

АРХИТЕКТУРА НА ИНФОРМАЦИОННИТЕ СИСТЕМИ, ПРЕДСТАВЕНИ В WEB

Веселина Недева
Технически колеж - Ямбол

Докладът представя съвременните архитектурни решения за изграждане на Web базирани информационни системи. Основните модели на архитектурите, които намират място в доклада са клиент-сървър архитектури, използващи сървъри на бази от данни или Data Warehouse, глобални разпределени информационни системи и архитектурни решения, базирани на технологията Интернет/Интранет.

Ключови думи: информационни системи, Web технологии, Складове за данни, OLAP, Интелектуален анализ на данни.

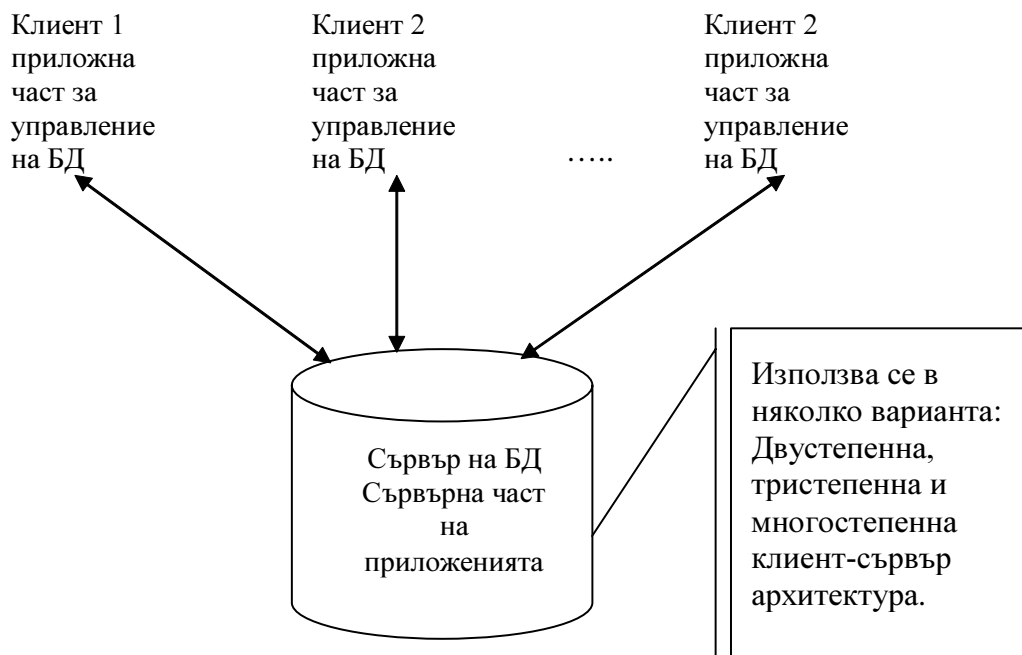
Key words: Information System, Web Technology, Data Warehouse, OLAP, Data Mining.

Архитектурата се представя във вид на съвкупност от модели, които се строят, за да може да се осмисли структурата и поведението на бъдещата система, да се облекчи процесът на нейното създаване и да се намали възможният риск, а също и документирането на предприеманите проектни решения. Разглеждани са особеностите на всяка архитектура и на методологията и инструментално-технологичните средства, поддържащи проектирането и разработването на информационните системи в съответната архитектура.

Архитектурите са разглеждани в различни комбинации, при което са съчетани с технологиите за съхранение на данните и технологии за комуникация в информационните системи (ИС). По такъв начин се получават различни архитектурни

решения, които могат да се приложат в зависимост от предназначението и целта системата, която трябва да работи в Web:

- архитектурни решения, базирани на технологията клиент-сървър, използващи сървъри на бази от данни;
- архитектурни решения базирани на концепцията СД;
- архитектура, предназначена за изграждане на глобални разпределени информационни приложения с интеграция на информационно-изчислителните компоненти на основата на обектно-ориентирания подход;
- архитектурни решения, базирани на технологията Интернет/Интранет.



Фиг. 1. Класическа клиент-сървър архитектура

Под клиент-сървърни приложения ще разбираме ИС, основана на използване на сървър на базата от данни (Фиг. 1.). Под сървър на БД ще разбираме програмно осигуряване на БД, което е независимо от съществуващите клиентски процеси и се изпълнява на отделен сървър. ИС в тази архитектура има следните характеристики:

- При клиента се изпълнява кода на приложенията, в който обезателно влизат компонентите, поддържащи интерфейса с крайните потребители, създаващи отчети и изпълняващи други специфични за приложението функции.

- Клиентската част на приложението взаимодейства със сървърната част на програмното осигуряване за управление на базите от данни.

Интерфейсът между клиентската част на приложението и клиентската част на сървъра на базите от данни по правило е основан на езика SQL. Клиентската част на сървъра на базите от данни, използвайки средствата за мрежов достъп, се обръща към сървъра на базите от данни, предавайки му текста на оператора на езика SQL.

Основно предимство на архитектурата клиент-сървър е нейната мащабируемост и способност за развитие. Увеличаването на мащаба на ИС не поражда принципни проблеми. Обикновено се заменя апаратурата на сървъра, а може и апаратурата на работните станции. При тези промени не се засяга приложната част на информационната система.

Клиент-сървър архитектурата е такава схема за изграждане на приложенията, обезпечаваща ефективно разпределение на логиката на програмната система между различни компютри в мрежата. Всеки елемент от компютърната мрежа има определен набор от параметри, най-важни от които са апаратните характеристики и програмното осигуряване. Развитата архитектура клиент-сървър позволява да се изгради приложението така, че всеки компютър максимално да използва силните страни на неговите специфични характеристики. Като правило, в сложните икономически системи това се достига с тристепенна схема за приложение (клиент-сървър-СУБД), когато клиентското приложение реализира само потребителски интерфейс, а бизнес-логиката се обработва от сървъра на приложенията.

Реализирането на тристепенна схема може да стане по различен начин. Създават се пълноценни сървърни приложения, използващи софтуер за управление на транзакции, за връзка с клиентските приложения и със СУБД. Често се прилага и друг подход, когато разработчиците използват технологията на хранящите процедури в СУБД като Sybase, Oracle и др. Вградените процедури по-

магат ефективно да се разпределят натоварванията между приложението-клиент и приложението-сървър. Те заместват софтуера за управление на транзакции, но не осигуряват необходимите средства за създаване и поддържане на мащабни проекти.

При изграждането на ИС можем да се ориентираме към архитектурата клиент-сървър с използване на софтуер за управление на транзакциите. При това общото разпределение на изчислителното натоварване е следното:

1. Сървърът на приложенията се занимава с отговорните задачи по изчисленията – бързи транзакции, генериране на отчети, поддържане логическата цялост на данните, реализация на алгоритмите за обработка, баланс на натоварването.

Сървърите на приложенията в ИС могат да бъдат два – сървър на аналитичната система (OLAP) и сървър за интелектуален анализ на данните (DM) (Фиг. 2.). Вторият сървър не присъства задължително в архитектурата. Това зависи от мащаба на фирмата, в която се реализира и от естеството на данните за обработка. Ако данните изискват сложна обработка с прилагане на методите на изкуствен интелект и едновременно с това обема на данните за обработка е порядъка на Терабайти, тогава се предвижда отделен сървър за ИАД.

Възможно е в информационната система да има един сървър на приложенията, в който се поддържа аналитичната система за анализ и системата за интелектуален анализ. На пазара се предлагат и настолни OLAP и системи за ИАД.

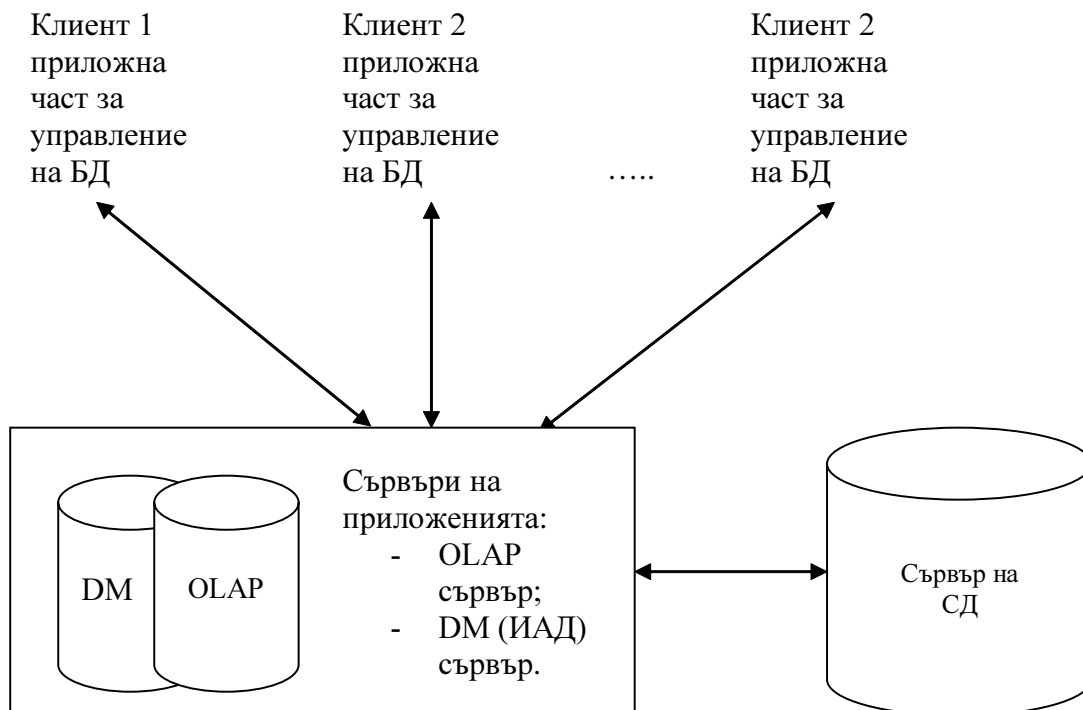
2. Сървър на Склада за данни (СД, Data Warehouse), на който се поддържат компонентите на склада за данни.

3. Приложението-клиент.

При това принципно са разделени интерфейсите на програмите и алгоритмите за обработка на данните.

Наличието на сървър на OLAP и ИАД и от друга страна на сървър на СД, предполага използването на подход, основан на концепцията на СД и OLAP. Общата архитектура на СД и OLAP включва няколко компонента – оперативните бази от данни, откъдето постъпва информацията, хранилището на СД и системата за аналитична информация.

Друго възможно архитектурно решение са глобалните разпределени информационни приложения с интеграция на информационно-изчислителните компоненти. Когато информационната система се разработва, опирайки се на принципа на отворените системи, всички нейни компоненти са мобилни и общото функциониране на системата не зависи от конкретното местоположение на компонентите.



Фиг.2. Многостепенна клиент-сървър архитектура

Такъв тип система притежава възможности за развитие. Възниква потребност от интегриране на независими и с различна организация информационно-изчислителни ресурси. Този проблем не може да се реши без приложение на технология за интегриране.

Решението на проблема за интегрирането на нееднородни информационни ресурси започва с интегрирането на нееднородни бази от данни. Направлението на интегрираните или федеративни системи от нееднородни бази от данни и мулти бази от данни, се появява във връзка с необходимостта от комплексно използване на системата на базите от данни, основани на различни модели на данните и управлявани от различни СУБД.

Основната задача на интеграцията на нееднородните бази от данни се явява предоставянето на потребителите на интегрирана система на глобалната схема на базите от данни, представени в някакви модели на данните, и автоматическото преобразуване на операторите за манипулиране на БД на глобално ниво, в разбираеми за съответстващите локални СУБД.

При строгата интеграция на нееднородните БД, локалните системи на БД загубват своята автономност. В системите с мулти-БД не се поддържа глобална схема на интегрираната БД и се прилагат специални способности за именуване на обектите на локалната БД, за да се осъществи достъп до тях. Като правило в такива системи на глобално

ниво се иска само подбор на данните. Така се запазва автономността на локалните БД.

Интегриране се налага за нееднородните БД, разпределени в изчислителната мрежа. Това в значителна степен усложнява реализацията. Допълнително към проблемите на интеграцията се налага да се решат и въпросите присъщи на разпределените системи за управление на базите от данни, като управление на глобалните транзакции, мрежова оптимизация на заявките и др. За външното представяне на интегрираните и мулти-БД се използва реляционният модел на данните. В последните години все по-често се прибегва до обектно-ориентираните модели, но на практика, в основата им е също реляционният модел. Основен недостатък на системите за интеграция на нееднородните бази от данни се явява това, че при това интегриране не се отчитат "поведенческите" аспекти на компонентите на приложната система. Става ясно, че даже и при наличие на развита интеграционна система, болшинството от указаните проблеми не се решават.

Съществен принос в създаването на съответстваща технология внесе международният консорциум OMG, създавайки редица документи, където се специфицират архитектурата и инструменталните средства за поддържане на разпределени информационни системи, интегрирани на основата на общия обектно-ориентиран подход. В базовия документ се специфицира еталонният модел на архи-

тектурата на разпределената информационна система.

Информационната система, съгласувана с архитектурата ОМА се представя като съвкупност от класове и екземпляри обекти, които функционират при поддръжката на брокер на обектните заявки (ORB - Object Request Broker). ORB, общите средства (Common Facilities) и обектните служби (Object Services) се отнасят към категорията на междинното програмно осигуряване (на средно ниво). Обектните служби са набор от услуги – интерфейси и обекти, които обезпечават изпълнението на базовите функции, необходими за реализацията на приложните обекти и обектите от категорията “обща средства”. Общите средства съдържат набор от класове и екземпляри обекти, поддържащи функции, които се използват в разни приложни области.

Интерфейсите на обекта-клиент и обекта-сървър трябва да бъдат определени на специален език IDL (Interface Definition Language).

Технологията на разпределените информационни приложения също може да бъде в основата на архитектурата на ИС. Това зависи от избора на софтуер за изграждане на СД. Някои от предлаганите софтуерни продукти (напр.Oracle), използват брокер на съобщенията, който е посредник при комуникациите между отделните разнородни бази от данни, включени в информационната система. Това се има предвид при избора на софтуер за реализация на СД в ИС.

Други архитектурни решения са базирани на технологията Интернет/Интранет. По своята същност информационната Интранет-система е информационна система, в която се използват методите и средствата от Интернет. Тя може да бъде локална, в смисъл изолирана от останалия свят на Интернет,

или да се свързва с виртуалната мрежа Интернет.

В общия случай Интернет-системите могат да използват всички методи и технологии, приложени в Интернет, но най-голямо внимание се отделя на хипермедийните възможности на Web. Затова има две основни причини:

1. С използването на езика за хипермедийно представяне на документите – HTML, може сравнително просто да се разработи удобна за използване информационна структура, която се обслужва от Web-сървър.

2. Използването на браузърите, ни освобождава от необходимостта да се създават собствени интерфейси за комуникация с потребителите, предоставяйки удобни и развити механизми за достъп до информацията.

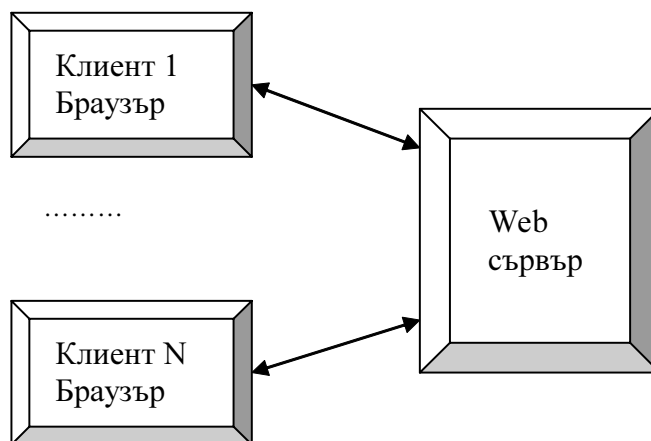
Независимо от своите преимущества Интранет-система с използване на средствата на Web създава и доста ограничения.

На първо място хипертекстовите структури трудно се модифицират. За да се направят промените, трябва да се преустанови работата на Web-сървъра, да се внесат промените в HTML кода и тогава да се продължи нормалното функциониране.

На второ място не винаги е достатъчно търсенето на информация в стил хипертекст. Базите от данни и съответстващите им средства за избор на данни, също често са необходими.

Класическата Интранет архитектура (Фиг.3.) не преодолява тези проблеми. Посочените трудности могат да бъдат решени с използването на по-развитите механизми на Web-технологиите. Те се усъвършенстват непрекъснато.

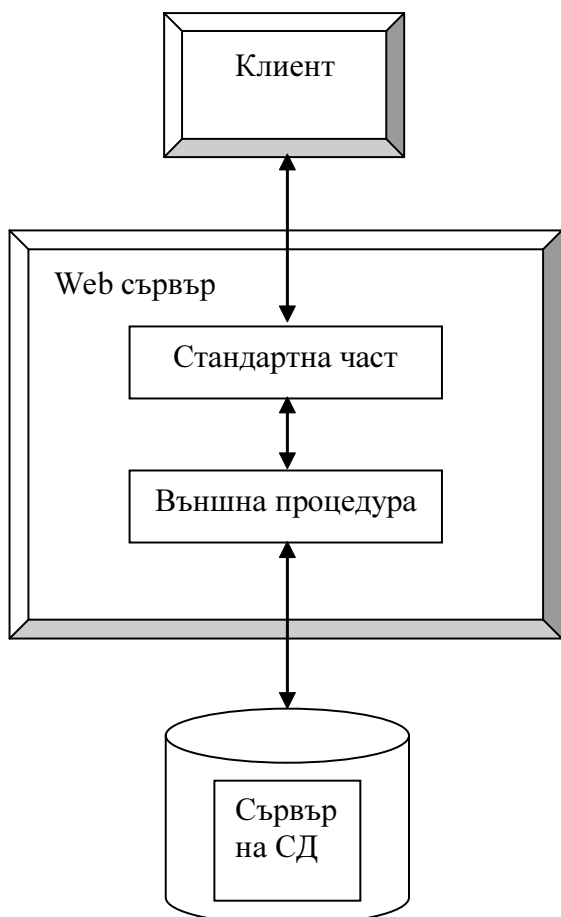
За взаимодействието на СД с Web-сървъра могат да се използват два подхода - CGI (Common Gateway Interface) и API (Application Programming Interface). И двата се основават на специални кон-



Фиг.3. Интранет архитектура

струкции в езика HTML, информиращи клиента-браузър, че ще получи специално съобщение. При получаването му сървърът извиква съответната външна процедура, получава резултатите и ги връща при клиента в стандартен HTTP формат.

Аналогична техника широко се използва за обезпечаване унифицирания достъп до базите от данни в Интранет-системите. Езикът HTML позво-



Фиг. 4. Складът за данни и взаимодействието му в Интранет

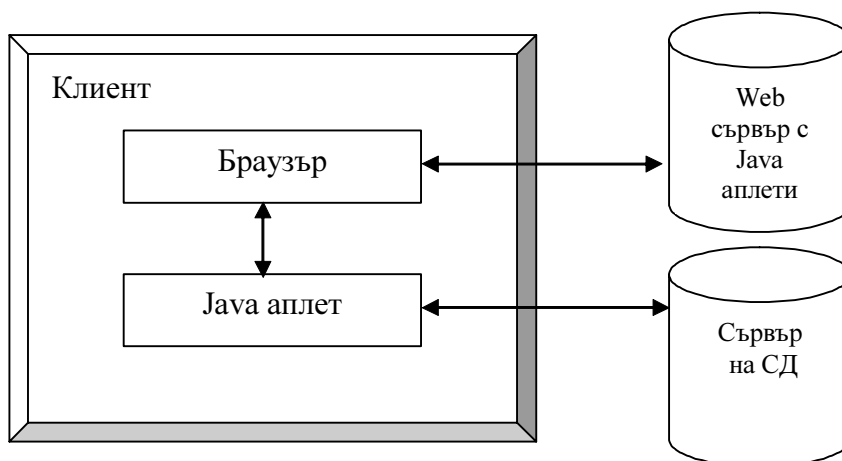
лява да се включват форми в хипертекстовите документи. Като правило към формата има прикрепена външна процедура на сървъра. При получаване на съобщения от клиента, сървърът извиква тази външна процедура, съдържаща параметрите на потребителя, попълнил формата. Такава външна процедура, в частност, може да изпълнява ролята на шлюз между Web-сървъра и сървъра на базите от данни. В този случай параметрите трябва да специфицират заявка на потребителите към базите от данни (Фиг. 4.).

При Интранет-системите е много важна ролята на езика Java. Мобилните кодове (аплети), получени в резултат на компилирането на Java-програми, могат да бъдат прикрепени към HTML-документ. Те постъпват на страницата на клиента, заедно с документа и се изпълняват или автоматически или по явно указание. Така аpletът също може да бъде шлюз към базите от данни. При тази техника достъпът до базите от данни се променя. (Фиг. 5.).

Интранет е мощно и удобно средство за приложение в информационните системи. Относителен недостатък на подхода може да се счита постоянно изменение на механизмите и развитие на стандартите на Интернет. От друга страна, ако информационната система е създадена на база на текущото ниво на технологиите и удовлетворява потребностите на потребителите, специалистите не са длъжни да я модифицират, поради появата на нови по-съвършени механизми.

В архитектурното решение за ИС могат да се обвързват няколко архитектурни концепции, които вече пояснихме. Те са разпределени по слоеве. Комбинациите между отделните концепции и слоеве могат да бъдат следните:

1. Двустепенна клиент-сървър архитектура в съчетание с архитектурата на СД в два слоя: сър-



Фиг. 5. Взаимодействие на сървърите на СД и Web с Java аплети

вър на СД и сървър на приложенията (OLAP). Тази архитектура може да бъде реализирана и в друг вариант: сървър на СД – сървър на приложенията (OLAP и ИАД);

2. Трестепенна клиент-сървър архитектура в съчетание с архитектурата СД в три слоя: сървър на СД – сървър на OLAP – сървър на ИАД и интерфейс приложенията.

3. Многостепенна клиент-сървър архитектура със следните слоеве: сървър на СД – сървър на OLAP и ИАД – брокер на заявките – сървър на приложенията.

4. Интернет/интранет информационна система в съчетание с архитектурата СД: сървър на СД – Web сървър – клиенти.

Многослойната клиент/сървър архитектура на ИС може да се базира на няколко сървъра, изпълняващи основни комуникационни и изчислителни функции в системата. Като такива могат да се използват Data Mining сървър, Multidimensional Database (MDB) сървър, OLAP сървър и Web сървър. Представената концепцията на многослойната клиент-сървър архитектура показва, че комуникациите между отделните работни станции в Интранет и достъпът до Интернет става по протокола HTTP посредством браузър. За функционирането на Интранет може да се налага включването и на други сървъри, както и такива за изграждане на защитни стени. Заявките на потребителите към сървърите на СД и OLAP се подготвят на SQL.

Разработените варианти за архитектурата и структурата на ИС са подходящи за приложение, както в големи корпорации с много филиали, така и в средни по големина фирми. Основание за този извод ни дават възможностите за приложение на отделните технологии, както в мрежови условия, така и за настолни компютри, което вече бе изяснено. Архитектурните решения също могат да

се подберат съобразно различните обстоятелства във фирмите. Цялостното изграждане на СД и завършване на ИС може да стане на етапи като се обхващат постепенно обектите и се проектира и реализира решаването само на част от задачите. Постепенно техният обхват се разширява на базата на увеличаване на обектите обхванати в СД, докато се стигне до окончателно изграждане и функциониране на ИС. СД може да се развие и да прерасне в корпоративен склад за данни на основата на вече изградената информационна база по маркетинг.

Литература

1. Архипенков, Сергей, Что, если...? Или экономическое моделирование средствами ORACLE Express, РУССКОЕ ИЗДАНИЕ ORACLE MAGAZINE - №3(5) 1997г.
2. Архипенков С.Я., Завьялов Б.П., Шеховцов А.И. Объектно-ориентированный подход в задачах моделирования сложных систем. Сб. Вопросы кибернетики, серия "Моделирование сложных систем и виртуальная реальность", Москва, 1997
3. Асадуллаев, Сабир, Фирменные архитектуры хранилищ данных, PC Week/RE'98
4. Вавилов К., С.Щербина, Web-интеграция корпоративных систем сайте <http://www.profi-club.kiev.ua/>, 2001
5. Минчева, Е., Технологии за разпределена обработка, Колор Принт, Варна, 2000
6. Сахаров А.А., Принципы проектирования и использования многомерных баз данных (на примере ORACLE Express SERVER), <http://www.its.kiev.ua/>, 2002
7. "Data Warehouse Strategies," The Forrester Report, Volume Eight, Number Six, September 1997. Copyright 1997,
8. DW Data Preparation, Conference Presentation, SYM8DWDData1098JHill, GartnerGroup, 1998

ARCHITECTURE OF THE WEB BASED INFORMATION SYSTEMS

Veselina Nedeva
Technical college – Yambol

ABSTRACT

The report presents contemporary architecture for building Web Information System. The basic architecture models that are presented in the report are client-server, using Data Base or Data Warehouse, global spread Information Systems and Internet-Intranet-based architectural decisions.