



ХИМИКОТЕХНОЛОГИЧЕН И МЕТАЛУРГИЧЕН УНИВЕРСИТЕТ
ДЕПАРТАМЕНТ ПО ФИЗИКОМАТЕМАТИЧЕСКИ И ТЕХНИЧЕСКИ НАУКИ
КАТЕДРА „ПРОГРАМИРАНЕ И ИЗПОЛЗВАНЕ НА
ИЗЧИСЛИТЕЛНИ СИСТЕМИ”

гл. ас. инж. Димитър Иванов Пилев

А В Т О Р Е Ф Е Р А Т

на дисертационен труд

СЪВРЕМЕННИ WEB-БАЗИРАНИ ИНФОРМАЦИОННИ СИСТЕМИ
И БАЗИ ОТ ДАННИ С ВРЕМЕВИ ХАРАКТЕРИСТИКИ

за придобиване на образователната и научна степен „доктор”
в област на Висше образование 4. Природни науки, математика и информатика,
професионално направление 4.6. Информатика и компютърни науки, научна
специалност 01.01.12 Информатика

Научен ръководител: доц. д-р инж. Анета Георгиева

Научно жури:

1. доц. д-р Ангел Дишлиев - председател
2. доц. д-р Антон Илиев - рецензент
3. доц. д-р Димитър Богданов - рецензент
4. проф. дмн Снежана Христова
5. доц. д-р Анета Георгиева

София, 2011

Пълният обем на дисертационния труд е 155 страници, от които заглавната страница, съдържанието и използваната литература са 14 страници. Дисертационният труд се състои от три глави (57 фигури, 36 таблици), заключение, публикации по дисертационния труд и списък с използваната литература.

Използваната литература включва 117 заглавия (106 на английски език и 11 на български език). Списъкът от публикации на докторанта по дисертационния труд включва 8 заглавия.

Представеният дисертационен труд е обсъден и приет на защита на заседание на разширен научен съвет на научното звено на катедра „Програмиране и използване на изчислителни системи“, състояло се на 16.05.2011 г.

Публичната защита на дисертационния труд ще се проведе на 15.07.2011 г. от 11:00 часа в зала 100, сграда „Б“ на ХТМУ.

Материалите са на разположение на интересувашите се на интернет страницата на ХТМУ и в отдел „Научни дейности“, стая 406, сграда „А“ на ХТМУ.

Въведение

Съвременните Web-базирани информационни системи (ИС) обслужват все повече области от човешката дейност. Те навлизат все по-широко не само в бизнеса, но и в социалната сфера, образованието, медицината и цялостното информационно обслужване на обществото. Непосредствените резултати от внедряването и използването на съвременните Web-базирани ИС са свързани с пестене на време и ресурси, с подобряване качеството на административното обслужване и информационното осигуряване на обществото.

Съществените компоненти на ИС са моделът на базата от данни (БД) и системата за управление на БД (СУБД).

В обикновените бази от данни не съществува концепция за времето - базата от данни има едно текущо състояние и към нея могат да бъдат извършвани запитвания само за това състояние.

В редица области е необходимо да се съхранява изменението на обработваните данни във времето. Широко използваният в базите от данни релационен модел на данните е в състояние да обхване само някои от аспектите на времето. Поради това са предложени както многобройни разширения на релационния модел, така и нови модели за работа с данни, вариращи с времето.

В настоящия дисертационен труд са изследвани известните в световен мащаб времеви релационни модели. При тяхната практическата реализация възникват редица проблеми. Значително се повишава обемът на съхраняваната информация. Усложняват се алгоритмите за извличане, добавяне и актуализиране на данните. Забавя се обработката на информацията. Често се налага съхранение на излишна информация, както и използването на допълнителни процедури за нейното редуциране.

В настоящия дисертационен труд е предложен и реализиран практически нов времеви модел на релационна база от данни. Разработени са алгоритми за добавяне, модифициране и изтриване на темпорална информация в БД.

Актуалност на темата

Традиционните модели на БД и СУБД могат да съхраняват и обработват само едно текущо състояние на моделираната предметна област. Обикновено това са текущи състояния, като при необходимост от промяна старите им стойности се унищожават. В реалния свят обаче, съществуват редица области, при които е необходимо да се осъществява достъп не само до текущото състояние на данните, но и до техните минали и дори бъдещи състояния. Такива области са счетоводство и финанси, банково дело, поддържане на данни за персонал, резервации на билети за транспорт, хотелски резервации и управление на проекти.

В световен мащаб, в областта на научните изследвания се забелязва голям интерес към темпоралните БД. С активното участие на няколко стотици учени през последните две десетилетия са написани повече от 2000 статии. Най - известните учени в областта на темпоралните БД са R. Snodgrass, C. Jensen, J. Clifford, S. Gadia, N. Lorentzos, N. Sarda, A. Steiner, G. Garani и др. Повечето от тези публикации са обхванати в серия от библиографии. Публикувана е енциклопедия [17], речник с терминологията [4], свързана с темпоралните БД. Разработени са и редица дисертации [9], [10], [11], [12], в които са разгледани основните концепции и проблеми, свързани с темпоралните БД. Ето защо, развитието на съществуващите и създаването на нови подходи, модели и алгоритми за работа с времеви данни води до по-ефективното решаване на голям брой задачи. Това прави изследванията в областта на темпоралните бази от данни изключително актуална тема.

От друга страна, до скоро съществуващите СУБД не позволяваха времева обработка на данните. За реализиране на темпорална поддръжка е необходимо промяна на сорс-кода на тези СУБД, което е твърде трудоемка задача. Актуалността на проблема се подчертава и от факта, че през септември 2010 г. комерсиалните СУБД Oracle и Teradata публикуваха нови спецификации на СУБД с времева поддръжка [83],[85],[86].

Цели и задачи на дисертационния труд

Понастоящем са предложени множество от несъвместими времеви модели. Някои от тях се основават на прости разширения на конвенционалния релационен модел, други модели представляват доста по - сложни нови предложения. Техните разлики се основават на различните подходи при моделирането на времевите

характеристики на данните. Това разнообразие усложнява значително много проблема, свързан с моделирането на времевите характеристики на данните.

Целта на дисертационния труд е да бъде създаден ефективен времеви модел на база от данни. Той трябва да може да се използва за моделиране на различни области от заобикалящата ни реалност, да постига гъвкавост при манипулиране на данните, съчетана с лекотата при представяне, използване и реализиране. Моделът да бъде разширение на конвенционалния релационен модел. По този начин времевият модел ще може да се използва съвместно с най-разпространените СУБД (Oracle, MySQL, SQL Server и др). Моделът е приложен при реализацията на конкретна информационна система.

Във връзка с основната цел се поставят следните конкретни задачи:

- Да се дефинира тип ефективно време, което се използва за маркиране на данните в кортежа. Използването му може да доведе в някои случаи до намаляване на обема на необходимата памет;
- Да се дефинира ефективен темпорален модел на БД. Да се опише структурата на модела и да се опишат алгоритми за добавяне, изтриване и модифициране на времевите данни;
- Предложеният темпорален модел да се използва при разработване на конкретна система за информационно осигуряване на учебния процес във ВУЗ.

Глава 1. Модели на бази от данни с времеви характеристики

В обзорната първа глава накратко са представени традиционните модели и системи за управление на БД. Подробно са разгледани и анализирани моделите на БД с времеви характеристики. Показана е необходимостта от тяхното използване, начините за тяхното реализиране и основните подходи, използвани при времевите модели данни. Разгледани са известни релационни модели на данни с времева поддръжка.

Основни понятия, използвани при времевите модели данни

База от данни, която съдържа версиите на измененията, настъпили в предметната област, се нарича темпорална база от данни. Темпоралната база от данни предоставя вградена поддръжка за обработване на времеви данни.

При времевите бази от данни, наличието на повече от една времева ос е предложено с цел улавяне на различни аспекти на връзката между времето и данните. Поради това във времевите модели на БД се използват различни понятия за време.

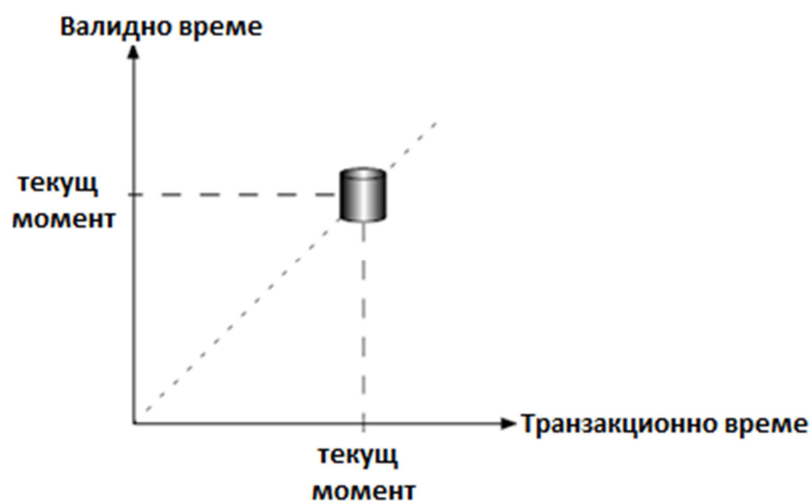
Време, дефинирано от потребителя. Това е такова време в БД, при което потребителят сам интерпретира времевата информацията (напр., стойностите, записани в атрибут, съхраняващ данни за рождени дни на служителите на дадена организация). СУБД третира тези времеви данни като отделен атрибут.

Валидно време. Това е времето, през което даден факт е верен в моделираната предметна област (може да бъде в миналото, настоящето или бъдещето). Валидното време улавя променящите се състояния на моделираната предметна област. По дефиниция всички факти имат валидно време. Тези факти могат да не бъдат записани в БД. Валидното време не зависи от записването на фактите в БД.

Транзакционно време. Това е времето, през което даден факт от моделираната предметна област е съхранен в БД. То започва от записването на факта в БД и завършва при изтриването на факта от БД. Допустимо е многократно записване и изтриване на факта. Изтриването на фактите от БД е логическо. Фактът остава в базата от данни, но престава да бъде част от текущото ѝ състояние. Транзакционното време улавя променящите се във времето състояния на БД, но не може да приема за край на периода бъдещ момент от време.

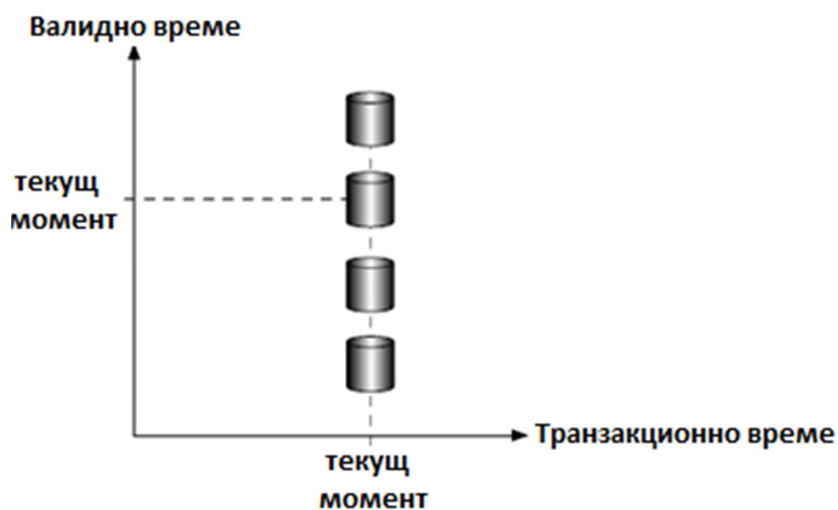
Както бе споменато, съществуват различни начини за представяне на времето. Въз основа на това са представени няколко вида БД, различаващи се по способността им за представяне на времева информация.

Моментна база от данни (Snapshot БД). Тя съхранява само едно моментно състояние на факта от реалния свят, обикновено настоящето. На Фиг. 1 е показана моментна БД, по осите на валидното и транзакционно време. При извършване на модификация в БД, текущото състояние се унищожава, като на негово място се записва ново. По този начин, след изпълнение на успешна транзакция, няма никакъв начин за възстановяване на предишното състояние на БД.



Фиг. 1 Моментна база от данни (Snapshot БД)

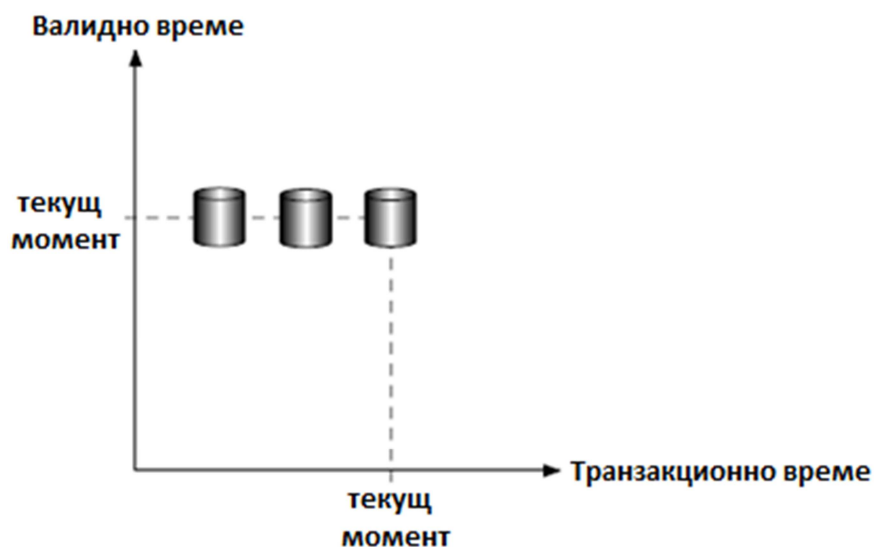
Исторически бази от данни. Те съхраняват историята на данните в реалния свят. Динамиката на реалния свят е уловена по оста на валидното време. Така могат да бъдат записвани минали, настоящи и бъдещи състояния на данните (Фиг.2).



Фиг. 2 Историческа база от данни

Историческите бази от данни изискват по-сложен език за заявки, позволяващ времеви запитвания за избор на конкретни състояния на данните в БД.

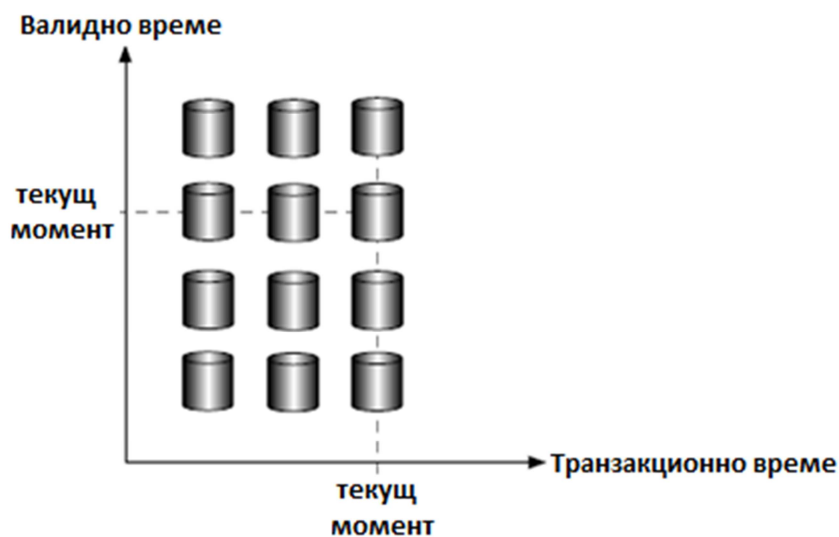
ROLLBACK бази от данни. Те записват промените в самата база от данни. Такава БД отчита само транзакционното време, като не позволява записване на бъдещи състояния.



Фиг. 3 Rollback база от данни

Всеки път, когато се извършва модификация на данните, системата записва новото състояние на БД (Фиг. 3). При тези БД, физически информацията никога не се изтрива.

Битемпорални бази от данни. Тези бази от данни са комбинация от исторически и Rollback бази от данни (Фиг. 4).



Фиг. 4 Битемпорална база от данни

При времевите модели на базите от данни фактите се представят, като се обозначава периодът им на валидност в реалния свят и/или кога са съхранени в базата от данни.

Понастоящем съществува голямо многообразие от времеви модели на БД. Тези модели се различават по вида на времето съхранявано в БД, по начина на маркиране на времето, по използваните елементи за маркиране на времето.

Времево маркиране на данните

Използват се два начина за времево маркиране на данните - маркиране на кортеж с време или маркиране на атрибут с време.

Обозначаването на кортежи с време обикновено се прилага при времеви релационни модели на данни, поддържащи само релации от първа нормална форма (1NF). В тези модели се добавят времеви отпечатъци на всеки кортеж в релацията. В случай на исторически данни, всеки кортеж се маркира с времеви период, означаващ кога съответният кортеж е валиден в реалния свят. В *rollback* базите от данни кортежите се свързват с периоди от транзакционно време. Кортежите при битемпоралните бази от данни се маркират с периоди на валидно и транзакционно време. Времеовото маркиране на данните се постига чрез добавяне на специални атрибути за време към традиционната (невремева) схема на релацията.

Основният недостатък на този подход е, че информацията е разпределена в няколко кортежа, като всеки кортеж представлява състояние на фактите от реалния свят в рамките на определен период от време. В този случай се получава голямо излишество на съхранявана информация.

Маркирането на атрибут с време преодолява недостатъците на маркирането на кортеж с време. При този подход се добавя времево маркиране на стойностите на всеки атрибут. По този начин данните в кортежа, които не са засегнати от промяна, не се повтарят. Така историята на стойностите на всеки атрибут се съхранява отделно. Характерно за този подход е, че основният модел на данните поддържа релации в непърва нормална форма (NFNF). Като негов недостатък може да се посочи използването на доста по-сложни алгоритми за генериране на заявки към БД в сравнение с подхода, използващ маркиране на кортеж с време.

Начини на маркиране на времето

Времевите модели се различават в зависимост от това дали се използва маркиране на кортежи или атрибути с време. Използването на един от двата подхода се определя от основния модел на данните. Обикновено данните могат да бъдат маркирани с времето чрез времеви момент, интервал от време или времеви елементи

(като последните представляват съвкупност от несъседни и неприпокриващи се времеви интервали). Времевите реляционни модели, реализирани в първа нормална форма (1NF), използват времево маркиране, чрез добавяне на допълнителни атрибути към кортежа, имащи атомарни стойности. Моделите, поддържащи релации от NFNF, позволяват използването на по-сложни стойности на атрибутите.

Глава 2. Разработване на ефективен темпорален модел

Във втора глава е представен нов времеви модел на БД, наречен Ефективен темпорален модел (ETM).

Направени са дефиниции за ефективно време, ефективна релация и ефективен темпорален модел.

Ефективно време. Това е времето, през което фактите от моделираната предметна област са едновременно валидни и записани в БД. За начало на времевия период при ефективното време се задава текущото в момента на записа на кортежа в БД време. Краят на времевия период е бъдещ момент, задаващ края на валидност на фактите.

Ефективна релация. Това е релацията, която използва период от ефективно време за маркиране на данните в кортежа.

Ефективен темпорален модел (ETM). Дефинираме ETM като хомогенен модел на данни, използващ ефективно време при маркиране периода на валидност на данните в кортежа.

При ETM всички данни в кортежа се характеризират с едно и също време. Релацията винаги е в първа нормална форма (1NF). Релационната схема се образува като към стандартните невремеви атрибути, описващи свойствата на субекта, са добавени два задължителни атрибута: E_s - начало на ефективния период и E_e - край на ефективния период. Ефективният период $t_e = [E_s, E_e)$ е със затворена долна и отворена горна граница.

Дефинирана е следната структура на релационната схема при ETM:

$$R = (A_1, A_2, \dots, A_n, E_s, E_e)$$

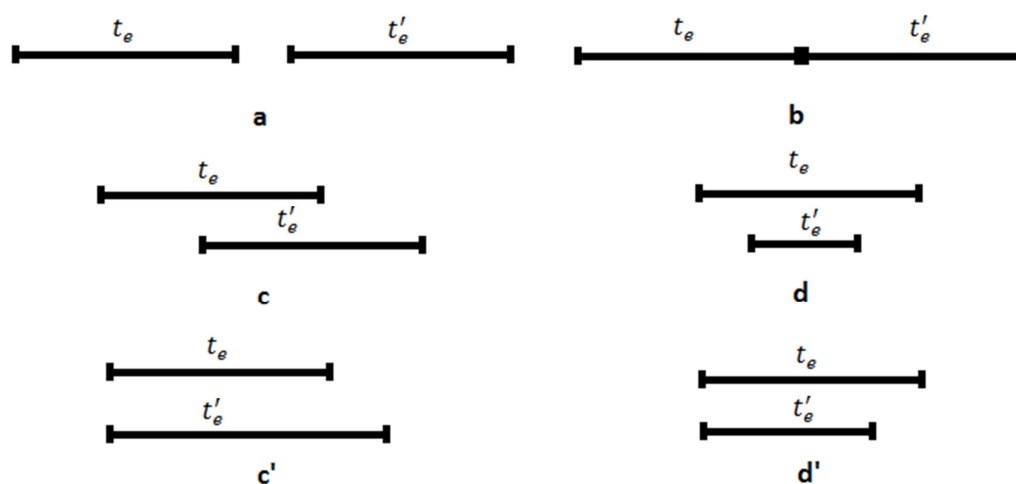
Представена е и семантиката на всяка една от трите форми на актуализиране на данните в модела.

Добавяне на данни

Добавянето на данни в ефективната релация се осъществява, когато желаем да запишем незаписани в БД факти (a_1, \dots, a_n) , които са валидни за определен период от време. От момента на записване на фактите в БД нататък ефективното време съвпада с валидното време. Това означава, че фактите са валидни в моделираната реалност и са записани в БД. След записа на данни се получава нова актуализирана версия на релацията.

Нека t_e представлява периода от време, с който са асоциирани стойностите на записаните в БД невремени атрибути (a_1, \dots, a_n) , $t_e = [E_s, E_e)$. С t'_e означаваме новия времеви период, маркиращ същите стойности на невремени атрибути (a_1, \dots, a_n) , които трябва да бъдат записани в БД, $t'_e = [E'_s, E'_e)$.

Като се отчита семантиката на ефективното време, винаги $E_s \leq E'_s$. При актуализиране на данните в релацията са възможни само част от предложените от Allen [41] операции (предикати) за работа с времеви интервали.



Фиг. 5 Възможни варианти на разположението на интервалите при добавяне на данни в релацията при ЕТМ

За представяне на семантиката на актуализиране на данните в модела са дефинирани следните функции за работа с ефективните времеви интервали.

Функцията $meets(t_e, t'_e)$ извършва проверка за срещане на два интервала:

$$\begin{aligned}
 &meets(t_e, t'_e) \\
 &\quad - \text{return } true \quad \text{if } (E_e = E'_e) \\
 &\quad - \text{return } false \quad \text{else}
 \end{aligned} \tag{1}$$

Функцията $overlap(t_e, t'_e)$ извършва проверка за частично припокриване на два интервала:

$$\begin{aligned}
 &overlap(t_e, t'_e) \\
 &\quad - \text{return } true \quad \text{if } (E'_e < E_e) \\
 &\quad - \text{return } false \quad \text{else}
 \end{aligned} \tag{2}$$

Съществуват три случая при добавяне на нови данни в релацията (3).

Първи случай: Нека стойностите на независещите от времето атрибути (a_1, \dots, a_n) не са били записвани до момента в релацията или са част от предишни нейни състояния. В такъв случай не съществува запис, при който ефективният период от време да се припокрива с новия период (Фиг.5a). При такава ситуация добавяме нов кортеж в релацията.

Втори случай: Ако стойностите на атрибутите (a_1, \dots, a_n) са били записани в релацията и съществува запис, при който периодът му на валидност t_e се среща с новия t'_e , т.е. $E_e = E'_e$ (Фиг.5b), ефективният период, с който е маркиран кортежа, се актуализира до $[E_s, E'_e)$.

Трети случай: Ако стойностите на атрибутите (a_1, \dots, a_n) са част от текущото състояние на релацията (ефективният период, чрез който данните в кортежа са маркирани, все още е активен) се изисква модифициране на данните, а не тяхното добавяне (Фиг.5c, Фиг.5d).

$$insert(r, (a_1, \dots, a_n), t'_e) =$$

$$\begin{cases} r \cup \{(a_1, \dots, a_n | t'_e)\} & \text{if } \nexists t_e ((a_1, \dots, a_n | t_e) \in r) \vee \exists t_e ((a_1, \dots, a_n | t_e) \in r \wedge \neg overlap(t_e, t'_e) \wedge \neg meets(t_e, t'_e)) \\ r - \{(a_1, \dots, a_n | t_e)\} \cup \{(a_1, \dots, a_n | changeETP(t_e, t'_e))\} & \text{if } \exists t_e ((a_1, \dots, a_n | t_e) \in r \wedge meets(t_e, t'_e)) \\ r & \text{в противен случай} \end{cases} \tag{3}$$

За актуализирането стойностите на данните (a_1, \dots, a_n) , които са част от текущото състояние на релацията, се използва функцията

$changeETP(t_e, t'_e)$ (4). Тази функция променя по един и същи начин ефективния времеви период, чрез който данните са маркирани в кортежа, независимо от това дали имаме срещане или припокриване на стария и новия период (t_e и t'_e).

$changeETP(t_e, t'_e)$

$$\begin{aligned} & - \text{return } [E_s, E'_e) \quad \text{if } (meets(t_e, t'_e) \vee overlap(t_e, t'_e)) \\ & - \text{return } 0 \quad \text{else} \end{aligned} \quad (4)$$

Изтриване на данни

Изтриването на даден кортеж се състои в неговото логическо изтриване от текущото състояние на релацията (5).

$delete(r, (a_1, \dots, a_n)) =$

$$\begin{cases} r - \{(a_1, \dots, a_n|t_e)\} \cup \{(a_1, \dots, a_n|delETP(t_e))\} & \text{if } \exists t_e((a_1, \dots, a_n|t_e) \in r \wedge CT \in t_e) \\ r & \text{в противен случай} \end{cases} \quad (5)$$

CT – текущото в момента на изпълнение на заявката към БД време.

Логическото изтриване на кортежа се осъществява чрез коригиране на времевия период, с който стойностите на невремевите атрибути (a_1, \dots, a_n) са маркирани. Това коригиране се реализира чрез задаване на текущото в момента на изтриване на кортежа време за край на ефективния период от време. За тази цел използваме функцията $delETP(t_e)$ (6).

$delETP(t_e)$

$$\begin{aligned} & - \text{return } [E_s, CT) \quad \text{if } (CT \in t_e) \\ & - \text{return } 0 \quad \text{else} \end{aligned} \quad (6)$$

Ако стойностите на атрибутите (a_1, \dots, a_n) не съществуват или, ако съществуват, но не са част от текущото състояние на релацията, изтриването на кортежа не би имало ефект.

Модифициране на данните

Модифицирането на съществуващ кортеж се извършва чрез операцията $update$ (7). Тя се дефинира като последователно извършване на операциите $delete$ (5) и $insert$ (3).

$$update(r, (a_1, \dots, a_n), t'_e) = insert(delete(r, (a_1, \dots, a_n)), (a_1, \dots, a_n), t'_e) \quad (7)$$

При така предложената семантика значително се намалява излишеството на съхраняваните данни. Намалява се и обемът на съхранената в БД информация. Можем да придобием точна представа за валидността на данните във времето. Същевременно се спазват условията за пълнота и непротиворечивост при съхранението на данните в релацията:

- Релацията в модела не съдържа кортежи с идентични независещи от времето атрибути, маркирани с припокриващи се или срещащи се времеви периоди. При такива кортежи автоматично се обединяват времевите им периоди.
- Нов кортеж може да бъде добавен, само когато в релацията не съществува кортеж със същите стойности на независещите от времето атрибути, при който $CT < Ee$.
- При всеки добавен нов кортеж в релацията, началото на времевия период E_s съвпада с текущото време CT .
- Само кортежи, които са част от текущото състояние на релацията могат да бъдат актуализирани и изтривани.

За практическата реализация на модела са предложени алгоритми за добавяне и изтриване на данните в ефективната релация. Използвайки тези алгоритми, лесно можем и да модифицираме данните в релацията.

Алгоритъм за добавяне на данни в ефективната релация

За реализиране на алгоритъма са дефинирани следните функции:

$MaxEndPeriod(r, (a_1, \dots, a_n))$ - връща най-голямата стойност на атрибута E_e , маркиращ (a_1, \dots, a_n) в релацията r .

$MaxEndPeriod(r, (a_1, \dots, a_n))$

return SELECT max(E_e) FROM r WHERE $A_1 = a_1$ AND $A_2 = a_2$... AND $A_n = a_n$ (8)

$InsertTuple(r, (a_1, a_2, \dots, a_n), e'_e)$. Тази функция въвежда кортеж в ефективната релация r . Характерно за функцията е, че освен стойностите на невремевите атрибути (a_1, \dots, a_n) , се задава и края на маркиращия ги ефективен период от време e'_e . За

начало на периода се използва текущото в момента на изпълнение на функцията време, означено с CT .

$InsertTuple(r, (a_1, a_2, \dots, a_n), e'_e)$

$INSERT INTO r (A_1, A_2, \dots, A_n, E_s, E_e) VALUES (a_1, a_2, \dots, a_n, CT, e'_e)$ (9)

$UpdateTuple(r, (a_1, a_2, \dots, a_n), e'_e, e_m)$. Тази функция извършва актуализиране на ефективния период от време, маркиращ кортежа с атрибути (a_1, \dots, a_n) . Актуализирането се осъществява като за край на времеви период се задава e'_e . Тъй като в релацията r може да съществуват повече от един кортеж с невремени атрибути (a_1, \dots, a_n) , при актуализирането се отчита и края на периода, зададен с e_m , където e_m е резултатът, върнат от функцията $MaxEndPeriod(r, (a_1, \dots, a_n))$ (8).

$UpdateTuple(r, (a_1, a_2, \dots, a_n), e'_e, e_m)$

$UPDATE r SET E_e = e'_e WHERE A_1 = a_1 AND A_2 = a_2 \dots AND A_n = a_n AND E_e = e_m$ (10)

Като входни параметри на алгоритъма за актуализиране на ефективната релацията се подават релацията r , която се актуализира, стойностите на невремените атрибути (a_1, \dots, a_n) , както и края на маркиращия ги ефективен времеви период e'_e .

$InsertET(r, (a_1, a_2, \dots, a_n), e'_e)$:

IF $CT < e'_e$

$e_m = MaxEndPeriod(r, (a_1, \dots, a_n));$

IF $e_m == NULL$

$InsertTuple(r, (a_1, a_2, \dots, a_n), e'_e);$

ELSE

IF $e_m < CT$

$InsertTuple(r, (a_1, a_2, \dots, a_n), e'_e);$

ELSE IF $e_m == CT$

$UpdateTuple(r, (a_1, a_2, \dots, a_n), e'_e, e_m);$

Този алгоритъм може да се опише по следния начин:

1. Ако зададената стойност на e'_e удовлетворява изискванията на модела ($CT < e'_e$), се преминава към т.2, в противен случай изпълнението на алгоритъма се прекратява. Преминава се към т.б.
2. Намира се най-голямата стойност e_m на атрибута E_e , използвана за маркиране стойностите на атрибутите (a_1, \dots, a_n) в релацията r . Преминава се към т.3.
3. Ако стойността на e_m е null се добавя кортеж с атрибути (a_1, \dots, a_n) , маркирани с времеви период $[CT, e'_e]$ в релацията r . Преминава се към т.б.
4. Ако стойностите на атрибутите (a_1, \dots, a_n) не са част от текущото състояние на r и край на маркиращия ги период E_e не съвпада с текущото време се добавя кортеж с атрибути (a_1, \dots, a_n) , маркирани с времеви период $[CT, e'_e]$ в релацията. Преминава се към т.б. В противен случай се преминава към т.5.
5. Ако стойностите на атрибутите (a_1, \dots, a_n) не са част от текущото състояние на релацията и край на маркиращия ги период E_e съвпада с текущото време се актуализира ефективният период от време, маркиращ невремеви атрибути (a_1, \dots, a_n) в релацията. За негов край се задава e'_e .
6. Край на алгоритъма.

Алгоритъм за изтриване на данни в ефективна релация

Изтриването на даден кортеж (a_1, \dots, a_n) в ефективната релация r се извършва като за край на маркиращия го ефективен времеви период се задава текущото в момента на изпълнение на заявката за изтриване време. Могат да бъдат изтривани само кортежи, които са част от текущото състояние на релацията r .

$DeleteET(r, (a_1, \dots, a_n)) :$

$UPDATE r SET E_e = CT WHERE CT < E_e$

$AND A_1 = a_1 AND A_2 = a_2 \dots AND A_n = a_n$

При реализацията на предложените алгоритми е използван стандартен SQL. Това позволява те да се прилагат независимо от СУБД, чрез която е реализирана времевата база от данни, използваща ефективно време за маркиране на данните в кортежа.

Предимства на ЕТМ

Значително се намалява обемът на съхраняваните данни, като същевременно не се губи полезна информация. При актуализиране на данните в ефективната релация, особено когато това актуализиране е свързано с промяна на маркиращия ги времеви период, не се извършва добавяне на нов кортеж в релацията. Това води до намаляване излишеството на съхраняваната в БД информация, което е характерно за темпоралните модели на БД.

Данните в релацията винаги са обединени - атрибутите не могат да имат множество стойности в даден момент от време.

Бързо и лесно можем да придобием точна представа за времето, през което фактите от моделираната предметна област са едновременно валидни и записани в БД.

Намалява се сложността на използваните алгоритми при манипулирането на данните в базата.

Намаляват се и разходите, свързани с администрирането на БД. При записване на данните в базата се посочва и краят на периода (в бъдеще време) на тяхната валидност. След изтичане на този период кортежът логически се изтрива от БД.

Глава 3. Приложение на ЕТМ при система за информационно осигуряване на учебния процес във ВУЗ

На база на ЕТМ е разработена Web-базирана система за информационно осигуряване на учебния процес във ВУЗ. Системата предоставя информация за оценки от проведените изпити и допълнителна статистическа информация за успеваемостта на студентите по групи, дисциплини, потоци и др. Реализиран е WAP достъп до ресурсите на системата. Информацията е достъпна през Интернет чрез стандартен Web - браузер или минибраузер на мобилен телефон /GSM/.

Разгледани и реализирани са съвременните средства за защита на данните в информационната система – достъп чрез парола, цифров сертификат и криптирана комуникационна сесия.

Предложеният в Глава 2 ЕТМ е използван и за реализиране на Web-базирана информационна система за експресно публикуване на кратки текстови съобщения в

Интернет от широк кръг оторизирани потребители. Реализиран е достъп до системата през стандартен браузер и през минибраузер на GSM.

Web-базирана система за информационно осигуряване на учебния процес във ВУЗ

Разработването на Web-базирана информационна система е свързано със създаването на конкретен модел на базата данни. Моделът трябва да бъде предназначен за обслужването на студенти от конкретен ВУЗ, имащи различни специалности, изучаващи различни дисциплини, групирани в различни потоци, с различна форма на обучение /редовно, задочно и др./. Същевременно моделът трябва да позволява обновяването и публикуването на данните да се осъществява от точно определени оторизирани за конкретни периоди от време лица (преподаватели), например за един семестър. Само така може да се осигури бърз и ефективен начин за обновяване на информацията в системата.

Правото на достъп на конкретни преподаватели до определени ресурси е свързано с фактора “време”. От съществено значение е времето, през което даден преподавател има реално право на достъп и същевременно е записан в БД. Данните за студентите също могат да се променят с времето.

За моделирането на данните в системата е използван предложеният времеви релационен модел (ETM). По този начин, чрез използване на ефективно време, се автоматизира администрирането на системата. При преустановяване достъпа на даден преподавател, свързаните с него данни логически се изтриват от текущото състояние на БД. Същевременно историята се запазва и може лесно да се проследи.

Предложените в Глава 2 алгоритми за актуализиране на данните при ETM са използвани за реализиране на допълнителен (повърхностен) слой към една от най-известните СУБД (MySQL). Това дава възможност за поддържане и обработване на данни с времеви характеристики. Повърхностният слой е написан на PHP. Разработени са класове, позволяващи добавяне, изтриване, модифициране и извличане на времева информация от БД. Осигурено е автоматизирано въвеждане на данните в системата.

За удобство на потребителите, в разработената система е реализиран достъп до ресурсите през стандартен Web – браузер и през минибраузер на мобилен телефон. Това води до повишаване качеството и ефективността на системата.

При Web-базираните информационни системи е особено важно да се гарантира сигурността на информацията. За целта са анализирани и реализирани съвременните средства за защита - достъп до ресурсите чрез парола, цифрови сертификати и криптирани комуникационни сесии.

Генериране на цифров сертификат за осъществяване на оторизиран достъп до ресурсите на ИС

Реализиран и конфигуриран е Web - сървър с поддържане на https протокол. За целта е генериран цифров сертификат чрез програмата OpenSSL [55], който се използва от Web – сървъра. В резултат са създадени файлове с генерирания цифров сертификат (server.crt) и частния ключ към него (server.key). Файлът server.crt съдържа сертификата (в случая самоподписан сертификат) на сървъра, който трябва да бъде предоставен на потребителите. Във файла server.key се намира частният ключ, необходим в процеса на декриптиране на информацията от сървъра. Използвайки споменатите по-горе два файла, е конфигуриран виртуален хост, позволяващ достъпът до информационната система да се осъществява чрез https протокол (<https://piis.uctm.edu>).

Генерирани са цифрови сертификати за осъществяване на оторизиран достъп до ресурсите на ИС.

Създаден е собствен главен /root/ сертификат (ca.crt, базиращ се на X.509 стандарт [113]) на катедра “ПИИС”, явяваща се като сертифицираща организация (СА) [114]. Публичният ключ в този сертификат е използван за верифициране на подписа на катедрата, а частният ключ (ca.key) се използва за подписване на данните.

Web-сървърът е конфигуриран, така че да изисква подписан от него сертификат за достъп до определен ресурс (<https://piis.uctm.edu/security/>). С помощта на openssl е създаден частен RSA [112] ключ с големина 1024 бита и е попълнена заявка за сертификат. В резултат на това са генерирани файловете client.key и client.csr, съдържащи съответно частния ключ и заявката за сертификат. Заявката е подписана от СА на сървъра. В резултат на това се генерира файлът client.crt, съдържащ подписания от сървъра клиентски сертификат.

На този етап, чрез стандарта PKCS#12 [115], използвайки подписания от сървъра клиентски сертификат (client.crt) и частния ключ към него (client.key), е

генериран сертификат (client.p12). Сертификатът е защитен с парола и се предоставя на клиента заедно със сертификата на подписващата СА.

Реализирани са два варианта за осъществяване на оторизиран достъп до ресурсите на ИС чрез използване цифров сертификат:

- достъп до страницата за включване на потребителя чрез цифров сертификат;
- директен достъп чрез записаната в сертификата информация за конкретен потребител.

Web-базирана информационна система за експресно публикуване на кратки текстови съобщения в Интернет

Разработен е и втори пример с предложени ЕТМ, реализиращ информационна система за експресно публикуване на кратки актуални текстови съобщения в Интернет. Системата дава оторизиран достъп за публикуване. Информацията е достъпна за всички. В този случай необходимостта от използване на ЕТМ е свързана с автоматизираното администриране на системата и автоматизираното показване само на актуалните (валидните за даден период) съобщения. Същевременно в БД се съхранява историята на публикуваните съобщения.

Достъпът до ресурсите (четенето и публикуването на съобщения) може да става както от стандартен Web-браузер, така и от браузера на мобилен телефон (GSM, PDA). Тази допълнителна възможност повишава ефективността на системата.

Реализирането на системата се основава на използването на LAMP (Linux, Apache, PHP, MySQL) технология. Системата включва модули за генериране на начална страница, за прочитане на съобщение, за включване в системата, за разпознаване вида на заявката, подадена от Web-браузер или от мобилен телефон, за генериране на изображение, за анимиране.

Разработеният интерфейс се състои от 2 части - публичен интерфейс, предназначен за четене на информацията без ограничения на потребителите и частен интерфейс, предназначен за потребителите, имащи права за публикуване.

Успоредно с разработения интерфейс за стандартен Web-браузер е създаден и интерфейс за минибраузер на мобилен телефон /GSM/. Така реализираният интерфейс е напълно съобразен с възможностите на мобилните устройства. Отчетен е малкият

размер на екрана и отчасти ограничената функционалност при някои от моделите на мобилните апарати.

Характерно за разработената система е това, че достъпът до информацията става от един и същ адрес, както при стандартния, така и при мобилния браузер. Системата автоматично разпознава устройството, от което е подадена заявката.

Заключение

В дисертационния труд е направен обстоен анализ на предложените до този момент времеви модели на бази от данни. Разгледани са основните концепции и подходи, използвани при моделирането на времевите характеристики на данните, както и подходите за тяхното реализиране. Показани са проблемите, свързани с моделирането на времето в базите от данни. Предложен е ефективен времеви модел (ETM), отчитащ реалния период от време, през който фактите от моделираната предметна област са едновременно валидни и записани в базата от данни. Предложени са алгоритми за актуализиране на данните. Предложеният ефективен времеви модел е приложен при реализирането на Web-базирана информационна система за осигуряване на учебния процес във ВУЗ. Системата е защитена от неправомерен достъп чрез съвременните средства за защита. Реализиран е WAP достъп до ресурсите.

Научно-приложни приноси

1. Предложен е нов времеви модел на БД, наречен ефективен темпорален модел (ETM). Предложено е ефективно време (ET) за маркиране на данните. Това е времето, през което данните от моделираната предметна област са едновременно валидни и записани в базата от данни. Моделът е общовалиден при маркиране на данните в кортежа с ефективно време.
2. Описана е семантиката на всяка една от трите форми за актуализиране на данните в предложения модел ETM - добавяне, изтриване и модифициране на данни в базата. Дефинирани са функции за работа с ефективните времеви интервали.

3. Разработени са алгоритми за актуализиране на времевата информация в БД. Алгоритмите са реализирани с използването на стандартен SQL. Това позволява приложимостта им при различни СУБД.

Приложни приноси

4. Предложените алгоритми за актуализиране на данните при ЕТМ са използвани за реализиране на допълнителен (повърхностен) слой към СУБД MySQL. Това дава възможност за поддържане и обработване на данни, маркирани с ефективно време.
5. ЕТМ е приложен при реализиране на Web-базирана система за информационно осигуряване на ВУЗ. Реализиран е WAP достъп до ресурсите на системата. Реализирани са съвременните средства за защита на данните в информационната система.
6. Предложеният ЕТМ е използван и за реализиране на Web-базирана информационна система за експресно публикуване на кратки текстови съобщения в Интернет през стандартен браузер и чрез мини-браузер на GSM.

Предложеният ЕТМ и Web – базираната ИС са разработени във връзка с изпълнението на договор № ДО 02-287/2008 на тема “Съвременни Web-базирани информационни системи и бази от данни с времеви характеристики“, сключен с Министерството на образованието и науката на Република България след спечелване на конкурс „Подпомагане развитието на научния потенциал във висшите училища“.

Разработката за WAP достъп до ресурсите на ИС [102] беше одобрена и представена на 6-то Научно-образователно Експо ”Българските образователни и изследователски институции в основата на иновативните ИКТ проекти” в рамките на международния технически панаир “Есен 2007” (24 – 29 септември 2007 г., палата 8, етаж 2, PC World Софтуерна изложба).

Разработената Web-базирана информационна система за експресно публикуване на кратки актуални текстови съобщения в Интернет, реализирана чрез използване на

ЕТМ, е докладвана на Осма национална научно-практическа конференция на младите учени в България, май, 2010 г., където беше наградена с Първа награда.

Научни публикации по дисертационния труд

1. Д. Пилев, А. Георгиева, Web – базирана информационна система с времева база от данни, Юбилейна научна конференция 40 години катедра АП, София, 18-19 април 2011, ISBN 978-954-465-043-8
2. D. Pilev, A. Georgieva, Temporal Based Mobile Information System, Proceedings of the International Conference Automatics and Informatics'10, Bulgaria, Sofia, October 2010, ISSN:1313-1850
3. Д. Пилев, Мобилна информационна система за експресно публикуване на кратки текстови съобщения, Сборник доклади от “Осма национална младежка научно-практическа сесия”, София, 11-12 май 2010, ISSN:1314-0698
4. D. Pilev, A. Georgieva, Comparative Analysis and Implementation of the Methods for Web Application Digital Certificate Protection, Proceedings of the International Conference Automatics and Informatics'09, Bulgaria, Sofia, October 2009, ISBN-1313-1850
5. D. Pilev, A. Georgieva, Web Application Security-Analysis and Implementation of the Methods for Crypted Data Sessions and Data Transfer, Proceedings of the International Conference Automatics and Informatics'08, Bulgaria, Sofia, October 2008, ISBN-1313-1850
6. D. Pilev, A. Georgieva, Implementation of WAP Access to the Information System for Ensuring of the Education Process in UCTM – Sofia, Proceedings of the International Conference Automatics and Informatics'07, Bulgaria, Sofia, October 3-6 2007, ISBN-1313-1850
7. D. Pilev, A. Georgieva, Information Provisioning Web Application for the Education Process in UCTM, Sofia, Proceedings of the International Conference Automatics and Informatics'06, Bulgaria, Sofia, October, 2006, ISBN-10:954-9641-48-1, ISBN-13:978-954-9641-48-6

8. А. Георгиева, Д. Пилев, Динамичен WEB-сайт за получаване на информация относно резултати от проведени изпити и брой събрани кредити на студентите от ХТМУ-София, в сб. доклади от Втора национална конференция по електронно обучение във висшето образование, Китев, 2006 г, ISBN-10:954-07-2413-9, ISBN-13:978-954-07-2413-3

Използвана литература

1. Snodgrass, R., The Temporal Query Language TQuel. ACM Transactions on Database Systems, 1987: p. 247-298.
2. J. Clifford, C.E.D., T. Isakowitz, C. .S. Jensen, and R. T. Snodgrass, On the Semantics of 'Now' in Databases. ACM Transactions on Database Systems, 1997: p. 171-214.
3. Y. Wu, S.J., and X. S. Wang., Temporal Database Bibliography Update. p. 338–366.
4. (eds.), C.S.J.a.C.E.D., A Consensus Glossary of TemporalDatabase Concepts. February 1998 Version.
5. (eds.), J.C.a.A.T., J. Clifford and A. Tuzhilin (eds.). Workshops in Computing Series. Springer-Verlag, 1995.
6. O. Etzion, S.J., and S. Sripada (eds.). Temporal Databases: Research and Practice. LNCS 1399, Springer-Verlag, 1998.
7. Patrick, J.F.R.a.J.D., Temporal Semantics in Information Systems-a Survey. Information Systems, 1992.
8. (editor)., J.M., SQL/Temporal. July 1996.
9. Jensen, C.S., Temporal Database Management. 2000.
10. Kumar, A., Temporal access methods for network management and other environments. 1998.
11. Kharti, V., Bridging the spatio-temporal semantic Gap: A theoritical framework, evaluation and a prototipe system. 2002.
12. STEINER, A., A Generalisation Approach to Temporal Data Models and their Implementations. PhD Thesis,SWISS FEDERAL INSTITUTE OF TECHNOLOGY Z URICH, 1998: p. 15-51.

13. J. Bair, M.H.B., C. S. Jensen, and R. T. Snodgrass, Notions of Upward Compatibility of Temporal Query Languages. *Wirtschaftsinformatik*, February 1997: p. 25–34.
14. Celko, J., Joe Celko's Data and Databases: Concepts in Practice. Morgan Kaufman Publishers, 1999.
15. Loizou, M.L.a.G., A Guided Tour of Relational Databases and Beyond. Springer-Verlag, 1999.
16. C. Zaniolo, S.C., C. Faloutsos, R. T. Snodgrass, V. S. Subrahmanian, and R. Zicari, Advanced Database Systems. Morgan Kaufmann Publishers, 1997.
17. (Editors), C.S.J.a.R.T.S., Temporal Database Entries for the Springer Encyclopedia of Database Systems. A TIMECENTER Technical Report, May 22, 2008.
18. Özsu, M.T.L., Ling (Eds.) Encyclopedia of Database Systems. September 2009.
19. C. j. Date, H.D., Nikos A. Lorentzos, Temporal data and the relational model. 2003.
20. SNODGRASS, R.T., Developing Time-Oriented Database Applications in SQL. Morgan Kaufmann Publishers, 2000.
21. Тужаров, Х., Базы данни.; Available from:
<http://www.tuj.asenevtsi.com/DB2007/index.htm> (посетен през март,2008).
22. Navathe, R.E.a.S., Fundamentals of Database Systems. Benjamin/Cummings Publishing Company, 1994.
23. Saake, A.H.a.G., Datenbanken: Konzepte und Sprachen. International Thomson Publishing, 1995.
24. MySQL, C.-. ;Available from: <http://www.mysql.com/> (посетен май март,2011).
25. Oracle, C.-. ;Available from: <http://www.oracle.com/> (посетен през май,2011).
26. Codd, E.F., A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM*, 1970: p. 377-378.
27. Russell, D.G., Relational Algebra Napier University, Edinburgh.
28. Padron-McCarthy, T., Lecture Notes: Relational Algebra.; Available from:
<http://www.databasteknik.se/webbkursen/relalg-lecture/index.html> (посетен през януари,2010).
29. Darwen, C.J.D.a.H., A guide to THE SQL STANDARD. Addison-Wesley Publishing Company, 1993.
30. Simon, J.M.a.A.R., Understanding the new SQL: A Complete Guide. Morgan Kaufmann Publishers, 1993.
31. Snodgrass, R., The TSQL2 Temporal Query Language. Kluwer Academic Publishers,101 Philip Drive, Assinippi Park, Norwell, Massachusetts 02061, 1995.

32. Bohlen, M., Temporal Database System Implementations. SIGMOD RECORD, 1995: p. 53-60.
33. Hawking, S.W., A Brief History of Time. Bantam Books, New York, 1988.
34. Vaishnav, S.K.G.a.J.H., A Query Language for a Homogeneous Temporal Database. In Proceedings of the International Conference on Principles of Database Systems, 1985: p. 51-56.
35. Yeung, S.K.G.a.C., Inadequacy of Interval Timestamps in Temporal Databases. Information Systems, 1991: p. 1-22.
36. Tansel, J.C.a.A.U., On an Algebra for Historical Relational Databases: Two Views. SIGMOD RECORD, 1985: p. 247-265.
37. A. Tansel, J.C., S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass, Temporal Databases: Theory, Design, and Implementation. Benjamin/Cummings Publishing Company, 1993.
38. Jensen, C., A Consensus Glossary of Temporal Database Concepts. Technical Report R 93-2035, Aalborg University, 1993.
39. Gadia, S.K., A Homogeneous Relational Model and Query Languages for Temporal Databases. ACM Transactions on Database Systems, 1988: p. 418-448.
40. Gadia, S.K., Toward a Multihomogeneous Model for a Temporal Database. In Proceedings of the International Conference on Data Engineering, 1986: p. 390-397.
41. Allen, J.F., Maintaining Knowledge about Temporal Intervals. Communications of the ACM, 1983: p. 832-843.
42. R. Snodgrass, C.E.D., C.S. Jensen, N. Kline, M.D. Soo, L. So, and J. Whelan, The MultiCal System. October 1993.
43. Toman., J.C.a., Temporal Logic in Information Systems. In Logics for Databases and Information Systems, Kluwer Academic Publishers, 1998: p. 31-70.
44. Snodgrass, C.E.D.a.R.T., Timestamp Semantics and Representation. Information Systems, 1993: p. 143-166.
45. Snodgrass, C.E.D.a.R.T., Efficient Timestamp Input and Output. Software—Practice and Experience, 1994: p. 89-109.
46. Bohlen, C.S.J.C.D.e.M., The Consensus Glossary of Temporal Database Concepts. 1998.
47. Ahn, R.S.a.I., A Taxonomy of Time in Databases. Proceedings of the ACM SIGMOD International Conference on Management of Data, 1985: p. 236-246.

48. Lorentzos, N., The Interval-extended Relational Model and its Application to Valid time Databases. *Temporal Databases: Theory, Design, and Implementation*, Benjamin/Cummings Publishing Company, 1993: p. 67-91.
49. Ahmed, S.B.N.a.R., TSQL : A Language Interface for History Databases. *Proceedings of the Conference on Temporal Aspects in Information Systems*, 1988: p. 113-128.
50. Yeung, S.K.G.a.C., A Generalized Model for a Relational Temporal Database. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1988: p. 251-259.
51. A. Tansel, E.A., and G. Ozsoyoglu, Time-By-Example Query Language for Historical Databases. *IEEE Transactions on Software Engineering*, 1989: p. 464-478.
52. Garnett, A.T.a.L., Nested Historical Relations. *SIGMOD RECORD*, 1989: p. 283-293.
53. C. S. Jensen, L.M., and N. Roussopoulos, Incremental Implementation Model for Relational Databases with Transaction Time. *IEEE Transactions on Knowledge and Data Engineering*, 1991: p. 461-473.
54. M. R. Stonebraker, L.R., and M. Hirohama, The Implementation of POSTGRES. *IEEE Transactions on Knowledge and Data Engineering*, 1990: p. 125-142.
55. Ahn, R.S.a.I., Temporal Databases. *IEEE Computer*, 1986: p. 35-42.
56. R. T. Snodgrass, M.H.B.o., C. S. Jensen, and A. Steiner, Adding Valid Time to SQL/Temporal. *SQL/Temporal Change Proposal. ANSI X3H2-96-501r2, ISO/IEC JTC1/SC21/WG3 DBL MAD-146r2*, 1996.
57. Christian S. Jensen, M.D.S., and Richard T. Snodgrass, *Unifying Temporal Data Models*. 2000.
58. Ariav, G., A Temporally Oriented Data Model. *ACM Transactions on Database Systems*, 1986(499-527).
59. Johnson, N.L.a.R.G., TRA: A Model for a Temporal Relational Algebra. In *Proceedings of the Conference on Temporal Aspects in Information Systems*, 1988: p. 99-112.
60. Sarda, N., Algebra and Query Language for a Historical Data Model. *The Computer Journa*, 1990: p. 11-18.
61. E.A., and G. Ozsoyoglu, Adding Time Dimension to Relational Model and Extending Relational Algebra. *Information Systems*, 1986: p. 343-355.

62. Croker, J.C.a.A., The Historical Relational Data Model (HRDM) and Algebra Based on Lifespans. In Proceedings of the International Conference on Data Engineering. IEEE Computer Society Press, 1987: p. 528-537.
63. Garnett, A.T.a.L., Nested Historical Relations. SIGMOD RECORD, 1989: p. 284-293.
64. Warren, J.C.a.D.S., Formal Semantics for Time in Databases. ACM Transactions on Database Systems, 1983: p. 214-254.
65. Wu, G.T.J., SERQL: An ER Query Language Supporting Temporal Data Retrieval. In Proceedings of the 10th International Phoenix Conference on Computers and Communications, 1991: p. 272-279.
66. Nair, S.K.G.a.S.S., Temporal Databases: A Prelude to Parametric Data. In A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass, editors, Temporal Databases: Theory, Design, and Implementation, Benjamin/Cummings Publishing Company, 1993: p. 28-66.
67. Snodgrass, L.E.M.a.R.T., Evaluation of Relational Algebras Incorporating the Time Dimension in Databases. ACM Computing Surveys, 1991: p. 501-543.
68. Chomicki, J., Temporal Query Languages: A Survey. Proceedings of the First International Conference on Temporal Logic, 1994: p. 506-534.
69. Ahmed, S.N.a.R., Temporal Extensions to the Relational Model and SQL. In A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass, editors, Temporal Databases: Theory, Design, and Implementation. Benjamin/Cummings Publishing Company, 1993: p. 92-109.
70. Sarda, N., HSQL: A Historical Query Language. In A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass, editors, Temporal Databases: Theory, Design, and Implementation. Benjamin/Cummings Publishing Company, 1993: p. 110-138.
71. Bohlen, M., Managing Temporal Knowledge in Deductive Databases. PhD thesis, Departement Informatik, ETH Zurich, 1994.
72. Soo, M.H.B.o.a.R.T.S.a.M.D., Coalescing in Temporal Databases. A TIMECENTER Technical Report, 1997.
73. Clifford, J., A Model for Historical Databases. In Proceedings of Workshop on Logical Bases for Data Bases, 1982.
74. S. Jones, P.M., and R. Stamper, LEGOL 2.0: A Relational Specification Language for Complex Rules. Information Systems, 1979: p. 293-305.

75. Ahmed, S.B.N.a.R., A Temporal Relational Model and Query Language. Information Sciences, 1989: p. 147-175.
76. Sarda, N., Extensions to SQL for Historical Databases. IEEE Transactions on Knowledge and Data Engineering, 1990: p. 220-230.
77. Snodgrass, R., The Temporal Query Language TQuel. In Proceedings of the International Conference on Principles of Database Systems, 1984: p. 204-212.
78. Snodgrass, R., An Overview of TQuel. In A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass, editors, Temporal Databases: Theory, Design, and Implementation. Benjamin/Cummings Publishing Company, 1993: p. 141-182.
79. G. D. Held, M.S., and E. Wong, INGRES - A Relational Database Management System. In Proceedings of the AFIPS National Computer Conference, 1975: p. 409-416.
80. Tansel, A., A Generalized Relational Framework for Modeling Temporal Data. In A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass, editors, Temporal Databases: Theory, Design, and Implementation. Benjamin/Cummings Publishing Company, 1993: p. 183-201.
81. Yong Tang, L.L., Rushou Huang, Yang Yu, Bitemporal Extensions to Non-temporal RDBMS in Distributed Environment. The 8th International Conference on Computer Supported Cooperative Work in Design Proceedings, 2003.
82. Maintaining and querying time data in PostgreSQL. 2007; Available from: <http://pgfoundry.org/projects/temporal/> (посетен през май, 2011).
83. Database, O., Workspace Manager Developer's Guide. September 2010.
84. Database, O., Oracle Database 11g Workspace Manager Overview. September 2009.
85. Corporation, T., Teradata Temporal.
86. Corporation, T., Teradata Temporal Capability. 2010.
87. Richard T. Snodgrass, M.H.B., Christian S. Jensen and Andreas Steiner, Adding Transaction Time to SQL/Temporal. change proposal, ANSI X3H2- 96-502r2, ISO/IEC JTC1/SC21/WG3 DBL MAD-147r2, November 1996.
88. Snodgrass, R.T., A Case Study of Temporal Data. Teradata Corporation, 2010.
89. R, J.Y.P.a.J., A Heterogeneous Temporal Nested Relational Data Model. InTeRel Report No. 5, Department of Computer Science, Birkbeck College, University of London, 1992.

90. Snodgrass, G.O.a.R.T., Temporal and Real-Time Databases: A Survey. IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, AUGUST 1995. VOL. 7(NO. 4).
91. Kristian Torp, C.J., Richard Snodgrass, Effective timestamping in databases. The VLDB Journal, 1999: p. 267-288.
92. Soo, M.H.B.o.a.R.T.S.a.M.D., Coalescing in Temporal Databases. Technical Report R-96-2026 Aalborg University September 1996. Proceedings of the 22nd International Conference on Very Latge Data Bases.
93. Christian S. Jensen, M.D.S., and Richard T. Snodgrass, Unifying Temporal Data Models. Temporal Database Management ,dr.techn. thesis by Christian S. Jensen, 2000.
94. Snodgrass, C.E.D.a.R.T., Timestamp Semantics and Representation. Information Systems, 1993. 18(3): p. 143-166.
95. А. Георгиева, Д.П., Съвременни Web-базирани информационни системи и бази данни с времеви характеристики. Сключен с МОН договор № ДО 02-287/2008, декември 2008.
96. D.Pilev, A.G., Information Provisioning Web application for the Education Process in UCTM Sofia. Proceedings of the International Conference Automatics and Informatics'06, 2006
97. Ернандес, М.Х., Проектиране на бази от данни. 2004.
98. Madjarov, I. WAP - Мобилни комуникации и Интернет.; Available from: <http://mulmedp.tu-sofia.bg/bookhtml/www/www9.htm> (посетен през септември,2009).
99. W3Schools. WAP / WML Tutorial. Available from: <http://www.w3schools.com/wap/> (посетен през май,2011).
100. Георгиев, В. Лекционен курс WAP/WML. Available from: <http://is.fmi.uni-sofia.bg/t3/prs/tmp-wap/site/index.htm> (посетен през май,2009).
101. WAP 2.0 Technical White Paper.; Available from: http://www.wapforum.org/what/WAPWhite_Paper1.pdf (посетен през август,2009).
102. D.Pilev, A.G., Implementation of WAP Access to the Information System for Ensuring of the Education Process in UCTM - Sofia. Proceedings of the International Conference Automatics and Informatics'07, Bulgaria, Sofia, October 3 - 6 2007.
103. А. Георгиева, Д.П., Web базирана система за информационно осигуряване на учебния процес в ХТМУ - София с реализация на WAP - достъп. Образователно

- ЕКСПО 2007 в рамките на Международния технически панаир „Есен 2007”,
Пловдив, 24 - 29 септември 2007.
104. Тужаров, Х. Информационна сигурност; Available from:
<http://www.tuj.asenevtsi.com/US2007/index.htm> (посетен през февруари,2011).
 105. Каео, М., Проектиране на мрежова сигурност. 2006.
 106. Wikipedia. Denial-of-service attack.; Available from:
http://en.wikipedia.org/wiki/Denial-of-service_attack (посетен през ноември,2008).
 107. Wikipedia. Domain Name System;Available from:
http://en.wikipedia.org/wiki/Domain_name_system (посетен през ноември,2008).
 108. Kessler, G.C. An Overview of Cryptography; Available from:
<http://www.garykessler.net/library/crypto.html> (посетен през ноември,2008).
 109. Koren, D. SSL Tutorial.; Available from:
<http://www2.rad.com/networks/2001/ssl/index.htm> (посетен през ноември,2008).
 110. Jalal Feghhi, P.W., Digital Certificates: Applied Internet Security.
 111. Laboratories, R. PKCS #1 v2.1: RSA Cryptography Standard.; Available from:
<ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-1/pkcs-1v2-1.pdf> (посетен през
август,2009).
 112. Internet X.509 Public Key Infrastructure Certificate and CRL Profile.; Available
from: <http://www.ietf.org/rfc/rfc2459.txt> (посетен през август,2009).
 113. Wikipedia. Certificate authority; Available from:
http://en.wikipedia.org/wiki/Certificate_authority (посетен през ноември,2008).
 114. Laboratories, R. PKCS 12 v1.0: Personal Information Exchange Syntax. ; Available
from: <ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-12/pkcs-12v1.pdf> (посетен през
ноември,2008).
 115. w3schools. AJAX. ; Available from: <http://www.w3schools.com/ajax> (посетен през
март,2010).
 116. Project, j. JQuery. ; Available from: <http://jquery.com> (посетен през април,2010).
 117. Пилев, Д., Мобилна информационна система за експресно публикуване на
кратки текстови съобщения, в сб. доклади от Осма национална младежка
научно-практическа сесия, София,11-12 май 2010.