



Международно Висше Бизнес Училище
International Business School

КУРСОВА РАБОТА

ПО

ИНФОРМАЦИОННИ ТЕХНОЛОГИИ В БИЗНЕСА

НА ТЕМА

ОБРАБОТКА НА ЗАЯВКИ В ИНФОРМАЦИОННАТА СРЕДА НА ТРАНСПОРТНА КОМПАНИЯ АСИАНА ООД

Изготвил:

Проверил:

I. ТЕКСТОВО ОПИСАНИЕ НА РАБОТНИЯ ПРОЦЕС

Обект на разглеждане в настоящата разработка е АСИАНА, която е фирма занимаваща се с пътнически превози в т.ч. продажбата на самолетни билети и други пътнически услуги предлагани от компании. Също така АСИАНА продава самолетни билети от името на превозвачи, които не са членове на Международната асоциация по авиопревоз.

В настоящата разработка предмет на изследване е информационното осигуряване на онлайн продажбата на билети. Това става на базата на електронна онлайн резервация на страницата www.airmeti.bg.

Правейки резервация в системата за електронни онлайн резервации, пътникът се обвързва с поръчката. В отговор на приетата поръчка, пътникът следва да получи от фирмата потвърждение за поръчаната услуга под електронна форма, т.е. на имейла, посочен от него при резервацията, в което се съдържат данните за поръчаната превозваческа услуга, за превозвача, за евентуални допълнителни услуги и точната им цена.

В отговор на задължителните условия, както и в изпълнение на приложомото законодателства в областта на пътнически превоз, информационната система дава възможност на фирмата:

- a) да бъде навременно уведомявана за направена поръчка или резервация;
- b) потвърждава поръчката
- c) навременно да издава билет или друг пътнически документ,
- d) навременно да предва по електронен път съгласно Закона за електронния документ на необходимите документи на пътника.
- e) трансфериране на цената на пътническата услуга към сметката на фирмата.

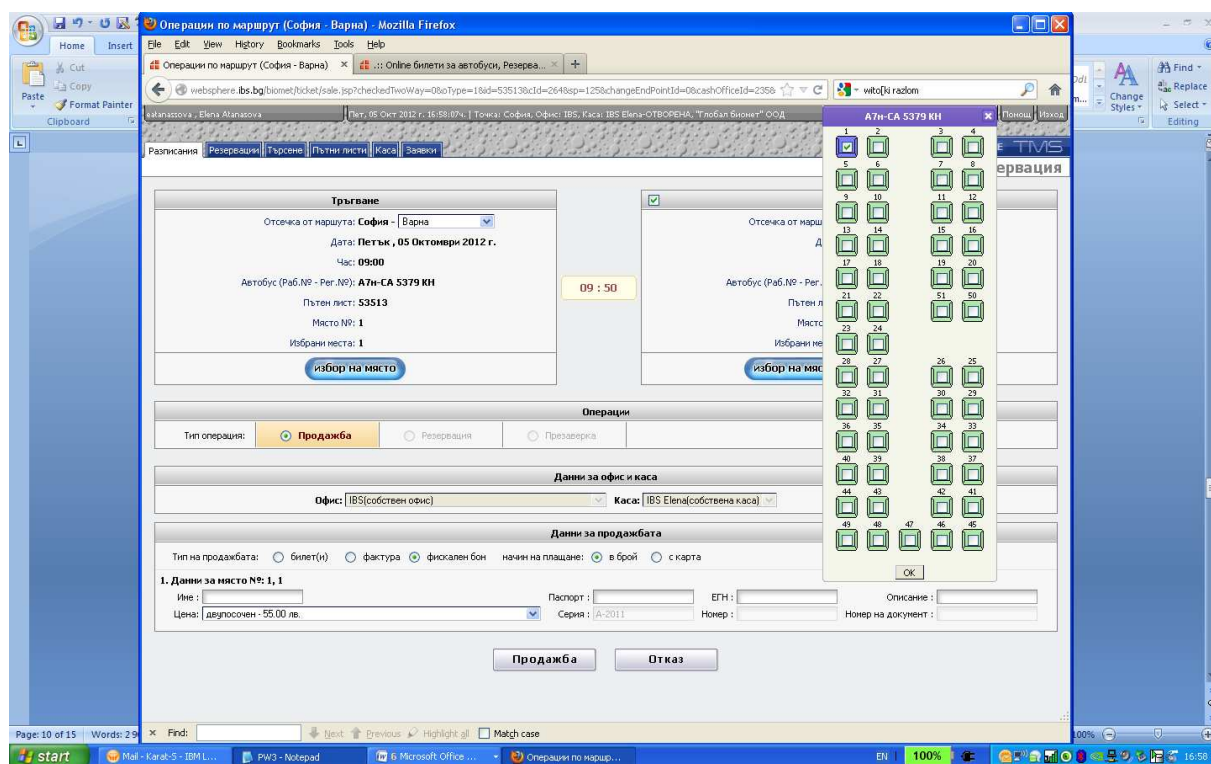
II. МОДЕЛИРАНЕ НА РАБОТНИЯ ПРОЦЕС

Компанията е избрала да използва TicketSphere Transportation Management System, която е изцяло професионална, уеб базирана система, която оптимизира

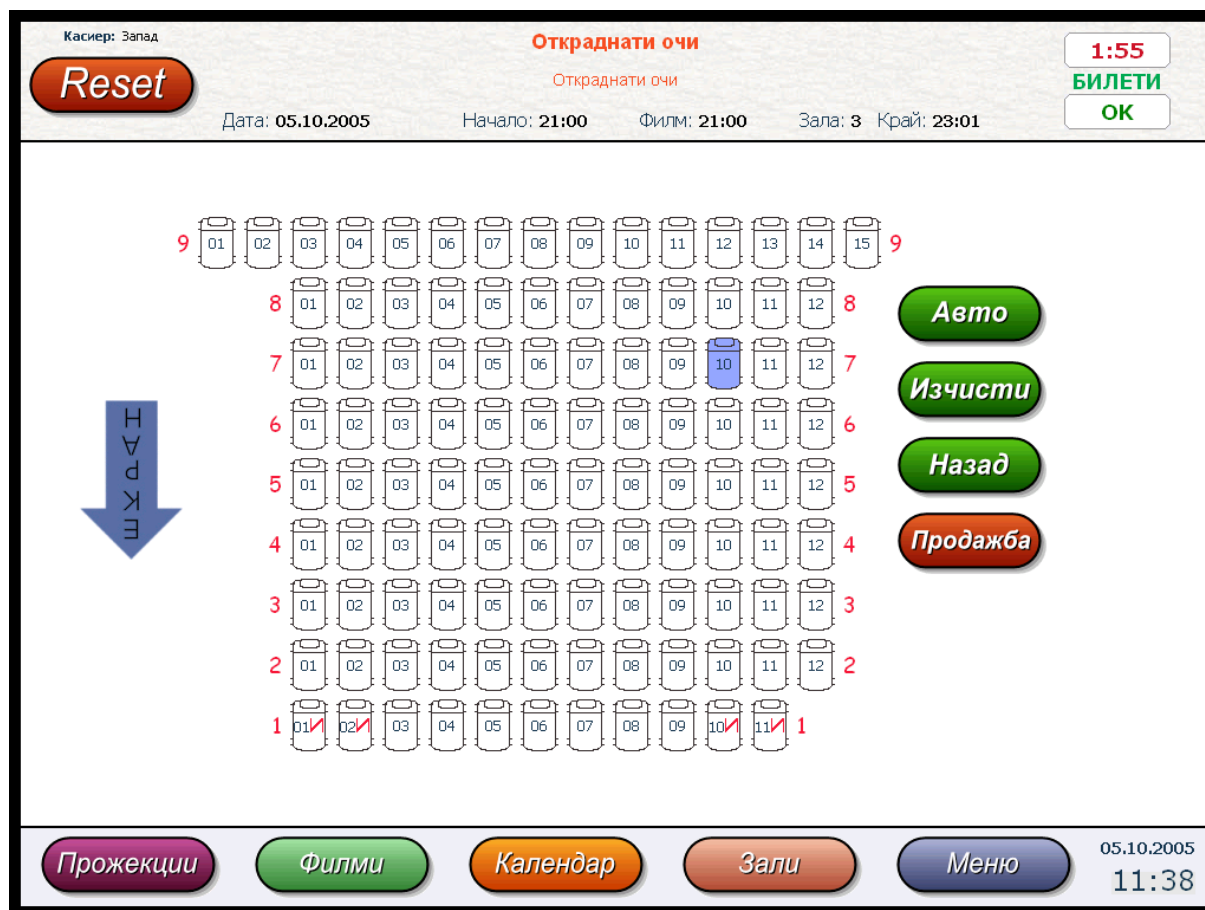
управлението и администрирането на ключови дейности при осъществяване на пътнически превози.

Системата е изградена от два основни модула **Каса** и **Администрация**, които могат да бъдат свързани с допълнителен модул за **Онлайн продажби**. В основата на нейното функциониране е обработката на заявките, които се получават от различните звена в системата.

През Касовия модул се извършват операциите от билетните каси, в това число, продажба, резервация, анулиране и връщане на билети. Всяка каса има подробен отчет, в който са отразени както сумите, така и отделните операции и документи издадени от съответната каса.



Административния модул, на системата ви дава възможност да управлявате оперативната информация в компанията, в това число маршрутите, разписанията, графичите, вида и броя транспортни средства, персонал, офиси, билети, цени и отстъпки. В този модул разполагате със секция със справки и отчети, които ви дават детайлен поглед върху дейността на компанията и касите в зависимост от различни критерии.



Интеграцията с Онлайн модула, позволява на транспортната компания да предлага на своите клиенти възможността да резервират и платят билета си директно през интернет.

Отделните модули комуникират помежду си в реално време и по този начин се осигурява моментална актуализация на базата данни при всяка извършена операция.

Проблемът за подобряване на работата на системите за управление на бази данни (СУБД) в момента е от все по-голямо значение. Увеличаване на броя потребители работещи с бази данни и увеличаване на размера на обработваната информация изискват съответното подобрение на базите данни за изпълнение в разумен срок от време за генериране на отговор на заявката на потребителя¹. Ефективна и икономически жизнеспособна алтернатива на еднопроцесорен СУБД се явява паралелно СУБД, работещо едновременно на множество процесори. Използването на паралелна база данни чрез съчетаване на няколко маломощни машини позволява да се

¹ Черепухин А.Н., Савинков Ю.А., Локишин М.В. Проблемм разработки больших информационных систем. // Современные проблемм информатизации в системах моделирования, программирования и телекоммуникациях: Сб. трудов. Вмп. 8. Под ред. д.т.н., проф. О.Я. Кравца - Воронеж: Издательство «Научная книга», 2003. - С. 120-121.

получи същото ниво на изпълнение на производството, както и в случай на една, но мощни машина, за получаване на резултати в скалируемост и надеждност на системата в сравнение с еднопроцесор база данни².

В момента, бази данни се използват в почти всички области на човешките дейности, свързани със съхраняване и обработка на информация. Напредъкът постигнат в областта на технологиите на бази данни се основава до голяма степен на релационните модели, предложени от Код, Е.³ в края на 60-те - 70-те години на XX век. По време на своята тридесет-годишна история релационните бази данни произтичат от изследване прототип, най-важните от които са System R⁴ и Ingres⁵, търговски продукти, които могат да съхраняват и обработват терабайти информация. Въпреки това, научната и практическа дейност на човека поставя все по-нови предизвикателства, които изискват обработката на много големи бази данни.

Появата на много големи бази данни е в резултат на разширяването на приложенията и базите данни. Примери за нови приложения на бази данни са електронна търговия⁶ вкл. електронните резервационни системи, цифровите библиотеки, геоинформационните системи, мултимедийни файлове, научните база данни⁷.

Днес се наблюдава нарастващото търсене на технологии за паралелни и разпределителни бази данни и от големи и от средни компании. Друга логична стъпка в посока на информационни системи от този клас е да се създаде единна база данни, за да даде възможност за бърз достъп и анализ на данни, както и за решаване на специфични за компанията (в зависимост от сферата на дейността) въпроси⁸.

² Коннолли Т., Бегг К, Страчан А. Базм даннмх: проектирование, реализация и сопровождение. Теория и практика. - М.: Вильямс, 2000. - 1120 с.

³ Copeland G.P., Keller T. A Comparison Of High-Availability Media Recovery Techniques // Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, Portland, Oregon, May 31 - June 2, 1989. - ACM Press, 1989. -P. 98-109.

⁴ Ballinger C., Fryer R. Born To Be Parallel: Why Parallel Origins Give Tera- data an Enduring Performance Edge // IEEE Data Engineering Bulletin. - 1997. -Vol. 20, No. 2. -P. 3-12.

⁵ Stonebraker M. Retrospection on a Database System // ACM Transactions On Database Systems. -1980. - Vol. 5, No. 2. -P. 225-240.

⁶ Kalakota R., Whinston A. Readings in Electronic Commerce. -Addison- Wesley, 1997.

⁷ Stonebraker M., Frew J., Gardels K., Meredith J. The Sequoia 2000 Benchmark // Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., May 26-28, 1993. -ACM Press, 1993.-P. 2-11.

⁸ Локишин М.В., Кравец О.Я. О некоторых принципах построения автома- тизированных информационных систем с применением СУБД. // Единое информационное пространство: сборник докладов

В действительност, единственото известно и ефективно днес решение за съхранение и обработка на много големи бази данни е използването на паралелни системи за бази данни⁹, с репликация на съхраняваната информация с оглед обезпечаване на паралелна обработка на информационните търсения в многопроцесорни системи.

Интензивната изследователска дейност в областта на паралелните база данни е започнала през 80-те години. През последните две десетилетия, паралелните системи бази данни са се преориентирали от изследователски прототипи към пълнофункционални търговски продукти, доставяни на пазара на информационни системи с висока производителност. Като примери за успешни проекти, създаване на паралелни системи бази данни могат да причислят гамата от продукти - DB2 Parallel Edition, NonStop SQL и NCR Teraciata¹⁰. Въпреки това, разходите на такава специализиран софтуер за тези системи е сравним, и в някои случаи по-високи от разходите на хардуерен компонент.

Тези системи интегрират стотици процесори и твърди дискове и състояние да се справят с десетки терабайта. Въпреки това, в паралелни системи бази данни се нуждаят от допълнителни изследвания¹¹, едно от тях - нататъшно развитие на методи за оптимизиране на ефективността на съхраняване и търсене (заявките) в паралелни и хетерогенни бази данни. В допълнение, много изследователи правилно отбелязва ограниченията в мащаба. С голям брой на процесорите в системите от този тип започват да се появяват конфликти на достъпа до споделената памет, която може да доведе до тежко влошаване на общата производителност на системата. В съответствие с това реалната мащабируемост така се наричат SMP системи с ограничен 32 процесор¹².

Международной на- учно-практической конференции (Днепропетровск, 3-4 декабря 2003 г.). ИПК ИнКомЦентра УГХТУ, 2003 г. - С. 140-143.

⁹ Eugene J. Shekita, Honesty C. Young, Kian-Lee Tan. Multi-Join Optimization for Symmetric Multiprocessors. // VLDB'93, Proceedings of 19th International Conference on Very Large Data Bases, August 24-27, 1993, Dublin, Ireland. -Morgan Kaufmann, 1993. -P. 479-492.

¹⁰ Лисьянский К, Слободяников Д. СУБД Teradata для ОС UNIX // СУБД. - • 1997. №5-6.-С. 25-46.

¹¹ Valduriez P. Parallel Database Systems: Open Problems and New Issues // Distributed and Parallel Databases. -1993. -Vol. 1, No. 2. -P. 137-165.

¹² Valduriez P. Parallel Database Systems: the Case for Shared-something // Proc. of the 9th Int. Conf. on Data Engineering, April 19-23, 1993, Vienna, Austria. - IEEE Computer Society, 1993. - P. 460-465.

III. ОПИСАНИЕ НА ИНФОРМАЦИОННАТА СИСТЕМА И ТЕХНОЛОГИИ ЗА РЕАЛИЗАЦИЯ НА ПРОЦЕСА

Единственото решение в рамките на заявки за търсене в големи бази данни, е репликацията на данни за целите на паралелна обработка. Свръх големите бази данни се характеризират с това, че количеството на данните, съхранявани в тях надвишава RAM паметта на разположение на системата. В резултат на това за ефективно натоварване на изчислителната мощност се изисква осигуряване на висока пропускателна способност на диска.

За разделяне на използваните методи за **съхранение на данни**, въз основа на обхвата не позволява тази операция (подразделение) да бъде динамична (за решаването на този проблем ще трябва да съхранява копие на отделните данни варира през всички възможни дялове)¹³. В допълнение, възможен е дисбаланс на натоварването на връзките на системния софтуер като не се позволява корекция. Методът на разделянето на данни на базата на хеш не позволява ефективна работа с данните, разделянето им на диапазони. Поставяне данни, базирани на хеширане се основава на определени априори и набор от атрибути. Този метод създава дисбаланс в натоварването по време на обвързване отношенията на работа¹⁴. Повече от тези методи се фокусират върху използването на системи, базирани на SMP системи.

Известно е, че системи, базирани на многопроцесорна обработка (MPP), позволяват да се постигне висока изчислителна мощ с по-ниски разходи и да осигури по-добра мащабираност в сравнение със системите за SMP. Основният недостатък на системите за MPP е ниската (в сравнение с SMP системи) скорост на взаимодействие между отделните изчислителни връзки¹⁵.

За да реши този проблем, компания АСИАНА ООД използва за всяка изчислителна връзка на MPP система собствено копие на данните. С оглед на това, можете да се реализира операция за паралелна обработка на данни, свеждайки до минимум взаимодействието между връзките на системата. Трябва да се отбележи, че

¹³ *Garcia-Molina H., Labio IV., Yang J.* Fast Incremental Maintenance of Approximate Histograms // VLDB'98, Proceedings of 24th International Conference on Very Large Data Bases, August 25-29, 1998, New York, USA. - Morgan Kaufmann, 1997. -P. 500-511.

¹⁴ *Walton C., Dale A., Jenevein R.*: A Taxonomy and Performance Model of Data Skew Effects in Parallel Joins. // VLDB'91, Proceedings of 17th International Conference on Very Large Data Bases, September 3-6, 1991, Barcelona, Catalonia, Spain. -Morgan Kaufmann, 1991. -P. 537-548.

¹⁵ *Коннолли Т., Бегг К, Страчан А.* Базм данних: проектирование, реализация и сопровождение. Теория и практика. - М.: Вильямс, 2000. - 1120 с.

този подход при рязките на MPP система може да се приложи за отделни системи SMP. Архитектурата наречена смесена или хибридна, позволява да се достигне до по-голяма производителност¹⁶, което се дължи на допълнителното прилагане на заявки за оптимизация, които днес са достатъчно развити.

Описвайки системата, основана на MPP архитектура, която позволява осъществяване на паралелна обработка на заявки за търсене в SQL¹⁷. Нека системата се състои от N връзки, които се изпълняват SQL Server, както и данните SQL Server не трябва да бъдат хомогенни. В този случай, да се позове на такава система трябва да използвате подмножество на език SQL, общ за цялата система SQL Server. Трябва да се има в предвид, че ако базата данни се използва от операторите на езика SQL, които се различават само по синтактична форма, с помощта на тривиалното трансформация става възможно да се използва синтаксисът SQL език, който не е общ за всички SQL Servers. В същото време, има дизайн, който не позволяват на тривиално да конвертирате от един милиард диалект на SQL в друг. Също ще приемем, че ние се интересуваме от един набор от данни, за N сървъри разпределителна система.

Под съгласуваност на данните на два езика, ние ще разбираме идентичност на резултата за всяко искане за предоставяне на данни, при и първия, и втория възел. За изпълнение на паралелна обработка на информацията за набор от данни те следва да се разглеждат в условно време на системата, за всяка комбинация от връзки. По-специално, това означава, че по време на започване на работа на системата данните трябва да бъдат синхронизирани.

Тази система може да се прилага няколко алгоритми за попълване/съхранение на данните. В най-простия случай, могат да бъдат данни за обновление на система за определен период от време, през който може да се добавят или променят съществуващите данни. Най-лесно приложимите методи за попълване на информацията, което е характерно за класическото съхранение на данните (Data Warehouse), където отсъства възможността да се актуализират по-рано натрупаните на данните. Може да се реализира актуализиране без спиране на работата на системата. Тогава актуализираните данни във всички връзки на разпределителната система трябва

¹⁶ *Shekhar S., Srivastava J., Dutta S.: A Formal Model of Trade-off between Optimization and Execution Costs in Semantic Query Optimization. 457-467, VLDB 1988.*

¹⁷ *Локишин М.В. Обработка запросов к большим объемам данных в СУБД // Актуальные проблемы информатики и информационных технологий: Материали Международной (VIII Тамбовской межвузовской) научно- практической конференции (9-10 сентября 2004 г.). Тамбов: Издательств- во ТГУ им. Державина, 2004 - С. 69.*

да се извърши в рамките на една разпределителна транзакция с оглед на промените в различни връзки на системата да остане достъпна за потребителите, които четат в синхронен режим. Такова поведение може да бъде постигнато с помощта на заключващ механизъм, реализиран в базата данни или въз основа на концепцията на времеви отпечатъци при боравене с искането за търсене като се взема под внимание времето на попълване в базата данни, които документи и записи след определено време се игнорират.

В разглежданата система е възможно да се приложи концепцията за смесени данни, OLAP (Online Analytical Processing) и OLTP (онлайн обработка на транзакции). За да се направи това, е препоръчително да се използва "мошен" сървър за поддръжка на OLTP част от базата данни, информация на който няма да бъде подложена на репликация, а останалата част на системата ще обезпечи функционирането на компонентите на OLAP за изпълнение на предложената схема. Цялата база данни ще се състои от едно копие на данни от OLAP и данните на OLTP сървър. Получаване на данни от частта OLAP на системата, както и от OLTP част може да се направи с помощта на бази данни, въз основа на концепцията за разделяне на данните. Целта на разделянето на данни може да бъде с оглед същите след края на период да не могат да бъдат редактирани. Съгласувани промени в произволни данни в системата като цяло е трудно в някой от предложените варианти за използване на системата, както и изпълнението на тази функция не е целесъобразно.

Трябва да се разбете, че предложената схема за изграждане на системата позволява ускоряване на изпълнението на SQL-заявки за търсене с понижаване на скоростта на изпълнение на искания за съхраняване / промяна / изтриване на данни (наричан по-нататък - модификации). Известно е, че има нужда от актуализация на репликационните данни, което изисква повече разходите и има някои ограничения. Последната от опциите за прилагането на системата представлява компромис между увеличаването на разходите за модификация на данните и увеличаване на скоростта на обработка на данни.

За описаната по-горе разпределителна система, ние предлагаме следното паралелно изпълнение на SQL-заявки за търсене. Потребителската заявка за търсене първоначално е била получена от един от връзките в системата, подготвена за паралелно изпълнение. Възможността за паралелно изпълнение предоставена от превръщането на изходната SQL-заявка в набор от запитвания е резултат от

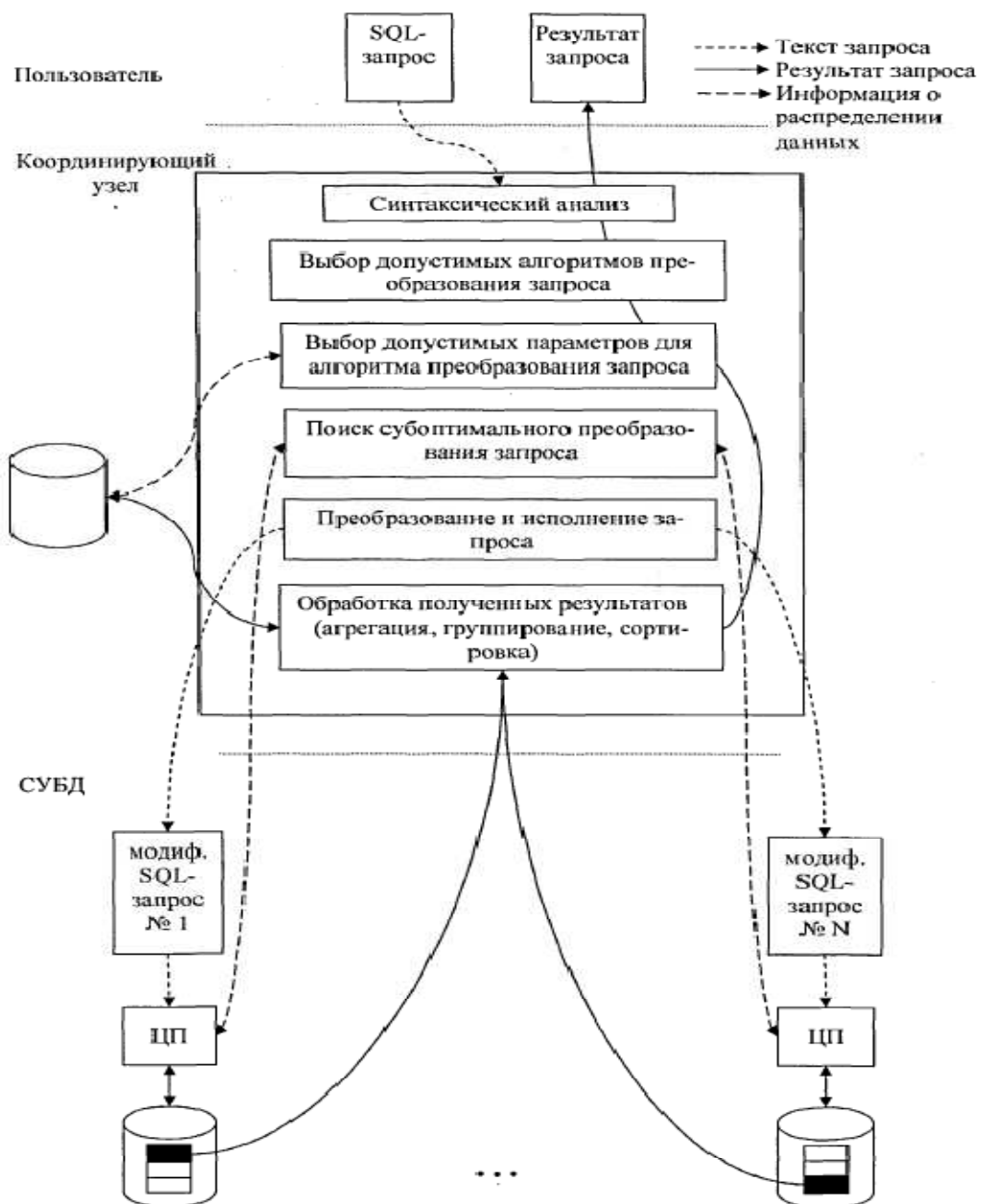
изпълнението еквивалентната на оригиналната¹⁸ (след комбиниране на резултатите от изпълнението на новите заявки, възможни са допълнителни изчисления на агрегатни функции и групиране и пренареждане на резултите).

Както ще бъде показано по-късно, набор от заявки, в резултат на преобразуването могат да бъдат изпълнени на всяка връзка в системата, независимо една от друга. Това произтича от изискването за идентичните копия на данните и еквивалентността произтичаща от първоначалната заявка. А наборът от правила за трансформация на запитвания за търсене се определя предварително и допустимостта на използването му или други се определят от синтактичните правила за конвертиране на първоначалното искане и наличната статистическа информация за атрибутите на използваните отношения в заявката. Този подход ни позволява динамично да изберем един алгоритъм за паралелно обработване на заявка за изпълнение въз основа на съдържанието на таблици в базата данни.

Ключовият момент в изпълнението на предложения подход е добавяне на посреднически звена за поредица от действия, съответстващи на за изпълнението на заявката¹⁹. По същество, практическата реализация на предложените решения представлява изграждане на система, основана на тристепенна архитектура (фиг. 7). В същото време от гледна точка на клиента обработката на заявката за търсене остава същата независимо действителния брой на сървърите, разпределени за да се справят с искането. Всички функции по координация на работата на отделните сървъри са с цел програмно обезпечение на средно ниво.

¹⁸ Волков В.В. Тести ТРС // СУБД. -1995. -№2. -С. 70-78.

¹⁹ Локишин М.В., Кравец О.Я. Параллельное исполнение запросов в реляционных базах данных с использованием репликации. // Информационные технологии моделирования и управления: Международны сборник научных трудов, Издательство «Научная книга», 2004 г. - С. 75-80.



Фиг.6 Архитектура на система за паралелна обработка на заявки за търсене

Хардуерна платформа, с която боравят посредническите звена може да действа като специално определена за тази цел, системни връзки за директно прилагане на потребителските заявки на СУБД.

Обобщавайки всичко посочено по-горе, следва да се посочат ползите от прилагането на този подход в Бата Шус ООД. Те са, както следва:

- Проектиране на работи с разнородни бази данни;

- Реализиране на възможности за постигане на висока производителност в резултатът от изчислението на заявката;
- Постигане на относителна простота на системата с паралелна обработка на заявки. В действителност системата прилага двойно процеса на оптимизация - за паралелно изпълнение на заявка между връзките на системата и процеса за локално изпълнение на искането на всяка връзка;
- Има възможност за прилагане на предложения принципите на паралелна обработка на базата на класическите методи на парализация (операции по сканиранията, присъединява и сортиране);
- Динамична промяна в броя на връзките, които могат да обработват искането по време на работа на системата;
- Динамична промяна на алгоритмите за трансформация на заявката. От гледна точка на статистическа информация, съхранявана в системата, е възможно да се променят динамично алгоритмите, които превръщат заявката. По този начин, няма отклоняване от декларирания принцип, залегнал в езика SQL, който описва какво трябва да се направи и как трябва да се направи - процесите на SQL-заявката.

IV. ИЗПОЛЗВАНА ЛИТЕРАТУРА И ИЗТОЧНИЦИ

1. Волков В.В. Тести TPC // СУБД. -1995. -№2. -С. 70-78.
2. Коннолли Т., Бегг К, Страчан А. Базм даннмх: проектирование, реализация и сопровождение. Теория и практика. - М.: Вильямс, 2000. - 1120 с.
3. Коннолли Т., Бегг К, Страчан А. Базм даннмх: проектирование, реализация и сопровождение. Теория и практика. - М.: Вильямс, 2000. - 1120 с.
4. Лисянский К, Слободяников Д. СУБД Teradata для ОС UNIX // СУБД. - • 1997. №5-6.С. 25-46.
5. Локишин М.В., Кравец О.Я. О некоторых принципах построения автоматизированных информационных систем с применением СУБД. // Единое информационное пространство: сборник докладов Международной научно-практической конференции (Днепропетровск, 3-4 декабря 2003 г.). ИПК ИнКомЦентра УГХТУ, 2003 г. - С. 140-143.

6. *Локишин М.В.* Обработка запросов к большим объемам данных в СУБД // Актуальные проблемы информатики и информационных технологий: Материалы Международной (VIII Тамбовской межвузовской) научно- практической конференции (9-10 сентября 2004 г.). Тамбов: Издательств- во ТГУ им. Державина, 2004 - С. 69.
7. *Черепухин А.Н., Савинков Ю.А., Локишин М.В.* Проблемы разработки больших информационных систем. // Современные проблемы информатизации в системах моделирования, программирования и телекоммуникациях: Сб. трудов. Вмп.
8. Под ред. д.т.н., проф. О.Я. Кравца - Воронеж: Издательство «Научная книга», 2003. - С. 120-121.
8. *Ballinger C., Fryer R.* Born To Be Parallel: Why Parallel Origins Give Tera-data an Enduring Performance Edge // IEEE Data Engineering Bulletin. - 1997. -Vol. 20, No. 2. -P. 3-12.
9. *Copeland G.P., Keller T.* A Comparison Of High-Availability Media Recovery Techniques // Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, Portland, Oregon, May 31 - June 2, 1989. - ACM Press, 1989. -P. 98-109.
10. *Eugene J. Shekita, Honesty C. Young, Kian-Lee Tan.* Multi-Join Optimization for Symmetric Multiprocessors. // VLDB'93, Proceedings of 19th International Conference on Very Large Data Bases, August 24-27, 1993, Dublin, Ireland. -Morgan Kaufmann, 1993. - P. 479-492.
11. *Garcia-Molina H., Labio IV., Yang J.* Fast Incremental Maintenance of Approximate Histograms // VLDB'98, Proceedings of 24th International Conference on Very Large Data Bases, August 25-29, 1998, New York, USA. - Morgan Kaufmann, 1997. -P. 500-511.
12. *Kalakota R., Whinston A.* Readings in Electronic Commerce. -Addison-Wesley, 1997.
13. *Shekhar S., Srivastava J., Dutta S.:* A Formal Model of Trade-off between Optimization and Execution Costs in Semantic Query Optimization. 457-467, VLDB 1988.
14. *Stonebraker M.* Retrospection on a Database System // ACM Transactions On Database Systems. -1980. -Vol. 5, No. 2. -P. 225-240.
15. *Stonebraker M., Frew J., Gardels K., Meredith J.* The Sequoia 2000 Benchmark // Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., May 26-28, 1993. -ACM Press, 1993.-P. 2-11.

16. *Valduriez P.* Parallel Database Systems: Open Problems and New Issues // Distributed and Parallel Databases. -1993. -Vol. 1, No. 2. -P. 137-165.
17. *Valduriez P.* Parallel Database Systems: the Case for Shared-something // Proc. of the 9th Int. Conf. on Data Engineering, April 19-23, 1993, Vienna, Austria. - IEEE Computer Society, 1993. - P. 460-465.
18. *Walton C., Dale A., Jenevein R.:* A Taxonomy and Performance Model of Data Skew Effects in Parallel Joins. // VLDB'91, Proceedings of 17th International Conference on Very Large Data Bases, September 3-6, 1991, Barcelona, Catalonia, Spain. - Morgan Kaufmann, 1991. -P. 537-548.