

АНАЛІЗ НАЙАКТУАЛЬНІШИХ СЕРВЕРНИХ СИСТЕМ УПРАВЛІННЯ БАЗАМИ ДАНИХ

© Коваленко В.В., Гуменюк В.О., 2007

Розглянуті ознаки, за якими можна аналізувати переваги й недоліки систем управління базами даних: продуктивність, масштабованість, доступність даних, функціональні можливості сервера, відкритість СУБД, наявність засобів розробки й адміністрування найактуальніших СУБД, *DB2, Oracle, Microsoft SQL Server, Informix*.

This article focuses on the criteria allowing to analyze the advantages and disadvantages of database management systems. The productivity, the scalability, the accessibility of data, the server functional capabilities, the data manager openness and the most relevant data management engineering tools and administration are analyzed.

Вступ. Сьогодні відомо більше двох десятків серверних систем управління базами даних. Найпопулярнішими є *DB2, Oracle, Microsoft SQL Server, Informix*.

Відомості про виробників перерахованих вище СУБД наведені в табл. 1.

Таблиця 1

Виробники СУБД

СУБД	Виробник	Url
Oracle	Oracle Corp.	http://www.oracle.com/
Microsoft SQL Server	Microsoft	http://www.microsoft.com/
Informix	Informix	http://www.informix.com/
DB2	IBM	http://www-4.ibm.com/

Для порівняння ефективності найвідоміших СУБД, розглянемо їх на такій безлічі критеріїв оцінки:

- масштабованість;
- продуктивність;
- доступність даних;
- функціональні можливості сервера;
- відкритість;
- засоби розробки.

Масштабованість

Термін "масштабованість" означає можливість роботи на найширшому спектрі апаратного забезпечення – від портативних комп'ютерів до серверів з масовим паралелізмом (*MPP*). Чим краща масштабованість, тим ширші можливості з нарощування системи в міру зростання потреб [1].

До поліпшень системних характеристик шляхом збільшення доступних обчислювальних ресурсів (кількості або швидкодії процесорів, числа дисків) можна зарахувати, наприклад:

- ріст числа обслуговуваних користувачів зі збереженням середнього часу відгуку;
- прискорення обробки одного запиту;
- збереження того ж часу обробки запиту при збільшенні обсягу таблиць [2].

Масштабованість СУБД *Informix* забезпечується за рахунок багатопотокової архітектури з підтримкою багатопроцесорної обробки (обслуговування клієнтів рівномірно розподіляється між всіма ресурсами, що є в наявності) і технології *PDQ* (*Parallel Data Query* – паралельне оброблення запитів). Обробка великих таблиць прискорюється пропорційно числу фрагментів, які розміщені на різних дискових пристроях. Дуже зручна можливість динамічної зміни обсягу й конфігурації ресурсів, які використовує сервер – числа віртуальних процесорів, дискових просторів баз даних. Відповідно до наявності ресурсів і потреб можна оперативнo регулювати інтенсивність паралельної обробки, змінювати правила фрагментації таблиць. Продуктивність інформаційної системи може збільшуватися шляхом розподілу даних й їхньої обробки між декількома серверами, зв'язаними мережею [8].

DB2 Universal Database може працювати як на портативному комп'ютері, що підтримує мобільного користувача, так і на машині з масовим паралелізмом, що підтримує трохи терабайт даних і тисячі користувачів; крім того, база даних, природно, здатна підтримувати різні конфігурації симетричних мультипроцесорних систем (*SMP*) і кластерів *SMP*. Важливо відзначити, що масштабованість означає збереження як у старших, так й у молодших версіях тих самих об'єктно-реляційних функцій. Завдяки цьому *DB2 UDB* ідеально підходить для зростаючих малих і середніх організацій; для великих підприємств, яким необхідно розгорнути дво- або трирівневі додатки, що охоплюють масштаби настільних систем, масштаби відділів і масштаби всього підприємства; а також для *Internet-провайдерів* і ділових партнерів, що обслуговують цих замовників.

В склад *DB2 UDB* для забезпечення масштабованості входять такі засоби:

- паралельна обробка на симетричних мультипроцесорних системах (*SMP*);
- підтримка кластерів і систем з масовим паралелізмом (*MPP*) [5].

Одна із загально визнаних переваг сервера *Oracle* – його високий ступінь масштабованості, як "горизонтальної", так й "вертикальної". Сервер *Oracle* у будь-якій конфігурації підтримує паралелізм під час виконання потоку операцій (він архітектурно спроектований для цього), в *SMP-архітектурі* для паралельного виконання окремих запитів потрібна інсталяція *Parallel Query Option*. Для кластерів й *MPP-систем Oracle* пропонує архітектуру, що дозволяє всім вузлам цих систем паралельно здійснювати доступ до однієї БД: щоб домогтися цього, досить установити *Parallel Server Option* [7].

Microsoft SQL Server забезпечує достатньо високі рівні масштабованості й доступності. Однак у цьому разі тут відсутня масштабованість сервера "у чистому вигляді", оскільки ріст його продуктивності залежить не тільки від апаратних засобів, але й операційного середовища, під керуванням якої працює СУБД [10].

Продуктивність

Кожна фірма-розроблювач ПО скаже, що продуктивність у її продукції висока. Але в кожному конкретному випадку продуктивність є результатом дії деяких оптимізуючих механізмів, індивідуальних для кожної СУБД. До особливостей архітектури *Informix DS*, які впливають на продуктивність інформаційної системи належать:

- багатопотокова архітектура;
- паралельна обробка;
- фрагментація таблиць й індексів;
- оптимізація виконання запитів;
- поділювана пам'ять;
- кеши словників даних і збережених процедур;
- власне керування дисковою пам'яттю;
- асинхронне введення/виведення;
- випереджальне читання [6].

До найістотніших параметрів *DB2 Universal Database*, що впливають на продуктивність СУБД, можна зарахувати:

- Асинхронне очищення сторінок. Можливість перекласти операції записування буферизованих сторінок із завдання виконання *SQL*-запиту на яке-небудь інше завдання дозволяє значно

прискорити відгук системи на запити. Асинхронне очищення сторінок забезпечує наявність достатнього вільного простору в буферах бази даних для обробки запиту. Ця функція дозволяє уникнути очікування синхронного записування модифікованих сторінок з буфера на диск для звільнення місця під дані запиту при його обробці.

- Послідовну вибірку, що попереджає. Швидкість виконання запитів, що вимагають великої кількості послідовних операцій введення/виведення на диск, може бути значно підвищена на випадок одночасного виконання операцій введення/виведення й обробки даних центральним процесором.

- Вибірку списку, що попереджає, яка дозволяє підвищити продуктивність обробки запитів, що звертаються до даних випадково або непослідовно.

- Розташування табличних областей (*tablespaces*) на декількох носіях. В UDB адміністратори можуть розділити базу даних на частини. При створенні таблиці можна визначити імена базової (*base*), індексної (*index*) і довгої (*long*) області таблиці. Використання індексних і довгих табличних областей дозволяє зберігати індекси й більші об'єкти *LOB* окремо від інших табличних даних, завдяки чому підвищується продуктивність СУБД. Одна таблична область може розташовуватися на декількох фізичних носіях.

- Безпосередній доступ до носіїв (робота із пристроями прямо). *UDB* дозволяє безпосередньо працювати з даними на пристрої, не витрачаючи ресурси на використання файлової системи, що підвищує продуктивність бази даних і цілісність інформації. Адміністратор має можливість визначити для табличної області режим безпосередньої роботи із пристроєм або використати звичайну файловою систему.

- Читання великих блоків. Функція дозволяє зчитувати за одну операцію введення/виведення декілька дискових сторінок, що скорочує навантаження на центральний процесор й, відповідно, покращує час відгуку на запити. Композитне секціонування забезпечує чудову керованість і високий коефіцієнт готовності, властивому ранговому секціонуванню, разом з перевагами розподілу даних, характерними для хеш-секціонування. Всі методики секціонування є прозорими для додатків, а стандартні оператори *DML* (*Data Manipulation Language*) виконуються для секціонування таблиць. Оптимізатор підготовлений до роботи із секціонованими таблицями, і розділами, які не містять ніяких стосовних до запиту даних, усуваються з області пошуку, що найчастіше приводить до істотного збільшення продуктивності [5].

Oracle 9i забезпечує *API* механізм прямого завантаження. Це забезпечує шлях для *ISV* і партнерів з розробки засобів керування системою для створення простих у використанні й швидких інструментальних засобів завантаження даних. В *Oracle 9i* включена безліч нових функцій. Розглянемо деякі з них. *Cache Fusion* – це технологія управління паралельним кешем, що координує ресурси бази даних у випадку одночасного доступу користувачів з різних вузлів кластера до тих самих даних. На основі технології *Cache Fusion* реалізована опція корпоративної редакції сервера – *Real Application Clusters*. Переваги полягають у тому, що немає необхідності використовувати жорсткий диск для передачі сторінок даних, оптимізується навантаження на процесор за рахунок зведення до мінімуму операції перемикавання контексту операційної системи, повністю використовуються можливості сучасних високошвидкісних міжвузлових шин кластера. Для створення резервних центрів *Oracle 9i* включає спеціальне рішення-*Oracle 9i DataGuard*. Припустимо, що в організації є основний комп'ютер або кластерний комплекс в одному будинку ще один комп'ютер, можливо, менш потужний, в іншому будинку (навіть на значній відстані). На ньому встановлена копія основної бази даних, він зайнятий тільки тим, що постійно синхронізує свою копію БД із основної. Як тільки основний комплекс виходить із ладу, запасний сервер переходить у робочий режим, його база даних стає доступна для роботи користувачам, які автоматично перемикаються на неї, при цьому копія БД повністю ідентична основній БД. Якщо з основним комп'ютером відбувся збій, то через деякий час можна зробити зворотне перемикавання й знову повернутися до роботи на основному комп'ютері. Всі операції з налаштування керування такою системою з резервуванням повністю автоматизовані. Системний адміністратор просто повинен віддати команду на перемикавання, а у разі збою це відбудеться автоматично [4].

Що стосується продуктивності СУБД *Microsoft SQL Server*, то, крім можливостей індексації, паралельного й розподіленого виконання запитів, у ній реалізовані такі механізми, як процесор запитів і продуктивне/інтелектуальне введення/виведення (*Big/Smart I/O*). Розглянемо їх докладніше [3].

Процесор запитів забезпечує обробку команд мовою *Transact-SQL-діалекту* стандартної мови *SQL* стосовно *SQL Server*. Найважливішими можливостями є:

- Використання декількох індексів на таблицю. Відсутність обмеження "одна таблиця – один індекс" дозволяє оптимізатору користуватися декількома індексами, наприклад, якщо умова пошуку в запиті задана одночасно за декількома полями. Над індексами можуть здійснюватися теоретико-множинні операції, наприклад, об'єднання або перетинання індексів, що спрощує обробку предикатів фільтрації з операторами *or* або *and*, а також може застосовуватися для динамічного створення покриваючого індексу.

- Поряд із традиційним алгоритмом дозволу з'єднання таблиць (*JOIN*) – вкладеним циклом (*nested loop*) – оптимізатор може застосовувати найчастіше ефективніші стратегії – злиття (*merge join*) і хеширування (*hash join*). Злиття застосовується, коли обидві, що з'єднуються, таблиці відсортовані по ключу з'єднання. Хеширування застосовується в тому випадку, якщо індекси задіяти не вдається.

- Процесор запитів звертається за даними до системи зберігання (*Storage Engine*) через інтерфейс *OLE DB*. Через цей же інтерфейс він може звертатися за даними й до будь-якої іншої *OLE DB-сумісним* джерелам даних, як локальним (що перебуває на цьому ж комп'ютері), так і вилученим. Отже, стандартні оператори *SELECT*, *INSERT*, *UPDATE* й *DELETE* можуть тепер в одному запиті з'єднувати таблиці з різних джерел даних. Цими джерелами даних можуть бути як *Microsoft SQL Server*, так й інші СУБД, а також нереляційні джерела, такі як *Exchange Server* або *Index Server*.

- Повнотекстовий пошук, забезпечується за рахунок інтеграції із системою повнотекстового індексування, з якої СУБД взаємодіє через *OLE DB*. Керування побудовою й підтримкою індексів здійснюється з головного засобу адміністрування *SQL Server* – *SQL Enterprise Manager*. Повнотекстові індекси зберігаються за межами баз даних *SQL Server*, у спеціально відведених файлових каталогах. Відновлення індексів виробляється вручну або автоматично, за розкладом. Побудова індекса можлива по символічних і текстових полях таблиць. В *SQL Server* також поліпшене керування кеширування, що підвищує продуктивність при читанні великої кількості даних [10].

Доступність даних

Якщо сервер зупинений для обслуговування або просто відбувся збій у роботі системи, користувачі можуть виявитися "відрізнаними" від потрібної їм інформації. Сервер *Informix DS* володіє можливостями, які дозволяють підвищити надійність *IC* і практично відмовитися від планових простоювань:

- віддзеркалювання дискових областей;
- відновлення некритичних для роботи сервера даних в оперативному режимі;
- повне тиражування даних сервера;
- розвинені засоби збереження даних;
- фрагментація таблиць (у разі відмови одного диска зберігається часткова доступність таблиці);
- інструменти спостереження за станом сервера;
- виконання більшості адміністративних завдань в оперативному режимі [9].

В *DB2 Universal Database* підтримуються різні засоби на рівні операційної системи, такі як *HASMP* для *AIX*, кластери *Sun Solaris* й *Wolfpack* для *NT/2000*. Адміністратори можуть виконувати резервне копіювання й відновлення на рівні окремих табличних областей, а не на рівні всієї бази даних. Якщо в табличній області підтримується єдина таблиця, то резервне копіювання й відновлення цієї області еквівалентно резервному копіюванню й відновленню на рівні таблиці. При резервному копіюванні табличні області можуть перебувати як в активному (*on-line*), так й у пасивному (відімкненому) режимі. При відновленні всі табличні області, за винятком відновлювальної, залишаються в режимі *on-line*. У разі помилки введення/виведення (відмови диска,

наприклад) від'єднаними виявляються тільки порушені табличні області (ті, що очікували відновлення). Інша база даних зберігає працездатність [5].

В *Microsoft SQL Server*, на жаль, деякі операції адміністрування рекомендується виконувати в режимі, коли користується один користувач. Отже, час від часу користувачі не будуть мати доступу до інформації в базі даних. Правда, такі простоювання можна звести до мінімуму, однак факт залишається фактом [3].

Функціональні можливості сервера

Можливості СУБД *Informix DS*, крім розглянутих вище, передбачають такі засоби: процедури, які збережені, тригери, курсори, каскадне видалення даних, підтримку цілісності, рівні ізоляції читання (брудне читання, підтвержене читання, стабільне читання, читання, що повторюється) [6].

В *DB2 Universal Database* підтримуються такі ключові об'єктно-реляційні функції (всі вони реалізовані відповідно до стандартів *SQL3* і тому являють собою відкритий, а не внутрішньо-корпоративний підхід).

- Обумовлені користувачем типи даних (*UDT*). За їхньою допомогою користувачі можуть створювати нові типи даних, які будуть представлені в базі з використанням вбудованих типів.
- Обумовлені користувачем функції (*UDF*). За їхньою допомогою в запити можна включати потужні обчислювальні предикати й предикати пошуку для фільтрації непотрібних даних безпосередньо у їхнього джерела.
- Великі об'єкти (*LOB*). За допомогою об'єктів *LOB* користувачі можуть зберігати в базі даних дуже великі двійкові або текстові об'єкти (розміром у декілька гігабайт).
- Обумовлені користувачем табличні функції (*Table UDF*). Користувачі *SQL* можуть тепер звертатися до даних, що зберігаються не в реляційному форматі, і при тому повною мірою використати всі можливості побудови запитів реляційної бази даних;
- Обумовлені користувачем функції *OLE (OLE UDF)*. Технологія *OLE (Object Linking and Embedding* – зв'язування й впровадження об'єктів) входить в архітектуру *Microsoft* для *Windows*. *DB2* надає підтримку контролера *OLE* для доступу до даних сервера *OLE* за допомогою обумовлених користувачем функцій *UDF*. За допомогою цих зовнішніх функцій *UDF* дані із серверів *OLE* можуть передаватися через *DB2* у запити *SQL* [5].

Oracle має велику кількість функціональних можливостей, таких як досить розвинута мова програмування *PL/SQL*, і механізми тригерів, збережених процедур і функцій тощо [4].

До складу *Microsoft SQL Server* входить досить потужна мова роботи з даними *Transact-SQL*, що є розширенням стандартного *SQL*. Однак сумісність зі стандартом *ANSI/ISO SQL-92* не є повною, хоча він і розглядається як кращий діалект *SQL*. *Transact-SQL* підтримує такі об'єкти БД, як збережені процедури, тригери, підтримка цілісності тощо. На жаль, відсутні механізми каскадного видалення й відновлення даних по зовнішніх ключах [3].

Відкритість

Цей термін включає оцінки по багатьох напрямках. Ступінь відкритості визначає ступінь інтегрування СУБД і продуктів, створених на її основі, у різноманітних апаратних, програмних, адміністративних, національних й інших середовищах, що надзвичайно важливо як для побудови *IC* як у теперішній час, так і для її розвитку в майбутньому. Перелічимо деякі властивості, що характеризують відкритість *Informix*:

- доступність на безлічі платформ, включаючи *Sequent, HP, Sun, IBM, Siemens Nixdorf, NCR*;
- підтримка, крім *UNIX*, операційних систем *Windows NT* й *NetWare*;
- можливість включення баз даних *Informix* у розподілені різноманітні *IC*, побудовані на основі апаратно-програмних платформ і СУБД різних виробників;
- інтегрування *Informix* із системами централізованого керування й адміністрування, такими як *Tivoli Management Environment (TME), HP OpenView, IBM NetView*;
- багатомовна підтримка [8].

Доступ до баз даних *DB2 Universal Database* можна одержати практично із будь-якого клієнта за допомогою всіх основних типів мереж. Крім того, *DB2 Universal Database* підтримує більшість галузевих стандартів, що дозволяє працювати із системою, використовуючи безліч існуючих інструментальних засобів і додатків.

Сервери *DB2 Universal Database* і шлюзи *DB2 Connect* можуть працювати на таких платформах: *AIX, Linux, HP-UX, OS/2, Solaris, Windows NT, Windows 2000*. [5]

Відкритість *MS SQL Server*, як і масштабованість, досить відносна. СУБД *SQL Server* "відкрита" для інших продуктів *Microsoft: MS Office, MS Visual Studio, MS Internet Information Server* і т.д. При цьому, однак, їхній виробник не без підстав затверджує, що оскільки всі ці програмні комплекси розроблені однією компанією, то взаємодія між ними здійснюється ефективніше, ніж з аналогічними продуктами інших фірм [3].

Засоби розробки

Засоби розробки й доступу для користувача в *Informix*, особливо об'єктно-орієнтований інструментарій групової розробки прикладних систем із графічним інтерфейсом *Informix NewEra*, оцінюються експертами як високорозвинені інструменти, що відповідають сучасним вимогам. Крім того, *Informix* підтримується багатьма інструментальними системами незалежних виробників [6].

В *DB2 Personal Developer's Edition* й *DB2 Universal Developer's Edition* програмістам надається середовище розробки, що дозволяє створювати для баз даних додатки, що звертаються до даних і керуючі ними у реляційних СУБД *IBM*. У версію *DB2* входять інструментальні засоби для розроблювачів, документація й приклади вихідного коду додатків на платформах робочих станцій, підтримуваних серверами *DB2 Universal Database*. *DB2 Personal Developer's Edition* містить у собі все необхідне для розробки додатків на платформах *Linux, OS/2* й *Windows*, зокрема набори інструментальних засобів розроблювача (*Software Developer's Kit, SDK*).

DB2 Universal Database є однією з найпростіших у використанні й керуванні СУБД свого класу. Вона містить у собі повний набір графічних інструментальних засобів, які задовольняють потреби адміністраторів баз даних і прикладних програмістів. Крім того, до її складу входять інструментальні засоби, які допомагають виконувати разові запити й створювати звіти для користувачів [5].

Засоби розробки *Oracle* представлені такими продуктами, як *Developer, Designer, Developer Server, JDeveloper, WebDB, Oracle Reports*. Розглянемо кожний із цих продуктів.

При розробці додатків на *Oracle Developer* користувач може:

- створювати форми й звіти, використовуючи будь-які БД: *SQL Server, Informix* й *DB/2*;
- перекладати додатки більш ніж на 40 різних мов;
- переносити додатки в *Web*, використовуючи стандарти *Java* й *HTML*;
- запускати додатки в *Windows* й *UNIX*.

Для розв'язання аналітичних завдань, пов'язаних зі складними розрахунками, прогнозуванням, моделюванням сценаріїв "якщо" тощо, застосовується особлива технологія багатомірних баз даних. Вона реалізована у вигляді сімейства *OLAP-продуктів Oracle Express*, що включає сервер багатомірних БД, спеціальні засоби розробки в середовищі клієнт-сервер й *Web*, а також готові аналітичні додатки.

Oracle Express Server – це система керування багатомірними базами даних або сервер багатомірних даних, який забезпечує зберігання великих обсягів інформації, підтримує багатомірну модель даних, яка найефективніше відбиває подання кінцевих користувачів про предметну область. Для динамічного доступу до реляційних баз даних використовується модуль *Express Relational Access Manager*, що реалізує довільні схеми зберігання даних – *ROLAP* (реляційний *OLAP*), *MOLAP* (багатомірний *OLAP*) або *HOLAP* (гібридний *OLAP*).

Express Server вже довгий час лідує в тестах продуктивності *OLAP-серверів (ABP-1)*, демонструючи при цьому найвищу масштабованість за обсягом кількості користувачів. *Oracle Express Server* легко інтегрується в загальну архітектуру інформаційної мережі організації, надаючи зручні кошти зв'язку з існуючими реляційними базами даних інших джерел через локальну або

вилучену мережу. *Oracle Express Server* працює на всіх популярних обчислювальних платформах від персональних комп'ютерів до *UNIX-серверів* великих машин.

Oracle Express Analyzer – засіб, призначений для самостійної побудови звітів, аналізу багатомірних даних *Express*, публікації результатів на *Web*. Всі одержувані інтерактивні таблиці, діаграми, графіки розташовуються на сторінках, які можна об'єднати у багатосторінкові книги-брифінги. Результати аналізу легко можна відіслати по електронній пошті або експортувати на корпоративний *Web-сайт*.

Отже, *Express Analyzer* включає в себе простий зручний доступ до даних з можливістю обміну результатами аналізу, підвищуючи ефективність спільної роботи користувачів.

Oracle Express Objects – об'єктно-орієнтоване середовище швидкої розробки *OLAP-додатків* у середовищі клієнт/сервер для багатомірних баз даних *Oracle Express*. Воно призначене для професійних розроблювачів, забезпечує високу гнучкість. *Express Objects* дозволяє створювати багатомірні додатки будь-якого ступеня складності з використанням як стандартних візуальних елементів *Windows*, так і спеціалізованих інтерфейсних об'єктів для доступу і маніпуляції багатомірними даними *Express* (наприклад, таблиця, графік або селектор) [4].

MS SQL Server також надає широкі можливості розробки баз даних. До складу цієї СУБД входять такі засоби розробки:

- *SQL Server Query Analyzer*. Використовується для розробки й оптимізації запитів, а також дозволяє дивитись докладний план виконання запиту. План відображається в графічному вигляді за допомогою піктограм і спливаючих текстових підказок. Конструкції мови *SQL* у тексті запиту виділяються різними квітами на основі синтаксичного аналізу. Показуються відносні (у відсотках) вартості виконання етапів, а також окремих запитів стосовно загальної вартості пакета. Результати запиту можуть бути представлені в табличній формі. *Query Analyzer* може також видати рекомендації з побудови індексів, які оптимізують виконання цього запиту.

- *Visual Database Tools*. Цей набір графічних інструментальних засобів призначений для малювання діаграм "сутності-зв'язку" і розробки складних запитів. За допомогою *Microsoft Visual Database Tools* можна:

- приєднати й досліджувати будь-яку базу даних, сумісну з *ODBC (Open DataBase Connectivity)*;
- створювати й змінювати бази даних, використовуючи діаграми;
- розробляти, виконувати й зберігати складні запити;
- додавати, змінювати й знищувати дані, що зберігаються в таблицях БД;
- проектувати такі об'єкти, як таблиці, тригери, збережені процедури для систем керування базами даних *Microsoft SQL Server* й *Oracle*;

- "перетягувати" об'єкти баз даних на проєктовані носії інтерфейсів (наприклад, на виготовлені *HTML-сторінки*) і зв'язувати їх з елементами керування засобами візуального проєктування [3].

Варто зазначити, що *SQL Server* поступається іншим розглянутим СУБД по двох важливих показниках: програмованість і засоби розробки. При розробці клієнтських додатків на *Java*, *HTML-програмних* засобів *SQL Server* часто виявляється недостатньо, і в цьому сенсі користуватися цією СУБД складніше, ніж *DB2*, *Informix* або *Oracle*. [9]

На підставі аналізу розглянутих критеріїв оцінки СУБД можна скласти табл. 2 ефективності СУБД із виділенням переваг і недоліків.

Таблиця 2

Ефективність СУБД із виділенням переваг і недоліків

СУБД	Переваги	Недоліки
1	2	3
<i>Informix Dynamic Server</i>	Широкий набір архітектурних особливостей, що забезпечують високу продуктивність, внутрішні механізми підтримки масштабованості, багаті можливості вбудованого мови маніпулювання даними, підтримка декількох апаратних платформ й операційних систем	Чим більше можливостей, тим більше тонкощів у налаштуванні сервера для того або іншого завдання. Адміністрування системи вимагає високого рівня професіоналізму. Набагато складніше функціонал і сама робота в <i>Informix DS</i> , ніж в <i>MS SQL Server</i>

1	2	3
MS SQL Server	Ідеально підходить для операційної системи <i>Windows</i> . Крім багатого набору програмних засобів для розробки й адміністрування, привабливість серверу надає наявність високоінтелектуального процесора запитів і добре розвиненого діалекту мови <i>SQL (Transact-SQL)</i>	Такі важливі параметри СУБД, як продуктивність і масштабованість, залежать від операційного середовища – <i>Windows</i> . Сервер орієнтований головним чином на інтеграцію з іншими продуктами <i>Microsoft</i> ; можливостей інтеграції із програмним забезпеченням інших виробників набагато менше. В основі СУБД лежить принципова відмова від підтримки операційних систем, що не належать сімейству <i>Windows</i>
DB2 Universal Database	Висока продуктивність, багаті можливості масштабування, графічний інтерфейс розроблювача й адміністратора баз даних, багатоплатформеність; підтримка об'єктно-реляційної парадигми й стандарту <i>SQL3</i>	На відміну від <i>MS SQL Server</i> , дуже складний і незручний інтерфейс і функціонал
Oracle 9i	Орієнтація на <i>Internet</i> , підтримка великої кількості апаратних і програмних платформ; багаті можливості для розроблювачів (об'єктно-реляційна БД, <i>PL/SQL</i> і т.д.)	Відносно висока ціна, тяжке адміністрування; широкі можливості сервера вимагають високу кваліфікацію розроблювачів й адміністраторів

Висновки. На підставі аналізу можна зробити такі висновки, що майже всі серверні СУБД мають можливість:

- реалізації на декількох платформах;
- використання зручних адміністративних утиліт;
- здійснення резервного копіювання даних;
- підтримки декількох сценаріїв реплікацій;
- підтримки паралельної обробки даних у багатопроцесорних системах;
- підтримки *OLAP* і створення сховищ даних;
- виконання розподілених запитів і транзакцій;
- використання різних засобів проектування даних для створення своїх об'єктів;
- підтримки засобів розробки й генераторів звітів як власного виробництва, так й інших виробників;
- підтримки як мінімум публікації даних в *Internet*.

Внаслідок проведеного аналізу можна стверджувати, що використання *MS SQL Server* ідеально підходить для операційної системи *Windows*, має у наявності високоінтелектуальний процесор запитів і добре розвинений діалект мови *SQL (Transact-SQL)*. *Oracle 9i* орієнтується на *Internet*, підтримує велику кількість апаратних і програмних платформ, має багаті можливості для розроблювачів (об'єктно-реляційна БД, *PL/SQL* тощо). СУБД *Informix* здатна обслуговувати одночасно працюючі додатки оперативної обробки транзакцій і системи підтримки прийняття рішень для локальних і розподілених баз даних з великою кількістю користувачів. *DB2 Universal Database* поєднує в собі високу продуктивність систем обробки транзакцій у режимі *on-line*, об'єктно-реляційні розширення, удосконалені засоби оптимізації з можливостями паралельної обробки й підтримкою дуже великих баз даних.

Існуючі можливості серверних СУБД відбивають сучасні тенденції розвитку інформаційних систем, такі, як використання багатопроцесорних систем і розподіленої обробки даних, створення розподілених систем, зокрема з використанням технологій *Internet*, застосування засобів швидкої розробки додатків, створення систем підтримки прийняття рішень із використанням аналітичної обробки даних, а також усе більше підвищуються вимоги до надійності інформаційних систем.

1. Корнєєв В.В., Гарєєв А.Ф., Васютін С.В., Райх В.В. Базы даних. Интеллектуальная обработка информации // *Нолідж*. – 2006. – С. 97–127. 2. Крейг С. Маллінс Адміністрування баз даних: Повний довідковий посібник з методів і процедур // *Database Administration. The Complete Guide to Practices and Procedures* // КУДІЦ-Образ. – 2003. – С. 79–115. 3. Поль Дюбуа *MySQL* // Вільямс. – 2006. – С. 45–70. 4. Кэри Мілсап Джейфф Холт *Oracle. Оптимізація продуктивності* // Символ-Плюс. – 2006 – С. 129–143. 5. Джордж Бакларц, Білл Вонг *DB2 для UNIX, Linux, Windows й OS/2* // *DB2 Universal Database v7.1 for UNIX, Linux, Windows and OS/2* // Лорі. – 2004. – С. 113–125. 6. Грачьев А.Ю. Введення в СУБД *Informix* // *Діалог-МІФІ*. – 2005. – С.87 – 99. 7. <http://www.sql.ru>. 8. <http://www.informix.ru>. 9. <http://www.citforum.ru>. 10. <http://www.oracle.ru>.