu bezpośredniego działania opisanego wyżej, gdyż stosuje te same mechanizmy do sterowania wejściem. Główna różnica polega na tym, że zamiast prostych menu i formularzy do wprowadzania danych, do budowy interfejsu stosowanych jest wiele środków.

- **Interfejs wirtualnej rzeczywistości**

W niektórych aplikacjach, występujących na przykład w symulacji, ważne jest, aby użytkownik, korzystając z interfejsu, czuł się w środowisku podobnym do środowiska dostępnego w świecie rzeczywistym. Przykładem mógłby być interfejs emulujący fizyczne ułożenie publikacji w bibliotece. Użytkownicy takiego systemu mogliby wybrać określoną publikację, przechodząc przez wirtualną przestrzeń reprezentowaną w systemie.

- Podstawową ideą interfejsu typu zapytanie przez przykład (Query By Example; QBE) jest dostarczenie bazie danych, która jest z natury rzeczy nieproceduralna, interfejsu w postaci menu lub formularza.
- Interfejsy QBE są głównie przeznaczone dla początkujących użytkowników baz danych.
- Na początku użytkownik zaczyna dialog z interfejsem QBE do wyboru tabel z wyświetlonego menu tabel. Następnie wyświetla się szablon tabeli dla każdej wybranej tabeli. Później użytkownik wypełnia puste miejsca w tym szablonie, używając stałych funkcji lub wcześniej zdefiniowanych poleceń.

# Graficzne interfejsy użytkownika

---

- Oparte na zestawie formularzy i technologii drag&drop.
- Pozwalają na realizowanie zadań za pomocą myszki oraz poprzez ustalanie zestawu właściwości w oparciu o listy rozwijalne, pola własności oraz kreatory

## Interfejsy języka naturalnego

---

- Dane są zazwyczaj wydobywane z bazy danych za pomocą formalnego języka, np. SQL.
- Interfejs pozwala na zadanie pytania lub polecenia w języku zbliżonym do naturalnego.
- Część oprogramowania, która umożliwia tę formę dialogu z użytkownikiem, jest znana jako interfejs języka naturalnego (**Natural Language Interface**; NLI).
- NLI są nazywane, bardziej trafnie, interfejsami ograniczonych języków. NLI zazwyczaj musi zostać „wyćwiczony” pod kątem konkretnej aplikacji. NLI nie może „rozumieć” wszystkich zdań języka naturalnego, a tylko ograniczony jego podzbiór. Powinien uwzględniać terminologię odpowiedniej grupy użytkowników i jak ta terminologia łączy się z zawartością bazy danych.



- **Analiza leksykalna**

Jest to proces dzielenia zdania wejściowego na słowa. Lista słów, które NLI może zrozumieć, jest zwykle przechowywana w repozytorium, zwanym leksykonem.

- **Analiza składniowa**

Jest to proces przypisany zdaniu wejściowemu pewnej struktury gramatycznej. Taka struktura ma zazwyczaj postać drzewa składniowego.

- **Analiza semantyczna**

Jest to proces przypisania instrukcji wejściowej pewnego znaczenia i określenia, jakie akcje powinny zostać podjęte.

### – **Uruchamianie, zatrzymywanie i odłączanie bazy danych**

DBMS musi dostarczać polecenia służące do uruchamiania i zatrzymywania aplikacji baz danych. Można to robić na początku i końcu każdego dnia lub innego okresu.

Czasami całość lub część bazy danych wymaga wyłączenia, tzn. działa ona dalej, ale nie jest dostępna dla użytkowników z wyjątkiem administratora. Może tak być, na przykład, gdy trzeba zmienić strukturę schematu.

## – Określenie grup użytkowników

Każdy użytkownik ma zwykle przypisaną unikatową nazwę. Niektóre DBMS pozwalają administratorowi tworzyć grupy użytkowników lub „role” o określonych nazwach. Daje to możliwość ustalania wspólnego profilu dla zbioru podobnych użytkowników. Jeśli użytkownicy lub grupy użytkowników mają określoną nazwę, każdej nazwie użytkownika lub grupy można przydzielić odpowiednie właściwości.

## – Przydział haseł do użytkowników lub grup użytkowników

Jedną z podstawowych właściwości jest definiowanie hasła dla użytkowników lub grup użytkowników. Pozwala to wprowadzić do systemu bazy danych podstawowe zabezpieczenie. Hasła poszczególnych użytkowników mogą być zmieniane w każdym momencie przez ich właściciela. Hasła roli będą prawdopodobnie okresowo zmieniane przez administratora bazy danych lub przez wyznaczoną osobę z grupy użytkowników.

## Przyznawanie dostępu do danych

- » Tworzenie **perspektyw** - okien do bazy danych.
- » Nadawanie uprawnień określonym użytkownikom lub ich grupom do **wstawiania** danych do określonych tabel lub perspektyw.
- » Nadanie określonym użytkownikom lub ich grupom uprawnień do **usuwania** danych z określonych tabel lub perspektyw.
- » Nadanie określonym użytkownikom lub ich grupom uprawnień do **modyfikowania** danych w określonych tabelach lub perspektywach.
- » Nadanie określonym użytkownikom lub ich grupom uprawnień do **wyszukiwania** danych z określonych tabel lub perspektyw.
- » DBA, który może przyznawać uprawnienia użytkownikom lub ich grupom do wykonywania powyższych funkcji, może te uprawnienia również odbierać.

## Przyznanie dostępu do funkcji DBMS

- » możliwość dostępu do słownika danych;
- » możliwość tworzenia użytkowników;
- » możliwość tworzenia tabel i perspektyw;
- » możliwość tworzenia indeksów;
- » możliwość przyznawania i odbierania dostępu do tabel i perspektyw.

## Zarządzanie uprawnieniami użytkowników:

```
GRANT przywilej [lista_kolumn]
ON tabela.baza
TO uzytkownik [IDENTIFIED BY `haslo` ]
[WITH GRANT OPTION ]
```

- Przywileje: ALL [PRIVILEGES] , ALTER, CREATE, DELETE, DROP, INSERT, SELECT, SHOW DATABASES
- Lista kolumn: opcjonalna, gdy nadawane są uprawnienia do konkretnej tabeli, pozwala na precyzyjne nadanie uprawnień do różnych części bazy danych
- Użytkownik: składa się z trzech części: loginu, symbolu @, oraz domeny (np. localhost, lub IP np. 149.156.255.139). Dla dowolnego komputera w danej domenie użyjemy:  
login@`%.metal.agh.edu.pl` lub login@`149.156.%`
- IDENTIFIED BY `haslo` : opcja potrzebna przy tworzeniu nowego użytkownika
- WITH GRANT OPTION: opcjonalna, daje użytkownikowi prawo do nadawania uprawnień

## Zarządzanie uprawnieniami użytkowników (2):

**FLUSH PRIVILAGES;**

- Powoduje ponowne odczytanie tabeli z uprawnieniami, dzięki czemu zadziałają zmiany.

**SHOW GRANTS FOR** *uzytkownik;*

- Wynikiem jest wykaz uprawnień danego użytkownika.

**REVOKE** *przywilej* **ON** *tabela.baza* **FROM**  
*uzytkownik;*

- Odbieranie uprawnień danego użytkownika.

## Archiwizacja, kopiowanie i odtwarzanie danych

---

- DBMS dostarcza zwykle narzędzi pozwalających regularnie tworzyć **kopie zapasowe** całości lub części bazy danych. DBA musi też mieć możliwość kopiowania **dziennika transakcji**.
- DBA może tworzyć kopie zapasowe bazy danych i dziennika bez konieczności zatrzymywania systemu.
- Kopie te (łącznie ze słownikiem danych) są zazwyczaj zapisywane na nośnikach zewnętrznych takich jak taśma magnetyczna i mogą być wykorzystane do **odtworzenia** systemu bazy danych po awarii sprzętu lub oprogramowania. DBMS zwykle dostarcza narzędzi umożliwiających wykonanie takiego odtworzenia.
- W dłuższych odstępach czasu DBA może wyczyścić bazę danych z nieużywanych danych. Dane takie mogą być archiwizowane dla celów dokumentacyjnych. W ramach DBMS mogą być dostarczane narzędzia umożliwiające **kompresję takich** danych w celu łatwiejszego przechowywania dużych ilości danych.



## Tworzenie kopii zapasowej (BACKUP) :

- W celu wykonania kopii zapasowej (bezpieczeństwa) należy użyć programu `mysqldump` :

```
shell> mysqldump /path/to/dir [opcje] db_name
```

- a następnie uruchomić ponownie serwer MySQL:

```
shell> mysqld --log-bin[=file_name]
```

- dzięki czemu w pliku logu zapisane zostaną wszystkie modyfikacje począwszy od momentu użycia `mysqldump`
- Odzyskiwanie danych z kopii zapasowej polega na uruchomieniu skryptu wynikowego `mysqldump` (pamiętać należy o `'DROP TABLE IF EXIST'`) :

```
shell> mysql < backup_sunday_1_PM.sql
```

- a następnie odtworzeniu zmian z pliku logu:

```
shell> mysqlbinlog gbichot2-bin.000007 | mysql
```

- Integralność danych obejmuje ochronę bazy danych przed **uprawnionymi** użytkownikami.
- Zabezpieczenie danych obejmuje ochronę bazy danych przed użytkownikami **nieuprawnionymi**.
- W systemie bazy danych zabezpieczenie jest zwykle realizowane przy użyciu narzędzi umożliwiających tworzenie użytkowników i przydzielenie im **uprawnień**. Niektóre DBMS umożliwiają obecnie **szyfrowanie danych**, zwłaszcza przy ich transmisji. Szyfrowanie obejmuje kodowanie danych przy zastosowaniu algorytmu korzystającego z **kluczy szyfrujących** i deszyfrujących. Zaszifrowane dane nie mogą być odczytane bez dostępu do odpowiednich kluczy deszyfrujących.
- Integralność jest realizowana za pomocą **więzów integralności**. DBMS pozwala zwykle administratorowi przeglądać aktywne więzy, usuwać, włączać i wyłączać więzy oraz nadzorować wpływ więzów na efektywność modyfikacji danych.

## Import i eksport danych

---

- DBMS zapewnia zwykle narzędzia do importu danych (zwanego czasem **ładowaniem**) z innych systemów do bazy danych. Często dane te są przechowywane w pierwotnych formatach. Oznacza to, że dane te przed wykonaniem importu będą wymagały zastosowania narzędzi do konwersji danych.
- DBMS dostarcza też narzędzi do eksportu danych z bazy do innych systemów. Także tutaj może być konieczne wykonanie konwersji danych.

## Etapy procesu integracji danych

---

1. **Zrozumienie** – polega na zrozumieniu danych, które podlegają integracji. Duże znaczenie ma analiza metadanych, zajmujących się opisem poszczególnych źródeł danych, czyli analiza schematów, kluczy obcych (zależności referencyjnych), kluczy głównych, analiza innych warunków integralności.

Może ona dotyczyć opisu danych, na przykład: częstość wystąpień, selektywność, rozkład statystyczny wartości danych. Analizie również podlegają relacje pomiędzy osobnymi źródłami danych.

Ważne jest, czy zbiór integrowanych źródeł danych jest określony z góry, czy dopuszczalna jest dynamiczna zmiana tych zbiorów.

## Etapy procesu integracji danych

---

2. **Standaryzacja** - określa się najlepszy sposób reprezentacji i metody czyszczenia integrowanych danych. Dotyczy to zwłaszcza określenia schematu, według którego zintegrowane dane będą prezentowane użytkownikom.

Decyzje, które są tutaj podejmowane dotyczą sposobów reprezentacji danych i używanych nazw. Przykładem może być nazwa ulicy w adresie, czy ma być poprzedzony słowem „Ulica”, czy skrótem „ul.”, z małej lub z dużej litery, z kropką czy bez. Na etapie standaryzacji podejmowane są decyzje odnoszące się do sposobów postępowania z niespójnymi i niepełnymi danymi.

## Etapy procesu integracji danych

---

3. **Specyfikacja** – określa sposób jakim ma być dokonane przetwarzanie danych. Metody i techniki specyfikacji są ściśle związane z wyznaczonym systemem przetwarzania. Używane są tutaj narzędzia mapowania danych konkretyzujących powiązania pomiędzy źródłowymi i docelowymi danymi. Wynikiem mapowania jest uzyskane zapytanie, które transformuje dane źródłowe do żądanej postaci docelowej.

## Etapy procesu integracji danych

4. **Przetwarzanie** – na tym etapie dokonywana jest faktyczna integracja. Może być realizowana przez: materializację, federacje albo indeksowanie.
  - » **Materializacja** – opiera się na stworzeniu bazy zintegrowanych danych. Podstawową metodą materializacji jest **ETL** czyli **Ekstrakcja-Transformacja-Ładowanie**.
  - » **Federacja** – opiera się na kreowaniu wirtualnej reprezentacji integrowanych danych. Prezentacja danych odbywa się przez schemat docelowy. Nie jest on bezpośrednio związany z danymi, ale istnieją definicje, które określają powiązania tego schematu ze schematem źródłowym.
  - » **Indeksowanie** – polega na utrzymywaniu informacji pozwalającej na dotarcie do pełnych danych. Tworzy się indeksy kluczowych słów zawierające adresy URL dokumentów, gdzie te słowa są zawarte. Gdy wykonywane jest zapytanie, dopiero wtedy dokumenty te są dynamicznie pobierane.

- DBMS dostarcza zwykle narzędzi do **monitorowania efektywności** aplikacji baz danych.
- Pozwala to administratorowi uzyskać informacje na temat użycia bazy danych, takiego jak wykonywanie zapytań czy efektywność sterowania współbieżnością.
- Administrator może następnie wykorzystać te informacje do **strojenia aplikacji** baz danych w celu osiągnięcia lepszej wydajności. Można to zrobić, tworząc nowe **indeksy**, zmieniając **struktury** przechowywania danych lub strukturę **schematu**.
- Administrator będzie musiał co jakiś czas dokonywać przeglądu aplikacji bazy danych. Nowe wymagania mogą prowadzić do konieczności przeprojektowania niektórych elementów systemu. Przykładem może być potrzeba **dopasowania możliwości przechowywania** danych w bazie.