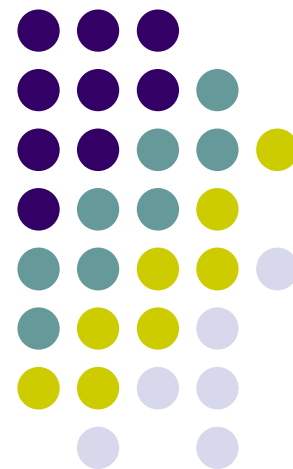


Мрежово програмиране

**Архитектури на
информационните
системи**



Архитектура на ИС



Основната цел на бизнес информационните системи (ИС) е да решават по определен начин информационни проблеми в различни области на бизнеса – търговия, транспорт, строителство, производство и т.н.

Архитектурата на ИС е абстрактен модел, който обхваща отделните елементи на системата и техните взаимодействия

Архитектура на ИС



Архитектурните описания на една информационна система (ИС) включват:

- Логическа архитектура - организация на софтуера на ИС
- Физическа (техническа) архитектура - хардуерът, на който е реализирана ИС, т.е. включените компютри и каналите за връзка между тях

Графично представяне на архитектурата



1. Диаграма на логическата архитектура
2. Диаграма на физическата архитектура
3. Диаграма на разгръщането (deployment diagram) – представя едновременно физическите компоненти на ИС и изпълняваните в тях софтуерни компоненти – т.е. съвместно представя физическата и логическата архитектура

Функции на ИС



- Потребителски интерфейс (Презентационни функции) – функции, реализиращи взаимодействието на потребителите с ИС
- Бизнес логика - функции по обработка на информацията в съответствие със специфичните правила за бизнес процеса, характерни за съответната приложна област (бизнес модел – търговия, транспорт и т.н.)
- Функции по управление, съхранение и извличане на данни от бази от данни или отделни файлове

Функции на ИС



- Бизнес модел – отговаря на съответната приложна област (търговия, транспорт и т.н.)
Състои се от определен брой бизнес процеси
- Бизнес процес – работен цикъл - повтаряща се последователност от действия, която може да бъде документирана (доставка, продажба)
- Бизнес логика - операциите по обработка на информацията в съответствие с правилата на бизнес процеса (ДДС, търговска отстъпка, % печалба, разходни норми и т.н.)

Потребителски интерфейс

Основната задача на интерфейса, е да преобразува задачата в нещо, което потребителя разбира.

> Намери общо продажби

> Намери общо продажби
4 продажби

Бизнес логика

Този слой изпълнява команди, взема логически решения, прави изчисления и преобразувания.

Дай списък на всички продажби от последна година

Събери всички прод. заедно

Данни

Тук информацията се запаметява и извлича при необходимост, връща се при бизнес логиката за обработка и после отново при потребителя.

Заявка

Прод.1
Прод.2
Прод.3
Прод.4

База данни

Съхранение

Функции на ИС



Бизнес процесът притежава:

- Цел – решава определени задачи
- Вход
- Изход
- Използва ресурси, в т.ч. и информационни
- Влияе хоризонтално на фирмата, т.е. може да засяга повече от една организационна единица – отдел доставки и склад
- Определен брой дейности, изпълнявани последователно (бизнес логика)

Архитектура «файл-сървър»



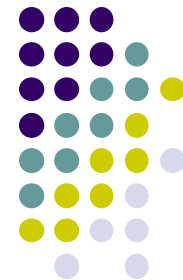
- **Достоинства:**

- многопотребителски режим на работа с данните
- удобство за централизирано управление на достъпа
- ниска стойност за разработване
- висока скорост на разработване
- невисока стойност за обновяване и изменения на ПО

- **Недостатъци:**

- проблеми при многопотребителския режим на работа с данните
- ниска производителност
- лоша възможност за добавяне на нови клиенти
- липсва надеждност на системата

Многослойна архитектура «клиент-сървър»



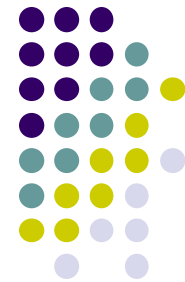
- **Достоинства:**

- клиентското ПО не се нуждае от администриране
- мащабируемост
- лесно конфигуриране
- висока безопасност и надеждност
- ниски изисквания към скоростта на канала между терминалите и сървъра на приложенията
- ниски изисквания към производителността и техническите характеристики на терминалите

- **Недостатъци:**

- сложно администриране и обслужване
- по-висока сложност за създаване на приложенията
- високи изисквания към производителността на сървърите на приложенията и сървъра за бази от данни
- високи изисквания към скоростта на канала (мрежата) между сървъра за бази от данни и сървърите на приложенията

Класификация на логическите архитектури



В зависимост от разположението и реализацията на трите функции, съществуват:

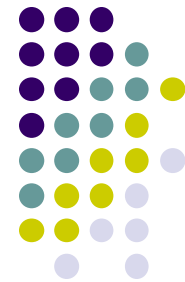
- Еднослойна архитектура
- Двуслойна архитектура
- Трислойна архитектура
- Четирислойна архитектура
- Сложна (N слойна) архитектура

Класификация на логическите архитектури



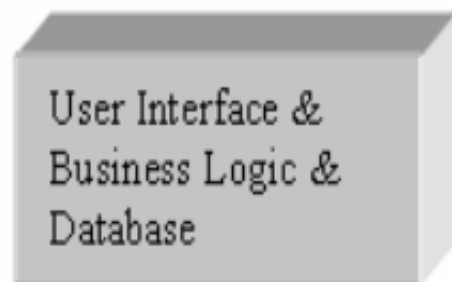
- Еднослойна архитектура - приложението се изпълнява върху един компютър, като трите вида функции са свързани
- Двуслойна архитектура - обособяване на два слоя:
 - Клиентски слой – обхваща потребителския интерфейс и бизнес логиката
 - Сървърен слой – реализира всички функции за работа с базите от данни

Класификация на логическите архитектури

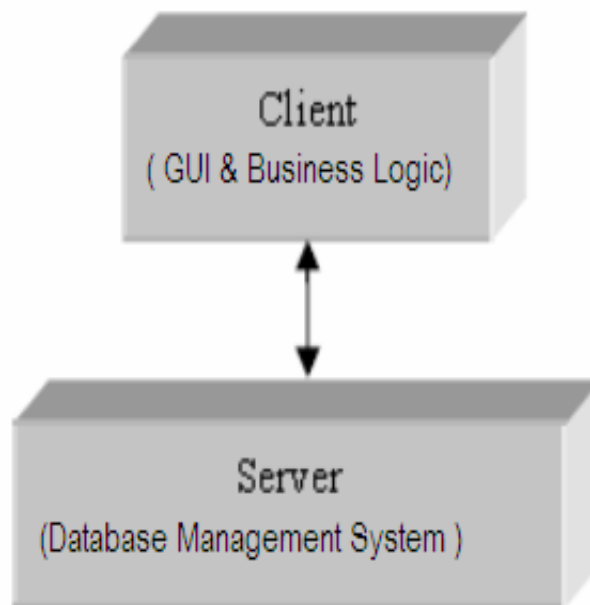


- Трислойна архитектура - обособяване на трите вида функции в три самостоятелни слоя:
 - Слой на потребителския интерфейс
 - Слой на бизнес логиката
 - Слой на управление на данните

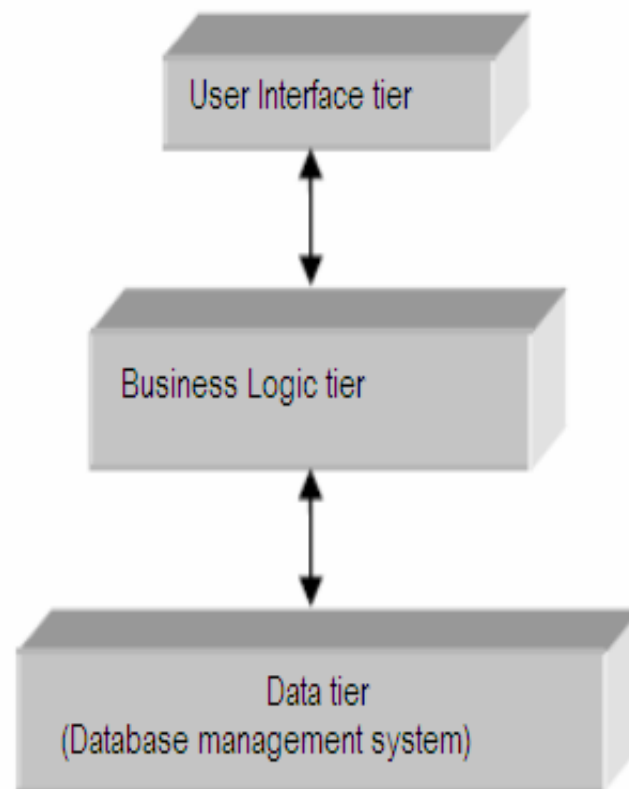
Класове логически архитектури



а) Еднослойна



б) Двуслойна



в) Трислойна

Двуслойна архитектура «клиент-сървър»



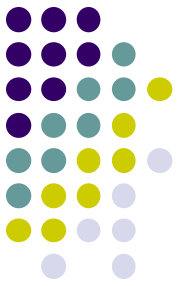
- **Достоинства:**

- Възможност за разпределяне на функциите на изчислителната система между няколко независими компютри
- Всички данни се съхраняват на защитен сървър
- Поддържа се многопотребителски режим на работа
- Гарантира се цялостност на данните

- **Недостатъци:**

- Неработоспособността на сървъра може да направи неработоспособна цялата изчислителна мрежа
- Сложно администриране
- Висока цена на оборудването
- Бизнес логиката на приложенията остава в клиентското ПО

Физическа реализация на трислойна логическа архитектура



- с “тънък” клиент – на клиентската работна станция е инсталиран потребителския интерфейс, а на сървъра – бизнес логиката и управлението на данните чрез СУБД
- с “дебел” клиент – на клиентската работна станция са реализирани потребителския интерфейс и бизнес логиката, на сървъра - само слоя за управление на данните, чрез определена СУБД

Физическа реализация на трислойна логическа архитектура



Клиентската работна станция с “тънък” клиент притежава:

- Ограничени технически възможности – процесор и памет
- Мрежова карта и входно-изходни устройства

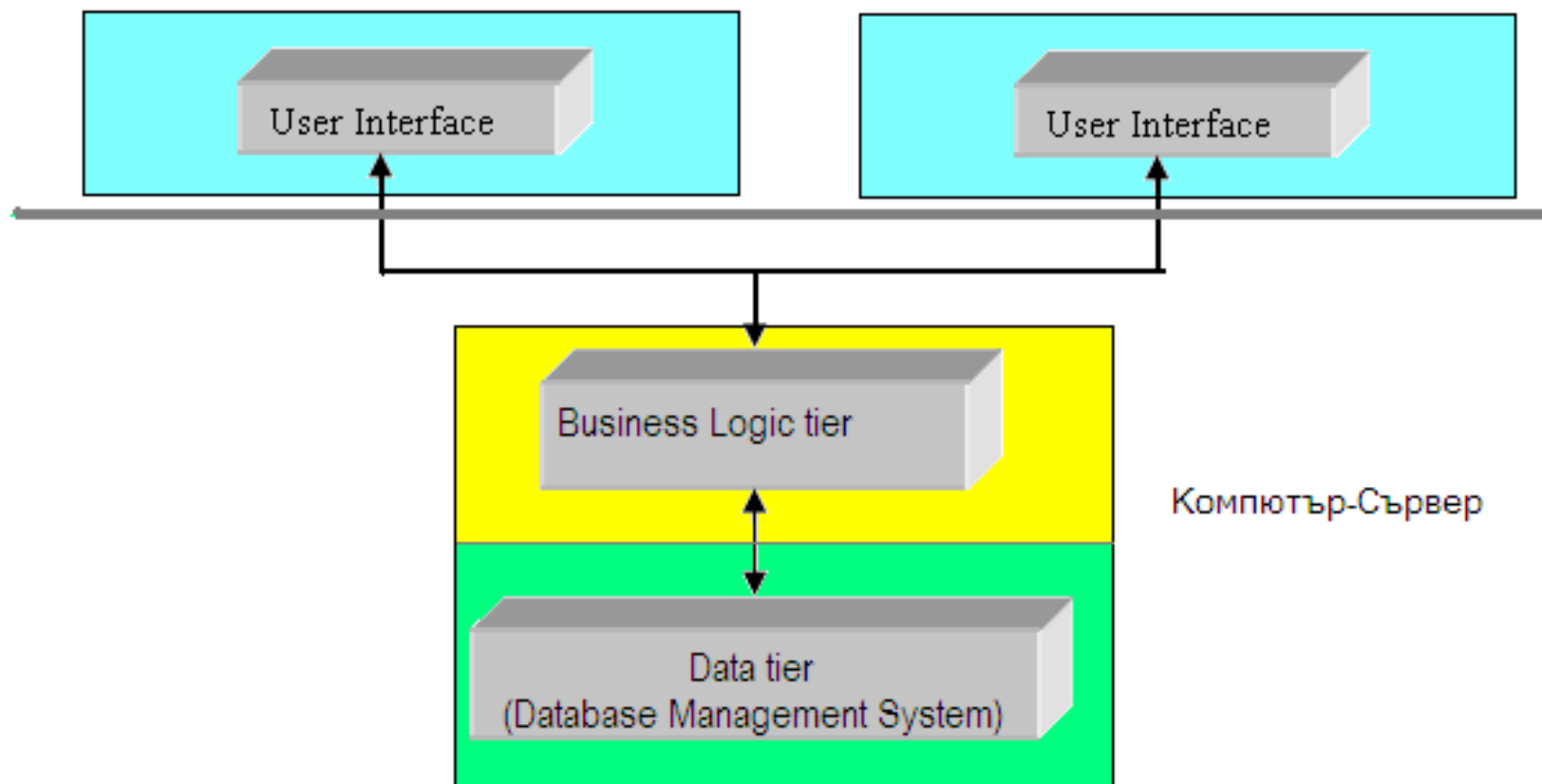
Предимства:

- По-ниски разходи за реализиране
- Не е необходимо локално администриране на данните

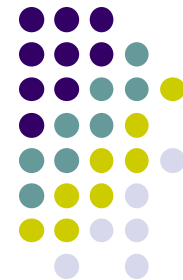
Трислойна клиент-сървър архитектура с “ТЪНЪК” КЛИЕНТ



Клиентски работни станции



Физическа реализация на трислойна логическа архитектура



Клиентската работна станция с “дебел” клиент притежава:

- Високи технически възможности – процесор и памет (оперативна и дискова)
- Мрежова карта и входно-изходни устройства

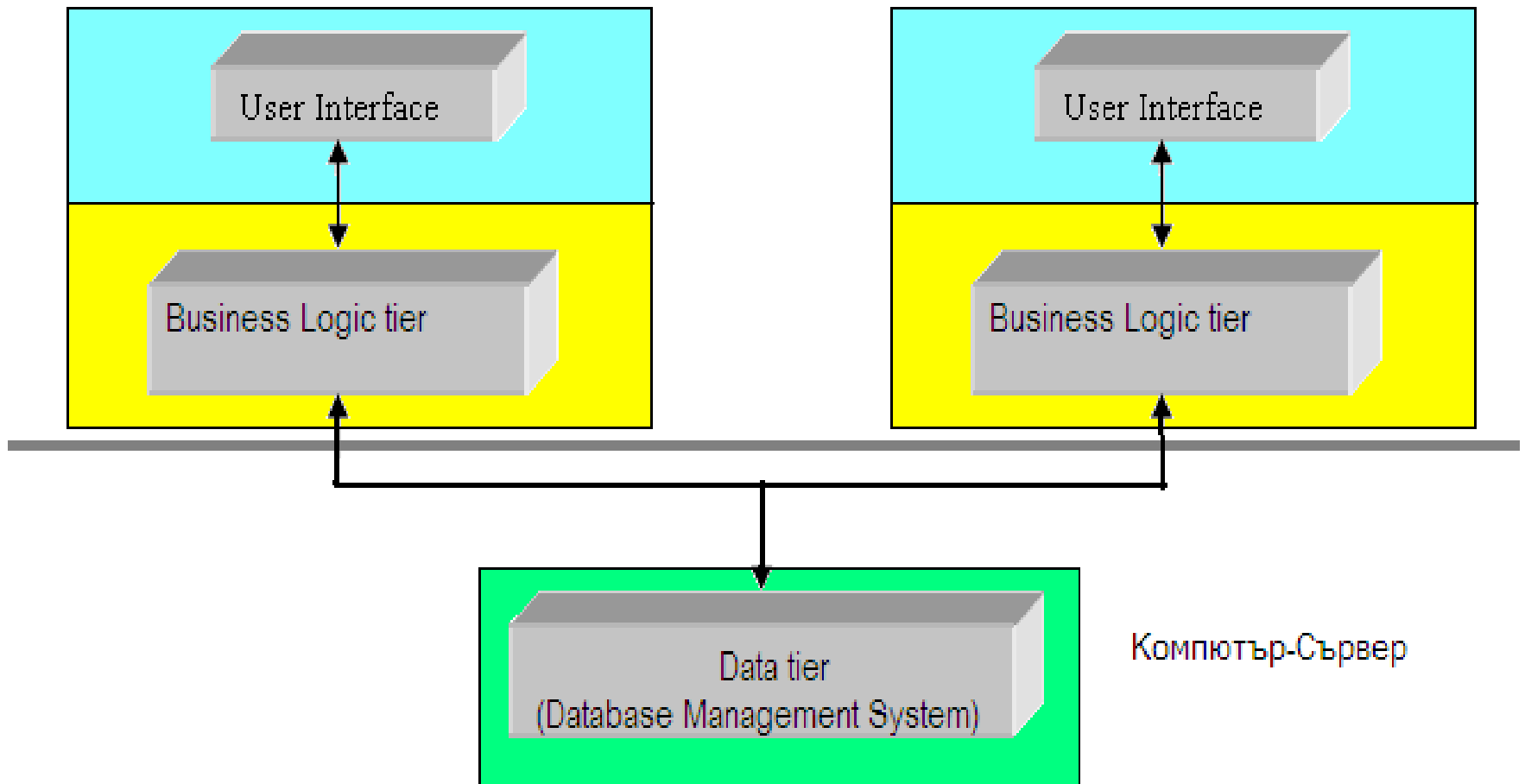
Предимства - не е необходимо локално администриране на данните (backup),

Недостатъци - по-високи разходи за реализиране

Трислойна клиент - сървър архитектура с “дебел” клиент



Клиентски работни станции



Физическо отделяне на бизнес логиката



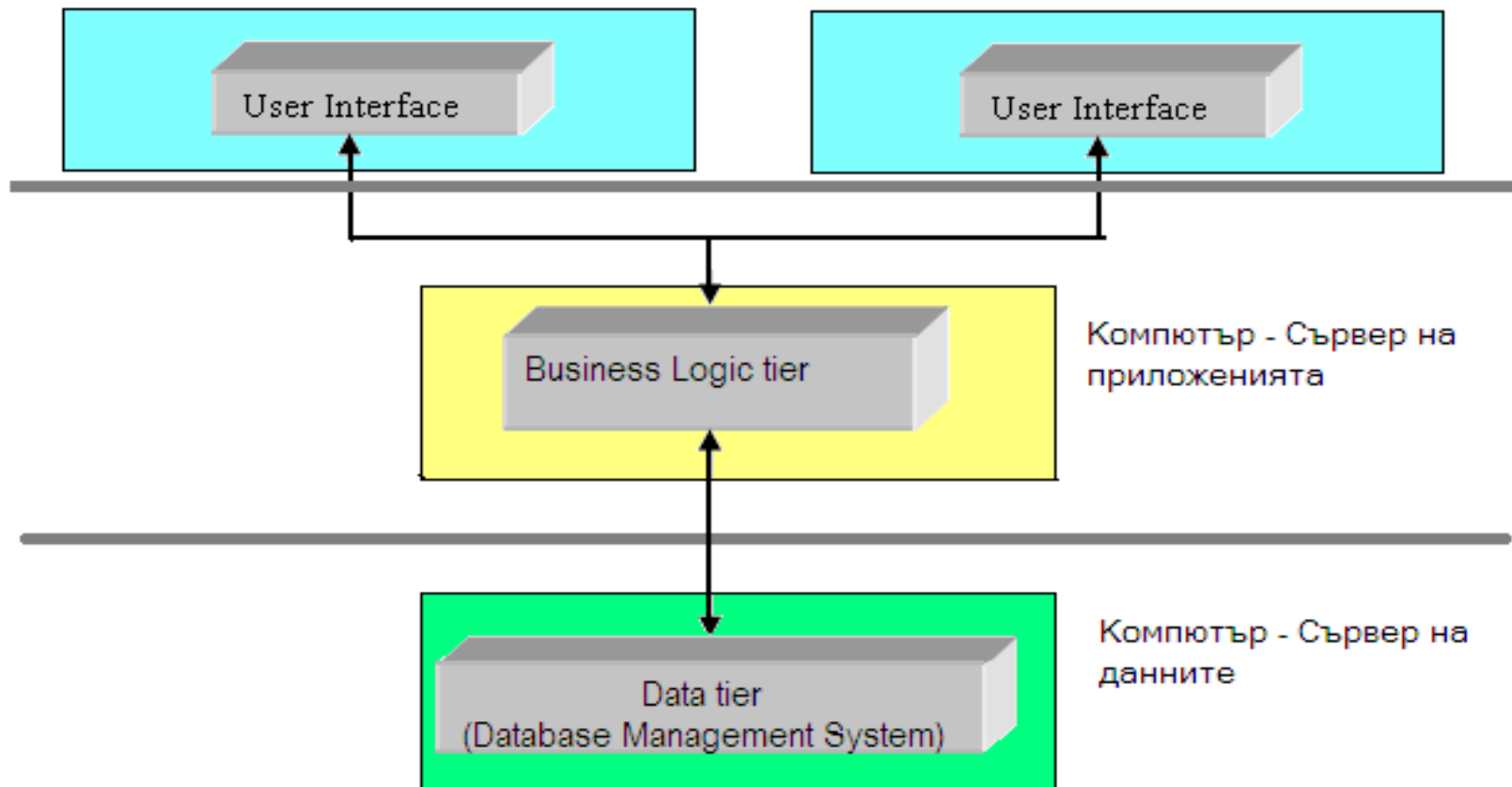
Сървър на приложенията (Application server) се използва двузначно като:

- Софтуер - приложно програмно осигуряване, обслужващо модулите с бизнес логиката - “софтуерен сървър на приложенията”
- Хардуер - компютър, на който са инсталирани модулите с бизнес логиката и обслужващото ги приложно програмно осигуряване - “компютър - сървър на приложенията”

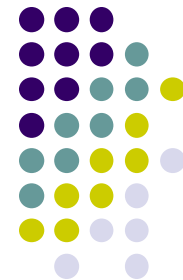
Трислойна архитектура със сървър на приложенията



Клиентски работни станции



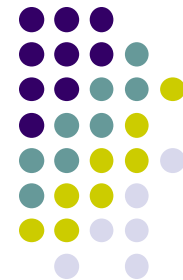
Трислойна архитектура при веб базирани приложения



Особености на архитектурата:

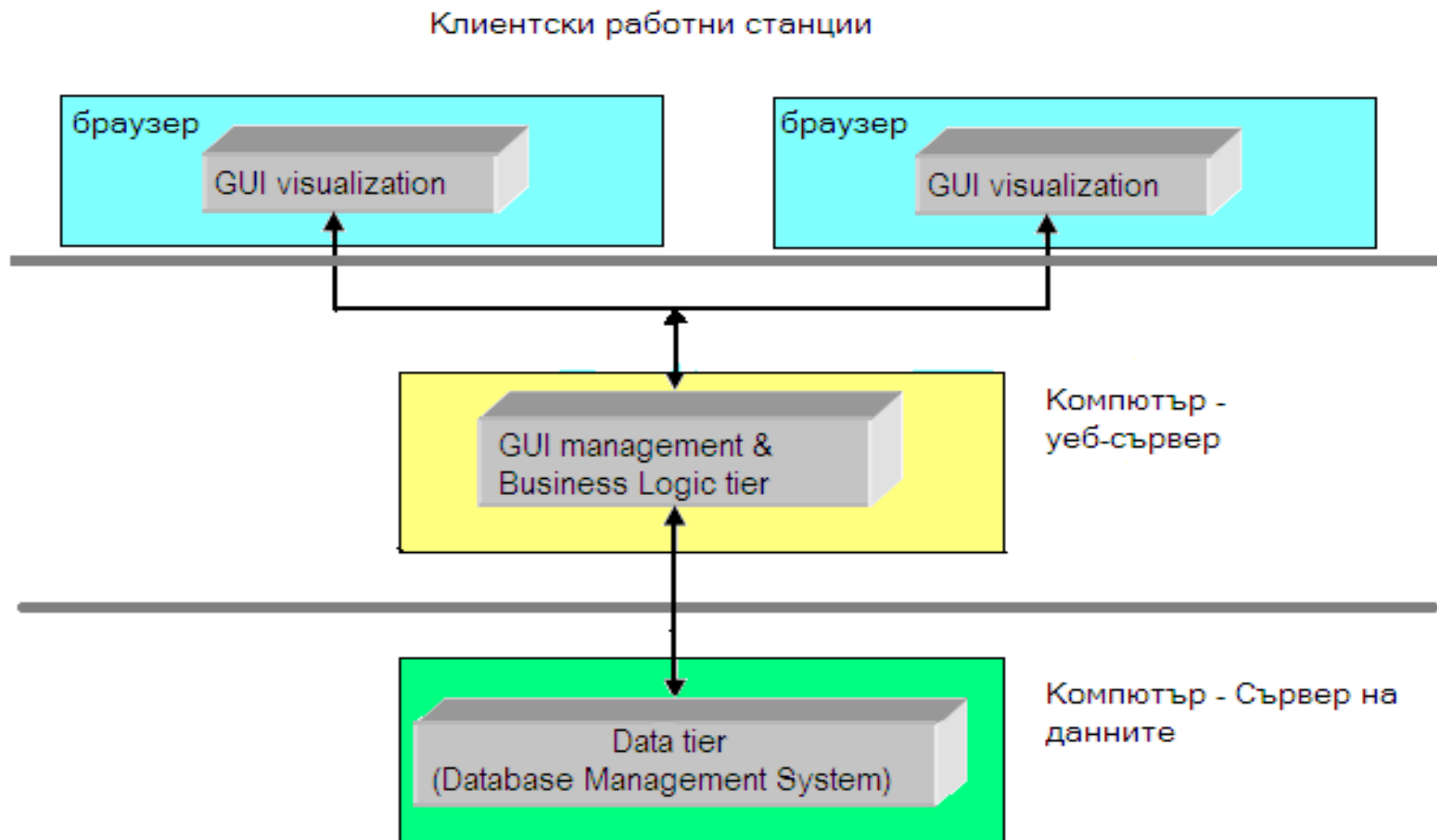
- Визуализирането на потребителския интерфейс се реализира чрез програма браузър – MS Internet Explorer, Mozilla FireFox, Opera и т.н.
- Включва веб сървър за управление на потребителския интерфейс и слоя на бизнес логиката
- Включва веб сървър за данни

Трислойна архитектура при веб приложения

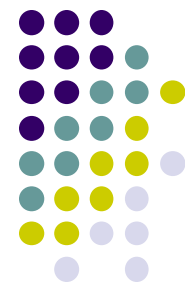


- Управлението на потребителския интерфейс не се разграничава строго от бизнес логиката
- Реализира се на някой от езиците, поддържани от веб сървърите - ASP (Active Service Pages), JSP (Java Service Pages), PHP и т.н.

Трислойна архитектура при веб приложения



Архитектура на уеб приложенията



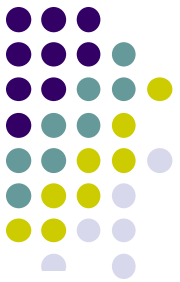
- Няма необходимост от използване на допълнително ПО на страната на клиента
- Възможност за добавяне на практически неограничен брой клиенти
- Централизирано място за съхраняване на данните
- Недостъпност при неработоспособност на сървъра или на комуникационните канали
- Работоспособен и при ниска скорост на уеб сървъра и комуникационните канали

4 слойна архитектура при веб базирани приложения

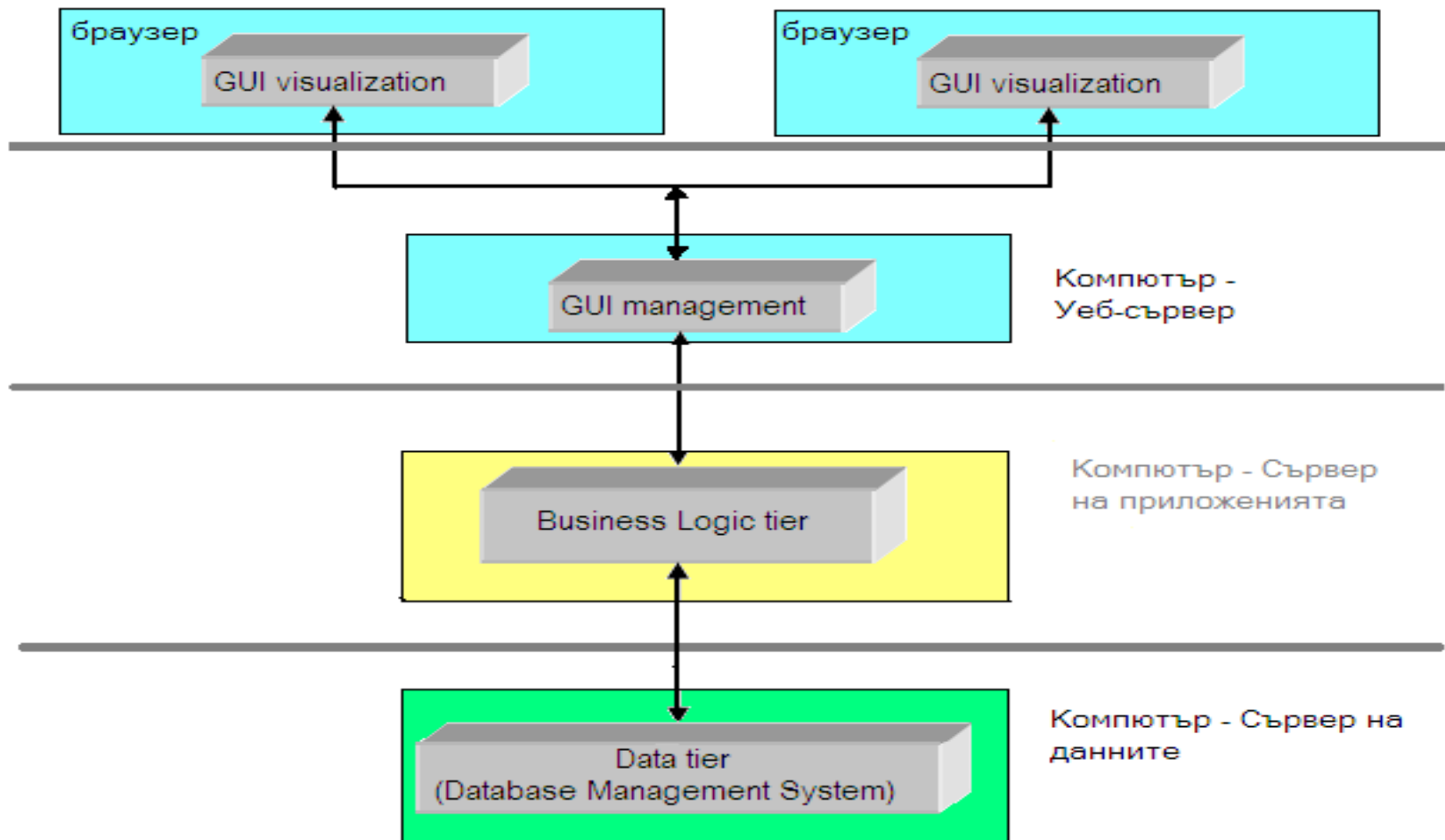


1. На клиентския компютър се визуализира потребителския интерфейс
2. На веб сървър се реализира управлението на потребителския интерфейс
3. Бизнес логиката се реализира като самостоятелен слой на компютър – сървър на приложенията
4. На веб сървър за данни се извършва управлението им

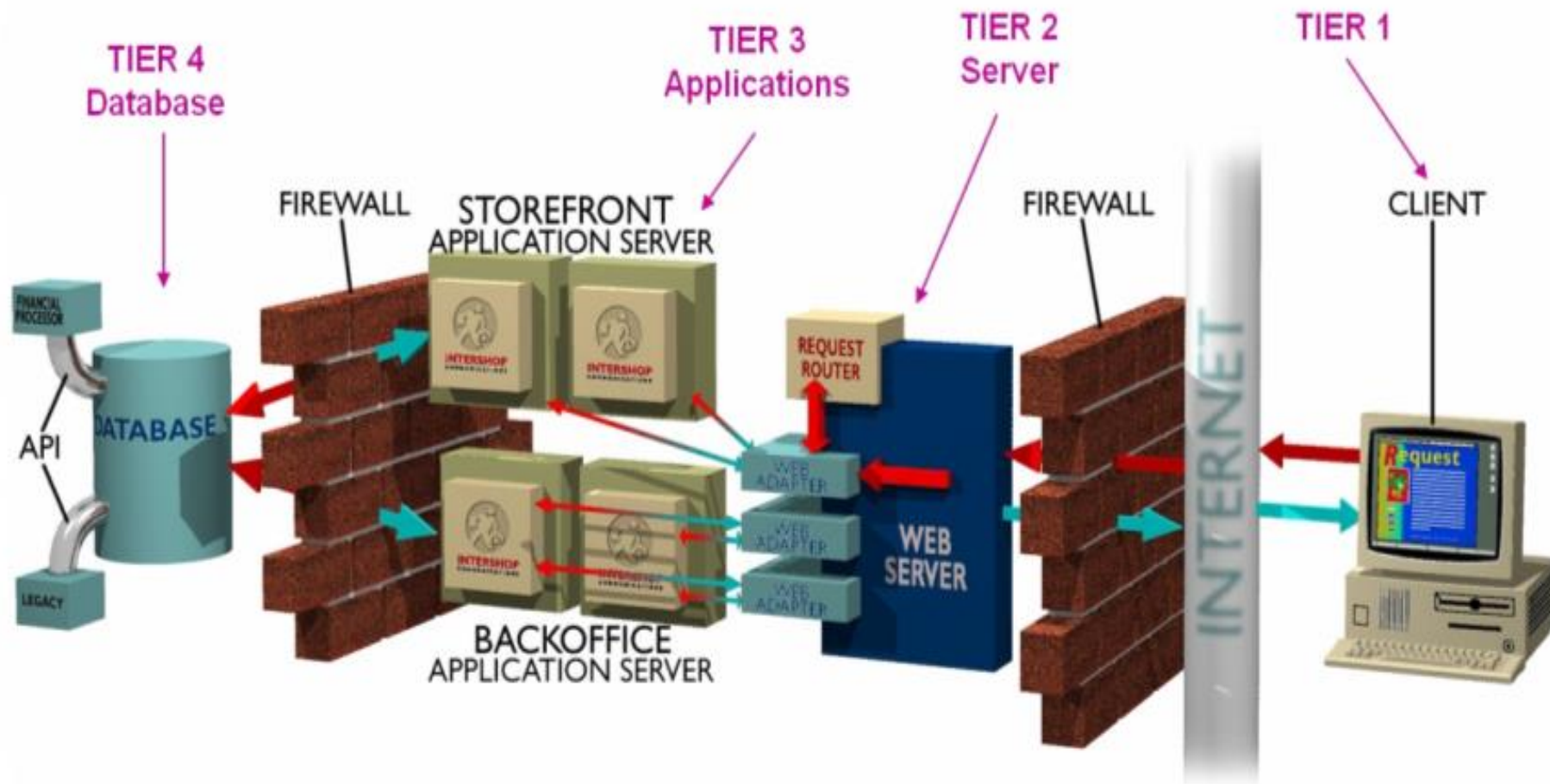
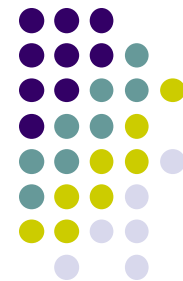
4 слойна архитектура при уеб базирани приложения



Клиентски работни станции



Защита на 4 слойна архитектура при веб базирани приложения



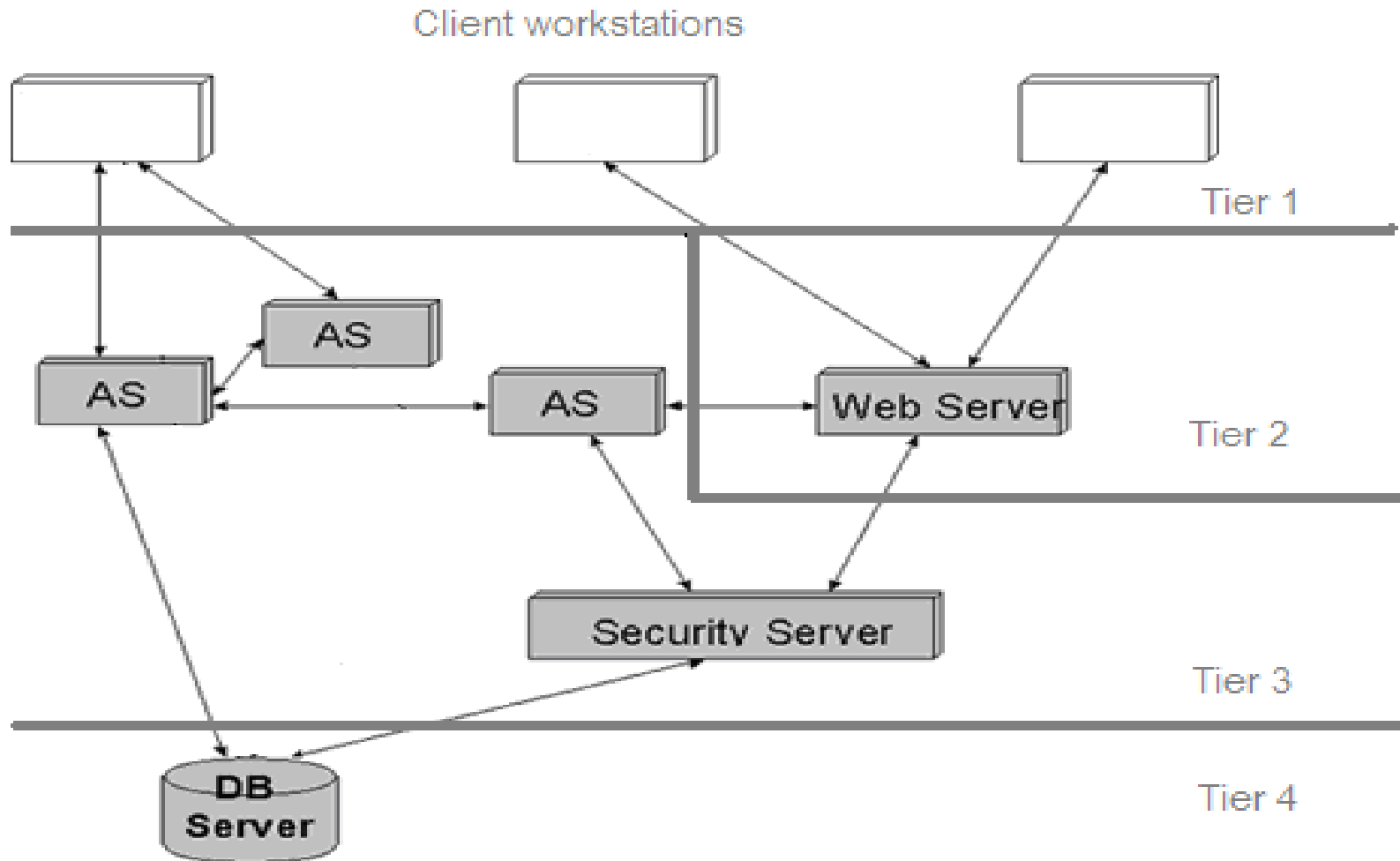
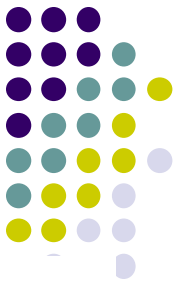
SOURCE: INTERSHOP

Сложни (N слойни) архитектури



- Бизнес логиката е реализирана в множество относително независими модули, които работят върху отделни компютри - сървъри на приложения, обменящи помежду си данни
- Едно обръщение на потребителя се обработва от голям брой модули, разположени на отделни компютри

Сложни (N слойни) архитектури

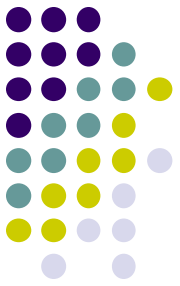


Предимства на приложенията с N слойна архитектура



- Модифицируемост - дава възможност да се променя функционалността на един от модулите, без това да влияе на другите модули
- Разширяемост - възможност на добавяне на нови функции чрез включване на нови модули с бизнес логика
- Многократна използваемост - възможност за изграждане на нови приложения чрез интегриране на съществуващи бизнес компоненти

Други важни особености при проектиране и изработване на ИС



От правилното изграждане на архитектурата на информационната система, зависи нейното по-нататъшно функциониране

При създаване на нова информационна система:

- първо се определя логическата архитектура – общата организация на софтуера според задачите, които трябва да решава информационната система
- после се определя физическата архитектура – хардуера, т.е. компютрите и връзките между тях

Всички подходи за организация на информационни системи за работа в мрежата се базират на общата архитектура клиент-сървър



- СУБД, основана на файл-сървърна архитектура
- Сървър за БД
- Информационна Интранет система



- До тук са разгледани начините и възможните архитектури на информационни системи, предназначени за оперативна обработка на данните, т.е. за получаване на текуща информация, позволяваща решаването на ежедневните проблеми на корпорацията

Склад за данни (DataWarehousing) и системи за оперативна аналитична обработка на данни (OLAP системи, от On-Line Analytical Processing)



- Edgar Codd, отчитайки потребностите на системите за динамична аналитична обработка на данни, е формулирал 12 основни изисквания към системите, поддържащи аналитични бази от данни

Многомерно концептуално представяне на данните

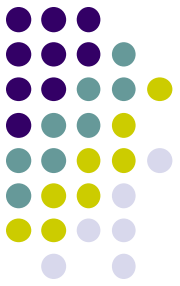


- Това изискване възниква поради причината, че бизнес потребителят разбира се си представя историята и дейността на корпорацията като многомерни (например едното измерение е времето, другото - поръчителите, третото - произведената продукция и т.н.). OLAP моделите са длъжни да поддържат това представяне и то трябва да се опира на възможностите на аналитичната база данни.



Прозрачност

- За бизнес потребителя не бива да е съществено къде конкретно са разположени средствата за динамичен анализ на данните. При разработването на OLAP системи следва да се придържа към подхода на отворените системи, което ще позволи разместването на средствата за анализ на кой да е възел в корпоративната мрежа.



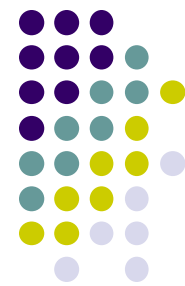
Достъпност

- Логическата схема, по която работи OLAP системата, трябва да се отразява в схемите на разнородни физически складове за данни. При достъп до данните трябва да се поддържа тяхното единно и съгласуванно представяне.

Съгласувана ефективност при производството на отчетите



- Тази ефективност не бива да деградира при увеличаването на броя на измеренията



Архитектура клиент-сървър

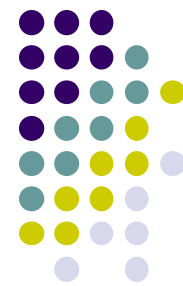
- Сървърният компонент на OLAP системата трябва да бъде достатъчно развит за да могат разнородни клиенти да се свързват с него с минимални усилия и загуби за допълнително "интегриращо" програмиране



Родова многомерност

- Структурните и операционните възможности за работа с всяко измерване на данните трябва да са еквивалентни. За всички измервания трябва да съществува само една логическа структура. Всяка функция, приложима към едно измерване, трябва да е приложима и към всяко друго измерване

Управление на динамично разредените матрици



- Сървърът на OLAP системата трябва да може ефективно да съхранява и обработва разредени матрици. Физическите методи за достъп трябва да са разнообразни, да включват прякото изчисление, В-дървета, хеширане или комбинации на тези методи

Поддръжка на многопотребителски режим



- OLAP системата е длъжна да поддържа многопотребителски достъп до данните, осигурявайки цялостност и безопасност на данните

Неограничени операции между измерванията



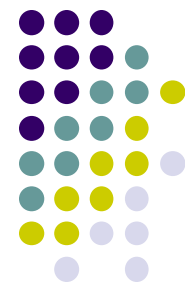
- При изпълнението на многомерен анализ на данните всички измервания се създават и обработват еднообразно. OLAP системата трябва да може да изпълнява съответстващите изчисления между измерванията.

Интуитивно манипулиране с данните



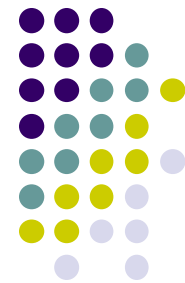
- Манипулациите, подобно на промяната на пътя на анализа или нивото на детайлизация, трябва да се изпълняват с помощта на прякото въздействие на елементите на OLAP модела без необходимост от използване на меню или други спомогателни средства

Гъбкава отчетна система



- Бизнес потребителят трябва да има възможност да манипулира с данните, да анализира и / или синтезира, а също така и да ги преглежда по начин, какъвто на него му харесва

Неограничен брой измервания и нива на агрегация



- OLAP сървърът е длъжен да поддържа не по-малко от 15 измервания за всеки аналитичен модел. За всяко измерване трябва да се допуска неограничен брой определяни от потребителите агрегати

Интегрирани разпределени приложения



- Като правило при разпределените изчислителни мрежи се налага да се интегрират нееднородни БД
- Най-много за създаването на съответстваща технология е допринесъл международния консорциум OMG, публикувал редица документи, в които се специфицират архитектурата и инструменталните средства за поддръжка на разпределени информационни системи, интегрирани на базата на общ обектно-ориентиран подход



- В базовия документ се специфицира еталонния модел на архитектурата (*OMA - Object Management Architecture*) на разпределена информационна система
- Съгласуваната с OMA архитектура на приложна информационна система се представя като съвкупност от класове и екземпляри от обекти, които взаимодействат чрез поддръжката на брокер на обектните заявки (*ORB - Object Request Broker*). ORB, общите средства (*Common Facilities*) и обектните услуги (*Object Services*) се отнасят към категорията на междинното програмно осигуряване (*middleware*) и трябва да се доставят съвместно



- *Обектните услуги* представляват набор от услуги (интерфейси и обекти), които осигуряват изпълнението на базовите функции, необходими за реализацията на приложните обекти и обектите от категорията "обща средства" (например специфицирани служба за именоване на обектите, служба за дълговременно съхраняване на обектите, служба за управление на транзакциите и т.н.)
- *Общите средства* съдържат набор от класове и екземпляри на обекти, поддържащи функции, използвани в различни приложни области (например средства за поддръжка на потребителския интерфейс, средства за управление на информацията и т.н.)

В основата на ОМА лежи базовия обектен модел *COM (Core Object Model)*, в който са специфицирани такива понятия, като обект, операция, тип, подтипизация, наследяване, интерфейс. Определени са начините за съгласувано разширяване на COM в различните обектни услуги.

