

Тема лекції 1:

Системи баз даних. Основні поняття, архітектура, моделі даних.

- ❑ Основні визначення
 - ❑ Розподіл обов'язків в системах з базами даних
 - ❑ Функції СУБД
 - ❑ Трирівнева архітектура організації бази даних
 - ❑ Архітектура програмно-технічних засобів
 - ❑ Класифікації моделей даних
-

Інформаційна система – це програмний продукт

Функції ІС:

- ☐ підтримка надійного зберігання інформації;
 - ☐ виконання певних перетворень інформації або обчислень;
 - ☐ надання користувачам зручного інтерфейсу.
-

Визначення бази даних

База даних – це сукупність логічно зв'язаних даних (і опис цих даних), яка відображає стан об'єктів та їх зв'язків в певній предметній області, динамічно змінюється у процесі свого функціонування і використання багатьма користувачами [Дейт].

База даних – це єдине велике сховище даних, яке один раз визначається, а потім функціонує у багатокористувацькому режимі [Ульман].

Інші визначення

Схема (зміст) бази даних – це опис бази даних

Стан (деталізація) бази даних– це сукупність інформації, що зберігається в базі даних в будь-який певний момент часу

Прикладні програми, застосування, додатки баз даних

- ☐ можуть працювати паралельно і незалежно одна від одної
 - ☐ структура даних відділена від додатків
 - ☐ незалежність від даних
-

Система управління базами даних

СУБД – це програме забезпечення підтримки інтегрованої сукупності даних, призначене для створення, зберігання, ведення і використання бази даних багатьма користувачами (застосуваннями).

Банк даних

- це система технічних, мовних, програмних, організаційно-методичних засобів, що призначені для накопичення, підтримки і багатокористувацького використання даних, які подані у вигляді баз даних.
-

Система баз даних

- це певна прикладна система, яка використовує базу даних і СУБД, що підтримує цю БД, призначена для вирішення конкретних завдань зберігання і обробки даних
-

Розподіл обов'язків в системах з базами даних

1. Адміністратори даних та адміністратори баз даних
 2. Розробники баз даних
 3. Прикладні програмісти
 4. Користувачі-клієнти
 - Пересічні користувачі
 - Досвідчені користувачі
-

Функції СУБД

- ❑ Зберігання, вибір та оновлення даних
 - ❑ Визначення структури бази даних, її ініціалізація і проведення початкового завантаження даних
 - ❑ Наявність каталогу (бази метаданих)
 - ❑ Служба підтримки цілісності даних
 - ❑ Підтримка транзакцій
-

Функції СУБД

- ❑ Сервіс контролю за доступом до даних
 - ❑ Служба підтримки незалежності програм від даних
 - ❑ Підтримка обміну даними
 - ❑ Сервіс відновлення бази даних
 - ❑ Сервіс керування паралельністю
-

Функції СУБД.

Додаткові служби, утиліти

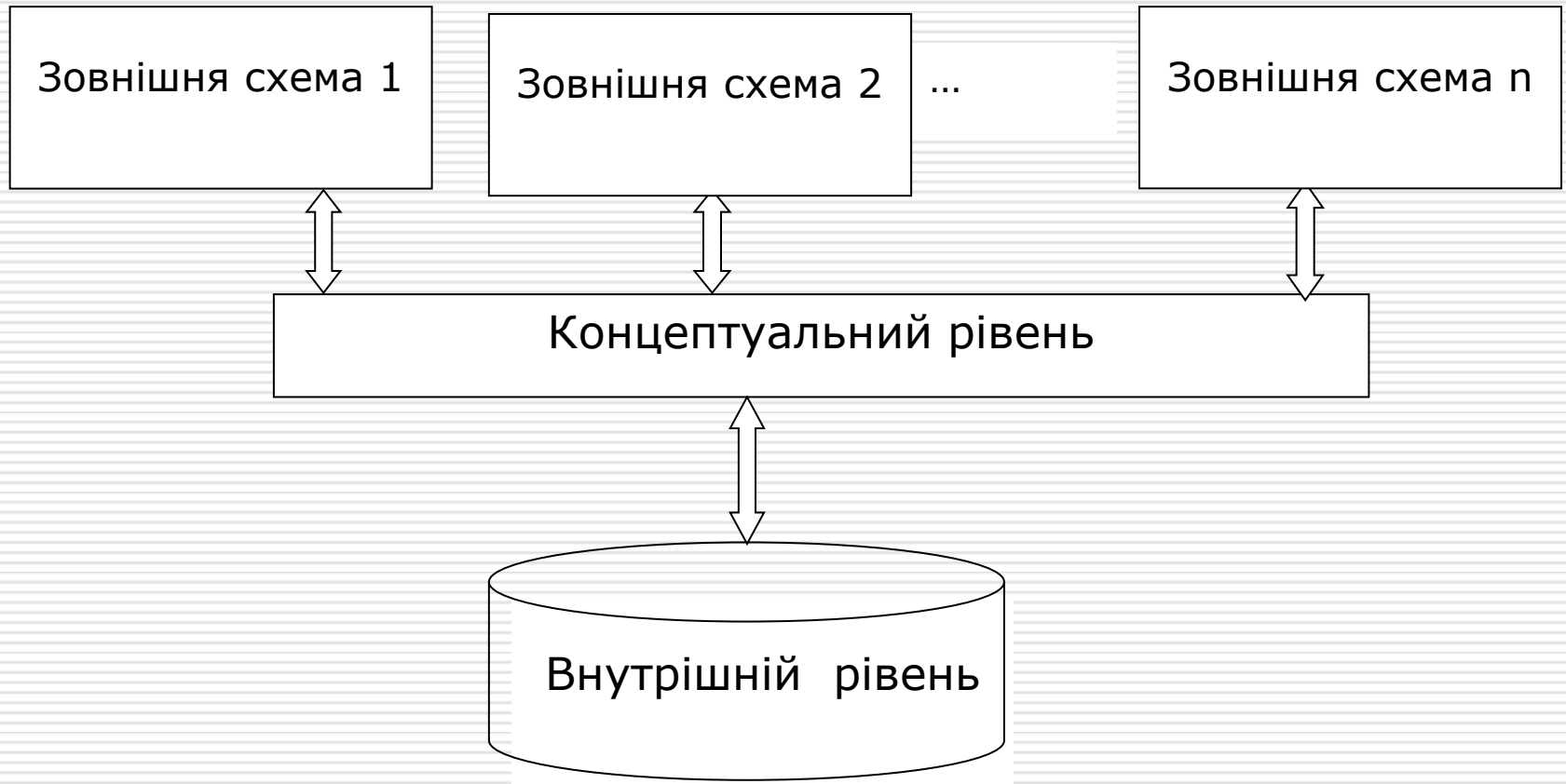
- ❑ утиліти імпортування та утиліти експортування бази даних;
 - ❑ засоби моніторингу, які слідкують за функціонуванням та використанням бази даних;
 - ❑ програми статистичного аналізу, які дозволяють оцінити продуктивність системи;
 - ❑ інструменти збору сміття та перерозподілу пам'яті.
-

Трирівнева архітектура організації бази даних

Вперше специфікована в 1975 році дослідницькою групою ANSI/X3/SPARC

- ❑ ANSI (American National Standard Institute) – Національний Інститут Стандартизації США,
 - ❑ X3 – його комітет обчислювальної техніки та обробки інформації,
 - ❑ SPARC (Standards Planning and Requirements Committee) – підкомітет ANSI/X3 з планування стандартів
-

Трирівнева архітектура ANSI-SPARC



Зовнішній рівень (external level)

- описує ту частину бази даних, яка відноситься до кожного користувача
 - складається з декількох різних зовнішніх схем (представлень) бази даних
 - зовнішні схеми можуть по-різному відображати одні і ті ж дані
-

Концептуальний рівень (conceptual level)

- ❑ здійснюється інтегрований опис предметної області
 - ❑ містить логічну структуру всієї бази даних
 - опис усіх сутностей, їх атрибутів і зв'язків;
 - підтримку цілісності даних;
 - обмеження, які накладаються на дані;
 - семантичну інформацію про дані;
 - ❑ підтримує кожну зовнішню схему
 - ❑ об'єднує дані, які використовуються усіма додатками, що працюють з базою даних
 - ❑ не містить ніяких відомостей про методи зберігання даних
-

Внутрішній рівень (internal level)

- ❑ описує фізичну реалізацію бази даних з урахуванням способів зберігання даних і методів доступу до них
 - ❑ містить наступну інформацію:
 - розподіл дискового простору для зберігання даних та індексів;
 - відомості про розміщення записів;
 - інформація про міри безпеки даних;
 - відомості про стиснення даних та методи їх шифрування.
-

Відображення

«концептуальний – зовнішній» і
«концептуальний – внутрішній»
дають змогу вирішувати проблему
логічної та фізичної незалежності
даних:

*будь-які зміни на одному з рівнів не
повинні спричиняти зміни на інших
рівнях, а лише має змінитися
відповідне відображення*

Логічна незалежність даних

- повна захищеність зовнішніх схем від змін, що вносяться в концептуальну схему
 - можливість змінювати логічне представлення бази даних без необхідності змінювати фізичні структури зберігання даних
-

Фізична незалежність

- ❑ захищеність концептуальної схеми від змін, що вносяться у внутрішню схему
 - ❑ зміни помітні лише в загальній продуктивності системи
-

Сучасна трирівнева архітектура бази даних

1. Рівень **представлення інформації** забезпечує інтерфейс з користувачем (людиною або програмою)
 2. Рівень обробки даних (рівень **бізнес-логіки**) визначає функціональність і працездатність системи в цілому
 3. Рівень **зберігання даних** забезпечує фізичне зберігання, додавання, модифікацію і вибірку даних; перевіряється цілісність і узгодженість даних; реалізація транзакцій
-

Архітектури програмно-технічних засобів

- ☐ файл-сервер
 - ☐ клієнт-сервер з бізнес-логікою на клієнті
 - ☐ клієнт-сервер з бізнес-логікою на сервері
 - ☐ N-рівнева архітектура
-

Відповідність логічної архітектури БД і програмно-технічної архітектури

Тип архітектури	Файл-сервер	Клієнт-сервер (бізнес-логіка на клієнті)	Клієнт-сервер (бізнес-логіка на сервері)	N- рівнева архітектура
<i>Представлення інформації</i>	Клієнт	Клієнт	Клієнт	Клієнт
<i>Бізнес-логіки</i>	Клієнт	Клієнт	Сервер БД	Сервер прикладних програм (комп. кластер)
<i>Зберігання даних</i>	Файл-сервер (або клієнт)	Сервер БД	Сервер БД	Сервер БД
Реалізація	Усі три рівні утворюють єдиний програмний модуль	Інтерфейс користувача і бізнес логіка утворюють єдиний модуль. Дані зберігаються на сервері	Усі бізнес-логіка реалізована у вигляді збережених процедур, які виконуються на сервері БД	Усі рівні реалізовані на різних машинах

Архітектура файл-сервер

- ❑ усі рівні системи представляють єдине і неподільне ціле
 - ❑ БД зберігається у вигляді файлу або набору файлів на файл-сервері
 - ❑ уся логіка вибірки, зберігання і забезпечення узгодженості даних покладається на клієнтську частину
 - ❑ обробка інформації ведеться на рівні окремих кортежів (записів)
-

Архітектура файл-сервер

□ **Переваги:**

- простота логіки;
- низькі вимоги до апаратного забезпечення і малий об'єм необхідної пам'яті;
- не вимагає надійних багатозадачних і багатокористувацьких ОС;
- невисока ціна СУБД.

□ **Недоліки**

- обмеженість мови і негнучкість середовища розробки додатків;
 - слабка масштабованість;
 - не забезпечує багатокористувацький режим роботи;
 - важко підтримувати цілісність і узгодженість даних;
 - необхідно вручну блокувати записи або таблиці;
 - низький рівень захищеності як зовнішньої (від зламу), так і внутрішньої (від програмних помилок), наприклад, індекси окремо від таблиць;
 - не має засобів шифрації мережевого трафіку;
 - створює велике навантаження на мережу.
-

Архітектура клієнт-сервер з бізнес-логікою на клієнті

- ❑ зберігання та вибірка та підтримка узгодженості даних покладається на сервер БД
 - ❑ вся бізнес-логіка і логіка представлення інформації виконуються на клієнтських машинах
 - ❑ продуктивність і збереженість даних залежить від сервера БД
 - ❑ клієнтська частина обмінюється даними з сервером за допомогою запитів SQL
 - ❑ обробка інформації ведеться на рівні множини кортежів (записів)
 - ❑ процес розробки розділяється на створення БД і написання клієнтської частини з бізнес-логікою
-

Архітектура клієнт-сервер з бізнес-логікою на клієнті

□ **Переваги:**

- висока продуктивність, стабільність і надійність при багатокористувацькому режимі роботи;
- легко організовується захист даних (шифрування мережевого трафіку SSH, SSL);
- універсальність мови визначення та маніпулювання даними.

□ **Недоліки**

- більша ціна СУБД (сервер БД продається окремо);
 - достатньо високі вимоги до кваліфікації розробників;
 - необхідні навички адміністрування сервера БД;
 - підвищені вимоги до пропускної здатності мережі;
 - підвищені вимоги до клієнтських місць (на них виконується рівень бізнес-логіки).
-

Архітектура клієнт-сервер з бізнес-логікою на сервері

- на сервер переноситься максимально можлива частина бізнес-логіки
 - можливість сучасних серверів БД виконувати збережені процедури на сервері
-

Архітектура клієнт-сервер з бізнес-логікою на сервері

□ **Переваги:**

- знижені, порівняно з попереднім класом систем, вимоги до пропускної здатності мережі та до клієнтських машин;
- простіший процес створення бізнес-логіки.

□ **Недоліки**

- підвищені вимоги до сервера БД, оскільки кожний сеанс «з'їдає» пам'ять з розрахунком граничного навантаження;
 - невисока мобільність системи на інші сервери БД.
-

N- рівнева архітектура

- ❑ основними елементами є сервери БД, сервер (кластер) прикладних програм і клієнтська частина
 - ❑ максимальне спрощення клієнта і сервера БД
 - ❑ тонкий клієнт являє собою деякий термінал типу браузера
 - ❑ уся бізнес-логіка оформляється у вигляді набору прикладних програм, які запускаються на сервері-кластері
 - ❑ сервери БД займаються лише задачами зберігання, додавання, модифікації та підтримки узгодженості даних
 - ❑ сервер програм з'єднаний з сервером БД окремим високошвидкісним сегментом мережі
-

N- рівнева архітектура

□ Переваги:

- підвищена захищеність;
- висока продуктивність;
- гнучкість розвитку та модифікації;
- простота адміністрування;
- можливість створення системи з масовим паралелізмом, оскільки сервери БД можуть бути різними, а сервером програм можуть служити декілька з'єднаних в кластер комп'ютерів.

□ Недоліки

- велика складність архітектури;
 - висока ціна рішення;
 - у деяких випадках поступається в продуктивності клієнт-серверним системам з бізнес-логікою на сервері.
-

Визначення даних і моделей даних

- ❑ «**Дані**» в концепції баз даних – це набір конкретних значень, параметрів, які характеризують об'єкт, умову, ситуацію або інші фактори.
 - ❑ Модель даних – це представлення «реальних» об'єктів, подій та існуючих між ними зв'язків. Це деяка абстракція, яка застосовується до певних даних, і в якій акцент робиться на найважливіших аспектах, а всі другорядні властивості ігноруються.
 - ❑ «**Модель даних**» в концепції баз даних – це інтегрований набір понять для опису даних, зв'язків між ними та обмежень, які накладаються на дані, в деякій інформаційній системі.
-

Класифікація 1 моделей даних

- ❑ об'єктні (object-based) моделі даних - опис даних на концептуальному та зовнішньому рівнях
 - ❑ моделі даних на основі записів (record-based) - опис даних на концептуальному та зовнішньому рівнях
 - ❑ фізичні моделі даних - опис даних на внутрішньому рівні
-

Класифікація 1 моделей даних.

Об'єктні (object-based) моделі даних

- використовуються такі поняття як сутності, атрибути і зв'язки.

Загальні типи об'єктних моделей даних:

- ☐ модель типу „сутність-зв'язок“, або ER-модель (Entity-Relationship model);
 - ☐ семантична модель;
 - ☐ функціональна модель;
 - ☐ об'єктно-орієнтована модель.
-

Класифікація 1 моделей даних.

Моделі даних на основі записів (record-based)

- База даних складається з декількох записів фіксованого формату, які можуть мати різні типи. Кожен тип запису визначає фіксовану кількість полів, кожне з яких має фіксовану довжину.

Загальні типи логічних моделей даних на основі записів:

- ☐ реляційна модель даних (relational data model);
 - ☐ мережева модель даних (network data model);
 - ☐ ієрархічна модель даних (hierarchical data model).
-

Класифікація 1 моделей даних.

Моделі даних на основі записів (record-based)

- **Реляційна модель даних** базується на понятті математичних відношень. Єдина вимога в реляційній моделі даних – це щоб база даних з точки зору користувача виглядала як набір таблиць, зв'язаних відношеннями. Однак це відноситься тільки до логічної структури бази даних, тобто до зовнішнього та концептуального рівня архітектури ANSI-SPARC. Дана вимога не відноситься до фізичної структури бази даних, яка може бути реалізована за допомогою різних структур зберігання.
 - В **ієрархічній моделі** дані представляються у вигляді деревовидної (ієрархічної) структури. Подібна організація даних є зручною для роботи з ієрархічно впорядкованою інформацією. Однак, при оперуванні складними логічними зв'язками ієрархічна модель стає дуже громіздкою.
 - В **мережевій моделі** дані організовуються у вигляді довільного графа. На відміну від реляційної моделі, зв'язки тут моделюються наборами, які реалізуються за допомогою вказівників. Недоліком мережевої моделі є жорсткість структури і складність її організації
-

Класифікація 1 моделей даних.

Фізична модель даних

- оперує категоріями, які відносяться до організації зовнішньої пам'яті та структур зберігання даних
 - використовуються різні методи розміщення даних, що базуються на файлових структурах:
 - організація файлів прямого та послідовного доступу;
 - індексні файли;
 - інвертовані файли;
 - файли, які використовують різні методи хешування;
 - взаємозв'язані файли
 - сторінкова організація даних (сучасні СУБД). Фізичні моделі, що базуються **на сторінковій організації** є найбільш перспективними.
-

Класифікація 2 моделей даних

- інфологічні (семантичні) моделі
 - даталогічні моделі
 - фізичні моделі
-

Класифікація 2 моделей даних.

Інфологічні (семантичні) моделі

- виражають інформацію про предметну область у вигляді, який не залежить від вибраної СУБД
- відображають інформаційно-логічний рівень абстрагування, який пов'язаний з фіксацією та описом об'єктів ПО, їх властивостей і зв'язків у зручній для розробників та інших користувачів формі
- використовуються на ранніх стадіях проектування БД

Найпоширеніші типи інфологічної моделі:

- діаграми Бахмана
 - модель „сутність-зв'язок” (*ER-модель*)
-

Класифікація 2 моделей даних. Даталогічні моделі

- підтримуються конкретною СУБД

Типи даталогічних моделей:

- ☐ документальні моделі
 - ☐ фактографічні моделі
-

Класифікація 2 моделей даних. Даталогічні → Документальні моделі

відповідають відображенню
слабоструктурованої інформації,
орієнтованої на вільні формати
документів та тексти на первинній мові

Типи документальних моделей:

- ☐ *дескрипторні моделі*
 - ☐ *моделі, орієнтовані на формат документу*
 - ☐ *тезаурусні моделі*
-

Класифікація 2 моделей даних. Даталогічні → Документальні → ***Дескрипторні*** моделі

- Найпростіші з документальних моделей, широко використовувались на ранніх стадіях використання документальних баз даних.
 - Кожному документу відповідає дескриптор. Він має жорстку структуру і описує документ у відповідності з тими характеристиками, які потрібні для роботи з документальною БД.
-

Класифікація 2 моделей даних.

Даталогічні → Документальні → Моделі,
орієнтовані на формат документу

пов'язані зі стандартними мовами розмітки:

- ❑ HTML: у якості елементу гіпертекстової бази даних, яка описується HTML, використовується текстовий файл, який може легко передаватись по мережі з використанням протоколу HTTP.
 - ❑ XML описує цілий клас об'єктів даних, що називаються XML-документами. Вони використовуються у якості засобу для контролю за правильністю складання документів.
-

Класифікація 2 моделей даних.
Даталогічні → Документальні →
Тезаурусні моделі

- ❑ основані на принципі організації словників
 - ❑ містять певні мовні конструкції і принципи їх взаємодії у заданій граматиці
 - ❑ ефективно використовуються в системах-перекладачах, в яких за тезаурусними моделями реалізовано принцип зберігання інформації
-

Класифікація 2 моделей даних. Даталогічні → Фактографічні моделі

- теоретико-графові моделі
 - теоретико-множинні моделі
 - об'єктно-орієнтована модель
-

Класифікація 2 моделей даних.
Даталогічні → Фактографічні →
Теоретико-графові моделі

- відображають сукупність об'єктів реального світу у вигляді графа взаємозв'язаних інформаційних об'єктів.

В залежності від типу графа виділяють:

- ☐ ієрархічну модель
 - ☐ мережеву модель
-

Класифікація 2 моделей даних. Даталогічні → Фактографічні → ***Теоретико-множинні*** моделі

- перехід від роботи з елементами даних (графові моделі) до роботи з макрооб'єктами

Основні моделі:

- ☐ реляційна модель
 - ☐ модель бінарних асоціацій
-

Класифікація 2 моделей даних.
Даталогічні → Фактографічні →
Об'єктно-орієнтована модель

Розширяє визначення сутності з метою включення в нього не тільки атрибутів, які описують стан об'єкта, а й дій, які з ним пов'язані, тобто його поведінку. В такому випадку говорять, що об'єкт інкапсулює стан та поведінку .

Дякую за увагу
