

ПРИНЦИПИ И МЕТОДИ ЗА ИЗГРАЖДАНЕ НА РАЗПРЕДЕЛЕНИ БИЗНЕС ИНТЕЛИГЕНТНИ ЕКОСИСТЕМИ

Валентин Кисимов, Камелия Стефанова*

1. ВЪВЕДЕНИЕ В БИ ЕКОСИСТЕМИ

Голямата несигурност в пазарната среда, особено през последната година, налага извършването на адекватни промени в управлението на всяка една организация. За да се осъществява постоянното въвеждане на промени, всяка компания все по-осъзнато започва да разчита на бизнес интелигентните (БИ) решения.

От своя страна, БИ системи силно еволюират – от решение използвано от ограничен брой специалисти в организацията, до мащабно изградена цялостна екосистема, където всеки в компанията, на всички нива, във всички отдели, изпитва предимствата от ежедневното използване на резултатите от работата на БИ системи.

Според създателя на термина "екосистема" – Arthur Roy Clapham [32], той представя комбинация от физически компоненти и биологически компоненти на една среда. С развитието на това понятие се достига до "жив организъм", който предоставя всички възможности за развитие и усъвършенстване. Основателят на екологията – Eugene Odum [32], доразвива идеята, като представя екосистемата като логическа единица, която включва всички организми, взаимодействащи си с физическата ѝ среда. На тази база, ние създаваме понятието "Бизнес Интелигентна екосистема" (БИ система), която включва всички съставни компоненти на БИ системите ("организмите" на БИ екосистемата), както и взаимодействието с бизнес средата ("физическата среда" на БИ екосистемите), за да може да функционира и се развива една БИ система. Създаването на една компютърна БИ система не е еквивалентно на създаването на БИ екосистема. На чистата компютърна БИ система, състояща се от технологии и ИКТ процеси, ѝ липсва връзката с конкретната бизнес среда на фирмата или организацията, за която е предназначена, за да може в резултат на взаимодействие на двете да се породят условия, водещи до създаване на специфични резултати, приложими за конкретната фирма или организация, които пък водят до развитието на БИ системата, както и до резултати, подпомагащи нейното бъдещо усъвършенстване. За разлика от БИ екосистемата, на чистата компютърна БИ система ѝ липсват бизнес аналитиците, които да насочват формирането на резултатите; липсват ѝ отношението и възприемчивостта на бизнес лидерите към резултатите, които БИ системата произвежда; липсват ѝ бизнес решенията за

* Валентин Кисимов е доктор на икономическите науки, доцент и зав.-катедра "Информационни технологии и комуникации", тел.: 81-95-203, vkisimov@gmail.com; Камелия Стефанова е доктор по икономика, доцент в катедра "Информационни технологии и комуникации"; тел. 81-95-533, stefs@cablebg.net . Студията е разработена в рамките на проекта по договор с МОМН Д002-116/2008.

реинженеринг на БИ системата с оглед създаване на нови резултати от работата на самата БИ система, служещи за по-ефективно удовлетворяване нуждите на бизнеса, липсват и формите за допълнително обучение на бизнес аналитиците и бизнес лидерите за използване на възможностите и резултатите на БИ системата. Една от същностите на БИ екосистемата е да има изграден БИ Център за компетентност, който да управлява всички взаимодействия на БИ системите ("организмите" на БИ екосистемата) с бизнес средата ("физическата среда" на БИ екосистемите). Естествено е, че БИ екосистемата трябва да осигури възможността за преобразуването на големи количества корпоративни данни в полезна информация, благодарение на която да бъдат предприети печеливши и ефективни бизнес дейности. Внедряването на БИ екосистема е дълъг процес и изисква извършването на много анализи, големи инвестиции и постоянни условия за динамични изменения както във функционирането на самата БИ система, в настройката на генерираните резултати, така и в гъвкавостта и динамичната ориентация на обслужващия и управленския екип на БИ системата и на цялостното бизнес ръководство.

БИ екосистемата може да се представи чрез 2 взаимно интегрирани части – *БИ система* и *Център за компетентност по БИ*. Това интегриране не е само техническо – на ниво ИКТ, но е и на равнище процеси, и на равнище човешки взаимоотношения.

2. СЪЩНОСТ НА РАЗПРЕДЕЛЕНИЯ ЦЕНТЪР ЗА КОМПЕТЕНТНОСТ ПО БИЗНЕС ИНТЕЛИГЕНТНИ СИСТЕМИ

За да доведат до желаните резултати, БИ проектите трябва да интегрират данните и процесите, както и изискванията и приоритетите към информационното осигуряване на организацията и нейните отделни структурни звена [29]. Тази дейност, от своя страна, изисква специализирани на много високо организационно равнище знания и умения. За съжаление, повечето фирми и институции срещат трудности в организирането и управлението на тези дейности, както и при намирането на подходящите хора, които да извършват това.

Организациите все по-добре започват да разбират, че БИ системи изискват много повече от самото внедряване на съответната технология. Необходим е задълбочен стратегически подход към изграждането на БИ дейността, който се отнася както до технологичните решения, така и до хората, процесите и организационната култура.

Ролята на Центъра за компетентност по бизнес интелигентност (ЦКБИ) в организациите е да направлява използването на БИ технологиите, да дефинира стандартите за бизнес интегриране и управление, както и да определя приоритетите на БИ проекти, свързани с доразвитие на БИ системата и нейния реинженеринг. Успехът на БИ системи зависи от формирането, организирането и управлението на ЦКБИ [29].

ЦКБИ изпълнява ключова роля в синхронизирането на БИ инициативи. Той реализира възможността за сътрудничество между бизнес процесите и обединяване на БИ проекти с цел премахване изпълнението на изолирани и ограничени БИ приложения.

ЦКБИ координира създаването и изпълнението на БИ стратегия. Той подпомага организацията в управлението и изпълнението на съществуващите задачи, като намалява проблемите и повишава ефективността на работата. Координирането на дейностите по бизнес анализ и отчитане гарантира успешното предаване и използване на информацията и добрите практики в рамките на цялата организация, така че да се гради върху успехите и научените уроци.

ЦКБИ представлява най-силният инструментариум за подпомагане трансфера на знания и засилване на аналитичните умения в организацията [30]. ЦКБИ ръководи и обучава специалистите за придобиване на нови способности, за реализиране на иновативност и проява на творчество. Центърът е много важен и за превръщането на анализа в ръководещ фактор на дейността, гарантиращ по-добро използване на информацията и по-високи нива на възвръщаемост на инвестициите.

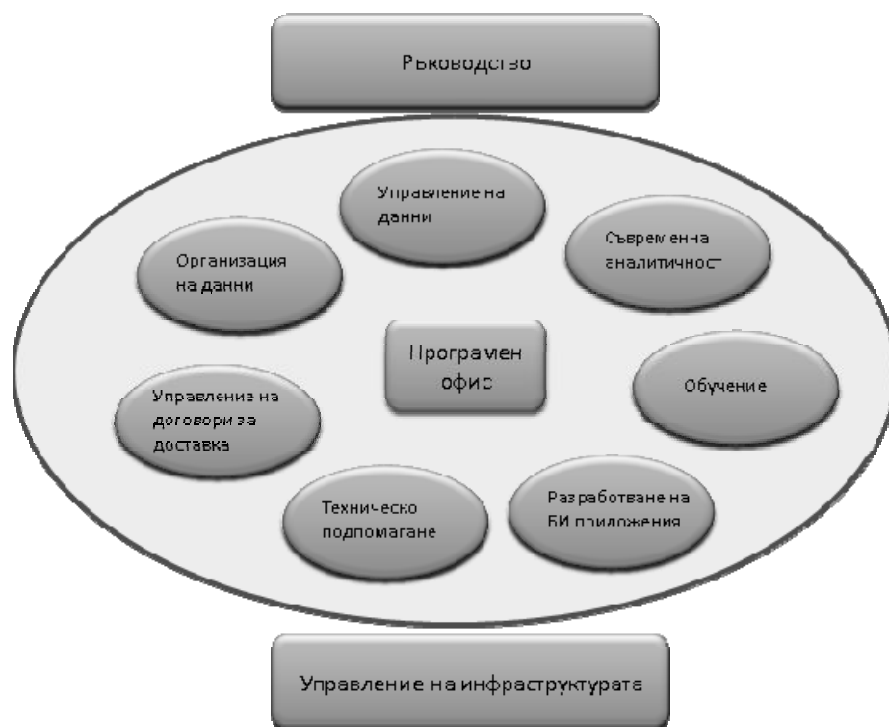
2.1. Дейност и функции на ЦКБИ

Първата стъпка в изграждането на ЦКБИ е да се разбере същността на организацията, нейните бизнес приоритети, информационните структури и потоци, възможностите за обмен на информация и процесите за взимане на управленски решения [31]. Видът и структурата на информацията са много важни за определяне на специфичната роля, обхват и функционалност на ЦКБИ, който да може ефективно да подпомага реализирането на бизнес целите на организацията.

Основните функционални дейности на ЦКБИ са изобразени на Фигура 1.

Функцията *"Програмен офис"* представлява сърцевината на ЦКБИ. Чрез него се организират и координират всичките дейности на центъра. Той изпълнява ролята на интерфейс към бизнес структурите в предприятието. Тази функция отговаря за дефинирането на БИ стратегия и следи за успешното ѝ прилагане във времето. Най-важната цел на тази функция е да гарантира, че БИ проекти се изграждат в посока и осигуряват необходимата подкрепа за постигането на стратегията за развитие на организацията. Специалистите от различните отдели на организацията могат да получават чрез тази функция съвети за това как да използват БИ анализи и как да интерпретират резултатите. Целта на Програмния офис е да се постигне единно управление и наблюдение на изграждането на БИ стратегия и въвеждането на цялостен съдържателен подход при внедряването на БИ системи – стандарти, технологии, управление на знанието, добри практики, бизнес аналитичност и оценка. Тази функция има и задачата да запознава организацията с новите тенденции и технологии в развитието на БИ и съответно отговаря за това, да оценява как те биха помогнали за повишаване на нейната конкурентоспособност.

Функцията *"Организация на данни"* отговаря за дейностите, свързани с нуждите на потребителите от резултатите от работата на БИ системи. Тя отговаря за развитие на методите за съхраняване на данните, тестването и поддръжката, а също така и за цялостното проектиране на складовете от данни и за интегрирането на проекти.



Фиг. 1

Функцията "Управление на данни" се грижи за администриране на техническите метаданни и гарантиране на съответствието им с бизнес метаданните. Тази функция отговаря за осигуряване на качеството на данните, управлението на метаданните, оценката и избора на стандарти и модели на данни.

Функцията "Съвременна аналитичност" се специализира в статистическо моделиране и извличане на закономерности от данните. Тя извършва решаването на сложни аналитични запитвания, задавани от бизнес структурите. В допълнение, тя извършва изследвания и експерименти с моделите, за да установи корелациите и взаимозависимостите между данните.

Функцията "Обучение" отговаря за квалификацията и подготовката на специалистите-потребители на БИ концепции, използващи специфични за организацията БИ приложения, като осигурява задълбочени специализирани консултации по БИ въпроси. В рамките на тази функция центърът отговаря да осигури професионално обучение за конкретните БИ продукти.

Функцията "Разработване на БИ приложения" отговаря за използването и развитието на всяко приложение на БИ системи, за тяхното тестване, подготовка за внедряване и поддръжка. В рамките на тази функция се разработва и вътрешният интерфейс за връзка със складовете от данни. В тази функция са концентрирани и дейностите по реинженеринг, свързани с динамичното модифициране и усъвършенстване на БИ системата "в движение" – по време на нейното функциониране.

Функцията "*Техническо подпомагане*" се изразява в специализирано техническо съдействие на бизнес специалистите във връзка с използването на приложенията за решаване на БИ проекти. Тази функция е насочена към изясняване на възникнали проблеми, анализиране на детайлите и обсъждане със специалистите от бизнес структурите за взимане на правилните решения. Ако проблемът не може да бъде решен вътрешно, то той се адресира към доставчиците на софтуер.

Функцията "*Управление на договори за доставка*" отговаря за всички въпроси, свързани с договорите и лицензите за покупка и използване на БИ софтуер в организацията. В рамките на тази функция се включва изготвянето на споразумения за услуги с доставчиците на БИ системи, както и оценката на различните продукти.

2.2. Принципи за разпределяне на функциите на ЦКБИ

С развитието на БИ екосистеми в организациите се повишава очакването за оптимизиране на различните видове процеси за взимане на решения по всичките нива на управленската йерархия. Това означава, че БИ решенията трябва да могат да подпомагат оценката на изпълнението, повишаването на продуктивността и съответните процеси за взимане на решения от всички управляващи от различните нива и роли в рамките на цялата организация, обхващайки:

- *Стратегическите решения* на организационно ниво – дефиниране и следене на организационните цели, агрегирани тактически решения за управление на оперативните дейности: дългосрочно планиране, финансово консолидиране, анализ на клиентите и пазарни директиви, оценка на двигателите за повишаване на продуктивността, аналитичност на разходите.
- *Оперативните решения*, свързани с нивата за реализация на бизнес дейностите – подпомагане осъществяването на оперативните приложения: управление и оценка на доставките и логистиката, контрол и инвентаризация, ефективност на производство, изпълнение, продажби, финанси.
- *Тактическите решения* на ниво работни групи – фокусиране върху решения, свързани с ефикасността на процесите: взаимодействия в рамките на процесите за изпълнение на бизнес дейностите, централизирано споделяне на данните, оптимизация на работните потоци, бърз и сигурен анализ на информацията.

Следенето на изпълнението на дейността в организацията на основата на дефинирани цели може да подпомогне въздействието върху процесите за взимане на решения, информирайки съответните нива в рамките на йерархията на управлението. Във връзка с наличния капацитет за изграждане и управление на БИ системи, съответната организация е в състояние да оцени възможността кои от нивата може да извършва самостоятелно, за кои да ползва консултантска помощ и кои би трябвало да аутсорсва.

3. ПРИНЦИПИ ЗА РАЗПРЕДЕЛЯНЕ НА ИКТ АРХИТЕКТУРАТА НА БИ СИСТЕМИ

Архитектурата на Бизнес интелигентните системи (БИС) се състои от множество компоненти, изградени на базата на Информационни и комуникационни технологии (ИКТ). Тези компоненти могат да се интегрират като една самостоятелна система, разположена в единен изчислителен център – без разпределение, или да се разделят и инсталират за работа в отделни физически сървъри, разположение на различни географски места, т.е. да формират разпределената система. По-долу се представят принципите на разпределение, предложени в настоящата студия, като първоначално се обръща внимание на архитектурните компоненти, които могат да бъдат разпределени.

3.1. Архитектурни компоненти

3.1.1. Склад от данни

Съществуват множество дефиниции за Склад от данни. Те обхващат различни аспекти на това понятие – "Хранилище от организирани електронни данни" [1]; "Колекция от данни от различни източници, организирани да подпомагат вземане на решения" [2]; "Интегрирани данни от различни оперативни системи, които типично се зареждат през определени интервали" [3]; "Колекция от данни, структурирани и проектирани за бъдещи заявки, вземане на решения, извличане на данни и откриване на знания" [4]; "Тематично ориентирана, интегрирана, времевариантна, неизменчива съвкупност от данни, подпомагаща вземането на решения" [5]. Всичките тези избрани дефиниции покриват същността на термина "Складове от данни", използван в настоящата студия.

Складът от данни може да се разположи в няколко компютъра и може да съдържа няколко бази от данни, както и да съдържа информация от многобройни източници в различни формати. Достъпът до Склада от данни е прозрачен за потребителя (бизнес анализатора), който може да използва прости команди, за да открие и анализира необходимата информация. Складът от данни съдържа също и мета данни – информация за това как е организиран самият склад от данни, където може да се открие информацията в него и какви са връзките между данните. Мета данните в Складовете от данни осигуряват въведените данни да са в правилния формат и да притежават съответните зависимости с другите данни в Складовете от данни. Складът от данни обикновено съдържа огромно количество данни, организирани така, че да подпомагат вземането на решения. Складът от данни се обновява с информация както в пакетен режим през определени периоди, така и в онлайн режим.

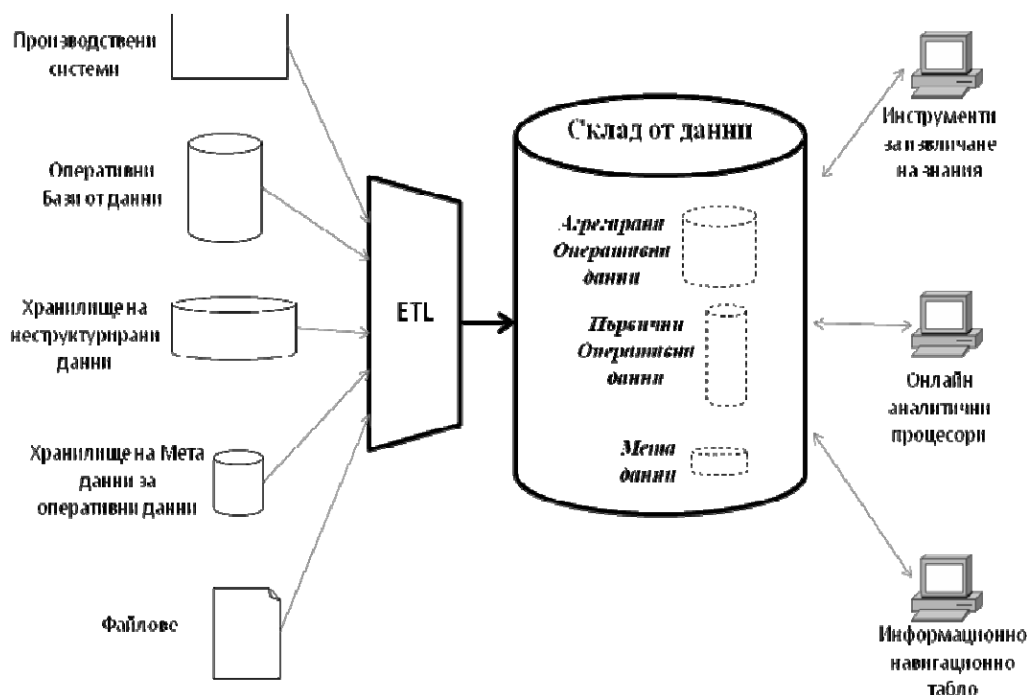
По принцип данните могат да се класифицират от архитектурна гледна точка в следните единици, всяка от които може да бъде разпределена компонента:

- Оперативни данни (например продажба на стока за определена цена, в определено време, в определен град);
- Агрегирани данни (например сумата на продадените стоки за един месец);

- Схема на база от данни (например таблиците за продажби, цени, места за продажби със спецификация на техните колони, техните първични ключове и връзките между тях);
- Мета данни (например информацията за продажбите се състои от 3 таблици, разположени в два физически сървъра с определени IP адреси, със задължителни параметри номер на продажба и цена, с цена на продажба в диапазона 50-100 лева и пр.);
- Бизнес правила (например въвеждането на цената на продажбата трябва да бъде в определения диапазон и да се записва в таблицата Цени, която се намира на физически сървър с адрес определен в мета данните).

Данните в Склада от данни се зареждат чрез процеса ETL (Extract, Transform, Load), който включва извличане на данните от първичните източници, почистване и форматиране, проверка за дублиране, проверка за съответствие с ограниченията, както и зареждането им в склада. ETL извлича данни от всички възможни източници на данни – файлове в мейнфреймове и UNIX машини, оперативните бази от данни, в които производствените транзакции зареждат своите резултати, данни от Неструктурирани хранилища от документи (Content management systems), както и от хранилища на мета данни за оперативните данни във фирмата. Данните на Склада от данни се използват от другите компоненти на Бизнес интелигентните системи – Съставители на отчети и Информационно навигационно табло (Dashboards), Инструменти за извличане на знания (Data Mining), Онлайн аналитични процесори (OLAP) и информационни системи на самите предприятия. Концептуалната архитектура на Склада за данни е показана на фигура 2.

Складът от данни представлява компютърна система, състояща се от един или няколко сървъра, използваща Система за управление на бази от данни – за структурирани данни, а също и Система за управление на съдържание от данни – за неструктурирани данни. В някои Складове от данни се използват също така и файлове за съхранение на данни. Дизайнът на схемата на базата от данни, използвана в Склад от данни, е съобразен с това какви отчети ще се получават, какви аналитични процеси ще се изпълняват и какъв тип решения се очакват да произлязат от данните. Данните в Складовете от данни могат да се разделят на части, наречени Екстракти от данни (Data Marts). Схемите на Екстрактите от данни често се обединяват в така наречени "кубове от данни". Няма стандарт, който да определя точно в каква форма да бъдат тези схеми за Екстракти от данни. Складовете от данни не са най-подходящото хранилище на неструктурирани данни. Поради това количеството на неструктурираните данни в съвременните Складове от данни е все по-ограничено.



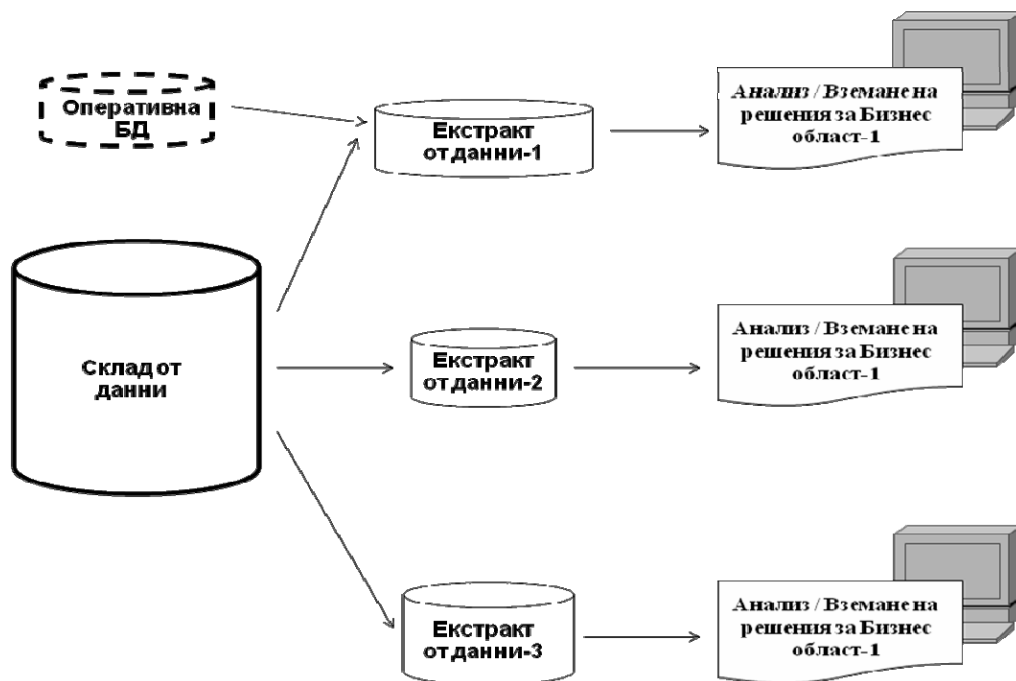
Фиг. 2

Конвенционалните системи за бази данни използват силно унифицирани (нормализирани) формати на данните, за да се осигури бърза обработка и онлайн оперативни справки. Форматите, които се използват в Складовете от данни, обикновено са по-свободни. Това ускорява получаването на резултати, които са многократно по-комплексни, но пък не се формират достатъчно бързо за съответствие на онлайн режима. Онлайн аналитичните процесори работят с "денормализирани" бази данни, въпреки че има инструменти, които работят със специални схеми за Складове данни, съхранени в трета нормална реляционна форма (third normal form), т.е. в нормализирани реляционни бази от данни. Данните, които се вкарват в Складовете данни, обикновено са във форма на обединение, агрегиране или селектиране по определен признак (например време). Процесът на такова преобразуване се осъществява чрез ETL инструментите. При такова преобразуване се осъществява също и премахване на несъвместимостта между отделните данни. Поради разделянето на Складовете от данни от Оперативните бази от данни, оперативните производствени системи на предприятието, работещи с оперативните бази от данни, няма да се забавят в тяхното изпълнение от работата на Складовете от данни и ползването на информацията в тях. Поради забавяне на процеса на извличане на данните от Оперативните бази от данни, както и агрегирането на оперативните данни, заредени в Складовете от данни, то данните в Складовете от данни нямат оперативен характер.

От гледна точка на функционирането Складовете от данни може да се класифицират в:

- Офлайн оперативни бази от данни, представляващи копие на оперативните бази от данни (снимка на данните в определен момент от време), но работещи в офлайн режим от оперативни системи;
- Офлайн складове от данни, които се зареждат от Оперативните бази от данни през определен регулярен момент от време и имат различна схема от Оперативните бази от данни;
- Складове от данни в реално време, при които данните от оперативните бази от данни създават актуализиране в тях при всяка транзакция с оперативните бази от данни;
- Интегрирани складове от данни, при които се използват повече от един склад от данни, всеки с отделна структура на данните, но обработващи единни заявки за справки и отчети.

Екстрактът от данни представлява подмножество на данните в една организация, за да поддържа определена бизнес функция или да обслужва определено бизнес звено в организацията – фигура 3.



Фиг. 3

Обикновено Екстрактът от данни е подмножество на Складовете от данни, като рядко представлява подмножество на Оперативни бази от данни. Много често Екстрактът от данни се разглежда като Мини-склад от данни – физически или логически. Когато Екстрактът от данни се разглежда логически като разделен от Склада от данни, то физически той се намира на друг сървър, различен от този, на който се намира Складът от данни. Възможно е обаче един Склад от данни да се разглежда

като съставен от няколко Екстракта от данни и тогава всичките Екстракти от данни са разположени на сървъра на Склада от данни. Естествено, съществува и смесена схема, при която едни Екстракти от данни се разполагат в сървъра на Склада от данни, а други се разполагат на отделен физически сървър. Съществуват и практически положения, при които съдържанието на Екстракт от данни се формира не само от данни в Склада от данни, но и от данни от някои Оперативни бази от данни, стига да се покриват изискванията за данни, нужни за провеждането на съответния анализ. Логически Екстрактът от данни се създава при наличие на необходимост от анализ на определени бизнес дейности (например дейностите на едно бизнес звено в организацията). По тази причина може да се приеме, че Екстрактът от данни се създава за обслужване на определен вид бизнес анализ, докато Складът от данни се създава въз основа на това какви оперативни данни съществуват в организацията и как те да се групират за последващо използване за голям брой бизнес справки или за фирмата/организацията като цяло.

Съвкупността от Оперативни бази от данни, Склада от данни и Екстрактите от данни може да се разглежда като съвкупност от съставните части на Обединена архитектура на данни на БИС.

Съществуват следните причини за създаване на Екстракти от данни:

- Необходимост от определена схема на данните за даден вид бизнес анализи;
- Лесно програмиране на анализите, които използват специално предназначена схема на данни;
- Лесен достъп до често нужни за използване данни;
- Създаване на обобщен поглед върху данните за определена потребителска група;
- Осигуряване на кратко време за достъп до нужни за анализи данни;
- По-евтина поддръжка на сървър за Екстракт от данни, отколкото на комплексен сървър (сървъри) за Склад от данни;
- Увеличава се сигурността на данните, защото дадена група от потребители имат достъп само до определен Екстракт от данни, а не до целия Склад от данни.

3.1.2. Онлайн аналитични процесори

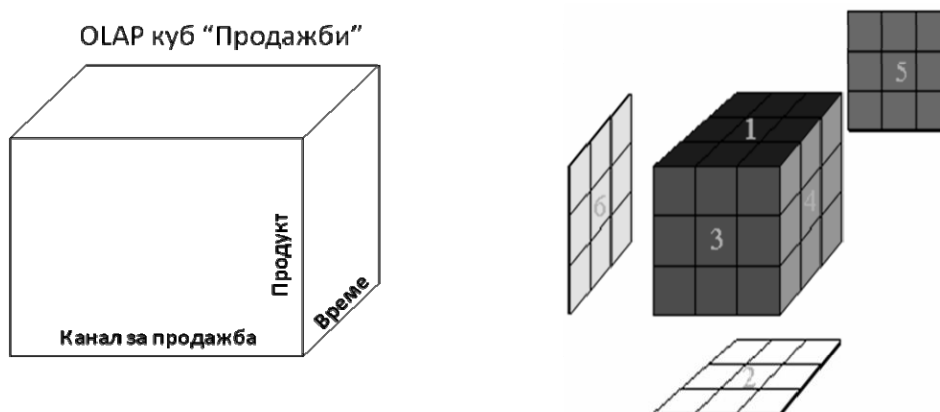
През 1993 година лидерите на релационните бази от данни – Код & Код, формулират термина OLAP ("On-Line Analytical Processing"), свързан с въпросите на разработването на многомерни модели. Идеята за обработка и анализ на многомерни модели съществува от 1962 г. след публикацията на книгата на Кен Айверсън "Език за програмиране" и поражда APL езика, внедрен от IBM и работещ със специални екрани, клавиатури и принтери. Въпреки своите специфики, APL е използван в много бизнес приложения от 80-те години, включително за редица икономически модели, които имат подобни функции на днешните OLAP системи. УНСС (тогава ВИИ "Карл Маркс") е единственото научно звено в България по това време, използвало APL за икономическо моделиране.

В началото на 80-те години се появява продукта System W на Comshare, който е първият продукт, работещ не с бази от данни, а с така наречения "хиперкуб-подход", и е ориентиран към разработката на финансови приложения от крайния потребител. От 2001 година Oracle обяви създаването OLAP Services (услуги) в Oracle9i. Oracle9i OLAP Services включва версия на софтуерния продукт Express engine, известен със своя академичен произход и наричан Аналитичната работна среда, както и нов ROLAP (Relational OLAP) инструмент. По онова време се създават технологически решения, свързани с въвеждането на много нови концепции, които все още не се използват широко, като например пълни непроцедурни правила, многомерно показване на цял екран и редактиране на данни, автоматична рераккалкулация и (партидна) интеграция на релационни данни и пр. Също така се създава и продуктът Essbase на Hyperion Solution, който не е директен наследник на System W, но е повлиян от неговия финансово ориентиран, напълно преизчислен подход с хиперкуб. В последствие се намесват продуктите за електронни таблици (Lotus и Excel), които работят с многомерни масиви от данни. В момента създателите на Системите за управление на релационни бази от данни (такива като Oracle, IBM, Microsoft,) предлагат OLAP продукти като градивни компоненти над техните СУРБД.

Същински архитектурен компонент на Онлайн аналитичните процесори (OLAP) е Analysis сървърът. OLAP се състои от две части – OLAP сървър и OLAP клиент. OLAP сървърът представлява средство за паралелна обработка на данни, създадено специално за целите на извличане на информация от релационни бази данни. Трансформирането на тези данни в ценна бизнес информация с възможности за анализ в множество измерения се извършва обикновено от OLAP клиента. OLAP клиентът е софтуерна компонента, която може да се зареди на хардуерния сървър, на който работи OLAP сървърът, а може да се зареди и на отделен персонален компютър. Самият OLAP анализ често се представя и визуализира като многомерен куб. Всяка страна на куба представлява отделно измерение на провеждания анализ. Всяко измерение на куба може да бъде различна категория бизнес данни, като например OLAP кубът "Продажби" може да се представи със следните видове данни – продукти, време, географски региони или канали за продажба – фигура 4. Многомерният куб много често има повече от 3 измерения, като в този случай носи названието "хипер куб".

Онлайн аналитичните процесори извършват многомерни аналитични запитвания върху данните – хипер OLAP кубове [8]. OLAP кубът се състои от *измерения* и конкретни стойности на данните, наречени *мерки*, изразяващи се чрез числови факти. Във всяко измерение може да се дефинира йерархия. Например за категорията време, йерархията може да включва година, тримесечие, месец и ден. Всяко измерение на куба се състои от Наименование и Мета-данни, свързани с него. Метаданните на куба описват от кои измерения се състои всеки куб и структурата на всяко измерение, като се създават или по схема "звезда", или чрез йерархически разширяема структура от релационни таблици. Всяко едно от измеренията може да се представи като йерархия от различни вариантности, например времето може да се представи чрез йерархията Ден->Месец->Година, или чрез Седмица-Тримесечие-Година.

Мерките са получени от таблицата на фактите, а измеренията са получени от таблицата на измеренията. Бизнес анализаторът, работещ с OLAP кубове, би желал да има "основни данни", около които да се извършват различни вариантни решения. Всяка клетка на куба се състои от многомерна структура, съдържаща данните, свързани с всяко от измеренията.



Фиг. 4

Съществуват интегрирани бизнес системи, такива като ERP (Enterprise Resource Planning) системи, при които OLAP свързът е интегриран към цялостната система. Това е възможно благодарение на използването на дефинициите на OLAP кубовете и техните измерения от ERP системата. За увеличаване на ефективността на OLAP клиента съществува възможност за експортиране на резултатите от анализа в електронни таблици. Това намалява разходите за допълнително обучение на потребителите, създава единен вид на Графическия потребителски интерфейс и съкращава времето за работа. С помощта на интегрираните средства за визуализация управленските справки много лесно и бързо могат да бъдат видоизменяни за различните цели на анализа.

Бизнес анализаторите чрез многомерния анализ могат да разберат смисъла на данните, съдържащи се в оперативните складове от данни (бази от данни). Анализаторът може да навигира в базата от данни чрез наблюдение на определено подмножество от данните, променяйки (добавяйки) измеренията на куба и калкулациите за формиране на измеренията. Този ръчен и итерационен процес, свързан с обхождане през данните, създаване и наблюдение на агрегации от данни и придвижване нагоре-надолу по нивата на агрегиране, както и с формиране на примерни кубове и наблюдение на отделните страни на кубовете, се нарича *формиране на страни и кубове (slice and dice)*. При опериране по създаването на OLAP кубовете се извършват следните операции:

- Създаване на резен (Slice). Резенът е двумерна матрица, при която всички останали измерения имат само една стойност. При тази операция се избират две измерения за създаването на резена, при фиксиране на определена стойност за всяко от останалите измерения.

- Създаване на зар (Dice). Зарът (по аналогия на тримерния куб) се изгражда от множество резени, при които се използват данни за едно или няколко от измеренията, които са имали фиксирани стойности при създаването на самите резени.
- Придвижване надолу/нагоре в йерархията на данни (Drill down/up). Тази операция извършва придвижване надолу по йерархията на агрегиране на данните до достигане на елементарните данни, или придвижване нагоре по йерархията на агрегацията на данни, достигайки по-високи нива на агрегация.
- Формиране на измерение (Roll-up). Тази операция въвлича в изчисления всички данни, отнасящи се към едно измерение.
- Промяна на ориентацията на измерение (Pivot). Операцията се използва при визуализиране на резултатите от анализа върху OLAP куба при извеждане на екрана или при формиране на отчет.

Съществуващите OLAP продукти предоставят множество методи за създаване на кубове, както и за обвързване на кубове. Обвързването на кубовете се прави с цел преодоляване на проблема, наречен "оскъдица" (sparsity). Оскъдицата се получава, когато с добавянето на измерение към куба се получават клетки, в които няма данни за дадено измерение. Оскъдицата може да се нарече още "недостатъчна празнота" на куба и нейното наличие води до изразходване на изчислителни ресурси за обработка на част от куба, в който няма данни. Например за всеки приход може да съществуват данни, свързани с отделен клиент и продукт, но разходите за всеки отделен приход може да не съществуват. В такъв случай, вместо да се създава един куб с определена празнота, то е по-добре да се създадат няколко свързани куба, всеки от тях без оскъдица. Обвързването на кубовете осигурява съвместимост на данните между тях.

Обикновено анализът, извършван върху OLAP куб, взема 0,1 от времето което би било необходимо, ако се използват данните от оперативните бази от данни [9]. Това се дължи на агрегирането на данните в кубовете в сравнение с оперативните бази от данни. Агрегирането се извършва върху таблицата на фактите, с прилагане на различно ниво на детайлност (granularity). Броят на агрегациите зависи от броя на измеренията, използвани в куба. Обикновено първоначално се изчисляват само няколко агрегации, а другите се изчисляват при необходимост. Процесът кои агрегации да се изчислят се нарича избор на вариант на решение. При работа с OLAP анализи се извършва баланс между броя на агрегациите, определящ времето за получаване на OLAP резултатите, и времето за формиране на отделните агрегации. В резултат на това цялостното време за създаване на OLAP резултата трябва да е оптимизирано, като за целта се използват множество различни видове алгоритми.

Както при базите от данни, когато един потребител започне промяна в таблица, тази таблица (или съответният ред от нея) трябва да се заключи за достъп от други потребители и такова заключване трябва да има и на ниво OLAP куб, или на част от куб. Съществува също и възможност след промяна на данните в куб да се генерира съобщение, че резултатите от изчисленията анализи (например обща сума) не са вече валидни поради промяна в данните. Съществуват обаче и възможности всички изчисления да се извършват динамично, по време на провеждане на самия анализ.

OLAP сървърите се различават според начина за съхранение на OLAP кубовете и могат да бъдат:

- Релационни OLAP (ROLAP)

При тях данните и измеренията се съхраняват в релационни бази от данни под формата на таблици. Съдържанието на отделните клетки на куба също се съхранява в релационни таблици. ROLAP не изисква предварително изчисляване на агрегация и измерения. Всички резултати се получават чрез SQL оператори, които извършват нужните калкулации. Схемата на базата от данни, представляваща ROLAP, се различава съществено от схемата на оперативните бази от данни, защото те удовлетворяват различни изисквания. По тази причина подготвителната работа за опериране с ROLAP е значително по-малка, докато времето за провеждане на един анализ е съществено (за всеки провеждан анализ се извършват съответните калкулации).

- Многомерни OLAP (MOLAP)

MOLAP е алтернатива на ROLAP технологията, независимо че и двата аналитични инструменти имат за цел да извършат анализа на данни чрез използване на многомерен модел на данните. MOLAP се различава значително по това, че се изисква предварително изчисляване и съхраняване на информация за куба – операцията, известна като *преработка*. MOLAP съхранява тази информация в оптимизирани многомерни масиви, а не в релационни таблици (както е при ROLAP).

- Хибридни OLAP (HOLAP)

HOLAP представляват комбинация от ROLAP и MOLAP. При HOLAP системите част от данните на куба се съхраняват в MOLAP склад, а част в релационни таблици. В различните програмни реализации това разделение варира. Едно примерно разделение е следното – агрегираните данни да се съхраняват в MOLAP склад за по-бързо търсене, докато неагрегираните (детайлни) данни се съхраняват в релационни таблици.

Дълго време не съществуваше стандартизиран език за достъп до OLAP кубове (така, както например съществува езикът SQL като стандартен език за дефиниране и манипулиране на данните в релационни таблици). През 1997 се създаде езикът MDX за работа с OLAP кубове. Този език бе усъвършенстван през 2001 с добавяне на XML като база за езика, чрез което MDX се явява де-факто индустриален стандарт за опериране с OLAP кубове.

3.1.3. Извличане на знания

Извличането на знания е процес на извличане на скрити модели (шаблони) от данни с бизнес значение. Извличането на знания се извършва от източник на данни – Склада от данни. Имайки предвид, че средно на всеки 3 години количеството на данните в една организация се удвоява [10], се налага съществуването на инструмент за преобразуване на данните в информация. Докато Извличането на знания може да се използва за формиране на модели от конкретните данни, то може да съществуват други конкретни данни, които не влизат в нито един модел. По същия начин извличането на знания от конкретни данни няма да разпознае формиран модел, ако конкретните данни не го покриват напълно. Както всеки друг инструмент,

Извличането на знания функционира ефективно при наличие на подходящата суровина – в този случай показателни и представителни данни, които трябва да се извлекат от Склада от данни. Освен това откриването на конкретен модел в даден набор от данни, не означава непременно, че моделът е представителен за всички данни в Склада от данни и по-специално за всички Екстракти от данни във и извън Склада. Ето защо важна част от процеса на извличане на знания е проверката и валидирането на формираните модели на данни върху съответните Екстракти от данни.

Основната цел на Извличането на данни е да се помогне на анализа на събраните данни за идентифициране на определено поведение [11]. В такива данни обикновено съществуват неизвестни взаимоотношения. Определени модели от данни могат да са валидни за едно подмножество от данните, подавани за Извличане, като може да не са валидни за друго подмножество от данните. За да се преодолее този проблем, бизнес анализаторът трябва да е екипиран с различни подходи, един от които е Избор на модел, създавайки експериментален проект на модела посредством създаване или премахване на определени корелации.

Извличането на данни може да открие модели на данни сред първичните данни, които са му подадени. По тази причина първичните данни трябва да са достатъчно големи по обем и по възможност да отразяват процесите, случили се във времето. За източник на първични данни се използват Складове от данни (Data Warehouses) или Екстракти от данни (Data marts). Извличането на данни е итеративен процес, свързан с начално обучение на системата и последващо разпознаване на модела/моделите от данни. Съществуват следните 3 фази в Извличането на данни (всяка от тях може да е итеративна сама за себе си):

- Предпроцесорна обработка;
- Създаване на модели от данни;
- Генериране на резултатните модели от знания.

Първичните данни, преди да се подадат към Извличането на знания, трябва да се "почистят". Това почистване означава да се премахнат данните, третиран като "шум", и липсващите данни. По този начин почистените данни се намалява до нивото, наречено "Вектор на отличаването (Feature Vector)". Този Вектор на отличаването съдържа първичните данни върху които ще извършва извличане на знания. Например черно-бяло изображение на лице може да се снее чрез растер 100 пиксела x 100 пиксела, или общо 10 000 бита първични данни преди изчистване. Ако се работи с алгоритъм за разпознаване на лице, което използва положението и формата на очите и устата, то тези първични данни може да се почистят и да се създаде Вектор на отличаването, съдържащ данни само за очите и устата, които данни ще бъдат много по-малко от тези преди изчистването. По този начин мястото за съхранение на цифровия образ ще изисква много по-малко памет и времето за разпознаване на образа ще бъде много по-кратко. Затова почистването на данните не се извършва самоцелно, а се прави с оглед на бъдещото формиране на различните модели на данни. За едно множество от първични данни се дефинират два Вектора на отличаването – Трениращ вектор и Валидиращ вектор. Трениращият вектор има за цел да "обучи" алгоритмите за извличане на данни да разпознават определени мо-

дели на данни, докато Валидиращият вектор служи за реално разпознаване на валидни модели на данни сред пълното множество от данни. Всички тези действия по почистването на първичните данни може да се разглеждат като Предпроцесорна обработка.

Основният процес по Създаване на модели на данни се състои от следните 4 главни стъпки (представени чрез съответни групи алгоритми), като тези стъпки се прилагат върху данни определени като Трениращ вектор на отличаване:

- а) *Класифициране* – свързването на данните към предварително определени групи. Например имейл съобщенията могат да се класифицират на съобщения-спам и нормални съобщения. Използват се алгоритми като: Алгоритъмът на к-близкият съсед [12] (извършващ класификация на обекти на база на близките обучени примери в Трениращия вектор на отличаването); Наивна класификация на Байес [13] (Представява вероятностна класификация, основана на теоремата на Байес и предполагаща наличие/липса на определена функция на класа, несвързана с наличието/липсата на всяка друга функция. Например един плод може да се приеме че е диня, ако е зелен на цвят, кръгъл и с диаметър поне 25 сантиметра); и Невронни мрежи [14] (алгоритъм, опитващ се да функционира по подобие на биологическите невронни мрежи, състоящ се от изкуствени неврони, обработващи входяща информация на база на комуникационен подход за изчисления. Тази мрежа представлява адаптивна система, която си сменя комуникационната структура на базата на предишни преминали през нея данни и извършени процеси);
- б) *Групиране* – действа като Класифицирането, но предварително не са определени групите и алгоритъмът формира отделни сходни единици от данни в групи. Един от често използваните алгоритми е Дърво на решенията [16] (използващ граф под формата на разклонено дърво на решенията, като на всяко разклонение се съставят възможни следващи разклонения, използвани ресурси и тяхната цена, а също и шанс за получаване на определени резултати, както и за определяне на вероятност за получаване на определено решение);
- в) *Регресия* – опит да се намери функция, която моделира данните с най-малко грешки. Обикновено използва методите на Генетичното програмиране [15] (част от Изкуствения интелект, използваща методологията на еволюционните и генетичните алгоритми);
- г) *Асоциативно обучение по правила* – провеждащ търсене на връзки между променливи, например при регистриране на продажби в магазин може да се открие кои продукти се продават най-много след 18 часа и продажбата на кой продукт е свързана с продажба на друг продукт (за да се подредят стоките в магазина въз основа на тази логика за клиенти, идващи след 18 часа).

Посочените стъпки показват, че няма единен подход за формиране на моделите на данни. Освен това за определяне на даден модел от данни може да се приложат няколко подхода. Посочените по-горе цитирания са отправни точки за последващо откриване на задълбочени и детайлни знания за споменатите алгоритми, при необходимост за бъдещото им прилагане.

След като се осъществи Основният процес по формиране модели на данни работещ с Трениращия вектор на отличаване, обикновено се пристъпва към Генериране на резултатните модели от знания. Този процес представлява прилагането на откритите модели от данни върху Валидиращия вектор на отличаването, т.е. да се провери до колко създадените модели са валидни за голямото количество данни, намиращи се във Валидиращия вектор на отличаване, и да се генерират реалните модели от данни. Не всички открити модели на база на Трениращия вектор на отличаване са валидни за данните, представени от Валидиращия вектор на отличаване. Освен това трябва да се анализира как откриването на моделите от данни работи с реални големи обеми от данни и при необходимост да се "донастроят" съответните алгоритми за откриване на моделите.

Съществува сериозна разлика между аналитичните резултати, формирани от Онлайн аналитичните процесори (OLAP) и тези, формирани от Извличането на знания (Data mining). Докато Онлайн Аналитичните процесори предоставят добра картина "какво" се е случило, то Извличането на знания обяснява "защо" се е случило и "какви са тенденциите" да се случи отново или да се случи в променен вид. Освен това Извличането на знания осъществява класифициране на получения резултат с други подобни, асоциира подобни явления, подрежда в последователност кое след кое явление би се случило и предвижда общата картина за бъдеще.

3.1.4. Навигационно табло

Като основа за дефиниране на Навигационно табло в БИС може да се използва дефиницията на Few [17] – визуално изобразяване на най-важната информация, необходима за постигане на една или повече цели, консолидирана и подредена в един екран, така че да се извлече информацията с "един поглед". В съвременните Навигационни табла се добавят три нови функции:

- Интерактивност;
- Възможност за анализиране на източниците (данни и информация) на представените еднокранни резултати;
- Организиране на информация за "един поглед" за различните управленски нива в организацията.

Интерактивността осигурява не само показване на резултати, но и диалог с потребителя, т.е. той да изисква нова форма на същите резултати, нови резултати или някои от еднокранните резултати да могат да се представят в по-подробна форма. Възможността за анализиране на източниците на еднокранните резултати е форма на така наречения drill-down, т.е. слизване нива надолу на агрегация до достигане на първични данни, породили формирането на крайните резултати. Навигационните табла не са предназначени за един вид бизнес анализатори, например ръководителите на организациите; те са предназначени да подпомагат всички нива на бизнес управлението. По тази причина трябва да съществуват еднокранни обобщени резултати, предназначени за отделни специалисти, например да се създадат еднокранни резултати освен за управляващите от най-високо ниво, но и за финансовите мениджъри, за бизнес мениджърите по отделните бизнес направления в организа-

цията, за техническите мениджъри, за ръководство на оперативното управление и т.н. По този начин Навигационните табла трябва да могат да предлагат множество еднокранни обобщени данни за различни управленски нива.

Навигационните табла не са информационни табла, а средство за управление. Затова в тях би трябвало да има възможности не само да се констатира какво става или какво може да се случи в организацията, но и при необходимост да се стига до визуализиране на различни нива на детайлност с оглед вземане на съответни управленски решения. Например, когато Индикаторът за производителност е спаднал под допустимата за организацията граница, чрез drill-down подхода да може да се стигне до причините, породили намаляването на производителността и до съответните бизнес звена, отговорни за това състояние.

Навигационните табла на БИС обикновено се използват за показване на производствени показатели, определени от организацията, като произведена продукция, продаден продукция, ниво на продажби по региони, дефекти на хиляда продадени изделия, мнения на клиенти и доставчици и др. Обикновено тези показатели са изразени като Ключови показатели за изпълнение (Key Performance Indicators – KPI), и типичното табло представя няколко KPI, корелациите между тях и целия аспект на бизнеса [18].

Типичен екран на БИС Навигационно табло е показан на фиг.5

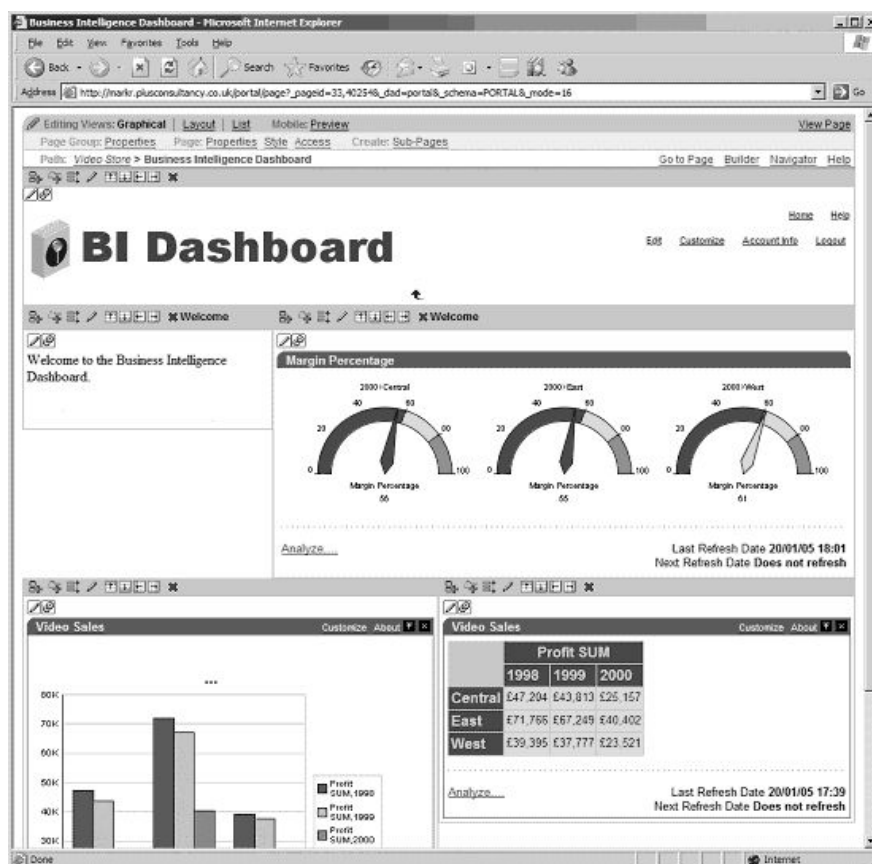
Стъпките по създаването на екран за визуализиране на Навигационно табло с използване на софтуерен продукт Портал могат да се сведат до:

а) Създаване на екранна страница на портала.

Порталът създава обединено представяне на екрани от различни Уеб сървъри, използвайки технологията на "portals", като в еднокранното представяне на Навигационното табло може да се включат Уеб страници и части от екрани (на клиент-сървърни приложения) от различни източници – всичките части на корпоративната информационна система. Тези данни може да се настройат според това къде в екрана на Портала трябва да се включат. В настоящата стъпка се извършва създаване на страница, нейното оформление чрез лого на организацията и основен текст за отгоре и отдолу на екрана, както и приветствие, представено в горната лява част на екрана. Всичко това е продиктувано от световните добри практики за създаване на Навигационно табло.

б) Създаване рамка за разполагане на информацията за основната страница на Портала – за визуализиране на графики, таблици и манометри.

Проектирането на еднокранното представяне на страница в Портала изисква да се проектира къде на екрана да се разположат графики, къде таблици и къде манометри (графични изображения на стрелкови уреди, които имат обикновено 3 области за съответно измерване – зелена, жълта и червена област). За всяка отделна информация се определя източникът, от който да се извлече информацията – IP адрес, URI адрес, както и частта от този URI, която ще се включи в екранната страница на Портала.



Фиг. 5

с) Посочване на конкретни типове данни за представяне при визуализиране.

По време на тази стъпка се избират типове данни, които ще се използват за всяка графика, таблица и манометър. Този процес се нарича кustomизиране с оглед на предоставяне на информацията за нужното ниво на управление.

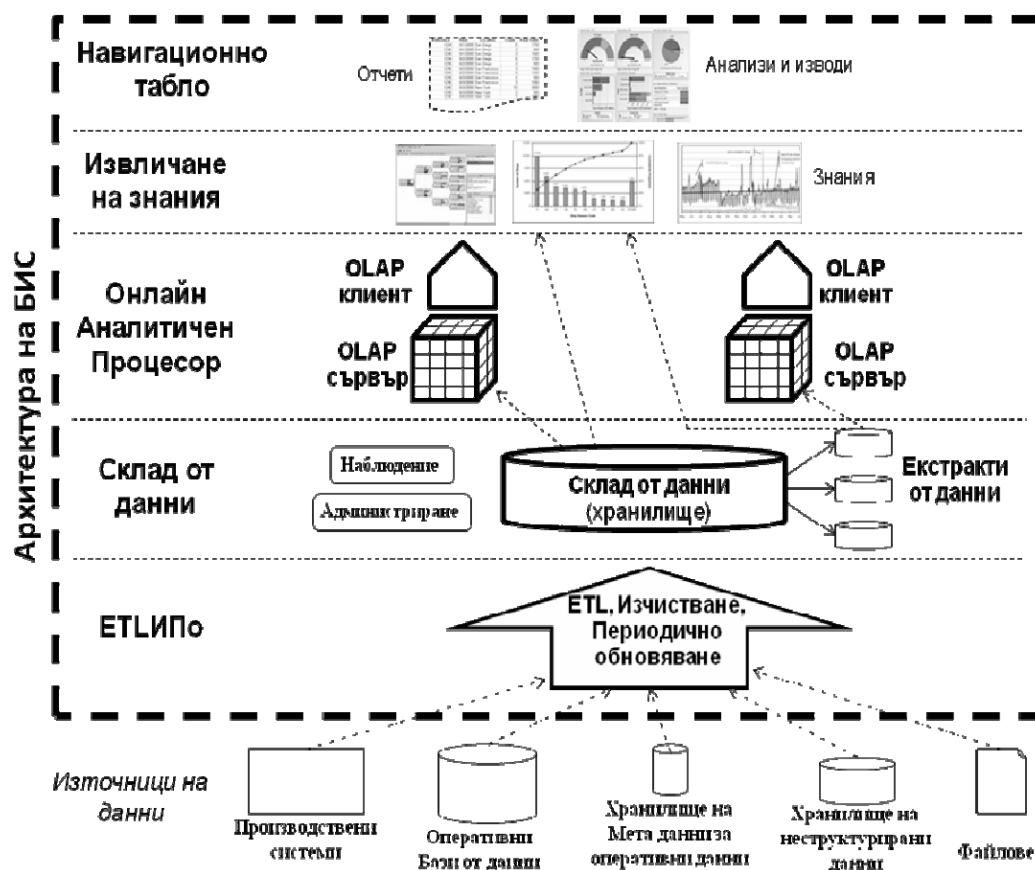
д) Създаване на контрол за достъп до еднокранната страница.

При необходимост информацията на екрана-страница може да се извежда само при подаване на коректно име и парола. Това се прави с оглед на факта, че такава интегрирана управленска информация, каквато се извежда на Навигационното табло, има конфиденциалност. За целта в екранната страница трябва да се задели място за въвеждане на име/парола и да се изгради списък с допустими стойности на име/парола.

Съществуват БИС, за които Навигационното табло не е така гъвкаво за изграждане на съдържание, и тези БИС допускат само определено конфигуриране на информацията.

3.2. Принцип на хоризонтално разпределение

Принципът се основава на постановката, че ИКТ архитектурата на БИС може да се представи графически отдолу нагоре, съответстващо на пътя на обработка на информацията, като отделните компоненти се подреждат по следния начин отдолу нагоре: ЕТЛИПо, Склад от данни, Онлайн аналитични процесори, Извличане на данни и Навигационни табла – фиг.6.



Фиг. 6

В посочената ИКТ архитектура най-отдолу се намират източниците на информация, които захранват БИС системата. Тази информация се обработва от БИС, като крайните резултати се получават на Навигационното табло, т.е. потребителите получават всички междинни и крайни резултати от работата на БИС чрез Навигационното табло. Принципът на хоризонтално разпределение представлява разделяне на ИКТ архитектурата на БИС чрез хоризонтални разделители, като един разделител е между два архитектурни елемента на БИС. Всеки архитектурен елемент представлява една архитектурна компонента, описана по-горе в студията, или ней-

на съставна част, но имаща пълно завършена функционалност. Прилагайки тези принципи, ИКТ архитектурата на БИС може да се раздели на следните отделни архитектурни елементи:

- a) ЕТЛИПо – извършващ процесите по Извличане, Трансформиране и Зареждане (Extract, Transform and Load –ETL), както и Изчистване и Периодично обновление. Изчистването се прави на регулярни периоди от време с оглед да се подадат към Склада от данни само бизнес валидни данни. За провеждане на изчистването може да се използват различни инструменти, такива като например IBM Information Server [19]. Зареждането на данните в Склада от данни става чрез друга периодичност – тази на обновлението, отговаряща на периодичността на актуализиране данните в Склада от данни. Двете периодичности трябва или да са синхронизирани, или Изчистването да е с по-къса периодичност, отколкото обновлението.
- b) Склад от данни, в които се включват и Екстрактите от данни, заедно с тяхното управление;
- c) Онлайн Аналитичен процесор (OLAP), чиито OLAP сървъри получават информация от Склад за данни;
- d) Извличане на знания, които получават информация също от Склад за данни, като елементът може да работи в паралел с Онлайн Аналитичен процесор за комплексни изводи или самостоятелно;
- e) Навигационно табло.

Всеки един от тези архитектурни елементи може да бъде зареден на отделен физически сървър и връзката с по-долно и по-горно ниво в ИКТ архитектурата може да се осъществява чрез WAN и/или LAN. По този начин разпределената ИКТ архитектура на БИС може се представи чрез 5 разделени архитектурни елемента максимум.

3.3. Принцип на вертикално разпределение

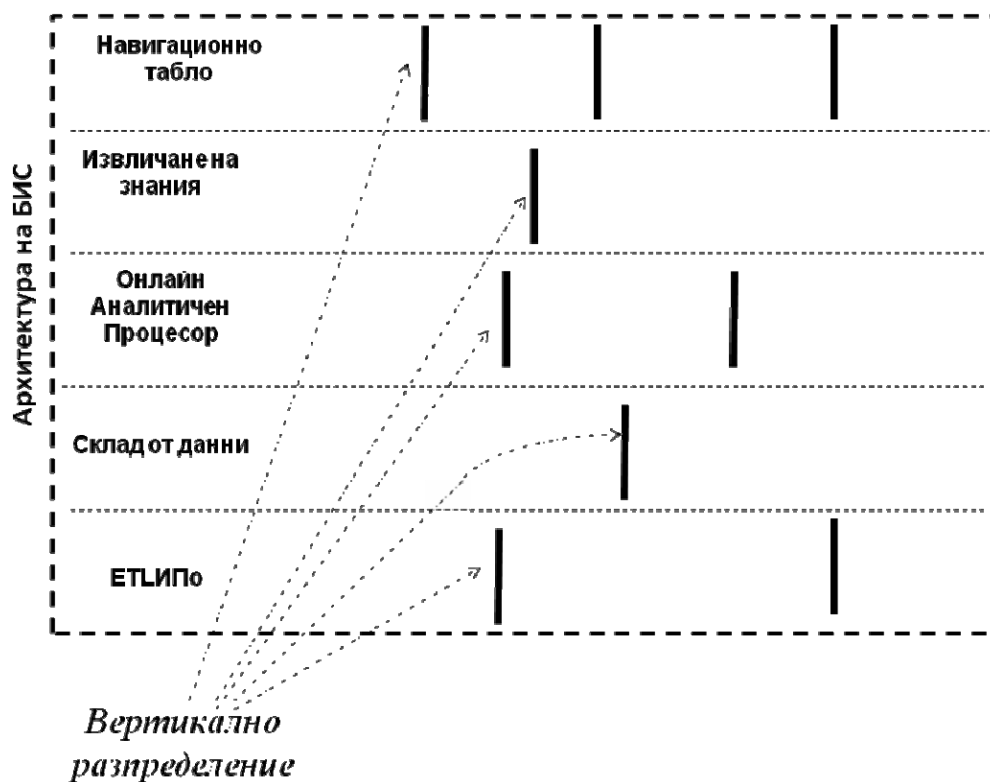
Принципът на вертикалното разпределение се основава на обстоятелството, че всеки архитектурен елемент от посочената по-горе ИКТ архитектура на БИС може да се раздели на поделементи, например Склада от данни да се раздели на няколко физически изградени архитектурни поделемента, или Онлайн аналитичният процесор да се раздели на няколко поделемента. Такова разделение графически се представя като вертикално разпределение, откъдето идва и названието на принципа – фиг. 7.

Архитектурният елемент "ЕТЛИПо" може да се раздели максимално на толкова елемента, от колкото източника се събира информацията за Склада от данни. Такова разпределение може да бъде също и на принципа на отделен поделемент за всеки тип източник.

Архитектурният елемент "Склад от данни" може да се разпредели по следните принципи:

- Разделяне на Склад от данни – Хранилище и Екстракти от данни, като отделните Екстракти може да се разположат в различни физически локации или да се обособят на принципа кое бизнес звено каква информация ще ползва;

- Самият Склад от данни да се разпредели на няколко физически сървъра, разположени физически на различни места, имащи една и съща схема на данни, или имащи различни схеми на данни;



Фиг. 7

Архитектурният елемент "Онлайн аналитични процесори" може да се раздели на:

- отделен Онлайн аналитичен процесор за всеки OLAP куб;
- един Онлайн аналитичен процесор за няколко OLAP куба;
- Онлайн аналитичен процесор за куб, разположен в Склад от данни и Онлайн аналитичен процес за куб, разположен в Екстракт от данни;
- Онлайн аналитичен процесор за всеки OLAP куб, разположен на отделен Екстракт от данни;
- Всеки OLAP куб в отделен OLAP сървър;

Архитектурният елемент "Извличане на знания" може да се раздели според:

- Фазата на обработка – Предпроцесорна обработка, Създаване на модели от данни и Генериране на резултатните модели от знания, като всяка фаза да се извършва чрез отделни физически сървъри, всеки от тях предназначен за дадената фаза;

- Отделните стъпки от Създаването на модели от данни, работещи с тотално различни алгоритми, като нужните за тях изчислителни мощности се изпълняват на отделни процесори.

Архитектурният елемент "Навигационно табло" може да се раздели на следните принципи:

- Няколко физически Уеб сървъра, създаващи отделни отчети, анали и изводи, и един сървър за Портал, обединяващ тяхната информация;
- За отделните нива на управление да се създава отделен физически Портал, който да осигурява нужната управленска информация според нивото на управление;
- За отделните бизнес направления в организацията да се създават отделни физически Портали, с възможност за drill-down до детайлни данни, отнасящи се само за съответните бизнес направления.

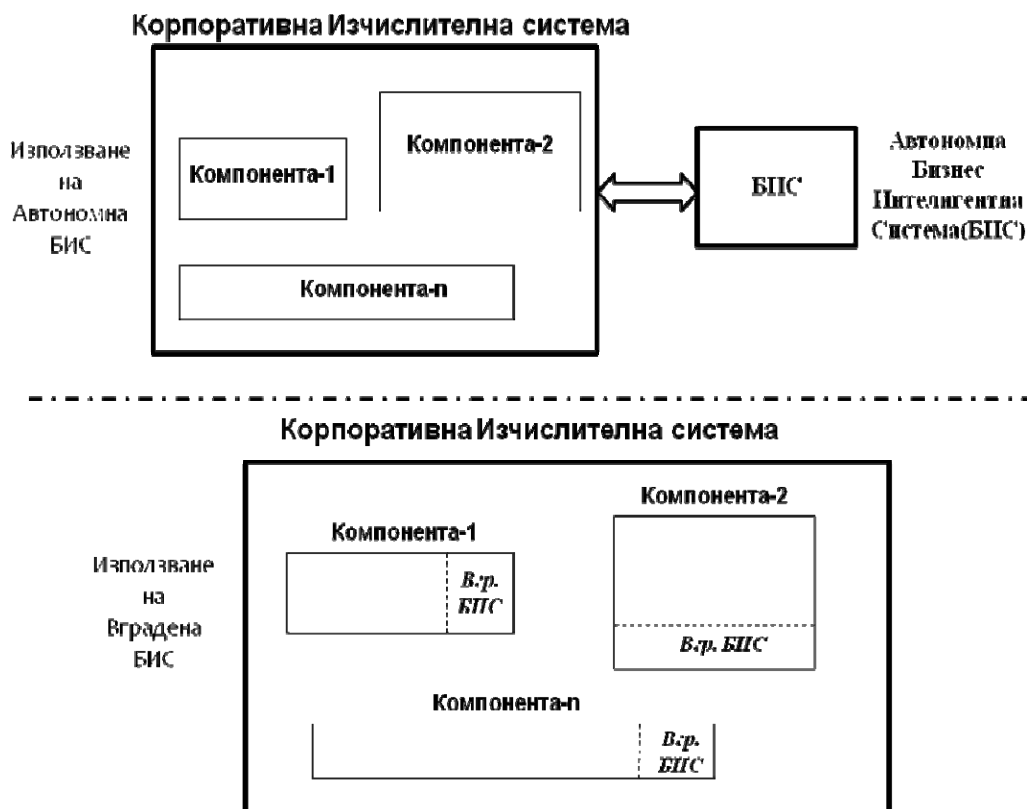
3.4. Принципи за разпределение с вградени компоненти

В съвременния свят все по-важно става вземането на бизнес решения да се извършва своевременно и на всяко ниво на управление. В същото време нуждата от бърза акомодация на динамично променящите се бизнес условия, нови законови разпоредби и повишеният акцент върху тяхното спазване принуждават фирмите да бъдат по-прецизни в процеса на вземане на решения. Тези противоречиви цели изискват многомерност, от една страна, и детайлност, от друга, на вземане на решенията във всички компоненти на Корпоративната изчислителна система. Освен това, тези решения, вземани на ниво на отделните компоненти, трябва да се синхронизират помежду си. Ключът към успеха е оперативните звена да могат да произведат повече решения, засягащи тяхната дейност. Типичен начин за забавяне на бизнес функционирането е да се изисква определени ключови оперативни решения да се вземат от високите управленски нива в организацията. Натискът за преместване на вземането на решения във всички нива на организацията създаде необходимостта от нов тип бизнес интелигентност и отчетност – вградената бизнес интелигентност. По този начин могат да се дефинират 2 главни типа Бизнес интелигентни системи – Автономни и Вградени. Основните архитектурни различия между тях са показани на фигура 8.

Всяка Вградена БИС е неразделна част от бизнес приложение или процес, а не отделно приложение или система, което е интегрирано, или което работи самостоятелно за формиране на резултати. Тя обикновено не е основен компонент на бизнес приложението, а спомагателен компонент, който предоставя резултати, представяни до крайния потребител във форма, съответстваща на самото приложение.

Вградената БИС е насочена за подпомагане на хората да разберат какво точно се случва в текущия момент, по възможност в реално време, в съответния бизнес процес или бизнес област. Затова се използват опростени форми на анализи и формиране на отчети, които са значително по-олекотени за работата от тези на Автономните БИС. Вградените БИС работят с данни в движение, а не с данни от Складове от данни. В повечето случаи данните се генерират от самото приложение, към кое-

то е свързана Вградената БИС. Вградените БИС не само оперират с данни, предоставени от самото бизнес приложение, но и формират резултатите във форма и вид, определен от самото приложение. По този начин потребителят не чувства разликата от стила на представяне на резултати, получени от функциите на самото бизнес приложение, както и тези, получени от Вградената БИС. Това помага на крайните потребители да се чувстват по-удобно с познатия Графичен потребителски интерфейс на приложението и им помага да увеличат своята производителност. Вградените БИС са много по-силно ориентирани към крайния потребител, отколкото Автономните БИС. Вградените БИС, от друга страна, са предназначени да бъдат част от бизнес процесите. При тях крайният потребител трябва да може да персонализира вида на съдържанието и формата на обобщените резултати, да формира варианти за решения, както и да осигури лесен достъп до данните, на основата на които са формирани крайните резултати.



Фиг. 8

Вградените БИС поставят много различни изисквания за разлика от традиционните Автономни БИС. Едно от тях е да има богат API, които да може да се използва от бизнес процеса, към който е вградена тази БИС. Тъй като Вградената БИС не разполага с данни в хомогенизирани склад, тя трябва да бъде в състояние да събира

данни от различни източници, включително бази данни, уеб-услуги и от работната памет на програмни обекти. В идеалния случай тя трябва да бъде в състояние да обедини и хомогенизира данните от различните източници преди да се подходи към тяхната обработка. Същевременно Вградената БИС трябва да има API, чрез който да отдава резултатите си към частта от бизнес процеса, която ще формира крайните резултати.

Архитектурата на Вградената БИС трябва да е съвместима с тази на бизнес приложението, към което се прикача. За разлика от обикновените програмни библиотеки, вградената БИС е по-скоро вградена система в самото бизнес приложение, а не добавена към него. Ако приложението е J2EE базирано, то БИС трябва да осигурява създаване на Servlets, JSPs и EJBs, които могат да бъдат добавени към бизнес приложението. Ако приложението е на база на .NET технологии, то БИС трябва да е също .Net-базирана система.

Вградените БИС трябва също да бъдат много адаптивни. Например заявка, отправена към бизнес приложението, трябва да може да се модифицира лесно като заявка, отправена към БИС. Всички цветове, шрифтове и т.н. трябва да бъдат лесно приспособими за типа Графичен потребителски интерфейс на бизнес приложението. Резултатите от БИС трябва да могат да се включват в стила на екраните на бизнес приложението. За да се постигне това, уеб приложенията обикновено използват хипервръзки, а десктоп приложенията работят чрез персонализиране на менюта и бутони в БИС средата.

Броят на Вградените БИС на пазара е сравнително по-малък от тези на Автономните БИС. Същевременно множество традиционни БИС имат маркетингово представяне, че могат да се вграждат към съществуващите ИКТ системи, но практически те не могат. Такива Автономни БИС имат свои бизнес процеси, имат свой облик на Графичния потребителски интерфейс и т.н. За преобладаващата роля на пазара на Автономните БИС има редица причини. Създателите на Автономните БИС са прекарвали много години в тяхното изграждане, работейки независимо от съществуването на множеството различни Корпоративни системи, и сега те се опитват да интегрират своите автономни системи към съществуващата ИКТ система на отделните фирми на основата на слабо свързани системи, без да се извършва вграждане на тези системи една в друга.

Докато Автономните БИС имат сложни функции за анализи и откриване на тенденции, създаване на отчети и механизъм за публикуване на резултатите, Вградените БИС притежават опростени функции, което им дава възможност да работят в реално време. Богатството на функции на Автономните БИС се явява проблем за Вградените БИС, защото те биха изисквали огромна изчислителна мощност на сървърите, на които ще работят, за да формират резултати синхронно с бързината на формирането на оперативните бизнес отчети. Вградените БИС подкрепят широко разнообразие на формати за отчети, имат поддръжка за графики и таблици, подкрепят множество изходни формати (HTML, PDF, Excel) и множество източници на данни. Те са проектирани от самото начало да бъдат включени в рамките на други системи, както и да си взаимодействат с приемаща и предаваща системи.

Вградените БИС биват два вида:

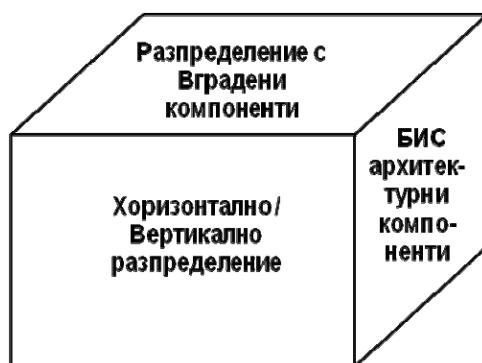
- Изпълняващи всичките функции на БИС и съществуващи като едно цяло;
- Изпълняващи само отделни функции на БИС, например "Извличане на знания".

В съвременния свят съществуват множество Вградени БИС с отворен код, които са във формата на единия или другия вид. Например съществуват продукти "Складове от данни", едните от които работят като Автономни Складове от данни, а другите работят като Вградени Складове от данни. Затова при изграждането на разпределени БИС на основата на Вградени БИС трябва внимателно да се избират съответните програмни продукти, които са от типа "вградени", а не само да претендират че са вградени.

4. ПРОЕКТИРАНЕ НА МЕТОДИ ЗА РАЗПРЕДЕЛЕНИЕ НА БИ СИСТЕМИ

4.1. Рамка за проектиране на методите за разпределение

Методите за разпределение се базират на трите принципа на разпределение, изложени по-горе: Хоризонтално разпределение, Вертикално разпределение и Разпределение с вградени компоненти. Използвайки тези три принципа, прилагани към БИС архитектурните компоненти, се получава Рамката за проектиране на методите за разпределение на БИС – фиг. 9



Фиг. 9

За всеки архитектурен компонент на БИС (например "Складове от данни") се създава Метод за разпределение, основаващ се на прилагането на принципите на хоризонталното разпределение, вертикалното разпределение и разпределението с вградени функции. И понеже се използват 4 архитектурни компоненти – Склад от данни, Онлайн аналитичен процесор, Извличане на данни и Навигационно табло, то се създават 4 метода за разпределение на БИС. Всеки метод обосновава създаването на набор от Системни елементи (СЕ), чрез които може да се изгради съответния БИС архитектурен компонент в разпределен вид. По този начин един метод обосновава множество Системни елементи, на които максимално може да се разпредели даденият БИС архитектурен компонент, т.е броят Системни елементи за даден ар-

хитектурен компонент показва максимално чрез колко елемента може да се проектира архитектурата на този компонент. От взаимодействието между хоризонталното и вертикално разпределение се получава матрица от елементи, към които, прилагайки вграденото разпределение, се получава съответен Системен елемент на БИС архитектурния компонент (например за архитектурния компонент "Склад от данни" се създават системни елементи, един от които е "Разпределение с използване на Екстракт от данни"). При практическа реализация на този компонент може да се използват както всичките обосновани и представени по-горе системни елементи за него, така и единно интегрирано решение. Това означава, че отделният БИС архитектурен компонент може да се създаде чрез едно интегрирано архитектурно решение или чрез решение, използващо множество Системни елементи, като техният максимален брой е определен от Метода за този архитектурен компонент. Например за архитектурния компонент "Склад от данни" методът му за разпределение определя 6 системни елемента и практическото реализиране на този архитектурен компонент може да варира от едно интегрирано архитектурно решение до използване на всичките 6 разпределени системни елемента. Тази вариантност определя възможностите и нивата за разпределено архитектурно решение на съответния архитектурен компонент. Един Системен елемент може да се създаде с прилагане или без прилагане на вграденото разпределение, т.е. той да се разглежда като тримерен или двумерен системен елемент. По този начин за всеки Системен елемент съществува вариантност – да се прилага със или без вградени компоненти (например да се използва Системен елемент с вграден компонент – "Разпределение с използване на Екстракт от данни работещ с Вграден Екстракт от данни, разположен в паметта", или Системен елемент без вграден компонент – "Разпределение с използване на Екстракт от данни, работещ със СУБД").

Характерът на Системния елемент се получава от взаимодействието на хоризонталното и вертикалното разпределение. Затова по-долу в студията, при разглеждането на отделните методи за разпределение, ще се представят методите и техните Системни елементи на база на хоризонтално и вертикално разпределение. Възможно е за даден Системен елемент в момента да не съществуват вградени компоненти, но след определено време в света да се създадат такива компоненти. Множеството вградени компоненти са с отворен код, а скоростта на разработка на софтуер с отворен код понякога е много голяма, като се има предвид огромното множество от разработчици в света, които могат да се включат в такава разработка. Освен това, за даден Системен елемент може да се прилагат няколко различни вградени компоненти, налични на пазара, което запазва методологическия характер на разработените методи и Системни компоненти, а практическото прилагане може да варира според конкретните условия на пазара. Ето защо разглежданите по-долу методи и Системни елементи обхващат БИС архитектурните компоненти, базирани само на хоризонтално и вертикално разпределение.

При разработването и представянето на отделните методи и Системни елементи, чрез които се извършва разпределението по дадения метод, е използвана следната система за наименуване на системни елементи:

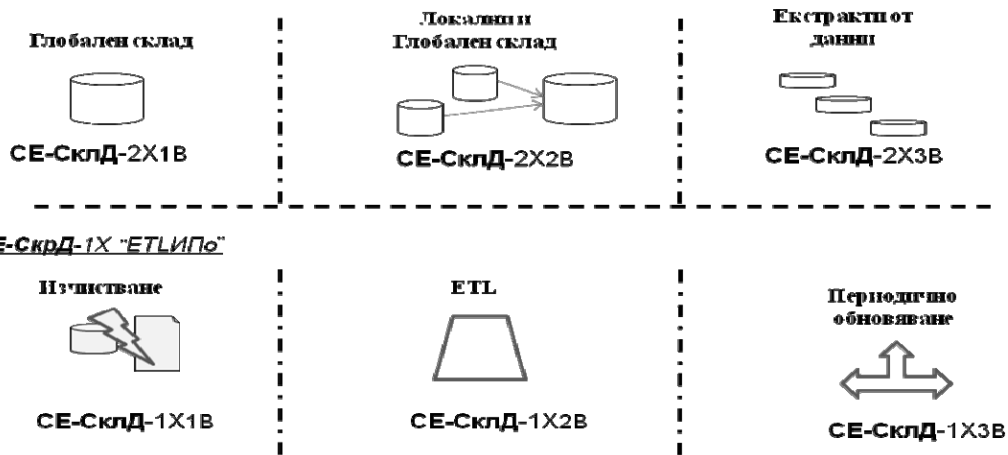
CE- α -nXmB

където "СЕ" означава Системен елемент, "α" е абревиация на съответния БИС Архитектурен компонент, за който е предназначен създаденият метод, "X" означава използван хоризонтален принцип на разпределение с "n" пореден номер, а "B" означава използван вертикален принцип на разпределение с "m" пореден номер. Използваните абревиации за Архитектурните компоненти са следните: "СклД" за Склад от данни, "OLAP" за Онлайн аналитичен процесор, "ИЗ" за Извличане на данни и "НТ" за Навигационно табло. При изписването на наименованието на елемента се прилага намаляване на фонта на B спрямо X, което означава съответно вертикално подниво на хоризонталното ниво в матрицата на системни елементи на метода. Например "СЕ-СклД-2X3B" означава метод за архитектурната компонента "Склад от данни", прилаган на второ хоризонтално ниво и трето вертикално ниво.

4.2. Метод за разпределение на Склад от данни

Методът, използван за разпределение на БИС архитектурната компонента "Склад от данни", и създаден на основата на представените по-горе принципи и чрез използване на проведените изследвания в литературата – [20],[21],[22], води до създаване на 6 системни елемента, представени на фигура 10.

СЕ-СклД-2X "Хранилища"



Фиг.10

Системните елементи за разпределение на архитектурната компонента "Склад от данни" са проектирани в 2 хоризонтални категории – ETLIPo (обхващащо процесите ETL, Изчистване и Периодично обновяване) и Хранилища. За всяко хоризонтално ниво са разработени по 3 вертикални нива, получавайки общо 6-те системни елемента. Тези системни елемента са:

- СЕ-СклД-1X1B – Изчистване;
- СЕ-СклД-1X2B – ETL;
- СЕ-СклД-1X3B – Периодично обновяване;

- **СЕ-СклД-2Х1В** – Глобален склад;
- **СЕ-СклД-2Х2В** – Локални и глобален склад;
- **СЕ-СклД-2Х3В** – Екстракти от данни.

Първите 3 и последният посочени системни елемента реализират съответните описани функции в началото на настоящата студия. Системният елемент "Глобален склад" означава, че се създава един Склад от данни за цялата БИС, независимо къде е разположен. Този склад може да бъде изграден чрез един или няколко сървъра, използвайки еднопроцесорни или многопроцесорни сървъри. Системният елемент "Локални и глобален склад" представлява форма на разпределение на Склада от данни на отделни логически единици разположени на различни географски места, като локалните складове съдържат или част от информацията на Глобалния склад, или са негово копие. Създаването на локални складове се налага поради съображения за бързодействие и/или надеждност. Изборът на метод **СЕ-СклД-2Х1В** или **СЕ-СклД-2Х2В** става не от съображения, свързани с разпределението на БИС, а от съображения за адекватност на предоставяните бизнес услуги (SLA) на предлаганата БИС.

Всеки системен елемент предоставя възможност да се реализира в отделна самостоятелна хардуерно-софтуерна форма, като информационните връзки между отделните хардуерно-софтуерни реализации на системните елементи се осъществяват чрез:

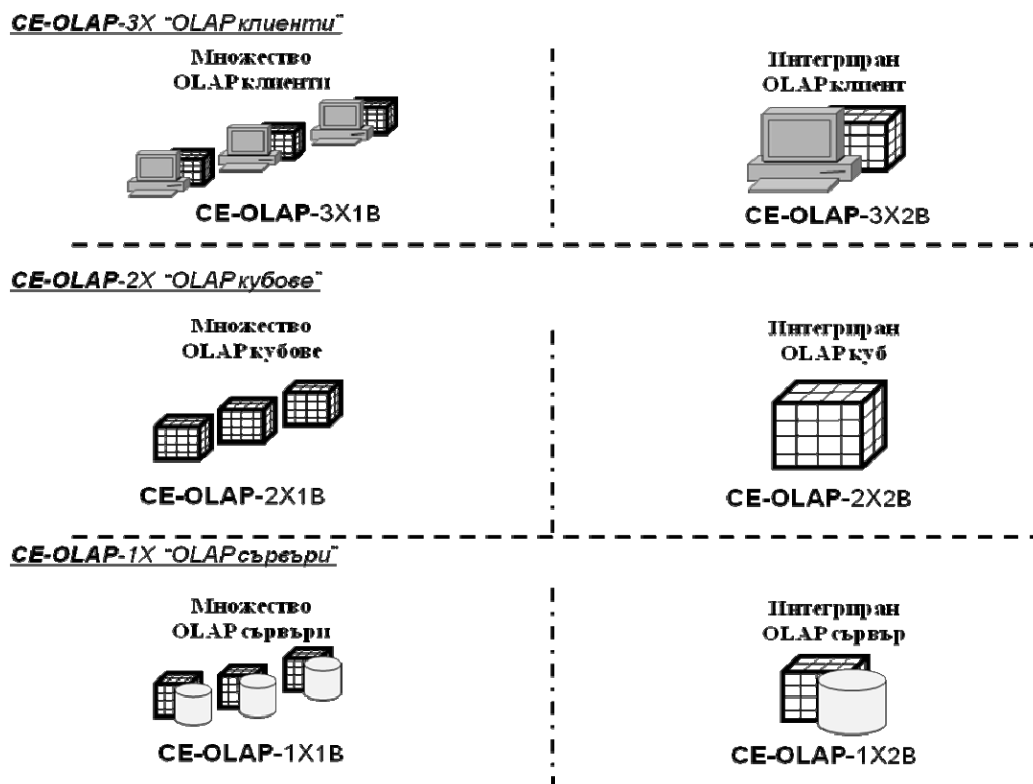
- Глобални мрежи (WAN) или;
- Локални мрежи (LAN).

Не всички мрежови протоколи за предаване на данни осигуряват нужното качество за предаване. Затова при използване на WAN е за предпочитане връзките между отделните хардуерно-софтуерни реализации на системните елементи да се надграждат с използване на системи за предаване на съобщения (например работещи с JMES, MQSeries, и др).

Методът за разпределение на Складовете от данни представлява избор на Системен елемент за всяко от йерархическите нива системни елементи – ЕТЛИПо и Хранилища. За конкретно решение на разпределена БИС и по-специално за разпределен Онлайн аналитични процесор, трябва да се изберат освен видовете системни елементи, но и видът комуникация между тях.

4.3. Метод за разпределение на Онлайн аналитични процесори

Проведеният анализ на съществуващите изследвания в областта на Разпределените Онлайн аналитични процесори – [21],[23],[24], както и представените по-горе принципи за разпределение на БИС, водят до създаването на Метод за разпределени Онлайн аналитични процесори. Този метод се състои в разпределяне на тези процесори на 6 Системни елементи, представени на фигура 11.



Фиг. 11

Системните елементи, на които може да се формира разпределеният Онлайн аналитичен сървър, се разделят според възможните разпределения на:

- OLAP сървърите;
- OLAP кубовете;
- OLAP клиентите.

Разработените 6 системни елемента са следните:

- **CE-OLAP-1X1B** – Множество OLAP сървъри;
- **CE-OLAP-1X2B** – Интегриран OLAP сървър;
- **CE-OLAP-2X1B** – Множество OLAP кубове;
- **CE-OLAP-2X2B** – Интегриран OLAP куб;
- **CE-OLAP-3X1B** – Множество OLAP клиенти;
- **CE-OLAP-3X2B** – Интегриран OLAP клиент.

Функционирането на всеки от посочените системни елементи – OLAP сървъри, OLAP кубове и OLAP клиенти, е според описаното по-горе в студията. Системният елемент "Множество OLAP сървъри" означава няколко физически сървъра, на които са инсталирани софтуерните компоненти на OLAP сървър. Броят на OLAP кубовете не съответства на броя на OLAP сървърите. За да съществува OLAP куб, е необходим поне един физически OLAP сървър. Същевременно един и същ OLAP куб

може да се инсталира на няколко физически OLAP сървъра, организирайки физически копия на OLAP куба. Същевременно е възможно обвързване на няколко куба с оглед провеждане на един анализ. Нормалното съотношение между OLAP сървър и OLAP куб е 1:1. За да се анализира един OLAP куб е необходим поне един OLAP клиент. Не може да съществува OLAP куб без да има съответен клиент, защото OLAP клиента е този който визуализира резултатите от данните, намиращи се в OLAP куба. Наличието на множество OLAP клиенти не винаги осигурява съществуването на физически сървъри за всеки OLAP клиент. Например възможно е да съществува за всеки OLAP куб отделен физически OLAP клиент, както и няколко OLAP клиента, работещи с отделни OLAP кубове, да се разположат логически в един физически сървър за тяхното функциониране.

За всеки от тези сървъри/кубове/клиенти може да се създадат системни елементи състоящи се от един интегриран сървър/куб/клиент, или от няколко разпределени такива. Тези сървър/куб/клиент представляват йерархическа структура, изградена отдолу нагоре според посоченото изброяване, като всеки един от тях може да участва в разпределената архитектура чрез съответен свой представител в съответен разпределен вид. Например би могло да съществува архитектура, състояща се от един OLAP сървър, осигуряващ функциониране на n OLAP куба и работещ с m OLAP клиента.

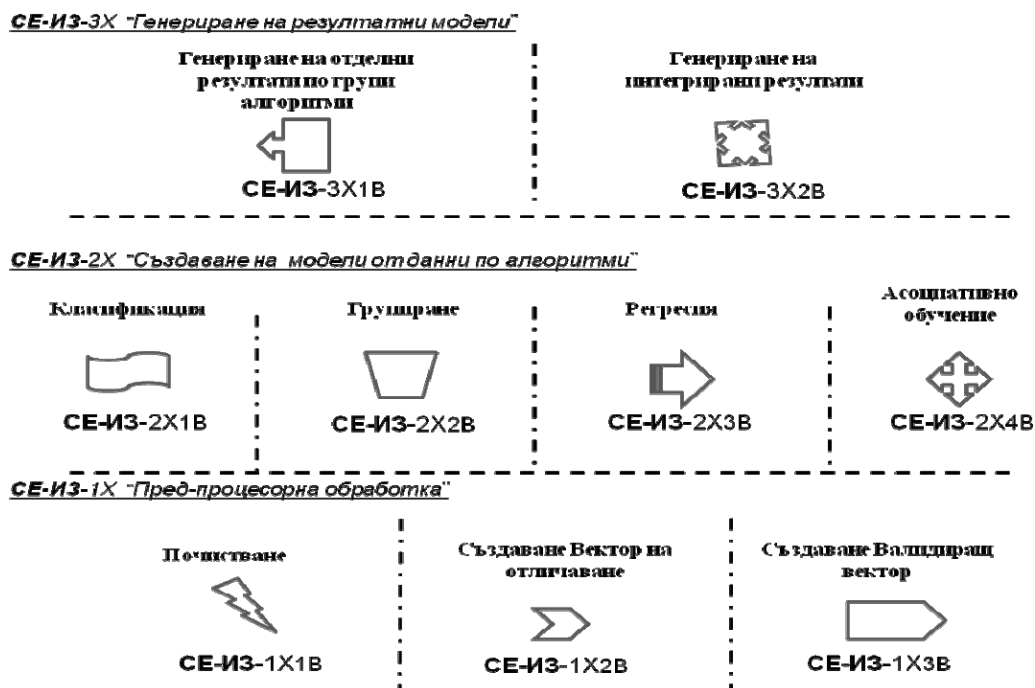
Разпределението между OLAP сървърите и OLAP кубовете е обикновено логическо, без да съществуват физически разпределени елементи, докато разпределението между OLAP кубовете и OLAP клиентите може да бъде чисто физическо, с използване на WAN или LAN за връзка между тях. Съотношението n OLAP сървъра към m OLAP кубове е винаги при $n \leq m$, както и съотношението между m OLAP кубове и q OLAP клиенти е винаги със спазване на зависимостта $m \leq q$.

Методът за разпределение на Онлайн аналитични процесори представлява избиране на съответния системен елемент за всяко от трите посочени йерархически нива. Проектантът на Разпределени БИС, когато започва да проектира Онлайн аналитични процесори, трябва да избере за всяко от трите йерархически нива съответен системен елемент. Освен това, той трябва да избере при кои системни елементи разпределението е физическо (с използване на самостоятелна хардуерно-софтуерна система) и при кои е логическо (да се разположат в една хардуерно-софтуерна система). Също така трябва да се проектира видът комуникация, която да се прилага при свързването на системните елементи, избрани за разпределен Онлайн аналитичен процесор.

4.4. Метод за разпределение на Извличане на знания

Извличането на знания е бизнес процес, който може да се изпълнява както в самостоятелна отделна хардуерно-софтуерна среда, така и като разпределено архитектурна среда. Съществуват определени изследвания – [25],[26],[27], чиито анализи доказват възможността на разпределяне на процеса на Извличане на знания. Представените по-горе в студията принципи на разпределение дават основата за създаването на Метода за разпределение на Извличане на знания, който осигурява разпределена архитектура. Този метод създава условия за изграждане на разпределена

архитектура, която може да се изгради от системни елементи, използващи някои от 9-те Системни елемента на метода – фигура 12.



Фиг.12

Методът е проектиран да работи с 3 йерархически групи системни елементи – "Предпроцесорна обработка", "Създаване на модели от данни по алгоритми" и "Генериране на резултатни модели". Всяка от посочените групи предлага два, три или четири (съответно) системни елемента, с които да се работи при архитектурното проектиране за всяка от групите. Разработените 9 системни елемента на метода са следните:

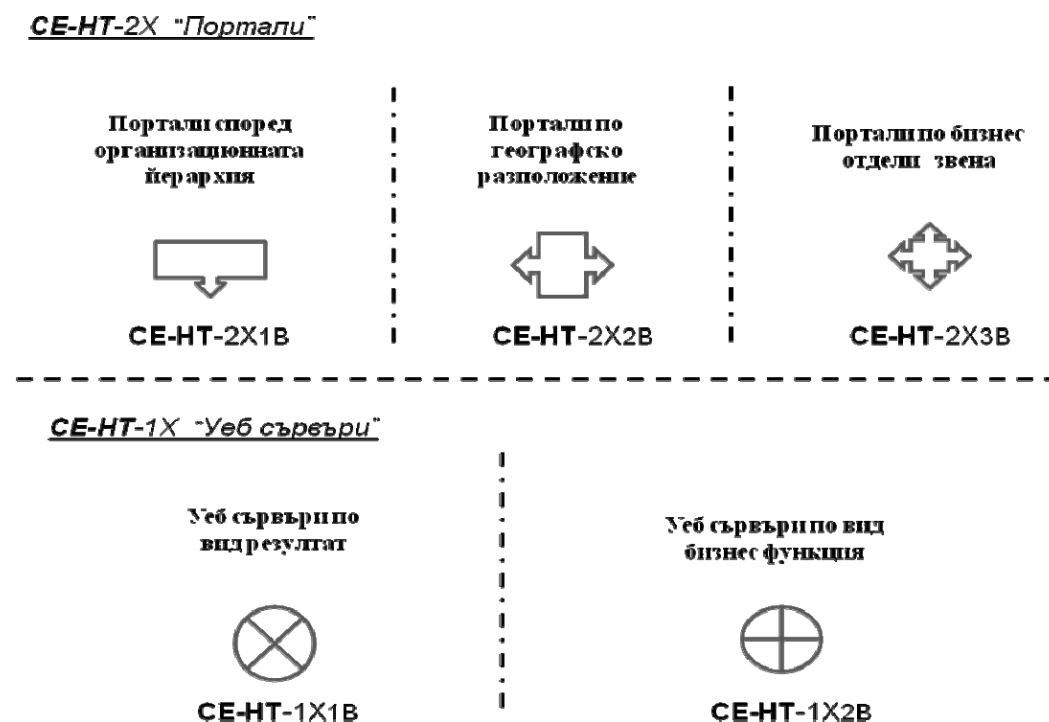
- **СЕ-ИЗ-1X1В** – Почистване;
- **СЕ-ИЗ-1X2В** – Създаване Вектор на отличаване;
- **СЕ-ИЗ-1X3В** – Създаване Валидиращ вектор;
- **СЕ-ИЗ-2X1В** – Класификация;
- **СЕ-ИЗ-2X2В** – Групиране;
- **СЕ-ИЗ-2X3В** – Регресия;
- **СЕ-ИЗ-2X4В** – Асоциативно обучение;
- **СЕ-ИЗ-3X1В** – Генериране на отделни резултати по групи алгоритми;
- **СЕ-ИЗ-3X2В** – Генериране на интегрирани резултати.

Предложеният метод предполага за всяка от представените йерархически групи да има системен елемент, включен в разпределената архитектура на Извличане на знания.

Функциите, които предоставят отделните системни елементи съответстват на тези, описани по-горе в студията. От предложените 9 системни елемента не всеки от тях трябва да се изпълни като самостоятелно хардуерно-софтуерно решение. Например за реализиране на системните елементи в групата "Създаване на модели от данни по алгоритми", може да се използват 1 сървър (всичките алгоритми работят на един физически сървър), 4 сървъра (всеки отделен тип алгоритъм работи на отделен физически сървър), или множество сървъри (един тип алгоритми се прилагат в няколко сървъра с оглед на по-висока производителност). Същевременно няколко системни елемента от различните групи може да се съвместят за работа в един физически сървър. Независимо в разпределеното архитектурно решение за Извличане на знания, между физическите сървъра трябва да има съответната мрежова комуникация (WAN или LAN).

4.5. Метод за разпределение на Навигационно табло

За проектиране на разпределено Навигационно табло на БИС е създаден метод, който е представен графически на фигура 13.



Фиг. 13

Предложеният метод се състои от две йерархически нива системни елементи – Уеб сървъри и Портали. Предназначението на групата системни елементи Уеб сървъри е да осигури нужната информация за Навигационното табло под формата на

Уеб интерфейси и Уеб услуги, т.е. това са група системни елементи, представляващи източници на информация за Навигационното табло. Втората група системни елементи – "Портали", има за цел да осигури интегрирането на информацията от различните Уеб сървъри, да осигури необходимия вид за представяне на информацията (look and feel) и възможностите за потребителска настройка на екранните изображения. Тази група системни елементи осигурява също така и крайното програмиране на форматите на отчетите, които ще се подготвят за отпечатване. При това разпределение трябва добре да се разграничи Уеб сървър от Портал.

Методът за разпределение на Навигационното табло се състои от следните 5 системни елемента:

- **СЕ-НТ-1Х1В** – Уеб сървър по вид резултат;
- **СЕ-НТ-1Х1В** – Уеб сървър по вид бизнес функция;
- **СЕ-НТ-1Х1В** – Портали според организационната йерархия;
- **СЕ-НТ-1Х1В** – Портали по географско разположение;
- **СЕ-НТ-1Х1В** – Портали по бизнес отдели/звена.

От наименованието на системния елемент и от представянето на функциите на Навигационното табло по-горе в настоящата студия става ясно предназначението на всеки системен елемент.

За всеки системен елемент могат да съществуват различни хардуерно-софтуерни реализации. Връзката между долното и горното йерархически нива системни елементи се осъществява с използване на HTTP протокол, който на ниско ниво може да използва WAN или LAN тип изчислителна мрежа.

ЗАКЛЮЧЕНИЕ

В студията е обосновано съществуването на Бизнес интелигентна екосистема, която в съвременните бизнес условия често се явява разпределена. Разпределената БИС е необходимо архитектурно решение, когато се отнася за малки и средни предприятия. Създадени са 3 принципа за проектиране на Бизнес интелигентна система (БИС), на чиято база е разработена Рамка за проектиране на Разпределена БИС. Избрани са 4 архитектурни компонента на БИС, за които са разработени Методи за проектиране на разпределени БИС системни компоненти. Създадени са 26 вида разпределени БИС системни компонента, като логически са представени възможните връзки между тях.

Литература

1. Data Warehouse, http://en.wikipedia.org/wiki/Data_warehouse
2. Data Warehouse, http://en.wiktionary.org/wiki/data_warehouse
3. Glossary of terms, Data Warehouse, http://www.oranz.co.uk/glossary_text.htm
4. Estrad software, Data mining Glossary, http://www.estard.com/data_mining/data_mining_glossary.asp
5. Bill Inmon, http://en.wikipedia.org/wiki/Bill_Inmon
6. Склад данни, <http://bg.wikipedia.org/wiki/%D0%A1%D0%BA%D0%BB%D0%B0%D0%B4%D0%B4%D0%B0%D0%BD%D0%BD%D0%B8>

7. E. Codd, S. Codd, C. Salley, Providing OLAP to User-analysts: An IT Mandate, Technical report, E.F.Codd & Associates, 1993, <http://www.fpm.com/refer/codd.html>
8. OLAP cubes, http://en.wikipedia.org/wiki/OLAP_cube
9. Online analytical processing, <http://en.wikipedia.org/wiki/OLAP>
10. P. Lyman, H. Varian, How much information?, <http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/>
11. Data mining, http://en.wikipedia.org/wiki/Data_mining
12. k-nearest neighbor algorithm, [http://en.wikipedia.org/wiki/Nearest_neighbor_\(pattern_recognition\)](http://en.wikipedia.org/wiki/Nearest_neighbor_(pattern_recognition))
13. Naive Bayes classifier, http://en.wikipedia.org/wiki/Naive_Bayes_classifier
14. Artificial neural network, http://en.wikipedia.org/wiki/Artificial_neural_networks
15. Genetic programming, http://en.wikipedia.org/wiki/Genetic_Programming
16. Decision tree, http://en.wikipedia.org/wiki/Decision_tree
17. S. Few, Dashboard confusion, Intelligent Enterprise 4, 2004, <http://www.intelligententerprise.com/showArticle.jhtml?articleID=18300136>
18. Mark Rittman, Building BI Dashboards with Oracle Database, Oracle Discoverer, and Oracle Portal, http://www.oracle.com/technology/pub/articles/rittman_dash.html
19. IBM Information Server architecture and concepts, <http://publib.boulder.ibm.com/infocenter/iisinfsv/v8r0/index.jsp?topic=/com.ibm.swg.im.iis.productization.iisinfsv.overview.doc/topics/cisoarchoverview.html>
20. L. Mustatea, Creating a Distributed Data Warehouse for Effective Workload Balancing and Data Sharing, TDW, Vol.18, 2004, <http://www.tdwi.org/Publications/WhatWorks/display.aspx?id=7316>
21. M. Akinde, M. Bohlen, T. Johnson, L. Lakschman, D. Srivastava, Efficient OLAP Query Processing in Distributed Data Warehouses, Advances in Database Technology — EDBT 2002, SpringerLink, 2002
22. R. Moeler, Distributed Data warehousing using Web technology, ISBN 0-8144-0588-6, AMACOM, 2000
23. C. Qiming, D.Jie, D. Umeshwar, H. Meichun, Distributed OLAP-based association rule generation method and system, USA Patent 7007020, 2006
24. Q. Chen, U. Dayal, A Distributed OLAP Infrastructure for E-Commerce, Fourth International IECIS Conference, Edinburg, UK, 2004
25. J. Zaki, Y. Pan, Special Issue on Parallel and Distributed Data Mining, International Journal of Distributed and Parallel Databases, Vol. 11, No. 2, March 2002
26. D. Talia, Grid-based Distributed Data mining systems, Algorithms and Services, University of Calabria, Italy, http://www.siam.org/meetings/sdm06/workproceed/HPDM/Domenico_talia_Invited_Session.pdf
27. C. Clifton, K. Wenliang, M. Atallah, Distributer Data Mining to protect Information Privacy, http://www.cs.purdue.edu/homes/clifton/DistDM/ppddm_nsf.pdf
28. К. Стефанова, В. Кисимов и кол., "Проектиране на Център за компетентност по Бизнес интелигентност", Международна научна конференция по повод 40-годишнината на специалност "Информатика", УНСС, София, 2007
29. Garner Group, The Current State and Future Direction of IT, Gartner Symposium, 31 July – 2 August 2006
30. PA Consulting Group, Business Intelligence Competency Centre: Deliver a step change in your performance management by establishing a Business Intelligence Competency Centre, 2007, www.paconsulting.com
31. White Paper: Business Intelligence Competency Centers: Creating a successful business intelligence strategy, 2006, www.sas.com
32. Ecosystems, <http://en.wikipedia.org/wiki/Ecosystem>

ПРИНЦИПИ И МЕТОДИ ЗА ИЗГРАЖДАНЕ НА РАЗПРЕДЕЛЕНИ БИЗНЕС ИНТЕЛИГЕНТНИ ЕКОСИСТЕМИ

Резюме

Студията е посветена на един нов подход в научните изследвания, насочен към създаване на разпределена Бизнес интелигентна екосистема. На основата на Разпределен център за компетентност на Бизнес интелигентните системи е дефинирана Екосистемата за Бизнес интелигентни системи. С оглед обслужване на малки и средни предприятия, системата е проектирана като разпределена. Представени са три принципа за разпределение – Принцип на хоризонтално разпределение, Принцип на вертикално разпределение и Принцип на разпределение с Вградени компоненти. На тяхна основа е създадена Рамка за проектиране на методите за разпределение, като са проектирани следните методи за разпределение – Метод за разпределение на Склад от данни, Метод за разпределение на Онлайн аналитични процесори, Метод за разпределение на Извличането на знания и Метод за разпределение на Навигационно табло.

DESIGN PRINCIPLES AND METHODS FOR DISTRIBUTED BUSINESS INTELLIGENCE ECOSYSTEM

Abstract

The paper is dedicated to a new approach in the scientific research, oriented towards Distributed Business Intelligent Ecosystems. This system is designed based on Distributed Business Intelligence Competence Centre. For the needs of Small and Medium Enterprises, the system is designed as a distributed. Three principles of distribution are presented – Principle of Horizontal distribution, Principle of Vertical distribution and the principle of Embedded distribution. A Design Framework is created on their basis, for the purpose of design of the following methods – Method for distribution of Data Warehouses, Method for distribution of OLAP engines, Method for distribution of Data Mining engines, and Method for distribution of Dashboards.