

Ю.В. Ришковець, П.І. Жежнич  
Національний університет “Львівська політехніка”,  
кафедра інформаційних систем та мереж

## МЕТОДИ СТРУКТУРНОГО ПОДАННЯ ЧАСУ В БАЗАХ ДАНИХ

© Ришковець Ю.В., Жежнич П.І., 2008

**Розглянуто часові проблеми, моделі часу та їхні інтерпретації. Проаналізовано представлення часу в базах даних як чітких, так і невизначених інтервалів, також введено узагальнену модель представлення даних типу дата/час.**

**In this article are considered temporal problems, models of time and their interpretation. Analysed presentation of time in databases as crisp and indefinite intervals, also entered generalized model of data type presentations.**

### Постановка проблеми

У сучасному інформаційному суспільстві час є важливою і невід’ємною компонентою сприйняття світу. Де б ми не знаходились, що б ми не робили, ми завжди опираємось на час. Спостерігаючи за часом, ми постійно намагаємось його зафіксувати, прив’язавши час до певної події чи їх сукупності. Тобто, використовуючи час, можна зафіксувати момент чи період виникнення тієї або іншої події відносно інших подій. Під вимірюванням часу розуміють введення часової шкали, користуючись якою можна було б співвідносити ці події. Під такою шкалою співвідношення подій розуміють календар.

Календар (від лат. calendae або kalendae, “календи” – назва першого дня місяця у стародавніх римлян), спосіб ділення року на зручні періодичні інтервали часу. Основними завданнями календаря є фіксація дат і вимірювання інтервалів часу. Календар дає змогу реєструвати дати природних явищ, історичні та суспільні події в їх хронологічній послідовності. Функції календаря використовуються в суспільній сфері і в побуті, де процентні платежі, заробітна плата та інші ділові відносини засновані на певних інтервалах часу. Багато статистичних і наукових досліджень також використовують часові інтервали.

Існує багато форм представлення часової інформації – археологічна, палеонтологічна, астрономічна, історична і т.д., але жодна з них не може бути представлена в часових базах даних за допомогою типу даних дата/час. Оскільки в часових базах даних відсутні специфічні типи представлення нестандартного часу в явному вигляді, то основна проблема полягає у представленні цих форм у часових базах даних [1]. У часових базах даних фактично лише деякою мірою використовується історична модель представлення часу, інші форми в явному вигляді відсутні.

У цій роботі ми запропонуємо огляд проблем, пов’язаних з часом, та запропоновані можливі шляхи їх вирішення, розглянемо представлення моделей часу та їх підструктур, введемо специфічні часові типи даних та узагальнену модель представлення даних типу дата/час.

### Зв’язок висвітленої проблеми із науковими завданнями

**Метою** статті є дослідження моделей часу.

**Наукова новизна** статті полягає у введенні специфічних часових типів даних.

**Практична цінність** статті визначається можливістю представлення у базах даних специфічних часових типів даних.

### Аналіз останніх досліджень

Розглянемо, як представляється час в сучасних операційних системах (ОС).

Час в UNIX зберігається у вигляді кількості секунд. Моментом початку відліку вважається опівніч (Universal Coordinated Time, UTC) з 31 грудня 1969 року на 1 січня 1970. Час в UNIX

узгоджується з UTC, зокрема, при оголошенні високосних секунд UTC відповідні номери секунд повторюються. Такий спосіб представлення часу дуже зручно використовувати при порівнянні дат (з точністю до секунди), а також для зберігання дат, причому за необхідності їх можна перетворити у будь-який зручний для читання формат. Недоліки в продуктивності можуть виявитися при дуже частому зверненні до окремих елементів дати, наприклад, до номеру місяця, номеру дня, номеру години, номеру хвилини і т.п., але в більшості випадків ефективніше зберігати час у вигляді однієї величини, а не набору полів.

Збій Unix-систем – дуже серйозна небезпека. Під керуванням Unix працює програмне забезпечення Internet і більшість критично важливих військових, фінансових систем та систем управління.

Проблеми, пов'язані з часовим форматом представлення даних, полягають у датах і особливостях мови C. Річ у тому, що цією мовою написані операційні системи типу UNIX, а також дуже багато програм під Microsoft Windows. Мова C обробляє час і дату за допомогою часу за Гринвічем (Greenwich Mean Time, GMT) у вигляді кількості секунд, починаючи з 1 січня 1970. Тип даних, який використовується в C для зберігання кількості секунд, не дає змоги працювати з числами, що перевищують значення 2 147 483 647.

Проблема полягає ще й у тому, що вікна ОС для коригування дати і часу також себе не виправдовують, коли використовуються архіви і програми, написані раніше 1970 року. Наприклад, Windows підтримує дати аж до 2099 року, а якщо спробувати встановити 2100 рік, то ОС встановить 1980 рік.

Більшість існуючого програмного забезпечення виконує певні операції зі значеннями дати і часу. Працювати з датами доволі складно; крім того, що доводиться оперувати строго форматованими даними, існує багато правил визначення їхніх допустимих значень і проведення коректних обчислень (потрібно врахувати високосні роки, національні свята і вихідні, діапазони дат і т.п.).

При аналізі часових проблем не можна не згадати про фактично першу часову проблему сучасного інформаційного суспільства – проблему 2000 року [1]. Проблема виникла тоді, коли в атрибуті типу дата/час, для збереження року було відведено тільки два розряди. Алгоритми, що працюють з датами в такому форматі, робили помилки, тому що вони не сприймали дати з 2000 р. як числа, значення якого більше, ніж дата в цьому столітті, тобто ті, що починаються з 19, а саме,  $2000 - 1998 = 2$ , проте  $00 - 98 = -98$  (або 98, якщо програма не працювала з від'ємними числами). Наприклад, 1 січня 1991 року в таких програмах представлялося як "01.01.91". Коли настало 1 січня 2000 року при двозначному представленні року, після 99 року настав 00 рік (тобто  $99 + 1 = 00$ ), що інтерпретувалося багатьма програмами як 1900, а це, своєю чергою, могло привести до серйозних збоїв в роботі критичних програм, таких як системи управління технологічними процесами і фінансові програми. Іншою проблемою могло стати те, що 2000 рік – це високосний рік, тоді як 1900 і 2100 – невисокосні, в деяких програмах це не враховувалось, внаслідок чого втрачався день. Для багатьох програм, призначених для планування і прогнозування, "Проблема 2000 року" настала раніше, ніж самого комп'ютера і поточної системної дати комп'ютера.

Метод представлення дат в Unix дає змогу створити часовий діапазон розміром в 68 років, кінець якого припадає на 2038 рік. Від'ємні значення використовуються для представлення дат з 1901 по 1970 рік. Остання дата, яка може бути представлена таким форматом в стандарті POSIX, це 03:14:07, вівторок, 19 січня 2038 року за всесвітнім часом (UTC). Наступний момент часу (19 січня 2038 року в 03:14:08 за UTC) може привести до помилкової інтерпретації цього числа як від'ємного, що подібно до заکیلцювання часу. Оскільки негативне число може бути сприйняте програмами як дата 1 січня 1970 року або 13 грудня 1901 року, залежно від реалізації. У результаті можуть бути проведені помилкові обчислення або отримані некоректні результати.

Проблемою 10000 року називають сукупність передбачуваних проблем, які можуть виникнути при роботі з програмним забезпеченням, яке представляє рік за допомогою лише 4 цифр. Такий підхід може привести до помилок і збоїв при переході від 9999 року до 10000 року. Ця проблема подібна до проблеми 2000 року. Але у деяких випадках обробка дат за межами 10000 року може знадобитися вже зараз – наприклад, в програмах, що оцінюють проекти довгострокового зберігання ядерних відходів.

Істотним доповненням до операційних систем є системи керування базами даних (СКБД), тому що саме вони дають користувачу змогу зберігати дані типу дата/час у додаткових форматах і ширшому діапазоні значень. З СКБД користувач безпосередньо не працює, він працює з прикладною програмою, яка, своєю чергою, працює з СКБД. Прикладна програма є зв'язною ланкою між користувачем та СКБД. Кожна СКБД підтримує декілька різних типів даних типу дата/час.

Розглянемо, які ж типи даних типу дата/час використовуються в сучасних системах керування базами даних.

Система керування базою даних MySQL [1] має декілька типів даних, призначених для зберігання календарних даних:

- **DATE** зберігає дату у форматі рік-місяць-день у діапазоні значень від '1000-01-01' до '9999-12-31';

- **TIME** може використовуватися не тільки для представлення часу дня, який має бути меншим за 24 години, але і для представлення тривалості розроблення чи інтервалу часу між двома подіями, який може бути набагато більшим за 24 години або навіть від'ємний, діапазон можливих значень від '-838:59:59' до '838:59:59'.

- **DATETIME** одночасно містить і дату і час, підтримує значення в діапазоні від '1000-01-01 00:00:00' до '9999-12-31 23:59:59';

- **TIMESTAMP** дає змогу зберігати комбіновані значення дати і часу у форматі `CCYYMMDDhhmmss`, починаючи з "1970-01-01 00:00:00" і закінчуючи невизначеною датою у 2037 році. Цей діапазон прив'язаний до таймера операційної системи Unix, відлік якого починається з 1 січня 1970 року і який є нульовою датою Unix.

- **YEAR** використовується для зберігання даних за роками, причому якщо формат року містить чотири символи, то рік зберігається в діапазоні 1901÷2155, а якщо два – то в діапазоні 1970÷2069.

SQL Server 2008 [1] дозволяє зберігати часову інформацію, використовуючи такі типи даних:

- **TIME** корисний для визначення часу дня, підтримує діапазон значень від 00:00:00.0000000 до 23:59:59.9999999;

- **DATE** використовується для зберігання дат без часової складової, значення можуть належати діапазону значень від 0001-01-01 до 9999-12-31;

- **SMALLDATETIME** – цей тип даних використовується для зберігання даних типу дата і час з точністю до 1 хвилини, діапазон значень від 1900-01-01 до 2079-06-06;

- **DATETIME** – дата і час зберігається в діапазоні від 1753-01-01 до 9999-12-31;

- **DATETIME2** призначений для зберігання точних значень дати і часу в діапазоні від 0001-01-01 00:00:00.0000000 до 9999-12-31 23:59:59.9999999;

- **DATETIMEOFFSET** дає змогу зберігати точне значення типу дата/час із зміщенням у діапазоні від 0001-01-01 00:00:00.0000000 до 9999-12-31 23:59:59.9999999.

СКБД Oracle [1] підтримує такі типи даних типу дата/час:

- **DATE** зберігає інформацію про рік, місяць, день, години, хвилини, секунди.

- **TIMESTAMP** представляється трьома типами даних. Тип **TIMESTAMP** зберігає значення дати і часу без інформації про часовий пояс з точністю до мільярдної частини секунди, тип **TIMESTAMP WITH TIME ZONE** зберігає разом з кожним значенням дати і часу інформацію про часовий пояс, а тип **TIMESTAMP WITH LOCAL TIME ZONE** зберігає значення дати і часу, які, як припускається, відповідають локальному часовому поясу.

- **INTERVAL** дає змогу представляти інтервали часу. Тип даних **INTERVAL DAY TO SECOND** використовується для представлення інтервалу часу у вигляді точної різниці між двома значеннями **DATETIME**, а тип **INTERVAL YEAR TO MONTH** – для представлення інтервалу часу у вигляді різниці між двома значеннями **DATETIME**, де істотними частинами є рік і місяць.

Розглянуті СКБД дають змогу представляти інформацію, пов'язану з часовою компонентою, у вигляді конкретних моментів чи інтервалів часу. Крім часових типів даних, сучасні СКБД підтримують моделювання історичних послідовностей календарних правил – різні часові пояси, зміну літнього і зимового часу і т.д.

### Виділення невирішених частин проблеми

Під час аналізу дати і часу, що використовуються в операційних системах, було виявлено, що всі проблеми, пов'язані з часом, є наслідком використання мови С, а саме, її неспроможності забезпечити всі сучасні вимоги для представлення і використання дати та часу.

Системи керування базами даних мають власні визначені типи даних дата/час, але і вони на сучасному етапі розвитку не спроможні забезпечити необхідний діапазон значень.

Крім проблеми обмеження значень часових типів даних в операційних системах та СКБД існує ще одна важлива проблема – представлення специфічних часових понять, наприклад, “шкільні канікули”, “фінансові роки”, “робочі години” і т.ін. При роботі з історичними системами, зокрема в музейній справі, використовуються такі поняття, як “ера”, “період”, “епоха”, “середні віки”, “на початку століття” та багато інших. Всі вони є невизначеними, тому що представляють інтервали часу як нечіткі множини, а представлення інформації такого виду не підтримується системами керування базами даних.

### Основний матеріал

Час є важливою компонентою в житті людини. Досліди, факти, об'єкти або події – всі вони пов'язані з деяким видом часу. Звичайні бази даних містять лише інформацію про поточний стан у модельованій дійсності. Такий підхід може привести до втрати інформації при її модифікації, тобто можна втратити попередній стан даних внаслідок їх зміни, а збереження хронологічної послідовності дасть змогу повертатися до попередніх станів. Особливо критичною така ситуація є під час роботи з історичними станами.

Кількість областей і програм, в яких присутня інформація, пов'язана з часом або залежна від часу, є дуже великою. Потреба в застосуванні часової компоненти існує у сферах банківської справи, фінансів, страхування, медицини, освіти чи музейної справи. Часто процес визначення часової функціональності не обробляється належним чином через складність керування часовими даними.

Часовий аналіз є дуже важливим, оскільки багато програм для своєї роботи потребують аналізу зміни інформації в часі. Знання, отримані таким чином, можна використовувати для планування чи прогнозування майбутніх подій або для відтворення подій минулого.

У людській уяві час представляється у вигляді прямої лінії, що тягнеться від нескінченного минулого часу до нескінченного майбутнього часу. Для баз даних контекст часто такий самий, але час природно зв'язується за величиною структури даних.

У часовій логіці запропоновано і введено фактично дві структурні моделі часу – дійсний час та час транзакцій. Дійсний час – це час зміни інформації, тобто момент часу, рідше інтервал часу, в який певний факт був, є чи буде істинним в модельованому світі. А час транзакцій – це час введення інформації до бази даних або час її редагування в базі даних.

На основі аналізу представлень часу можна виділити декілька узагальнюючих форм представлення часу. Отже, часову вісь можна представити такими структурами:

а) лінійний час (рис. 1):



Рис. 1. Представлення лінійного часу

б) розгалужений час (рис. 2):

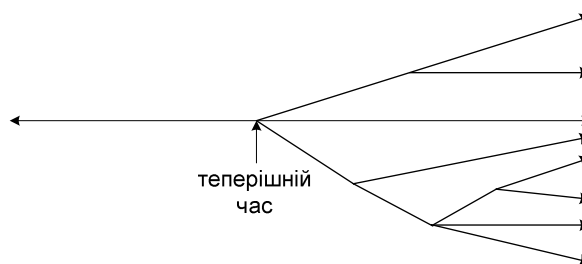


Рис. 2. Представлення розгалуженого часу

Лінійна модель використовується для загальних часових представлень, а коли результат певного процесу має різні альтернативи, наприклад, в плануванні, то тоді застосовується розгалужена модель часу. Існують три різні інтерпретації графіка часу лінійної моделі – дискретна, щільна і безперервна.

З дискретного погляду, час розглядається як лінія часу ізоморфна до цілих чисел. Часова вісь складається з часових періодів деякої фіксованої мінімальної тривалості, що називаються хрононами. Між кожною парою хрононів існує обмежена кількість інших хрононів.

Якщо ж розглядати час як щільність, то лінія часу ізоморфна до раціональних чисел. Між будь-якими двома хрононами існує необмежена кількість моментів.

А якщо час є безперервним, то лінія часу ізоморфна до дійсних чисел. Між кожною парою моментів є нескінченна кількість інших моментів.

Модель розгалуженого часу використовується тоді, коли є декілька можливих варіантів розвитку подій у майбутньому.

Розглядаючи певну підструктуру часу в контексті її періодичності, можна говорити про моделювання періодичності часу (рис. 3). Наприклад, тиждень починається з понеділка і закінчується неділю, він триває сім днів (понеділок, вівторок, серeda, четвер, п'ятниця, субота, неділя), кожний наступний тиждень також триває сім днів: починається у понеділок і закінчується у неділю. Тобто цей процес є періодичним, нескінченним і незворотним, з періодом 7 днів. Про аналогічну періодичність часу можна говорити і для таких підструктур часу, як день, місяць, рік, пори року та ін.



Рис. 3. Періодичність днів тижня

Часовий момент – це часова точка на дійсній часовій осі. Подія – це миттєвий факт. Час виникнення події – це мить (момент), в який подія відбувається в реальному світі.

Часовий інтервал – це час між двома моментами; він є спрямованою тривалістю часу. Тривалість – це кількість часу з відомою довжиною, яка не має специфічного початку чи закінчення моментів часу. Інтервали можна поділити на два види – визначений або чіткий і невизначений.

Час вимірюється за допомогою різних часових аспектів, які мають асоціюватися з фактом (рис. 4).

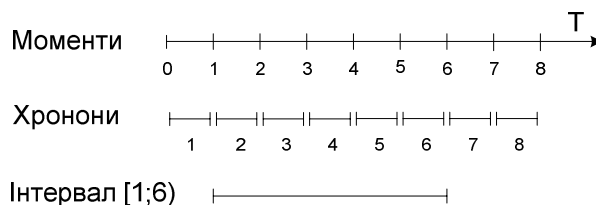


Рис. 4. Зв'язок між моментами, хрононами та інтервалами

З геологічного погляду, часова вісь представляється за допомогою ер (груп) та періодів (систем), як зображено на рис. 5. Кожна ера (їх всього 4) має свою кількість періодів.

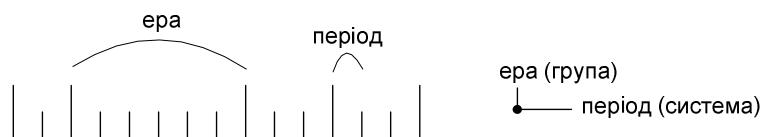


Рис. 5. Представлення часу в геології

В астрології час представляється за допомогою епох, кожній з яких відповідає певний знак зодіаку. Як правило, для представлення астрологічного часу використовується періодична модель даних.

Досліджуючи час з історичного погляду, тобто з погляду історичних періодів, його можна представити у вигляді ієрархії або за допомогою визначених і невизначених множин (рис. 6).

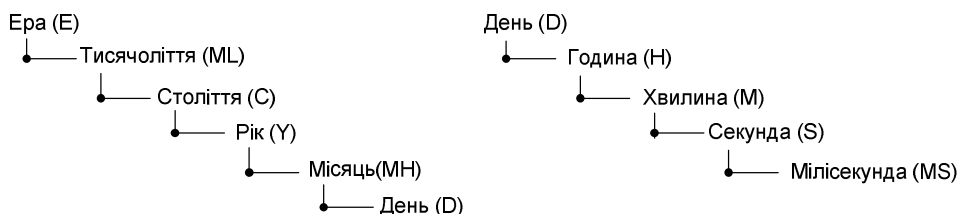


Рис. 6. Представлення історичного часу як ієрархічної структури

Коренем такої структури історичного часу є ера (E), яка містить невизначену кількість тисячоліть (ML), тобто вона є невизначеною множиною. Тому справедливі такі залежності:

$$\begin{aligned}
 E &= \{ML_1, ML_2, ML_3, \dots\} \\
 ML &= \{C_1, C_2, C_3, \dots, C_q\}, \quad q = \overline{1,10} \\
 C &= \{Y_1, Y_2, Y_3, \dots, Y_g\}, \quad g = \overline{1,100} \\
 Y &= \{MH_1, MH_2, MH_3, \dots, MH_p\}, \quad p = \overline{1,12} \\
 MH &= \{D_1, D_2, D_3, \dots, D_v\}, \quad v = \overline{1,31} \\
 D &= \{H_1, H_2, H_3, \dots, H_z\}, \quad z = \overline{1,24} \\
 H &= \{M_1, M_2, M_3, \dots, M_i\}, \quad i = \overline{1,60} \\
 M &= \{S_1, S_2, S_3, \dots, S_j\}, \quad j = \overline{1,60} \\
 S &= \{MS_1, MS_2, MS_3, \dots, MS_k\}, \quad k = \overline{1,1000}
 \end{aligned}$$

Для представлення інтервалів часу також можна використати нечіткі множини.

Специфічні невизначені множини для представлення часових інтервалів можуть використовувати дійсні числа. Невизначені інтервали визначаються, як правило, через їх функції належності. Функція належності відтворює базову множину дійсних чисел між 0 і 1. Це “невизначене значення” означає свого роду ступінь належності до нечіткої множини S. Базова множина для невизначених інтервалів часу є часовою віссю, представленою множиною R дійсних чисел. Дійсні числа дають змогу моделювати безперервний потік часу, який ми помічаємо в нашому житті. Невизначений інтервал часу представляється як невизначена підмножина дійсних чисел.

Чіткі та невизначені інтервали графічно зображаються, як на рис. 7.

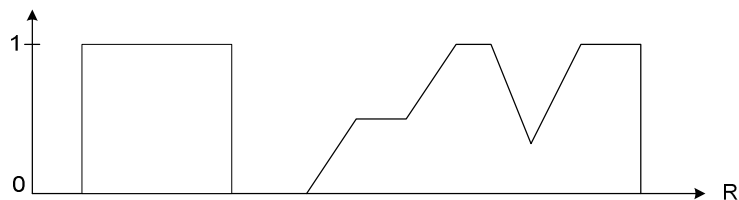


Рис. 7. Чіткий та невизначений інтервали

Крім того, невизначені інтервали можуть бути ще й нескінченними. Наприклад, термін “після цього року” може представлятися невизначеним значенням, яке збільшується від невизначеного значення “0”, яке відповідає даті 01.01.2008, до невизначеного значення “1”, якому відповідає дата 31.12.2008, і залишається в “1” до нескінченності (рис. 8).

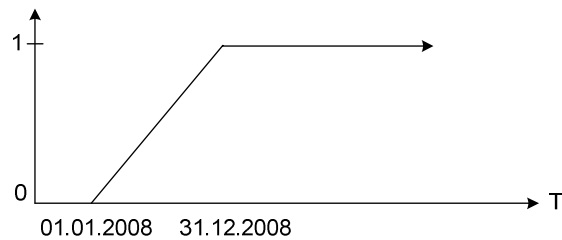


Рис. 8. Невизначений нескінченний інтервал “після цього року”

Як звичайні, так і невизначені інтервали можуть бути відкритими і закритими. Наприклад, розглянемо такий невизначений інтервал. Якщо маємо  $i(a)=0,5$ ,  $i(b)=1$ , а  $i(c)=0,5$  та  $i(d)=0$ , то  $i$  закритий на  $a$  і  $b$  і відкритий на  $c$  і  $d$ . Відкрита сторона невизначеного інтервалу позначається штриховою лінією (рис. 9).

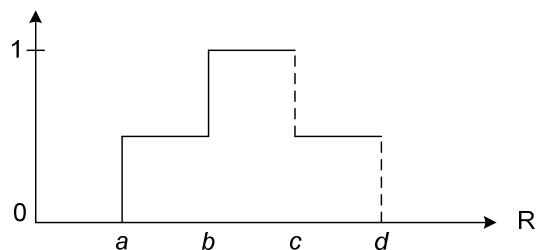


Рис. 9. Напіввідкритий невизначений інтервал

Розглянемо два суміжні інтервали “понеділок цього тижня” і “вівторок цього тижня”. Якщо обидва інтервали представляються як закриті інтервали, тоді опівніч належатиме як понеділку, так і вівторку, але виходячи із здорового глузду, таке неможливо. Тому більш правдиво і реалістично представляти такі інтервали як напіввідкриті інтервали, в яких значення, що лежить на межі двох інтервалів, належить, наприклад, першому з них. З цих міркувань у розглянутому прикладі опівніч належатиме лише понеділку, а не понеділку і вівторку одночасно.

Розглянемо детальніше проблему представлення невизначеного часу в історичних системах, зокрема, в музейній справі. В музейній справі часто оперують специфічними часовими поняттями, як-от, “середні віки” чи “XIV століття”, які по своїй суті є нечіткими визначеннями часу. Проблема полягає в тому, що сьогодні жодна з існуючих СКБД у своєму наборі типів даних не має такого типу дата/час, за допомогою якого можна було б представляти невизначені інтервали часу в базі даних. На жаль, існуючі СКБД містять типи даних типу дата/час лише для зберігання моментів та визначених інтервалів часу.

Одним із найважливіших факторів, які впливають на історичну цінність предмета, є час. Кожний експонат музею характеризується великою множиною атрибутів – назва, автор, час

створення, час побутування, стиль, матеріал і т.д. Частина з них пов'язана із часом безпосередньо, а частина ні. Так, коли говорити про час створення експонату чи час його побутування, тоді розуміємо, що йдеться про безпосереднє представлення часу. А якщо говорити про автора музейного експонату, то розуміємо, що йдеться насамперед про його прізвище, ім'я та по батькові, і вже потім пов'язуємо його з певною історичною епохою чи періодом часу. Час може представлятися у базі даних у вигляді як моментів, так і інтервалів часу, причому інтервали часу можуть бути визначеними і невизначеними.

Загалом час створення експонату представляється певним періодом часу – якщо він чіткий, тоді його початок та кінець є чітко визначені, а якщо він невизначений, то і його представлення в базі даних є нечітким. Тобто в першому випадку період створення експонату є чітко визначеним, наприклад, 01.01.1680-12.06.1685 рр., а в іншому – цей період є нечітким, тобто час представляється за допомогою специфічного значення, як-от, “XVII століття”.

Деякі музейні експонати датуються періодом “нашої ери”, а деякі – “до нашої ери” (рис. 10). Всі СКБД дають змогу представляти час, який належить лише нашій ері. Крім того, всі вони накладають свої обмеження на діапазон значень існуючих типів даних дата/час.

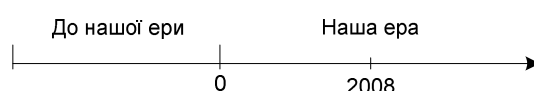


Рис. 10. Представлення лінії часу як послідовності ер

Для представлення в базі даних специфічних часових понять, пов'язаних з історичними експонатами, у вигляді розширення існуючих часових типів даних в базах даних пропонується ввести такі типи даних: “Ера” (E), “Період” (P) і “Точність періоду” (EP). Тоді доцільно буде ввести узагальнену модель представлення даних типу дата/час  $T = \{D, E, P, EP\}$ ,

де  $D$  – конкретний момент часу,

$E = \{ \text{До нашої ери, Наша ера} \}$ ,

$P = \{ \text{День, Тиждень, Місяць, Десятиліття, Століття, Тисячоліття} \}$ ,

$EP = \{ \text{Початок, Середина, Кінець, Половина, Третина, Четвертина, Поч.Сер.Кін.} \}$ .

## Висновки

Розглянуто представлення дати/часу в сучасних операційних системах. У процесі досліджень встановлено, що в операційних системах всі проблеми, пов'язані з часом, є наслідком використання мови С. Подібна ситуація виявлена і в СКБД. Хоч СКБД мають власні визначені типи даних типу дата/час, але і вони на сучасному етапі розвитку не спроможні забезпечити необхідний діапазон потрібних значень. Також вони не забезпечують представлення в базі даних невизначених типів даних типу дата/час.

Під час розгляду моделей часу встановлено, що деякі їх підструктури можна розглядати у контексті періодичності.

Розглянуто і показано взаємозв'язок одиниць представлення часової інформації.

Введено нові специфічні часові типи даних та узагальнену модель представлення даних типу дата/час.

1. Zhezhnych P., Peleschyshyn A. *Time Aspects of Information Systems // Proceedings of the IXth International Conference CADSM, 2007, P.530–533.* 2. Robinson B.A. *The Y2K crisis that never happened.* [http://www.religioustolerance.org/y2k\\_prob.htm](http://www.religioustolerance.org/y2k_prob.htm). 3. Дюбуа П. *MySQL, 3-е издание.: Пер. с англ. – М.: ООО “И.Д.Вильямс”, 2007. – 1168 с. (255–263).* 4. *Using Date and Time Data. SQL Server 2008 Books Online (February 2008).* [http://technet.microsoft.com/en-us/library/ms180878\(SQL.100\).aspx](http://technet.microsoft.com/en-us/library/ms180878(SQL.100).aspx). 5. *Representing Date and Time Data. Oracle® Database Application Developer's Guide – Fundamentals 10g Release 1 (10.1) Part Number B10795-01.* [http://download.oracle.com/docs/cd/B14117\\_01/appdev.101/b10795/adfnst\\_ty.htm#1006833](http://download.oracle.com/docs/cd/B14117_01/appdev.101/b10795/adfnst_ty.htm#1006833).