

АНАЛІЗ СИСТЕМ НАКОПИЧЕННЯ ТА ОБРОБКИ ДАНИХ ТУРИЗМУ

© Угрин Д., 2008

Проаналізовано проблеми опрацювання даних туризму та ефективність їх використання в засобах накопичення та обробки інформації.

In the article the analysis of problem of working of information of tourism is examined that efficiency of their use in facilities of accumulation and treatment of information.

Вступ

Зміни в навколишньому світі, посилення впливу інформаційних процесів на суспільство сприяють розвитку інформаційного середовища. Актуальність цього питання особливо зростає в період вступу світу до нової стадії цивілізації, вирішальну роль в якій відграють інформаційні засоби накопичення інформації та її обробки, суттєво змінюючи усі сфери життєдіяльності.

Сьогодні всі туристичні організації накопичують або вже нагромадили в процесі своєї діяльності великі обсяги даних. Ці колекції даних зберігають у собі розвинуті потенційні можливості щодо видобування нової аналітичної інформації, на основі якої можна й необхідно будувати стратегію фірми, виявляти тенденції розвитку ринку, знаходити нові рішення, що обумовлюють успішний розвиток в умовах конкурентної боротьби. Для деяких туристичних організацій такий аналіз є невід'ємною частиною їхньої повсякденної діяльності, але більшість, мабуть, тільки починає звертати на нього увагу.

1. Актуальність роботи

Із інтенсивним розвитком туристичного бізнесу та інформаційних технологій, які все більше почали застосовуватися в цій галузі, виникли й проблеми із використанням систем для обробки баз даних (БД).

Оскільки на початку розвитку туристичного бізнесу провідну роль відігравали системи управління базами даних (СУБД), вони більшою мірою могли задовольнити кожен туристичну фірму. Але прогресуючому підприємству туристичної сфери базами даних обійтись ставало все важче. Спроби будувати системи прийняття рішень, які зверталися б безпосередньо до баз даних систем оперативної обробки транзакцій (OLTP-систем), виявилися переважно невдалими. По-перше, аналітичні запити «конкурують» з оперативними транзакціями, блокуючи дані й викликаючи недостачу ресурсів. По-друге, структура оперативних даних призначена для ефективної підтримки коротких і частих транзакцій і в силу цього занадто складна для розуміння кінцевими користувачами й, крім того, не забезпечує необхідної швидкості виконання аналітичних запитів. По-третє, в організації, як правило, функціонує кілька оперативних систем, кожна зі своєю базою даних. У цих базах використовуються різні структури даних, одиниці виміру, способи кодування і т.д. Для кінцевого користувача (аналітика) задача побудови якого-небудь зведеного запиту за декількома подібними до баз даних практично нерозв'язна.

Розгляд проблеми опрацювання даних допоможе вирішити питання, яку систему краще використовувати для підвищення достовірності та надійності інформації, а, отже, підвищить ефективність рішень, вироблених системою.

2. Аналіз літературних джерел та постановка задачі

Проблема опрацювання даних у реляційних базах даних, системах прийняття рішень тощо постала ще у 70-ті роки минулого століття та поширювалася на різні об'єкти дослідження. Для того,

щоб забезпечити можливість аналізу накопичених даних, організації стали створювати сховища даних, які являють собою інтегровані колекції даних, зібрані з різних систем оперативного доступу до даних. Сховища даних стають основою для побудови систем прийняття рішень. Незважаючи на розходження в підходах і реалізаціях, всім сховищам даних властиві такі загальні риси:

Предметна орієнтованість. Інформація в сховищі даних організована відповідно до основних аспектів діяльності організації (турист, замовлення, тури тощо); це відрізняє сховище даних від оперативної БД, де дані організовані відповідно до процесів (виписка рахунків, сервіс туристичного закладу, оформлення рейсів й т.п.). Предметна організація даних у сховищі сприяє як значному спрощенню аналізу, так і підвищенню швидкості виконання аналітичних запитів. Полягає вона, зокрема, у використанні інших, ніж в оперативних системах, схемах організації даних. У випадку зберігання даних у реляційній СУБД застосовується схема «зірки» (star) або «сніжинки» (snowflake). Крім того, дані можуть зберігатися в спеціальній багатомірній СУБД – в n-мірних кубах.

Інтегрованість. Вихідні дані витягаються з оперативних БД, перевіряються, очищуються, приводяться до єдиного вигляду, у потрібному ступені агрегуються (тобто обчислюються сумарні показники) і завантажуються до сховища. Такі інтегровані дані набагато простіше аналізувати.

Прив'язка до часу. Дані в сховищі завжди прямо пов'язані з певним періодом часу. Дані, вибрані їх оперативними БД, накопичуються у сховищі у вигляді „історичних каталогів” відповідно до конкретного періоду часу. Оскільки туристична галузь має властивість швидко розвиватися, то це дає змогу аналізувати тенденції в розвитку туристичного бізнесу.

Незмінюваність. Потрапивши в певний „історичний каталог” сховища, дані вже ніколи не будуть змінені. Це також відрізняє сховище від оперативної БД, у якій дані увесь час змінюються, «дихають», і той самий запит, виконаний двічі з інтервалом у 10 хвилин, може дати різні результати. Стабільність даних також полегшує їхній аналіз.

Вищенаведені особливості були вперше сформульовані в 1992 році засновником сховищ даних Біллом Інмоном (Bill Inmon) у його книзі „Building the Data Warehouse”.

Типовими предметними областями, у яких виникає необхідність роботи з великими нагромадженнями даних на туристичному ринку послуг, є сфери:

- які обліковують фактор часу: туристичні бази з сезонним використанням, готелі, інтернет-сайти;
- які працюють із неструктурованими даними – туристичні агентства;
- діяльність яких пов'язана з ризиком у разі неточної інформації – тур-оператори, туристичні фірми.

3. Основний матеріал

Базу даних можна визначити як сукупність взаємопов'язаних даних (прості чи складені типи), що зберігаються разом на одному носії та описують якусь предметну область за наявності такої мінімальної надмірності, яка допускає їх оптимальне використання для одного або декількох застосувань. Розрізняють ієрархічні, мережеві, реляційні, часові (темпоральні), постреляційні (об'єктно-орієнтовні, з гніздуванням), розподілені та багатомірні бази даних.

Використання бази даних припускає роботу з нею декількох прикладних програм (застосувань), що вирішують завдання різних користувачів.

Логічним продовженням в розвитку баз даних стало сховище даних, яке розширило можливості роботи в інформаційній сфері. Сховище даних – поняття, яке запропоноване і розроблене для забезпечення проблем ефективного доступу до даних з різних джерел (баз даних) [3].

Проте для розробників важливішим є таке визначення.

Сховище даних (СД) – предметно-орієнтована, інтегрована, варіантна в часі, неруйнівна сукупність даних, призначена для підтримки прийняття керівних рішень. Основні вимоги до сховищ даних сформовані у книзі Гаврилової Т.А., Хорошевського В.Ф. «Базы знаний интеллектуальных систем» [4].

Основними компонентами сховища даних є такі:

- оперативні джерела даних;

- засоби проектування/розроблення;
- засоби перенесення й трансформації даних;
- СУБД;
- засоби доступу й аналізу даних;
- засоби адміністрування.

В основу концепції сховища даних покладено дві такі основні ідеї [6].

Інтеграція роз'єднаних деталізованих даних (деталізованих у тому сенсі, що вони описують деякі конкретні факти, властивості, події та ін.) в єдиному сховищі. У процесі інтеграції мають узгоджуватися і, можливо агрегуватися деталізовані дані. Дані можуть надходити з історичних архівів корпорації, оперативних баз даних, зовнішніх джерел.

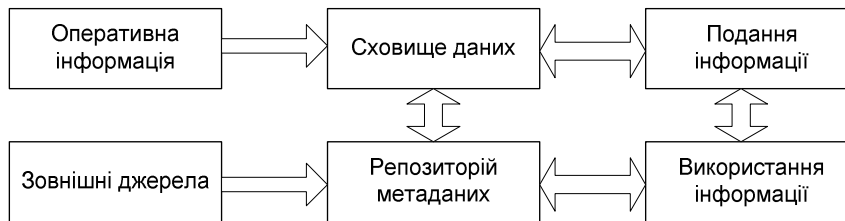


Рис. 1. Концептуальна модель сховища даних

Розподіл наборів даних і застосувань, що використовуються для оперативного опрацювання і для розв'язування задач аналізу, Ральф Кімболл (Ralph Kimball), визначає як сховище даних – «місце, де люди можуть отримати доступ до своїх даних». Він же сформулював і основні вимоги до сховищ даних:

- підтримка високої швидкості отримання даних зі сховища;
- підтримка внутрішньої несуперечності даних;
- можливість отримання і порівняння так званих зрізів даних (slice and dice);
- наявність зручних утиліт перегляду даних у сховищі;
- повнота і достовірність даних, що зберігаються;
- підтримка якісного процесу поповнення даних.

Задовольняти всі перераховані вимоги у межах одного і того самого продукту часто не вдається. Тому для реалізації сховищ даних зазвичай використовується декілька продуктів, одні з яких є власне засобами зберігання даних, інші — засобами їх витягання і перегляду, треті — засобами їх поповнення та ін.

Типове сховище даних, як вже говорилося, зазвичай відрізняється від традиційної реляційної бази даних. По-перше, традиційні бази даних призначені для того, щоб допомогти користувачам виконувати повсякденну роботу, тоді як сховища даних призначені для прийняття рішень. Наприклад, продаж путівки і приєднання до туристичного рейсу здійснюється з використанням бази даних, призначеної для опрацювання транзакцій, а аналіз динаміки путівок за декілька років, що дає змогу спланувати роботу з туристичними операторами, — за допомогою сховища даних.

По-друге, традиційні бази даних характеризуються постійними змінами у процесі роботи користувачів, а сховище даних відносно стабільне: дані у ньому зазвичай оновлюються за розкладом (наприклад, щотижня, щодня або щогодини — залежно від потреб). В ідеалі процес поповнення (або як далі ми будемо його називати — *завантаження*) є просто додаванням нових даних за певний період часу без зміни попередньої інформації, що вже міститься в сховищі.

І, по-третє, традиційні бази даних найчастіше є джерелом даних, що потрапляють у сховище. Крім того, сховище може поповнюватися за рахунок зовнішніх джерел, наприклад, статистичних звітів.

Сховища даних дають змогу краще використовувати дані в туристичному бізнесі. При високих темпах росту даних у цій галузі бази даних не зможуть вмещувати стільки інформації, а також ефективно керувати даними. Сховища даних пристосовані до зберігання та аналітичного опрацювання великих обсягів даних і переважно є інтеграцією реляційної та багатовимірної

моделей (корпоративна інформаційна фабрика Білла Інмона, шина Ральфа Кімбола, зведення даних корпорації TDAN). Вони мають розвинені засоби інтеграції даних з різних джерел та дають змогу працювати як з деталізованою, так і агрегованою інформацією.

Опишемо модель сховища даних [6]:

Сховищем даних (СД) назвемо шістьку

$$DW = \langle DB, rf, RF, rm, RM, func \rangle,$$

де **DB** – множина вхідних баз даних (реляційних, багатовимірних, об'єктно-орієнтованих, ненормалізованих тощо) (або множина відношень, їхніх схем та обмежень цілісності, які містять інформацію з вхідних баз даних), **rf** – множина відношень фактів, **RF** – схема **rf**, **rm** – множина відношень метаданих, **RM** – схема **rm**, **func** – множина процедур прийняття рішень.

Тоді *нові дані* (або *рішення*) – це результат застосування функцій сховища даних над відношенням фактів:

$$Design = func(rf, user_param).$$

де *user_param* – множина параметрів користувача, або вимог, які ставляться до рішення.

Оскільки відношення *rf* містить агреговану інформацію з відношень баз даних, то зв'язок між ним і відношеннями баз даних **DB** приводить до утворення так званого гіперкуба даних (моделі багатовимірного подання даних) – див. розділ 4.

У реляційній моделі гіперкуб даних подається за допомогою розгортання даних у множини відношень вимірів та відношень фактів.

Виміром назвемо множину відношень бази даних $V_i \in Universum(DB_i)$.

Кожен вимір містить напрямки консолідації даних, що складаються із серії послідовних рівнів узагальнення (рівнів ієрархії).

Відношення між вимірами – відношення, яке є зв'язком між певними вимірами та відношенням фактів:

$$V_1 \times V_2 \times \dots \times V_n \times rf \rightarrow rel.$$

На даний момент існують проблеми, які не вирішуються за допомогою баз даних і сховищ даних, хоча вони дають змогу опрацьовувати інтегровані дані, що побудовані на основі наперед допустимих моделей даних. У випадку туристичної сфери наперед не можна визначити, які саме моделі даних використовуватимуться. Тому виключно за допомогою баз даних та сховищ даних не можна організувати ефективної взаємодії між усіма об'єктами цієї предметної області. Крім того, розглядаючи сферу туризму, очевидно, що тут доводиться використовувати кілька моделей даних, що неможливо застосовувати у сховищах даних, оскільки у сховищах даних використовується тільки одна модель. Розробники часто зустрічаються з набором слабо зв'язаних джерел даних, і тому повинні кожного разу вирішувати низькорівневі завдання управління даними, що знову ж таки не дає змоги уніфікувати дані.

Визначимо список питань та ситуацій, які виникають при використанні баз та сховищ даних, що ускладнюють подальшу роботу, а іноді приводять до припинення роботи з даними:

1. Як вирішити питання під час надходження даних з різного роду джерел туристичної сфери та інтегрувати всі тексти, дані, код і потоки для їх подальшої модифікації.

2. Чи можна забезпечити можливості багатоконтрольності даних у системах управління базами даних та у Web-технологіях.

3. Через надходження великих обсягів даних та стрімкий розвиток туризму виникають випадки, що аналіз інформації може затримувати темпи розвитку туризму, а іноді і приводити до банкрутства. Бази та сховища даних не витримують великих обсягів інформації. Якщо їх завантажити до максимуму, то оброблятимуться дані дуже повільно. Схожа ситуація виникає під час аналізу, пошуку та інших важливих операцій.

4. Проблематика в єдиних стандартах опису метаданих та опрацюванні мультимедійної інформації.

5. Як організувати підтримку неточних та невчасних даних та реалізувати неточні запити.

6. Проблеми, які ускладнюють задачу маніпулювання потоками даних.

7. „Гнучкість” у розумінні формування вихідних даних на подану задачу користувача з потоків даних.

8. Проблеми оперативної та складної інтеграції даних, зокрема складноструктуровані.

9. Складності у сприйманні та застосуванні використання природних мов запитів до баз та просторів даних.

10. Не вживана в базах даних та мало розвинута в сховищах даних підтримка систем обробки поточкових даних.

11. Проблема в базах даних із каталогізацією отриманих первинних даних та даних вторинної обробки інформації, зберігання, доступу і модифікації інформації про стан, а також її комбінування з реальними поточковими даними.

12. Складність у застосуванні великої кількості мов для роботи з усіма різновидами даних, що призводить до затримки та конфліктації обробки даних.

Традиційні СУБД подають тільки одну точку (хоч і дуже важливу) в просторі рішень управління даними. Важливою точкою є «системи інтеграції даних». Насправді системи інтеграції даних і обміну даними традиційно призначаються для підтримки багатьох інших осмислених служб в системах просторів даних. Особливість полягає у тому, що в системах інтеграції даних потрібна семантична інтеграція до того, як можуть бути забезпечені які-небудь інші послуги. Тому хоч і відсутня єдина схема, якій відповідають всі дані, система повинна знати точні взаємозв'язки між елементами, що використовуються в кожній схемі. В результаті для створення системи інтеграції даних потрібна значна попередня робота [6].

Розглянемо приклад, який часто зустрічається в туристичній сфері. Кілька потужних туристичних організацій співпрацюють між собою, а отже, вони обмінюються досвідом та своїми програмними розробками. Зрозуміло, що кожна з вищесказаних організацій вже має свою готову модель, каталог тощо. Але вивчивши роботу своїх партнерів, можна запозичити від них щось, що б покращило функціонування туристичного підприємства, відшукати простіші способи аналізу, узагальнення, пошуку інформації. Використовуючи бази чи сховища даних, це питання вирішити неможливо, оскільки у них насамперед вимагається, щоб керування даними та адміністрування відбувалося з одного джерела. По-друге, виникають проблеми з неточними та невчасними даними, які часто зустрічаються в туристичному бізнесі (запізнення авіа- чи авторейсів, неточно вказані дані сервісу на відпочинковій чи оздоровчій базі). До важливих проблем належать також такі, як забезпечення можливості багатоконтрольності даних; інтеграції тексту, даних; підтримки систем обробки поточкових даних тощо.

Простір даних розглядають як нову абстракцію управління даними [16]. Як ключова задача робіт у області управління даними використовується платформа підтримки просторів даних (DataSpace Support Platforms, DSSP). DSSP забезпечує набір взаємозв'язаних послуг і гарантує розробникам можливість концентруватися на специфічних проблемах їх застосувань, а не на завданнях, що повторюються, виникають при потребі узгодженої і ефективної роботи з взаємозв'язаними, але роздільно керованими даними.

Серед вже згаданих проблем, які спричинили створення просторів даних як єдиної предметної області з опрацювання різних джерел даних та використання цих даних, найважливішими є [6, 7]:

1. Інтеграція тексту, даних, коду і потоків (частково вирішена завдяки введенню Коддом 12-ти правил побудов сховищ даних та їх подальшої модифікації).

2. Забезпечення можливості багатоконтрольності даних (у концепції сховищ даних вирішувалася через процедури *витягання, перетворення та завантаження даних* (extract, transform and load – *ETL*), а у випадку Інтернет із тисячами джерел інформації, поданої у різних форматах, ETL не придатна для використання).

3. Створення простих способів аналізу, узагальнення, пошуку й огляду електронних підбірок мультимедійної інформації, зокрема розроблення стандартів опису метаданих (сьогодні немає єдиних стандартів опису та опрацювання мультимедійної інформації).

4. Підтримка неточних та невчасних (тих, що надходять із запізненням або неочікувано) даних та реалізація неточних запитів. Сьогодні достатньо інтенсивно досліджується окремий випадок цієї проблеми, так звані *top-K-запити* та неточні запити. Знову ж таки, розроблені моделі та технології працюють лише із визначеними моделями даних (зокрема реляційною – *Fquery, Fuzzy Grouping* та *Fuzzy Lookup*). Стосовно невчасних даних взагалі відсутні методи, які б давали змогу не тільки фіксувати факт запізнення даних, але й на основі цих тимчасово відсутніх даних приймати рішення (ведуться роботи у галузі машин для обробки подій, проте сьогодні задекларовані лише проблеми, які призводять до задачі маніпулювання потоками. Роботи, що були проведені у середині 90-х років у галузі активних баз даних (*Active DBMS, ADBMS*) залишилися невикористаними через великий час відклику запитів та неврахування тимчасової відсутності даних).

5. Релевантність відповіді повинна залежати від користувача і від контексту. Потрібне середовище для накопичення і використання відповідних метаданих. Є напрацювання щодо визначення типу користувача на основі записів у журналі доступу до ресурсів.

6. Проблема інтеграції даних, зокрема надоперативних (частково вирішена оперативними сховищами даних – дані збираються та оперативно опрацьовуються, але залежать від зовнішньої структури) та частково структурованих (частково вирішено засобами пошуку неструктурованих даних).

7. Використання природних мов запитів до баз даних (так звані системи з природномовним інтерфейсом) – передбачають формування запитів до системи у вигляді запитальних речень природною мовою.

8. Підтримка систем обробки поточкових даних (наприклад, система *Postgres* Майкла Стоунбрейкера, високорівнева мова «*STREAMSQL*» з вбудованими орієнтованими на потоки примітивами і операціями).

9. Повинна бути можливість ефективного зберігання, доступу і модифікації інформації про стан, а також її комбінування з реальними поточковими даними.

10. Для інтеграції в системі повинна використовуватися однорідна мова для роботи з усіма різновидами даних.

Як бачимо, майже за усіма вказаними напрямками ведуться роботи. Але ці роботи неінтегровані, не передбачають єдиного опрацювання та жорстко прив'язані до моделі даних, що є цілком неприйнятним у контексті просторів даних. Тому проблема формалізації просторів даних є актуальною.

На відміну від СУБД, в ядрі *DSSP* потрібна підтримка декількох моделей даних, щоб природно підтримувалися якомога більше типів учасників.

Простір даних – це множина баз даних, сховищ даних, локальних сховищ та індексів, статичних, графічних матеріалів, засобів інтеграції, пошуку та опрацювання інформації [6, 7]. Як ключова задача робіт в області керування даними використовується *платформа підтримки просторів даних* (*DataSpace Support Platforms, DSSP*). *DSSP* забезпечує набір взаємозв'язаних послуг і гарантує розробникам можливість концентруватися на специфічних проблемах їх додатків, а не на завданнях, що повторюються, виникають за потреби узгодженої й ефективної роботи з взаємозв'язаними, але роздільно керованими даними.

Основні об'єкти простору даних та задачі, що розв'язуються з його допомогою, подано на рис. 2.

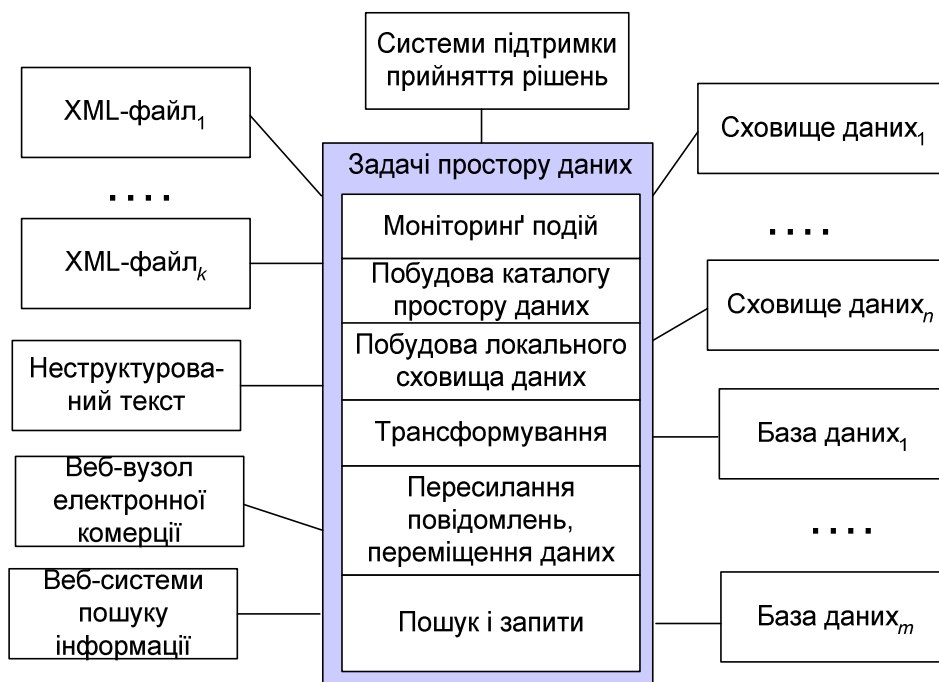


Рис. 2. Об'єкти простору даних та його задачі

Модель простору даних **DS** – це множина даних, поданих у різних моделях (баз даних **DB**, сховищ даних **DW**, статичних веб-сторінок **Wb**, неструктурованих даних **Nd**, графічних та мультимедійних даних **Gr**), локальних сховищ та індексів **ODW**, а також засобів інтеграції **Int**, пошуку **Se** та опрацювання інформації **Wo**, об'єднаних середовищем управління моделями **EM**.

$$DS = \langle DB, DW, ODW, Wb, Nd, Gr, Int, Se, Wo, EM \rangle$$

Моделі даних, що підтримуються у просторі даних, утворюватимуть ієрархію відповідно до їх виразної потужності [9]: реляційна, багатовимірна, об'єктно-реляційна моделі, розширена мова розмітки інформації (*Extensible Markup Language* — *XML*) зі схемою, середовище опису ресурсів (*Resource Description Framework* – *RDF*), стандартний засіб опису зв'язків між об'єктами даних – онтології, описані за допомогою *Web Ontology Language* – *OWL*, структурований текст (зокрема *HTML*), неструктурований текст.

Кожен учасник простору даних підтримує деяку модель даних і деяку мову запитів, відповідну цій моделі. Запит до такої моделі даних відповідає тому, що зазвичай підтримується у файлових системах відносно їх директорій: зіставлення імен, пошук в діапазоні дат, сортування за розміром файла тощо. На наступному рівні *DSSP* повинна підтримувати модель даних мультимножини слів для ефективного пошуку необхідної інформації за ключовими словами, отже, ми отримуємо деяку можливість бачення вмісту учасників простору даних. Нижче рівня моделі мультимножини слів в ієрархії може розташовуватися модель напівструктурованих даних, заснована на позначених графах.

За наявності деякого середовища ключова проблема полягає в знаходженні методів інтерпретації запитів різними мовами щодо учасників, що підтримують деякі моделі. Точніше,

проблема полягає в переформулюванні запиту, поданому складнішою мовою, для джерела, яке підтримує слабкішу модель даних, і навпаки, переформулюванні запиту, поданого простішою мовою, для джерела, яке підтримує виразнішу модель даних і мову запитів (наприклад, запит за ключовими словами до реляційної бази даних).

Висновок

З найбільшими труднощами розробники зіштовхуються, коли рішення необхідно приймати в реальному часі, тобто коли аналізуються потоки даних. При цьому немає часу на те, щоб попередньо зберегти і якось обробити дані – інформацію потрібно одержувати одночасно з надходженням даних. У сучасному світі, що прагне до глобалізації відносин, період між надходженням даних і вирішенням завдання неухильно скорочується. Усе більше процесів вимагає аналізу в режимі реального часу, коли будь-яка затримка є критичною. І в таких ситуація без допомоги ефективного інструментарію обробки даних, побудованого на інформаційних технологіях, людині не обійтися.

1. Date C.J. and Darwen H.: *Databases, Types, and the Relational Model: The Third Manifest (3rd edition)*. Reading, Mass.: Addison-Wesley. – 2006. 2. Гектор Гарсиа-Молина, Дженнифер Уйдом, Джефффри Ульман. *Системы баз данных*. – М.: Вильямс, 2003. 3. Inmon W. H. *Building The Data Warehouse (Second Edition)*. – NY, NY: John Wiley. – 1993. 4. Гаврилова Т.А., Хорошевский В.Ф. *Базы знаний интеллектуальных систем: Учебник*. – СПб.: Питер, 2000. – 384 с. 5. Codd E. F., Codd S. B., Salley C. T. *Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate*. – Codd E. F. & Associates. – 1993. 6. Пасічник В.В., Шаховська Н.Б. *Сховища даних: Навч. посібник*. – Львів «Магнолія 2006», 2007. 7. Шаховська Н.Б. *Модельовання сховищ даних з невизначеністю на основі реляційної моделі. Дис. ... канд. техн. наук*. – Львів, 2006. – 176 с. 8. *Garretts Summary of Principles of Dataspace Systems*, http://aravaipa.eas.asu.edu/wiki/index.php/Garretts_Summary_of_Principles_of_Dataspace_Systems#Overview. 9. Шаховська Н.Б. *Простори даних: поняття та призначення // Матеріали конференції CSIT-2007*. – Львів, 2007. – С. 269–277.