

де $\{ C_i \}$ - множина логічних умов, які визначають наступні гілки у графі виконання складного оператора S (мал.1), $T(S_i: \bar{X})$ — часова функція, пов'язана з виконанням оператора S_i .

Тип (R_2, E_2) кваліфікаційних функцій, які віддзеркалюють часові функції, визначається таким чином:

обмеження R_2 відсутнє, оскільки приймається довільна множина $\{ C_i \}$;

спосіб обчислення E_2 задається формулою:

$$\tilde{T}(S: \bar{X}) = \sum_{C_i = \text{істина}} \tilde{T}(S_i: \bar{X}),$$

де $\tilde{T}(S_i: \bar{X}) = T(S_i: \bar{X})$, якщо $T(S_i: \bar{X})$ -

арифметичний вираз.

Також пропонуються правила для побудови часових функцій та правила виведення часових функцій для довільних програмних конструкцій. На цій основі будуються алгоритми часового аналізу програмних систем.

Література:

1. *Gabrielian A., McNamee L., Trawick D.* The qualified function approach to analysis of program behavior and performance. //IEEE Transactions on software Engineering. - 1985. - SE-11, №8. -P.758-773.

ТРЬОХРІВНЕВА МОДЕЛЬ ІНФОРМАЦІЙНИХ СИСТЕМ В АРХІТЕКТУРІ КЛІЄНТ-СЕРВЕР

С. Спасітелева, І. Чичкань
(кафедра інформатики НаУКМА)

В основу трьохрівневої моделі покладено принцип, згідно з яким декомпозиція великих складних задач не тільки спрощує, а і прискорює їх виконання. В однорівневій моделі все, що відноситься до додатків, за винятком використовуваних ними зовнішніх даних, розташовано на локальній робочій станції.

У найбільш популярній двохрівневій моделі логіка обробки даних залишається на комп'ютері користувача, а операції з базою даних (БД) здійснюються на тому ж сервері, на якому ця БД зберігається.

У трьохрівневій або багаторівневій моделі додатки поділяються на підсистеми (сервіси), кожна з яких функціонує на окремому комп'ютері. Як правило, існують три групи подібних підсистем: для обслуговування користувачів (розташовані на локальних робочих станціях), для обслуговування додатків (виконуються на спеціалізованих серверах додатків) і для обслуговування даних (розташовані на серверах БД). Ці служби можуть працювати як на індивідуальних серверах, так і на мініЕОМ та мейнфреймах. Підвищення продуктивності в межах такої моделі відбувається за рахунок того, що сервери БД і додатків здатні одночасно обробляти запити одразу від декількох додатків.

У доповіді пропонуються методи та прийоми проектування трьохрівневої моделі для інформаційних систем вищих навчальних закладів.

При проектуванні інформаційних систем необхідно врахувати такі фактори.

1. Система забезпечує сучасний доступ до таких ресурсів, як БД та прикладні програми обробки ділової інформації в масштабах не одного із підрозділів, а установи в цілому. При цьому число паралельних під'єднань до БД перевищує 50, а число одночасно працюючих користувачів — більше 1000. Дані повинні бути постійно доступними всім виконуваним додаткам в масштабі установи. Визначати структуру зберігання даних в термінах суті та зв'язків між ними варто у відповідності до потреб підприємства в цілому, а не тільки його окремих підрозділів. Жоден додаток не може мати власних, тобто використовуваних лише ним даних.

2. Система обробки має бути пов'язаною з особливостями функціонування установи і повинна виконуватися не на клієнтських комп'ютерах, а на серверах додатків. При проектуванні рівня прикладної обробки необхідно враховувати структуру установи, виконувани нею найважливіші функції і організацію інформаційних потоків між підрозділами. Правильна декомпозиція додатку є ключем до забезпечення високої продуктивності систем.

3. Підвищення інтенсивності сумісного використання комунікаційних та інших допоміжних служб вимагають підсилення координації та контролю. Складність системи робить автоматичний моніторинг і відновлення після збоїв абсолютно необхідним.

4. У розроблюваному інформаційному середовищі необхідно забезпечити досить складні функції системного адміністрування, зокрема, необхідно забезпечити захист даних, керування повноваженнями доступу до даних тощо.

5. Для розробки додатків необхідно використовувати єдиний програмний інтерфейс для роботи з БД, такий як API-інтерфейс ODBC.

Для отримання дійсно корисної моделі розроблюваної системи її формування варто здійснювати на базі об'єктозорієнтованих методологій. Об'єкти цих моделей можна потім реалізувати у прикладних програмах для керування установою та надання послуг клієнтам. Нові додатки можуть використовувати ті ж самі моделі прикладної обробки і даних. Всі ці вимоги задовольняє трьохрівнева модель додатків клієнт/сервер.

PENAL FUNCTIONS AT DISTRIBUTION OF TEMPORARY RESOURCES

I. Chichkan (кафедра інформатики НаУКМА)

In a basis of organization of systems, which are adapted to rigid temporary restrictions, principle multiversions of programming is fixed. In [1] algorithm of distribution of a temporary resource, based on the decision of a discrete problem of optimum control, is offered.

In the present report the methods of penal functions offer in a number of cases to improve received analytical dependences of the characteristics, which allow to operate distribution of temporary resources. Numbering of the formulas corresponds [1].

To an estimation of weighed total criterion in view of infringement of phase restriction it is necessary to apply ideas of a method of penal functions.

We shall consider at restrictions (2) and (3) new criterion function of a kind

$$\Phi(N) = \Delta(N) + \frac{\Delta^*(N) - \Delta_{\min}(N)}{\varepsilon_1} f(T(N) - T_0) \quad (7)$$

Where not undifferential function

$$f(T(N) - T_0) = \left\{ \begin{array}{l} 0, \text{ if } T(N) - T_0 \leq 0; \\ T(N) - T_0, \text{ otherwise,} \end{array} \right\}$$

$\Delta^*(N)$ — meaning of criterion function (1) for the allowable decision of a problem (1) - (4);

$\Delta_{\min}(N)$ — an estimation from below of a minimum of function (1) at absence of restriction (4);