

Інститут спеціального зв'язку та захисту інформації НТУУ
«КПІ»

Спеціальна кафедра № 5

ЛЕКЦІЯ

з навчальної дисципліни
“Організація баз даних та знань -1”

Тема 4. Розподілені та паралельні бази даних.

Лекція 4/1. Архітектура розподілених баз даних.

Київ 2016

Лекція 4/1. Архітектура розподілених баз даних.

Учбові питання лекції:

Вступ.

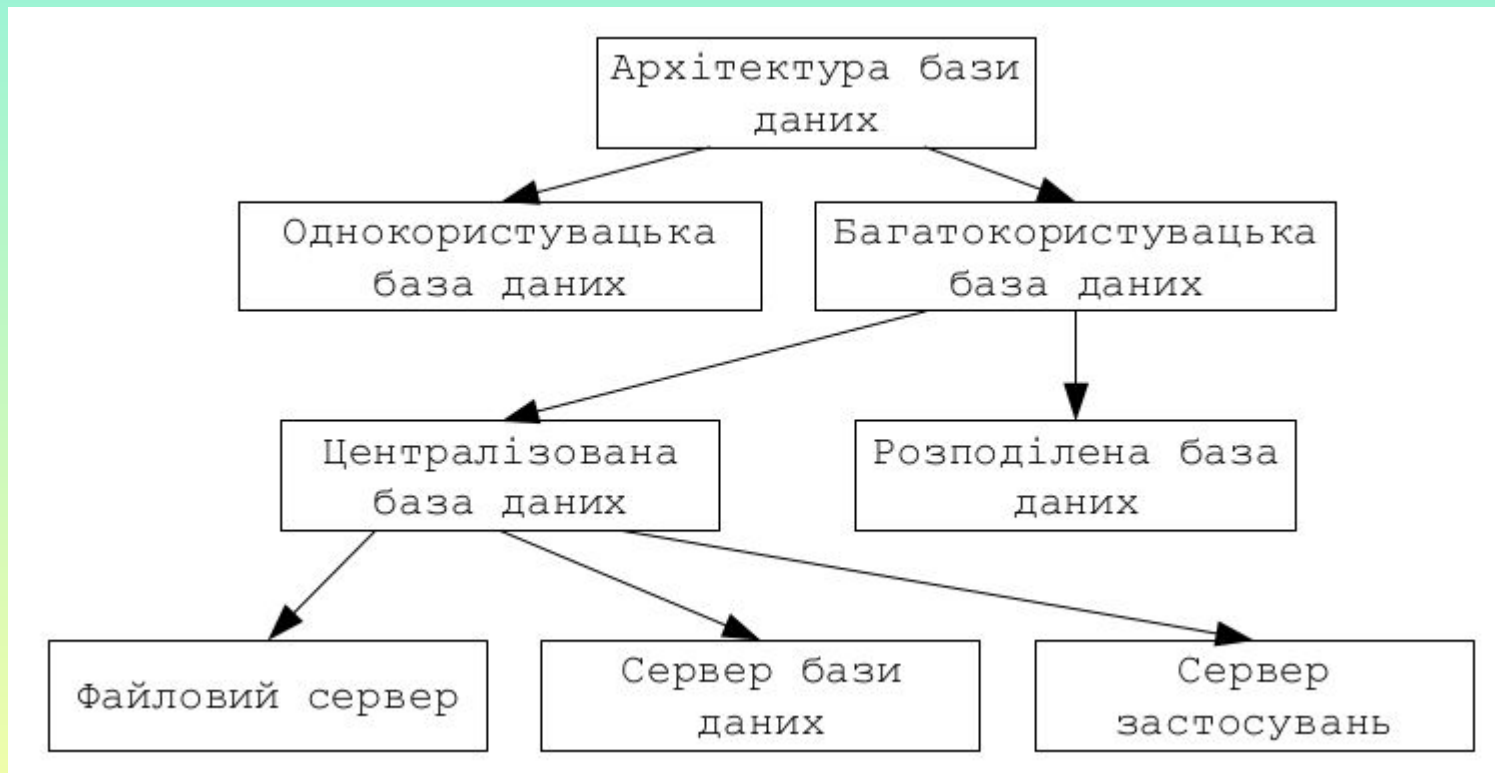
1. Класифікація розподілених БД.
2. Архітектура розподілених БД.
3. Обробка розподілених запитів.

Література:

1. Пасічник В.В., Резніченко В.А. Організація баз даних та знань. – К.: Видавнича група ВНУ, 2006. – 384 с.
2. Малыгина М.П. Базы данных: основы, проектирование, использование. 2-е изд. перераб. – СПб.: БХВ-Петербург, 2006. – 528 с.
3. Базы данных: лабораторных практикум. [Электронный ресурс]. Составитель И.Е. Агапов. – Воронеж, 2005. – 91 с.
4. Андон Ф., Резниченко В. Язык запросов SQL. Учебный курс. – СПб.: Питер; Киев: Издательская группа ВНУ, 2006. – 416 с.: ил.
5. Зубрилина Т.В., Юрьев В.Н. Базы данных. Проектирование реляционных баз и хранилищ данных с использованием CASE-технологий: Учеб. пособие. СПб.: Изд-во Политехи, ун-та, 2007. 44с.
6. Маклаков СВ. VPwin и ERwin. CASE-средства разработки информационных систем. М.: Диалог - МИФИ, 2000.
7. Новожилова М.В., Резнікова С.Ю., Інформаційна безпека систем управління базами даних: Навчальний посібник. – Харків:ХДТУБА, 2004. – 185 с.
8. Барригера Т.А., Харченко В.Ф. Базы данных и информационные системы. – СПб:

Режими роботи з базою даних

Якщо з БД працюють одночасно декілька користувачів, то в цьому випадку СУБД повинна забезпечувати коректну паралельну роботу всіх користувачів над одними і тими ж даними. Розрізняють розподілену обробку і розподілені БД.



Основні поняття і визначення

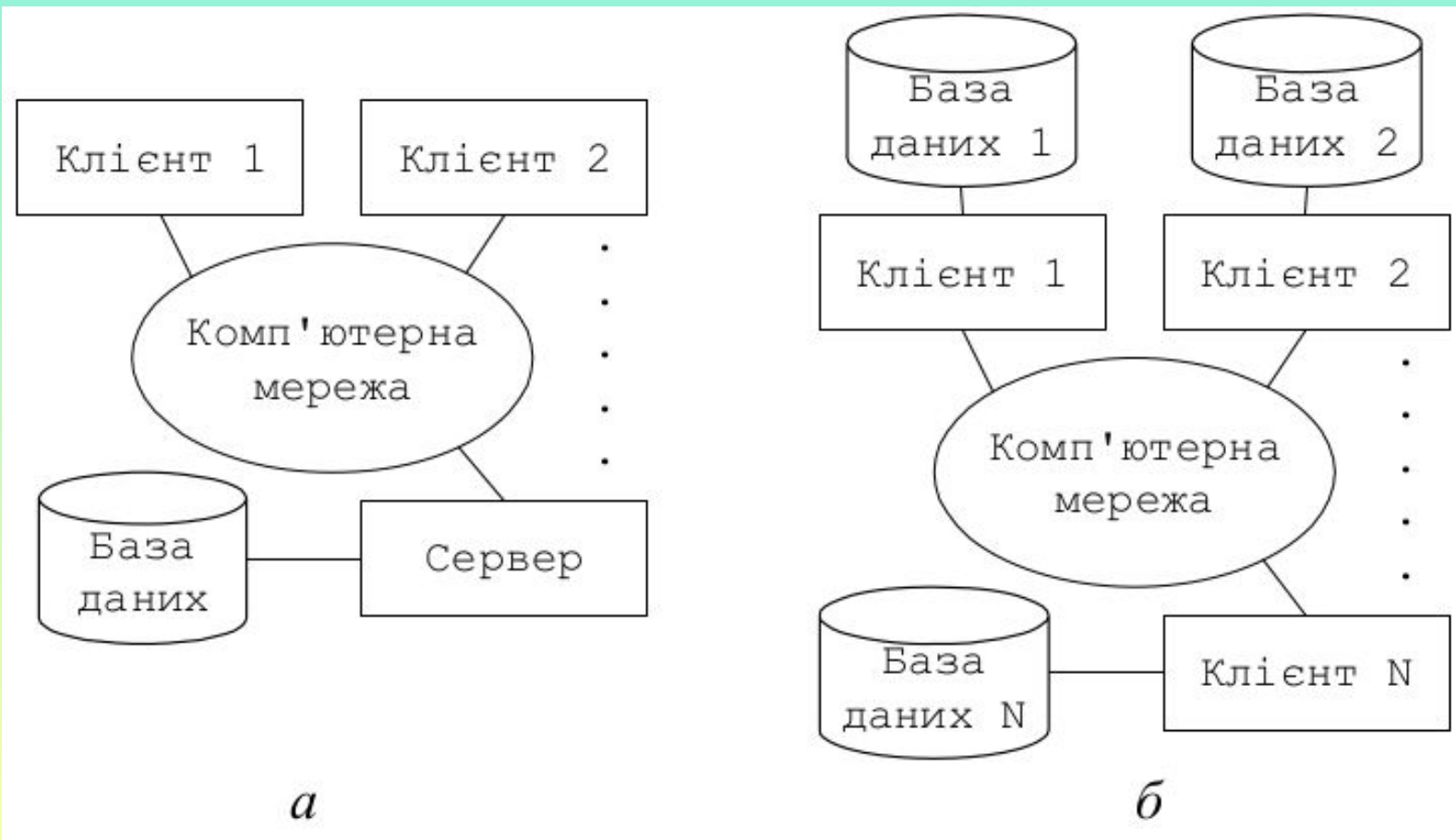
Розподілена обробка – це обробка з використанням централізованої бази даних, доступ до якої може виконуватись з різних комп'ютерів мережі. Ця топологія часто називається "клієнт-сервер". В цій системі одні вузли – клієнти, а інші – сервери.

Сервер – комп'ютер, який надає деякі послуги іншим комп'ютерам, обмін повідомленнями з якими здійснюється за допомогою мережі, що їх з'єднує. Послуги

Структура інформаційної СИСТЕМИ

а – розподілена обробка.

б – розподілена база даних.



Розподілена система

Якщо всі вузли розподіленої системи використовують той самий тип СУБД, то така система називається **гомогенною**.

Якщо вузли розподіленої системи використовують різні типи СУБД, які обробляють різні моделі даних, то така система називається **гетерогенною**.

Управління паралельною обробкою

В багатокористувацьких системах з БД одночасно можуть працювати декілька користувачів або прикладних програм. Для збереження цілісності даних і забезпечення безпеки БД в цих умовах застосовуються транзакції, які забезпечують роботу кожного користувача з узгодженим станом БД.

Транзакція – неподільна з точки зору впливу на БД послідовність операторів маніпулювання даними, яка

Транзакція

Властивості транзакції означають таке:

- **атомарність** означає, що транзакція виконується, як єдина операція доступу до БД і виконується або повністю або не виконується зовсім;
- **узгодженість** гарантує взаємну цілісність даних, тобто виконання обмежень цілісності БД після завершення роботи транзакції;
- **ізолюваність** означає, що транзакції,

Керування паралелізмом

Наступні конфліктні ситуації, які можуть призвести до одержання неправильного результату через взаємні перешкоди між деякими транзакціями, можливі при відсутності відповідного керування (слід зазначити, що транзакція, яка вносить перешкоду, сама по собі може бути правильної):

- проблема втрати результатів відновлення;
- проблема незафіксованої залежності;
- проблема несумісного аналізу.

Проблема втрати результатів відновлення

Розглянемо ситуацію транзакція А витягає деякий кортеж p у момент часу $t1$; транзакція В витягає деякий кортеж у момент часу $t2$; транзакція А обновляє деякий кортеж p (на основі значень, отриманих у момент часу $t1$) у момент часу $t3$; транзакція В обновляє той же кортеж p (на основі значень, отриманих у момент часу $t2$, які мають ті ж значення, що й у момент часу $t1$) у момент часу $t4$. Однак результат операції відновлення, виконаною транзакцією А буде загублений, оскільки в момент часу $t4$ вона не буде врахована й тому буде «скасована» операцією відновлення, виконаною транзакцією В.

Транзакція А	Час	Транзакція В
-		-
Добування кортежу p	$t1$	-
-		-
	$t2$	Добування кортежу p
-		-
Відновлення кортежу p	$t3$	-
-		-
	$t4$	Відновлення кортежу p

Проблема незафіксованої залежності

Проблема незафіксованої залежності з'являється, якщо за допомогою деякої транзакції здійснюється добування (або, що ще гірше, відновлення) деякого кортежу, який у цей момент обновляється іншою транзакцією, але це відновлення ще не закінчене. У такому випадку в першій транзакції будуть брати участь дані, які більше не існують.

Транзакція А	Час	Транзакція В
-		-
	t1	Відновлення кортежу р
-		-
Добування кортежу р	t2	
-		-
	t3	Скасування виконання транзакції
-		-

Транзакція А стає залежною від невиконаної зміни в момент часу t2

Проблема незафіксованої залежності

У прикладі транзакція А в момент часу t_2 зустрічається з невиконаним відновленням. Потім це відновлення відміняється в момент часу t_3 . Таким чином, транзакція А виконується на основі фальшивого припущення, що кортеж r має деяке значення в момент – часу t_2 , тоді як насправді він має деяке значення, що існувало ще в момент часу t_1 . У підсумку після виконання транзакції А буде попікований невірний результат

Проблема несумісного аналізу

Транзакція А	Час	Транзакція В
-		-
Добування РАХУНОК sum = 40	кортежу 1 t1	-
Добування РАХУНОК sum = 90	кортежу 2 t2	-
	t3	Добування РАХУНОК 3 кортежу
	t4	Відновлення РАХУНОК 30 → 20 кортежу 3
-	t5	Добування РАХУНОК 1 кортежу
-	t6	Відновлення РАХУНОК 40 → 50 кортежу 1
	t7	Завершення <u>виц.</u> Транзакції
Добування РАХУНОК sum = 110 (а не 120)	кортежу 3 t8	
	↓	

Проблема несумісного аналізу

Показані транзакції А і В, які виконуються для кортежів з рахунками (РАХУНОК). При цьому транзакція А підсумує баланси, транзакція В робить переклад суми 10 з рахунку 3 на рахунок 1. Отриманий у підсумку транзакції результат 110, мабуть, невірний, і якщо він буде записаний у базі, то в ній може виникнути проблема несумісності.

У цьому випадку говорять, що був виконаний несумісний запит. Слід

Обробка паралельних транзакцій

Для обробки паралельних транзакцій застосовується метод тимчасових міток і серіалізація транзакцій.

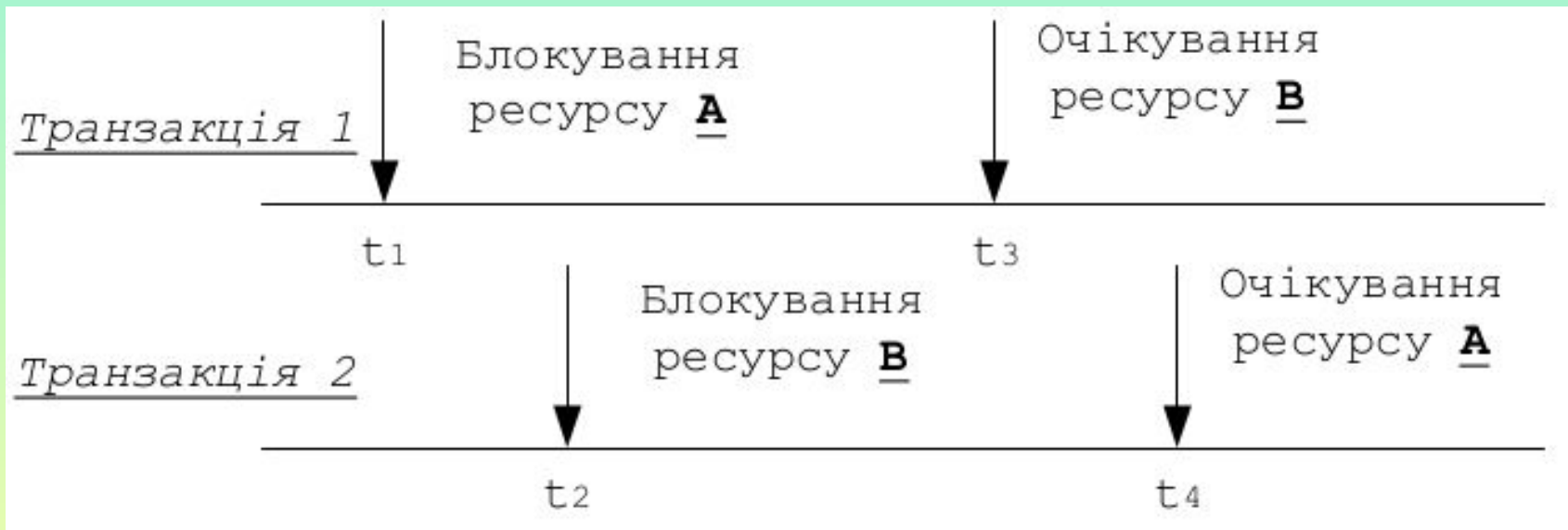
Блокування передбачає встановлення режиму доступу (монопольного або сумісного) до деякого ресурсу даних, що дозволяє виключити доступ до нього одночасно з даною транзакцією інших транзакцій, в результаті якого може бути порушена логічна цілісність даних БД.

Види блокування

- **нежорстке** блокування або роздільне блокування (Shared – S-блокування); об'єкт блокується для виконання операції читання; об'єкти в цьому випадку не змінюються у ході виконання транзакції і доступні іншим транзакціям також, але тільки в режимі читання;
- **жорстке** блокування або монопольне (eXclusive – X-блокування); об'єкт блокується для виконання операції запису, модифікації або вилучення. В цьому випадку

Взаємне блокування транзакцій

Застосування різних типів блокувань призводить до тупиків. Тупикова ситуація виникає тоді, коли дві і більш транзакції одночасно знаходяться у стані очікування, причому для продовження роботи кожна з транзакцій очікує завершення роботи іншої транзакції.



Блокування

Основою визначення тупикових ситуацій є побудова графа очікування транзакцій. Алгоритм виходу із тупика передбачає визначення транзакції-жертви. Після вибору такої транзакції виконується її відкат.

Для серіалізації транзакцій також застосовується двофазне блокування, яке полягає у такому:

– перед виконанням операцій з будь-яким

Сумісність для X- і S-Блокування

1. Припустимо, що в системі підтримується два типи блокувань: блокування без взаємного доступу (монопольне блокування), називана X-Блокуванням, і блокування із взаємним доступом, називана S-Блокуванням. Іноді ці блокування називають блокуваннями запису й читання відповідно.
2. Якщо транзакція А блокує кортеж р без можливості взаємного доступу, то запит іншої транзакції В с блокуванням цього кортежу р буде відмінний.
3. Якщо транзакція А блокує кортеж р з можливістю взаємного доступу, то
 - запит з боку деякої транзакції В на X-Блокування кортежу буде відкинутий;
 - запит з боку деякої транзакції В на S-Блокування кортежу р буде прийнятий (тобто транзакція В також буде блокувати кортеж р за допомогою S-Блокування).

	X	S	-
X	N	N	Y
S	N	Y	Y
-	Y	Y	Y

Символи S і X позначають блокування, а прочерк – відсутність блокування. N позначає конфліктну ситуацію (запит з боку іншої транзакції не може бути вдоволений, а сама ця транзакція переходить у стан очікування), а Y – повну сумісність (запит задовольняється).

Протокол доступу до даних

1. Транзакція, призначена для добування кортежу, насамперед, повинна накласти S-Блокування на цей кортеж.
2. Транзакція, призначена для відновлення кортежу, насамперед повинна накласти X-Блокування на цей кортеж.
3. Якщо запитуване блокування з боку транзакції У відкидається через конфлікт із деяким іншим блокуванням з боку

Серіалізація транзакцій

З керуванням транзакціями в багатокористувацькій СУБД зв'язані важливі поняття серіалізації (здатності до впорядкування) транзакцій і серіального плану виконання суміші транзакцій. Під серіалізацією транзакцій, що паралельно виконуються, розуміється такий порядок планування їх роботи, при якому сумарний ефект суміші транзакцій еквівалентний ефекту їх деякого послідовного виконання. Серіальний план виконання суміші транзакцій - це такий план, який приводить до серіалізації транзакцій. Зрозуміло, що якщо вдається добитися дійсно серіального виконання суміші транзакцій, то для кожного користувача, з ініціативи якого утворена транзакція, присутність інших транзакцій буде непомітно (якщо не вважати деякої затримки роботи з порівняння з однокористувацьким режимом).

Така здатність до впорядкування є загально визнаним критерієм правильності керування паралельною обробкою кортежів. Точніше кажучи виконання, що чергується, заданого безлічі транзакцій буде вірним, якщо воно впорядковане. Обґрунтованість цього твердження впливає з наступних зауважень.

1. Окремі транзакції вважаються вірними, якщо при їхньому виконанні БД переходить із одного несуперечливого стану в інший несуперечливий стан.
2. Отже, виконання транзакцій одна за іншою в будь-якому послідовному порядку (використовуються незалежні друг від друга транзакції) також є вірним.
3. виконання, що чергується, транзакцій, отже, є вірним, якщо воно еквівалентно деякому послідовному виконанню, тобто якщо воно підлягає упорядочиванню.

Для заданого набору транзакцій будь-який порядок їх виконання називається *графіком запуску*. Два графіка називаються **еквівалентними**, якщо при їхньому виконанні буде отриманий однаковий результат.

Завдання для самостійної роботи:

1. Повторити матеріал лекції.
2. Повторити основні елементи БД: алфавіт, лексика, синтаксис, семантика, прагматика.
3. Підготуватися до вхідного контролю у формі контрольної роботи по варіантах.