

Національний технічний університет України (КПІ)

На правах рукопису

УДК 681.3.06

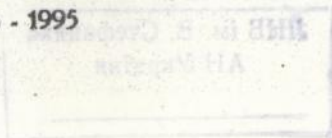
СУРЖИКОВА Аліна Володимирівна

**ПРОДУКЦІЙНИЙ МЕТОД
ГЕНЕРАЦІЇ ПРОГРАМ ІНФОРМАЦІЙНОЇ
ПІДТРИМКИ РІШЕНЬ**

Спеціальність 05.13.04 - "Автоматизовані системи
управління та системи обробки інформації"

**Автореферат дисертації
на здобуття наукового ступеня
кандидата технічних наук**

Київ - 1995



Дисертацією є рукопис.

Роботу виконано на кафедрі автоматизованих систем обробки інформації та управління Київського політехнічного інституту.

Науковий керівник: доктор технічних наук, професор
Гриша С. М.

Офіційні опоненти: доктор технічних наук, професор
Кулик Ю. В.
кандидат технічних наук
Кваліта П. Я.

Провідна організація: Інститут проблем математичних машин
і систем АН України, м. Київ

Захист відбудеться "19" червня 1995 р. о. 15 годині на
засіданні спеціалізованої Ради Д 01.02.08 при Київському політехнічному інституті за адресою: 252056, м. Київ, проспект Перемоги, 37,
вул. 56 корп. 14

З дисертацією можна ознайомитися в бібліотеці Київського політехнічного інституту.

Автореферат розіслано "11" травня 1995 р.

Вчений секретар
спеціалізованої Ради,
доктор технічних наук



Романенко В. Д.

ЛНБ ім. В. Стефаника
АН України

ЛННБ України ім. В. Стефаника



00754901 (Q)

Метою дисертаційної роботи є розробка нових моделей, методів та засобів, які дозволяють швидко, технологічно та якісно проектувати і реалізувати діалогові інструментальні системи обробки даних (СОД) для різних предметних галузей. Технологія розробки СОД повинна бути орієнтована на кінцевого користувача системи, тобто не повинна вимагати від людини, що створює СОД, знань програміста.

Для досягнення поставленої мети у роботі вирішено наступні задачі:

- виконано дослідження задач, що виникають в сфері управління виробництвом, умов експлуатації тих задач, що були автоматизовані, та труднощів, зв'язаних із внесенням змін до алгоритмів або зовнішнього інтерфейсу задачі (систем меню, форм документів, звітів та ін.) і, як наслідок - із коригуванням (а у випадку необхідності, повною переробкою) програмного забезпечення задачі;

- виконано аналіз технологічного процесу створення програмного забезпечення задач із використанням відомих CASE - технологій (CASE-систем) і показано неможливість використання таких технологій для задач з коротким "життєвим циклом" та незручність використання таких систем користувачем-непрограмістом;

- сформульовано загальні вимоги до алгоритмів для автоматизованої генерації програмного коду та ті вимоги, за якими стоять проблеми, що потребують пошуку шляхів більш досконалого вирішення;

- обґрунтовано концепцію подання даних про предметне середовище за допомогою комплексу об'єктів, що забезпечують інваріантність інформаційної технології до змін розмірів даних та структур;

- виконано теоретичне дослідження задачі формування віртуального відношення згідно з запитом користувача - показано можливість вираження процесу побудови кортежів через послідовність блоків пошуку та блоків аналізу результатів пошуку у відношеннях БД, причому, якщо правильно побудувати індекси, можна суттєво зменшити область перебору (пошуку) у порівнянні з традиційними алгоритмами (наприклад, у QBE) і для цього:

- виконано формалізацію правила побудови програмного кода за реляційними запитами та формалізацію правила побудови індексних файлів, результатом яких є запропоновані методи.

Автор захищає:

1. Узагальнення технологічних засобів створення програмного забезпечення за допомогою CASE-технологій.
2. Підхід до розробки модульної структури CASE-системи з використанням об'єктно-орієнтованої методології.
3. Комплекс взаємопов'язаних правил продукційного типу для задачі генерації програмного коду відповідно до запиту користувача, що дозволяє моделювати процес побудови програмного модуля за специфікацією задачі, оформленою мовою реляційної алгебри.
4. Систему продукцій для побудови сукупності опосередкованих індексів відповідно до структури запиту, що дозволяє ефективно вирішити проблему д'яблання користувача результатом.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. Робота спрямована на вирішення актуальної задачі, пов'язаної з розробкою моделей і алгоритмів для автоматизованого конструювання систем обробки даних, і яка дає можливість залучити користувача-непрограміста, що є споживачем відповідної інформації, безпосередньо до процесу створення майбутньої системи. Вирішення цієї проблеми також позбавляє користувача необхідності звертатися до програмістів для внесення змін у алгоритми чи структури і, як наслідок, дозволяє заощаджувати час та отримувати прибуток, що у ринкових умовах набуває особливого значення.

Методи досліджень. Реляційна алгебра, об'єктно-орієнтована методологія, теорія інформаційно-вартісного аналізу.

Наукова новизна досліджень полягає у тому, що розглядається структура CASE - системи, спрямованої на роботу безпосередньо користувача, а не на конструктора; у розробці та дослідженні відкритої системи продукцій для генерації програмного коду за специфікацією задачі, оформленою мовою реляційної алгебри; а також: виконано доказ повноти та несуперечності даної системи продукцій; запропонована реалізація запитів користувача за допомогою використання непрямих (опосередкованих) індексів; розроблена система продукцій для побудови

сукупності непрямих індексів відповідно до структури запиту; показано взаємозв'язок обох систем продукції.

Практична цінність результатів, які одержано в роботі складається в тому, що вказані ідеї були використані при розробці алгоритмічного та програмного забезпечення інтелектуалізованого інтерфейсу користувача (у компоненті, що забезпечує роботу в віртуальними поняттями), який впроваджено в промислову експлуатацію у ряді підприємств, підпорядкова их Міністерству охорони навколишнього природного середовища (м. Київ, м. Черкаси, м. Сарків, м. Донецьк та ін.), НДІ "Енергопрогрес", ОКБ "Південь" мініс-ва машинобудування, Ін-т української археографії та ін. Техніко - економічну доцільність використання підтверджено актами про впровадження. Передбачається використання здобутих наукових результатів при виконанні подальших госпрозрахункових работ кафедри АСОІУ з промисловими підприємствами.

Реалізація результатів роботи виконувалась у рамках програми 6.02.03/ 034-93 "INTEL" Державного комітету України з питань науки та техніки за темою "Розробка наукових основ та програмних засобів автоматичного конструювання комп'ютерних інформаційних технологій для організаційних та організаційно-технічних систем", а також у рамках договорів: Створення сервісних та захисних функцій у складі інтелектуалізованої БД про екологічні дослідження та розробки" між кафедрою АСОІУ КПІ та Міністерством охорони навколишнього середовища, "Адаптація та впровадження інтелектуалізованої асоціативної бази даних" між ТОВ "АСУ-КОМПЛЕКС" та ОКБ "Південь" та ін.

Апробація результатів роботи Основні положення було викладено на 1 Українській конференції з автоматичного керування АВТОМАТИКА - 94, Київ, 1994; на міжнародній науково-практ. конф. " Проблеми українізації комп'ютерів (Проблеми розвитку комп'ютерних інформаційних технологій в Україні)" - Львів, 1993; на 4-ій Міжнародній науково-практ. конф. " Україномовне програмне забезпечення УКРСОФТ - 94 " - Львів, 1994.

Публікації Основні результати роботи надруковані у п'яти публікаціях.

Структура і обсяг роботи. Дисертаційна робота складається з вступу, чотирьох глав, висновку, переліку основної літератури в 61 найменування і 1 додатку. Основний зміст роботи викладено на 109 сторінок машинописного тексту і вміщує 18 малюнків та 3 таблиці.

Перший розділ: по результатах теоретичного дослідження сформульовані основні етапи створення програмного продукту за допомогою існуючих CASE-систем. Визначено, яким чином долаються недоліки існуючих систем в розробковій технології. Охарактеризовані основні функціональні компоненти технології, їх взаємозв'язок. Сформульовано та обгрунтовано задачі дослідження, показано місце цих задач у важливому колі питань, щодо розробки алгоритмічного та програмного забезпечення завданої технології. У другому розділі описано запропоновану систему об'єктів як результат вирішення проблеми реалізації одного з найважливіших вузлів CASE-системи - інтелектуалізованого інтерфейсу користувача, що є водночас оболонкою для збору даних про предметне середовище та гнучким прототипом майбутньої системи. В третьому розділі розглянуто продукційну модель керування процесом вибору кортежів файлів БД при конструюванні віртуального відношення, доведено повноту і надійність моделі. В четвертому розділі описано систему продукцій побудови індексів файлів БД (доведено несуперечність правил системи продукцій) для виключення дублювання кортежів у віртуальному кортежі-результата. Наведено сукупність програмних структур, комбінуючи які за визначеними правилами, можна автоматично одержати модуль, що буде виконувати запит користувача.

ЗМІСТ РОБОТИ

Робота зв'язана з проблемами розробки інформаційної технології підтримки прийняття рішень а, отже, також з проблемами автоматизації розробки СОД. Орієнтованість на користувача, котрий є особою, що приймає рішення (ОПР), і, як правило, непрофесіоналом в галузі програмування, визначила основні вимоги, яким повинна відповідати система. Є очевидним, що для ОПР найбільш важливі - оперативність та достовірність. Крім того, не можна не враховувати, що коло задач, які необхідно вирішувати, постійно змінюється. Також очевидним є те, що не маючи у відповідний момент необхідної інформації, користувачу доводиться приймати інтуїтивне рішення, яке у випадку помилки може

викликати нові проблеми. Оперативність можна збільшити, якщо створити таке середовище, що дозволило би користувачеві-непрофесіоналу будувати необхідні структури даних, формулювати запити до БД, автоматично одержувати програмні модулі, що реалізують запити, не поглиблюючись у проблеми реалізації, а, отож, не відволікаючись від зав'язь основного роду діяльності. Іншими словами, такий інтерфейс повинен сполучати простоту та повноту. Це можливе тільки при відповідній інтелектуалізованості інструментальних засобів (тобто, мова йде про внутрішню складність інструмента, що робить його більш простим). Що до достовірності, то є сенс згадати наступне: метою існування кожного підприємства є прибуток. Тому потрібно враховувати, щоб витрати на доведення задачі до перевищували б ефекту від володіння повною інформацією. Отож, не знецінюючи важливість розробок, зв'язаних з вирішенням задач оптимізації, можна стверджувати, що в кожному разі існує точка оптимальної компромісної інформованості. І головний принцип, покладений до основи розроблюваної інтелектуалізованої інформаційної технології - максимальна інформованість найбільш простими засобами при повній відкритості системи. Тобто остаточне рішення - вдаватися до вирішення оптимізаційної задачі чи ні - залишається за самим користувачем. Але, враховуючи, що проблеми оптимізації - окремий напрямок наукових досліджень, і коло питань, що розглядаються, досить трудомістке і без того, то при розробці даної технології відповідні задачі не розглядалися як задачі користувачького інтерфейсу.

Як наслідок проведеного аналізу існуючих технологій було сформульовано загальні етапи CASE-розробки програмного забезпечення - від постановки задачі до супроводження, узагальнити переваги та недоліки таких CASE-систем. І нова CASE-технологія розроблялась з метою збереження всіх досягнутих переваг та звільнення від відомих недоліків.

Практично неможливо спроектувати систему оптимальної структури водночас. Для вирішення задач проектування необхідно мати дані про розміри, конфігурацію предметного середовища, про інтенсивність інформаційних потоків та ін. Достатньо точна інформація про це може бути одержана тільки в наслідок дослідної експлуатації прототипа системи. Тому необхідно мати деяку програмну оболонку, яка б могла сприйняти всю інформацію для її активного вимірювання під час роботи.

Така оболонка одержала назву *Інтелектуалізованого інтерфейсу користувача (ІІК)* (див. мал.1).



Мал. 1

Однією з задач, що потрібно було вирішити під час розробки ІК, була задача формалізації процесу переведення множини об'єктів, якою оперує користувач до множини об'єктів, з якою доводиться мати справу програмісту (проектувальнику) - конструктору відповідної СОД. Можливий варіант вирішення цієї проблеми описано в дисертаційній роботі. (Під об'єктом розуміється сукупність даних (характеристик об'єкта) та сукупність процедур, що описують "поведінку" кожного з об'єктів). Особливості запропонованого підходу є розглядання у ролі об'єктів програма асоціацій, які характеризують відносини між об'єктами користувача. Звідси і база даних, що була сформована для зберігання відповідних даних набула назву "Асоціативна БД (АсБД)".

Головна позитивна якість - певний (коло розглядаємих задач обмежене) універсальність (інваріантність до змін розмірів даних і структури) - інструментальної оболонки, яка фактично емулює роботу системи на етапі проектування (перепроєктування), перетворює її в невиправдано громіздкий засіб для експлуатації системи в робочому ритмі виробничого процесу.

Тому процес конструювання робочої версії системи нерозривно зв'язаний із генерацією програмних моделей, "налагоджених" на конкретні структури та алгоритми. Вивчивши і формалізувавши процес генерації,

можна надати в розпорядження користувачеві додаткові CASE-засоби - здійснювати генерацію в автоматичному режимі.

Ідея підключення програми - вимірювача чисельних параметрів предметного середовища пн., час дослідної експлуатації в деякій мірі також знаходила реалізацію у інших системах. Але "відкриття чорного ящика" та створення нової версії системи з урахуванням одержаної інформації знову таки покладалась на групу розробників-програмістів. У даній технології ці функції "переадресовані" інтелектуалізованому оптимізаційному генератору, функціональна структура якого подана на мал.2. Тут також відмічені алгоритмічні частини, що були розроблені і знайшли відображення у роботі як закінчені компоненти розроблюваної технології.

Система продукції побудови робочих індексів :

$$S_1 = \langle X, O, R, \Omega, I, I^{\infty} \rangle$$

де X - сукупність змінних - вузлів дерева запиту; O - множина операцій реляційної алгебри (РА); R - множина змінних - операндів операцій РА: $R = \{.u, u.\}$ ($.u$ - лівий операнд, $u.$ - правий операнд); Ω - сукупність операцій алгебри індексів: $\Omega = \{ \overset{\rightarrow}{\cup}, \overset{\rightarrow}{\cap} \}$ ($\overset{\rightarrow}{\cup}$ - об'єднання індексів, $\overset{\rightarrow}{\cap}$ - перетинання індексів); I - множина індексів попереднього рівня: $I = \{A, V\}$ (A - сукупність атрибутів, що включені в індекс, V - послідовність (порядок) атрибутів у ключовому виразі); I^{∞} - множина власних індексів: $I^{\infty} = \{A^{\infty}, V^{\infty}\}$.

Основні операції алгебри індексів:

1. Об'єднання індексів:

$$I_1 \overset{\rightarrow}{\cup} I_2 = \{A_1, V_1\} \overset{\rightarrow}{\cup} \{A_2, V_2\} = \{A_1 \cup A_2, V_1 \cup V_2\}$$

де $\overset{\rightarrow}{\cup}$ - об'єднання двох множин, виконуване за знаними правилами.

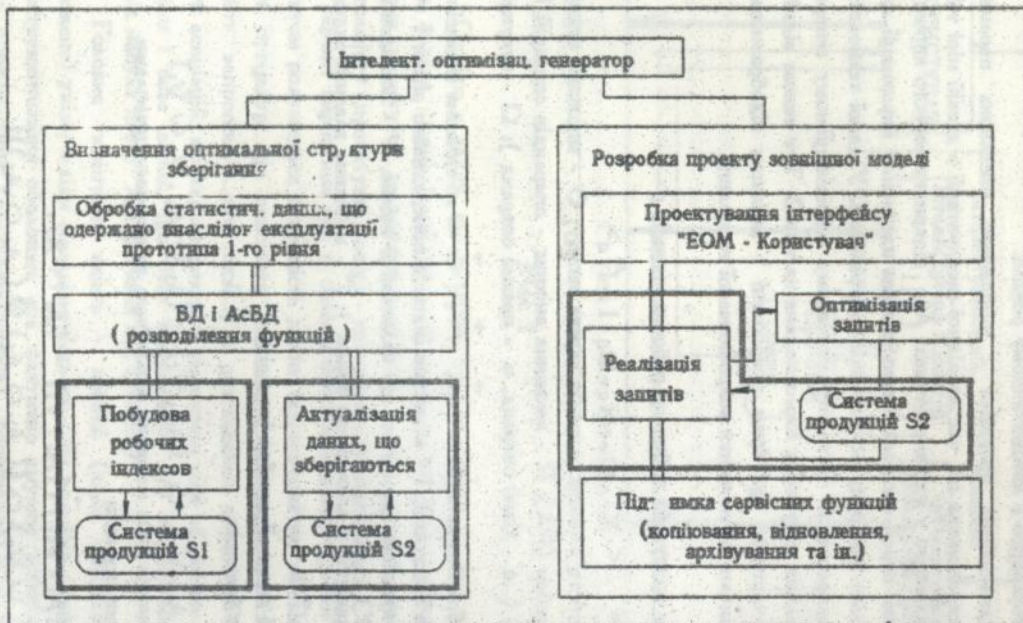
2. Перетинання індексів:

$$I_1 \overset{\rightarrow}{\cap} I_2 = \{A_1, V_1\} \overset{\rightarrow}{\cap} \{A_2, V_2\} = \{A_1 \cap A_2, (V_1 \cup V_2) \cap ((A_1 \cap A_2) \otimes (A_1 \cap A_2))\}$$

де $\overset{\rightarrow}{\cap}$ - перетинання двох множин, виконуване за знаними правилами.

Правила системи продукції :

$$u \in \{<, \otimes\} \Rightarrow I(.u) = I$$



- ключовий вираз індексу для лівого операнда (.u) в операціях "><" і "⊗" співпадає з ключовим виразом індексу попереднього рівня "><" визначає операцію "природного з'єднання", "⊗" - операцію "декартова добутка";

$$u \in \{><, \otimes\} \Rightarrow I(u) \equiv (I \overset{\rightarrow}{\cap} I^{\otimes}) \overset{\rightarrow}{\cup} I^{\otimes}$$

$$\text{де } I^{\otimes} = \{ \text{Sh}(X) \cap \text{Sh}(Y), \emptyset \}$$

- ключовий вираз індексу для правого операнда (u.) в операціях "><" і "⊗" може бути складений із ключових виразів індексу попереднього рівня та власного індексів за допомогою операцій алгебри індексів. Причому, якщо .u описує одну з операцій PA, то X - віртуальне відношення, що описується деревом запиту нижчого рівня з верхньою u; аналогічно, якщо u. також описує операцію PA, то Y - віртуальне відношення, що описується деревом запиту з верхньою u. (.u протилежному разі - .u ≡ X и u. ≡ Y);

Визначення.

Індексом попереднього рівня (головним індексом) I наведемо такий індекс, упорядкованість атрибутів в якому залежить від вищестоячої операції (вузла дерева).

Визначення.

Власним індексом I^{op} наведемо такий індекс, упорядкованість атрибутів в якому визначається тільки поточною операцією і не залежить від вищестоячої операції.

$$u \in \{\pi_s\} \Rightarrow I(u) \equiv I \overset{\rightarrow}{\cup} S$$

- операція π (проекція) змінює ключовий вираз головного індекса, додавая (при необхідності) атрибути з S;

$$u \in \{\cup, \setminus\} \Rightarrow I(u) \equiv I \overset{\rightarrow}{\cup} \{\text{Sh}(X), \emptyset\}$$

- ключовий вираз індекса для лівого операнда (.u) в операціях "∪" (об'єднання) и "∖" (різниця) можна одержати шляхом об'єднання головного індекса в послідовність атрибутів відношення X, переказаних у довільному ("фізичному") порядку. (X- або віртуальне відношення, або відношення БД. В останньому випадку: .u ≡ X).

$$u \in \{\cup, \setminus\} \Rightarrow I(u) \equiv \{\text{Sh}(X), \emptyset\}$$

- порядок слідування атрибутів в ключовому виразі індекса для правого операнда (u.) в операціях "∪" и "∖" повинен співпадати з порядком слідування тих самих атрибутів в ключовому виразі індекса для лівого

операнда. (Тому $Sh(X)$, а не $Sh(Y)$, незважаючи на те, що $A_1^{(X)} \equiv A_2^{(Y)}$ ($V_1^{(X)}$ и $V_2^{(Y)}$ в загальному випадку можуть не співпадати)).

Твердження.

Застосування будь-якого випадкового правила у всіх випадках веде до повної однозначності у порядку слідування атрибутів в ключовому виразі визначасмого операнда, що визначається, (несуперечність правила системи продукцій).

Система продукцій. $S_2: \quad S_2 = \langle \Theta, X, O, \Gamma, R, \rho \rangle$
(керує вибором кортежів у файлах БД)

де Θ - множина ознак результатів виконаних дій: $\Theta = \{f, \bar{f}, n, \bar{n}\}$, (f - перший кортеж знайдено (\bar{f} - не знайдено), n - наступний кортеж знайдено (\bar{n} - не знайдено), ϑ - визначеність ознаки конкретизується для заданого запиту); X - сукупність змінних - вузлів дерева запиту; O - множина операцій реляційної алгебри (РА); Γ - множина виконуваних дій: $\Gamma = \{\alpha, \beta\}$ (α - взяти перший кортеж, β - взяти наступний кортеж, γ - якщо визначеність дії не має значення); $R = R^1 \cup R^2$, R^1 - множина змінних - операндов операцій РА: $R^1 = \{.u, u.\}$ ($.u$ - лівий операнд, $u.$ - правий операнд); $R^2 = \{u\}$ - операція- вершина для розглядаемого операнда; $\rho = \rho^1 \cup \rho^2$ - сукупність правил системи продукцій: $\rho^1: \Theta(X, O) \Rightarrow \Gamma(R^1)$, $\rho^2: \Theta(X, O) \Rightarrow \Theta(R^2)$.

Нехай дерево запиту наведено у вигляді: $G = \{g^i\} = (g | G)$,

де $g^i = \{u, u, .u, u.\}$.

Твердження.

Кожне з правил системи продукцій ($\forall \rho \in \rho$) може бути виражено за допомогою однієї з нижчезказаних формул:

$$\vartheta(x^{i-1}, (g^i | G)) \Rightarrow \vartheta(x^i, (g^i | G))$$

$$\vartheta(x_1^{i-1}, (g^i | G)) \& \vartheta(x_2^{i-1}, (g^i | G)) \Rightarrow \vartheta(x^i, (g^i | G))$$

$$\vartheta(x_1^{i-1}, (g^i | G)) \Rightarrow \gamma(g_2^{i-1}) \Rightarrow \vartheta(x_2^{i-1}, (g^i | G))$$

де x^{i-1} - змінна-кортеж відношення, що є одним із операндів (лівим або правим) операції РА - вузла дерева запиту. (Якщо в формулі необхідні

послання на обидва операнди окремо - з'являються відповідні індекси x_1^{i-1} , x_2^{i-1} . Причому однаково, який із операндів - лівий, який - правий.) x^{i-1} може бути як кортежем певного відношення БД, так і кортежем віртуального відношення, складеного із x_1^{i-2} и x_2^{i-2} . (x^i - відповідно віртуальне відношення, побудоване із x_1^{i-1} и x_2^{i-1} . Тобто і описує розглядаєний вузол (операцію РА) дерева запиту, $i-1$ - вузол-потомок операції i , $i-2$ - вузол-потомок операції $i-1$ і т.д.).

Таким чином, $\vartheta(x^i, (g^i|G))$ визначається або кон'юнкцією $\vartheta(x_1^{i-1}, (g^i|G))$ и $\vartheta(x_2^{i-1}, (g^i|G))$, або одним з $\vartheta(x^{i-1}, (g^i|G))$; в свою чергу, $\vartheta(x_1^{i-1}, (g^i|G))$ - кон'юнкцією $\vartheta(x_1^{i-2}, (g_1^{i-1}|G))$ і $\vartheta(x_2^{i-2}, (g_1^{i-1}|G))$, або одним з $\vartheta(x^{i-2}, (g_1^{i-1}|G))$ і т.д. Через кінцевість числа рівней дерева запиту:

$$\vartheta(x^1, (g^1|G)) = \bigwedge_{j=1, J} \vartheta_j^1,$$

$$\text{де } \vartheta_j^k = \begin{cases} \vartheta(x_1^k, (g_j^k|G)) \\ \vartheta(x_2^k, (g_j^k|G)) \\ \vartheta(x_1^k, (g_j^k|G)) \ \& \ \vartheta(x_2^k, (g_j^k|G)) \end{cases}$$

l - вказує на корневий вузол дерева запиту, k - на листя дерева запиту (конкретні відношення БД), J - кількість вузлів, безпосередніми потомками яких є відношення БД.

Для одержання ϑ_j^k необхідно здійснити пошук в j -ом відношенні БД, для ϑ^l - проаналізувати всі ϑ_j^k .

Оскільки кожна умова ϑ однозначно визначає дію γ , то відповідно:

$$\gamma(g^1|G) = \bigwedge_{j=1, J(G)} \gamma(g_j^1|G), \quad J(G) \leq J$$

де $\gamma(g^j|G)$ - визначає пошук (формування) першого (або наступного) кортежа віртуального відношення; $\gamma(g^k|G)$ - пошук першого (або наступного) кортежа в j -ому відношенні БД.

Твердження.

За допомогою системи продукції можна сформувати послідовність блоків пошуку та блоків постобробки результатів пошуку, що дозволяє говорити про можливість перекладу їх у програмну конструкцію, тобто:

$$F: S(G) \Rightarrow A(G) \Rightarrow M(G)$$

або $\gamma(g_1^k) \Rightarrow \vartheta(g_1^k) \Rightarrow \gamma(g_2^k) \Rightarrow \dots \Rightarrow \gamma(g_{j(C)}^k) \Rightarrow \vartheta(g_{j(C)}^k)$

де $S(G)$ - множина правила системи продукції, які необхідні для реалізації запиту G ; $A(G)$ - послідовність блоків пошуку та блоків постобробки результатів пошуку; $M(G)$ - програмний модуль (блок кода), що реалізує запит G .

Твердження.

Наявність одномісних операцій (таких як π (проекція) та σ (селекція) хіба що опосередковано впливає на порядок виконуваних дій над відношеннями БД, і відповідні блоки можуть бути з'єднані або як постумови з блоками аналізу результатів пошуку або як передумови з блоками, що виконують пошук.

В дисертаційній роботі наведено достатньо загальний приклад того, як за допомогою системи продукції можна одержати модуль, що реалізує запит користувача.

Застосування зазначеної технології дозволяє:

- скоротити термін розробки (модифікації) СОД (за результатами експлуатації ПК - в 7-8 разів);

- скоротити кількість спеціалістів - розробників СОД (при певних навичках роботи з комп'ютером користувач-непрограміст може виконувати всі етапи робіт самостійно);

і, як наслідок:

- отримати прибуток, що в умовах ринку стає вирішальним фактором "виживання" як для програмного продукту (в даному випадку - CASE-системи), так і для тієї виробничої структури, де цей продукт експлуатується.

ОСНОВНІ РЕЗУЛЬТАТИ РОБОТИ

1. Виконане теоретичне дослідження недоліків існуючих CASE-технологій, сформульовані головні проблеми, що вирішуються за допомогою розроблюваної CASE-технології.
2. Розроблено систему примітивів (об'єктів) до компоненти CASE-системи - інтелектуалізованого інтерфейсу користувача, яка є водночас оболонкою для збору даних про розміри, конфігурацію предметного середовища та гнучким прототипом майбутньої системи; розкрито суть синтаксичних, семантичних, родо-видових та алгоритмічних асоціацій між об'єктами.
3. Запропоновано продукційний метод визначення ключових виразів індексів файлів БД, що дозволяє ефективно вирішувати проблему дублювання кортежів у результуючому віртуальному відношенні - є можливість зводжати пам'ять комп'ютера або поліпшити швидкість роботи (в залежності від того, де за умовами алгоритму повинен міститися стек попередніх (побудованих до цього) кортежів - в пам'яті чи на диску); доведена несуперечність правила описаної системи продукцій.
4. Сформовано систему продукційних правл., що описують процес побудови відношення-результата шляхом послідовного пересування покажчиків записів у файлах БД; доведені повнота та несуперечність правила системи продукцій.
5. Розроблено сукупність програмних структур, реалізуючих правила системи продукцій; показано на достатньо загальному прикладі як для конкретного запиту можна одержати автоматично сформований програмний модуль з використанням запропонованої сукупності програмних структур.
6. Запропоновано алгоритми оптимізації одержаного програмного модуля. Доведено, що завдяки оптимізації програмний модуль можна скоротити, зваживши виключно блоки пошуку та блоки постобробки результатів пошуку у відношеннях БД, в яких складено віртуальне відношення - результат.
7. Оскільки конструкції та функції, використовані в програмних структурах, є типовими для різних мов програмування: if-then-else, do-while-case, функції маніпулювання файлами, і, крім того, проміжний результат генерації програмного модуля - текстовий файл програми, що піддається подальшим коригуванням, то можна говорити про незалежність

одержаних результатів від середовища програмування та про користь одержаних результатів для програміста-професіонала.

8. Виконано реалізацію розроблених моделей і алгоритмів в складі алгоритмічного та програмного забезпечення інтелектуалізованої CASE-системи.

ПЕРЕЛІК НАУКОВИХ ПРАЦЬ

1. Гриша С.Н., Суржикова А.В. CASE - технологии с реляционной генерацией / Киевск. политехн. ин-т. - Киев, 1993. - 5 с. - Деп. в УкрНИИНТИ 26.05.93, N 1028 - Ук.93.

Виконано теоретичне дослідження проблем розвитку вказаних технологій, сформульовані головні питання, що плануються до вирішення за допомогою розробки нової CASE - технології, виконана постановка задачі генерації програмного коду відповідно до запиту користувача, описано укрупнений алгоритм вирішення цієї задачі.

2. Гриша С.Н., Суржикова А.В. Интеллектуализированные правила в реляционных CASE - технологиях / Киевск. политехн. ин-т. - Киев, 1993. - 8 с. - Деп. в УкрНИИНТИ 26.05.93, N 1029 - Ук.93.

Конкретизовані деталі алгоритму, що було описано у попередній роботі.

3. Гриша С.М., Суржикова А.В. Асоціативна БД як платформа інтелектуалізованої експертної оболонки / 1-а Українська конференція в автоматичного керування АВТОМАТИКА - 94, Київ, 1994.

Викладені основні міркування на користь асоціативного подання даних; моделі адаптації внутрішнього подання даних до особливостей предметного середовища; зміст синтаксичних, семантичних, родо-видових та алгоритмічних відносин між поняттями асоціативної БД; технології зовнішнього подання асоційованих понять БД.

4. Гриша С.М., Суржикова А.В. Застосування CASE - технології при розробці інтелектуалізованих інтерфейсів користувача баз даних / Міжнародна наук.-практ. конф. "Проблеми українізації комп'ютерів (Проблеми розвитку комп'ют. інформ. технологій в Україні)" - Львів, 1993. Розглядаються питання, що до програмної реалізації інтелектуалізованого інтерфейсу користувача.

5. Суржикова А. Продукція та модель генерації програм у інтелектуалізованій CASE - технології / 4-а Міжнародна науково-практ. конф. "Україномовне програмне забезпечення УКРСОФТ - 94" - Львів, 1994.

Суржикова А.В. Продукционный метод генерации программ информационной поддержки решений. Рукопись. 05.13.04 - "Автоматизированные системы управления и системы обработки информации". Киевский политехнический институт. Киев, 1995.

В диссертации рассмотрены проблемы, связанные с разработкой алгоритмического и программного обеспечения интеллектуализированной CASE-технологии. Технология решает вопросы автоматизированного проектирования и создания систем обработки информации. Разработан набор примитивов (объектов), обеспечивающих инвариантность информационной технологии к изменениям размеров данных и структур. Разработана и обоснована система продукционных правил, позволяющая моделировать процесс построения программного модуля по спецификации запроса к БД. Предложен продукционный метод задания ключевых выражений индексов файлов БД, позволяющий эффективно решать проблему устранения дублирования кортежей в результирующем виртуальном отношении.

Surzikova A.V. The Production Method of Program Generation as The Information Support for Decision. Manuscript. 05.13.04 - "Automatized Control Systems and Processing Information Systems". Kiev Polytechnical Institute. Kiev, 1995.

The problems, which are considered in the dissertation, deal with the elaboration of the algorithmic support and software for the Intellectualized Information Technology (by CASE). The questions of automatized designing and creation of the Processing Information Systems are decided by this technology. It has been proved in this dissertation the elaborated set of the objects guarantees the Information Technology invariance in connection with modification of the data and the data structures. The system of the production rules has been elaborated and grounded to be able to model the construction process of the program module by the query (to the database) specification. The production method to determine the key expression of the database index files are proposed. This method permits to decide the problem of duplicate records in the result virtual relation easy and effective.

Ключові слова: автоматизовані системи управління, інформаційні технології, CASE-технології і системи, системи обробки інформації.

ЛНБ ім. В. Стефаника
АН України

Подписано к печати 4.05.95. Формат 60x84, 1/16
Объем 1,0 печ. лист. Заказ № 50 Тираж 100

ОНТАИЦД УКРНЦЮ

118765 .

AB 32.349

AB 32.349

001 1800 10000 20.20.1 10000 10000 10000
10000 10000 10000 10000 10000 10000 10000
10000 10000 10000 10000 10000 10000 10000