

А.М. Ковальчук, доц.  
Житомирський інженерно-технологічний інститут

### АВТОМАТИЗАЦІЯ ПРОЦЕСУ СТВОРЕННЯ ТА АДАПТАЦІЇ ІНТЕРФЕЙСУ КОРИСТУВАЧА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЧИСЕЛЬНОГО АНАЛІЗУ МАТЕМАТИЧНИХ ЗАДАЧ

В даній роботі розглянуто проблему автоматизації процесу створення та адаптації інтерфейсу користувача програмного забезпечення чисельного аналізу математичних задач. Запропоновано структуру бази даних для автоматизованого формування інтерфейсу користувача на основі формального опису предметної області чисельного аналізу математичних задач.

Детальна структурна декомпозиція інтерфейсу користувача (ІК) програмного забезпечення (ПЗ) для предметної області чисельного аналізу математичних задач, яку представлено у вигляді формалізованого опису [1] надає можливість розробити автоматизовану процедуру побудови інтерфейсу користувача. Автоматизація процесу формування інтерфейсу користувача ПЗ базується на формалізованому описі області наближення даних, представленим у роботі [1], та в багатьох множині програмних інтерфейсів, розглянутих в [2] та оригінальній архітектурі ПЗ [3]. Програмні інтерфейси об'єднують вхідні та вихідні канали підсистем формування ІК та надають можливість використовувати програмні компоненти (підсистеми) різного виду та призначення для формування ІК ПЗ чисельного аналізу математичних задач. Для забезпечення процесу автоматизованого створення та адаптації інтерфейсу користувача необхідно розробити структуру бази даних (БД), яка буде відповідати формалізованому опису, та процедуру формування ІК на основі даної структури.

В архітектурі ІК для забезпечення адаптивності та уніфікованості прикладної програмної системи ІК розподілено на дві частини: першу – жорстку (або фіксовану) частину, яка створюється на етапі розробки програмного забезпечення; другу – адаптивну частину, що зберігається незалежно від програмного коду, та використовується безпосередньо під час виконання програми. На етапі розробки визначаються множини станів підсистеми, вхідних, вихідних, керуючих каналів підсистеми та множина допустимих маніпуляцій користувача. На рис. 1 наведено структурну схему розмежування фіксованої та адаптивної частин архітектури інтерфейсу користувача програмної системи (фіксована частина може бути реалізована підсистемою клієнтом, а адаптивна – сервером).

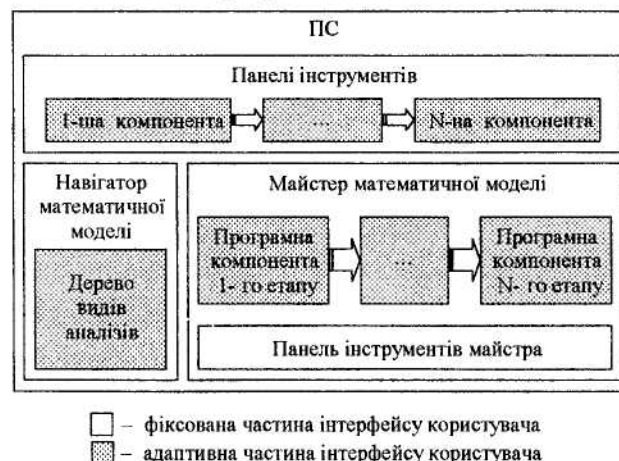


Рис. 1. Схема розмежування фіксованої та адаптивної частин ІК

До фіксованої частини інтерфейсу користувача було віднесено: каркас системи, навігатор математичної задачі, майстри опису математичної задачі та панелі змінних інструментів [4, 5]. На думку автора, фіксована частини інтерфейсу повинна відповідати за загальносистемні дії, а саме: збереження–зчитування даних, підключення зовнішніх бібліотек та створення адаптивної частини інтерфейсу користувача (формування навігатора математичної моделі, майстра моделі, підключення та активізацію програмних компонент, управління панелями інструментів тощо).

Адаптивна частина інтерфейсу програмного забезпечення повинна формуватися автоматизовано в процесі виконання програмного засобу, на основі інформації, яка зберігається окремо від програмного коду.

На основі запропонованих множин кортежів (довідникової інформації щодо деталей ІК) спроектовано структури даних для зберігання, поновлення та доповнення інформації про програмнонезалежну частину інтерфейсу для автоматизованого формування ІК. Зберігання інформації про компоненти та їх параметри, що необхідні для формування ІК, у базі даних дозволяє модифікувати інтерфейс користувача, не змінюючи програмний код. Для вирішення даної задачі було спроектовано та програмно реалізовано інструментальний засіб автоматизації процесу формування ІК.

В основу формування ієрархічних дерев математичних задач та видів чисельного аналізу було покладено множину математичних задач [6, 7].

Структура бази даних створена для формування адаптивного інтерфейсу користувача в ПЗ на основі множин кортежів, що введені в попередньому параграфі (існуюча незначна невідповідність структури БД та множин кортежів пояснюється бажанням підвищити ефективність формування ІК в програмній реалізації системи) (рис. 2).

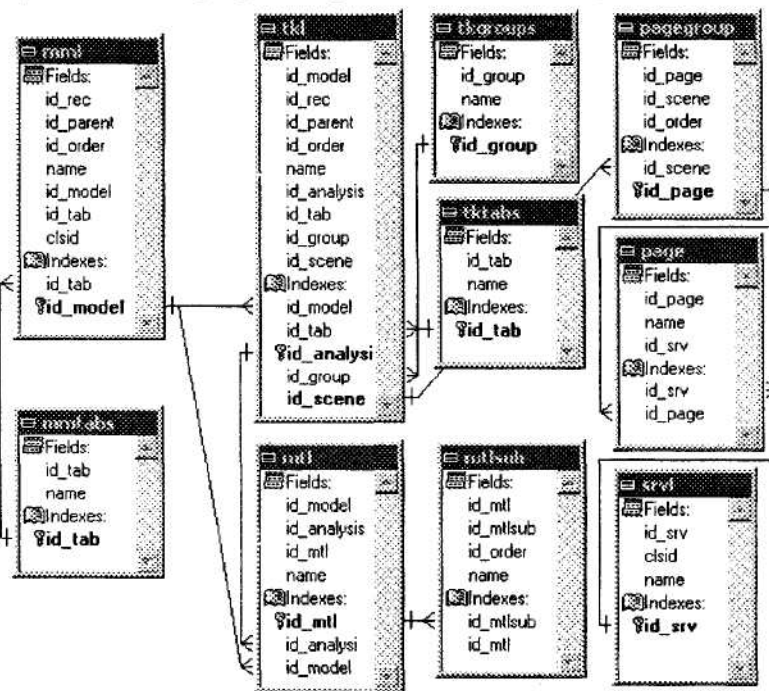


Рис. 2. Структура бази даних формування адаптивного ІК

У табл. 1 наведено перелік та вказано призначення всіх таблиць бази даних, яка використовується для автоматизованого формування адаптивного інтерфейсу користувача.

Таблиця 1

Перелік та призначення таблиць БД формування інтерфейсу користувача

| Назва     | Призначення   |
|-----------|---|
| mml       | Зберігання інформації для формування дерева математичних задач  |
| mmtabs    | Довідник назв для групування математичних задач   |
| tkl       | Зберігання інформації для формування дерева видів аналізів для аналізу конкретної математичної моделі |
| tkltabs   | Довідник назв для групування дерев видів аналізів   |
| tklgroups | Довідник назв для групування видів аналізів у дереві  |
| pagegroup | Довідник сценаріїв аналізу математичних задач   |
| page      | Довідник, що містить опис множини способів візуального представлення на етапі аналізу                 |
| srvl      | Довідник, що містить множину програмних компонент для візуального представлення етапу аналізу задачі  |
| mtl       | Довідник алгоритмів чисельного аналізу математичних задач   |
| mtlsb     | Довідник алгоритмів чисельного аналізу, що використовуються в інших алгоритмах                        |

Розглянемо забезпечення довідниковою інформацією алгоритмів формування візуальних частин інтерфейсу користувача програмного забезпечення. На першому аналізі задачі користувач вибирає тип математичної задачі, аналіз якої буде проводитись. Вибір виду математичної задачі користувач проводить за допомогою дерева математичних задач, для формування якого алгоритми використовують поля "id\_parent", "id\_rec", "id\_order" з таблиці "mml". На рис. 3 наведено приклад таблиці даних для формування дерева математичних структур та результат роботи методу.

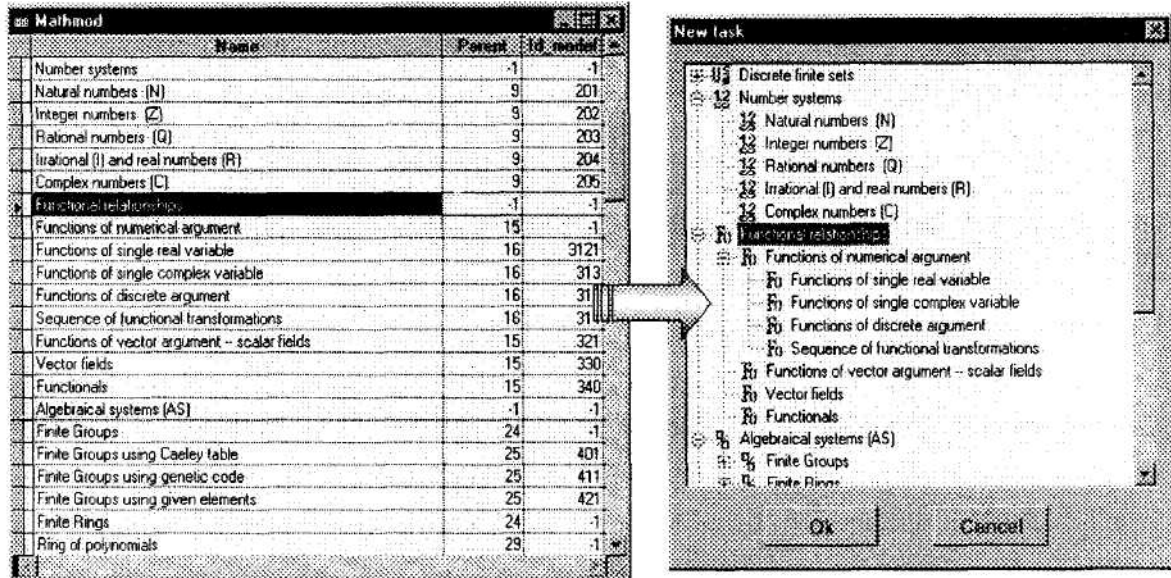


Рис. 3. Приклад формування дерева математичних задач

Ідентифікаційний номер математичної структури, що була вибрана користувачем для аналізу, використовується для подальшої адаптації інтерфейсу користувача. Записи з відповідним ідентифікаційним номером вибираються з таблиці "tkl", з відібраної підмножини видів аналізу формується дерево видів аналізу. Алгоритм формування дерева видів аналізу аналогічний до алгоритму формування дерева математичних задач, (рис. 4).

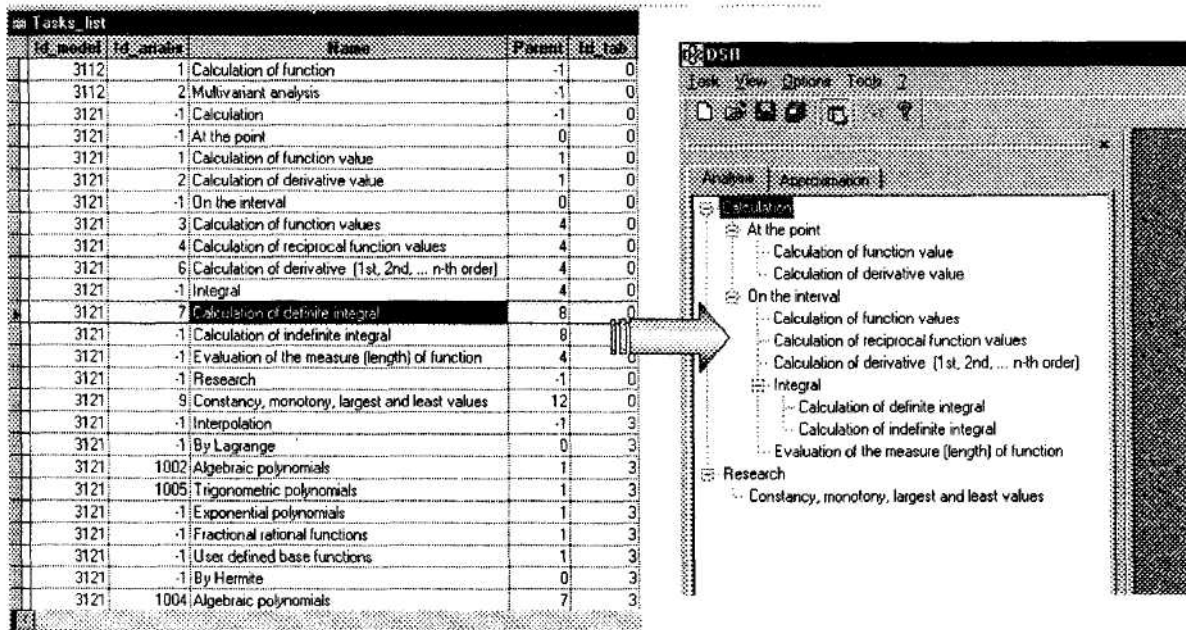


Рис. 4. Приклад формування дерева видів аналізу задачі

Далі відбувається групування видів аналізу за ознаками сумісності методів аналізу та способів візуального представлення задачі для аналізу. Наприклад види наближення даних включені до математичної задачі "Функціональні відношення" та згруповані на закладці "Апроксимація" рис. 4.

Визначення виду аналізу дозволяє виділити сценарій аналізу задачі, для чого використовуються зв'язки між таблицями "tbl" та "pagegroup". На наступному кроці алгоритму за значенням ідентифікатора сценарію з таблиць "page" та "srvl" отримуємо підмножини способів та засобів візуального представлення етапів аналізу в ІК. Для визначення методів чисельного аналізу використовують таблиці "mtl" та "mtlsub".

Збереження інформації про ІК у базі даних дозволяє модифікувати інтерфейс користувача, не змінюючи програмний код. Така можливість робить актуальним питання ефективної модифікації ІК ПС. Це в свою чергу означає, що для модифікації потрібно використовувати не стандартні засоби систем керування БД, а спеціально створений програмний засіб, який враховує запропоновану методику формування ІК. Для зберігання, поновлення та доповнення інформації в БД про предметно-орієнтовану частину ІК створено програмний інструмент "UIDBManager". Програмний інструмент реалізовано у вигляді клієнтської підсистеми, яка використовується не тільки для формування ІК, але й для модифікації БД за допомогою інструментального засобу "UIDBManager".

Створений інструментальний засіб підвищує швидкість та ефективність розробки та модифікації ІК програмного забезпечення наближення експериментальних даних за рахунок використання текстового опису компонент формування візуального інтерфейсу та їх параметрів у відповідних полях таблиць бази даних.

#### ЛІТЕРАТУРА:

1. Ковальчук А.М., Левицький В.Г. Розробка адаптивного інтерфейсу користувача програмної системи чисельного аналізу математичних задач // Вісник ЖІТІ. – 2002. – № 20. – С. 111–119.
2. Ковальчук А.М., Левицький В.Г. Компонентна архітектура адаптивного інтерфейсу користувача програмного інструменту розробки комп'ютерних практикумів // Вестник Херсонского государственного технического университета. – 2002. – № 14. – С. 291–295.
3. Колодницький М.М., Ковальчук А.М., Кур'ята С.М. Типологія та архітектура інтерфейсу користувача прикладної програмної системи "DSR Open Lab 1.0" // Проблеми програмування. – Київ. – 1999. № 2. – С. 111–120.
4. Kolodnytsky M., Kovalchuk A., Kuryata S., Levitsky V. "A Theory of Approximation of Functions" Courseware // Proceedings of the 5th Asian Technology Conference in Mathematics, December 17–21, Chiang Mai, Thailand. – 2000. – P. 86–95.
5. Kolodnytsky M., Kovalchuk A., Kuryata S., Levitsky V. The Mathematical Software Implementation for Computational Algebra and Number Theory // Proceedings of the 4th Asian Symposium on Computer Mathematics, December 17–21, Chiang Mai, Thailand. – 2000. – P. 291–294.
6. Колодницький М.М. Типологія математичних моделей технічних систем. Частина 1 // Вісник ЖІТІ. – 1997. – № 6. – С. 130–142.
7. Колодницький М.М. Типологія математичних моделей технічних систем. Частина 2 // Вісник ЖІТІ. – 1998. – № 7. – С. 208–218.

КОВАЛЬЧУК Андрій Михайлович – доцент кафедри автоматизації і комп'ютерних технологій Житомирського інженерно-технологічного інституту.

Наукові інтереси:

- комп'ютерні інформаційні технології;
- методи організації графічних інтерфейсів користувача прикладних програмних систем;
- використання обчислювальної техніки в навчальному процесі;
- моделювання та розв'язок задач за допомогою обчислювальної техніки.