

РОЗДІЛ 4. ПРОЕКТУВАННЯ АРХІТЕКТУРИ СППР

4.1. Основні підходи до проектування СППР

За категорією класифікації – концептуальна модель (схема) – існують такі підходи до проектування: інформаційний, підхід, що ґрунтується на знаннях, та інструментальний підхід.

Інформаційний підхід

З позицій інформаційного підходу СППР належать до класу інформаційних систем, основне призначення яких полягає в поліпшенні характеру діяльності управлінського персоналу підприємства (саме покращення характеру, а не у наданні потрібної інформації в певний час) за рахунок застосування засобів інформаційних технологій. У межах цього підходу запропоновано дві моделі СППР: «Спрага» та еволюціонуюча модель.

Основні компоненти СППР «Спрага»: інтерфейс «користувач – система», база даних (БД) і база моделей. Інтерфейс «користувач – система» забезпечує зв'язок з кожною базою. Він включає програмні засоби для управління базою даних, управління базою моделей, управління і генеруванням діалогу і повинен забезпечити виконання таких функцій: керувати різними стилями ведення діалогу; змінювати стиль діалогу за бажанням користувача; представляти дані в різних формах і видах; надавати гнучку підтримку користувачеві.

Структурну схему СППР «Спрага» зображено на рис. 4.1.

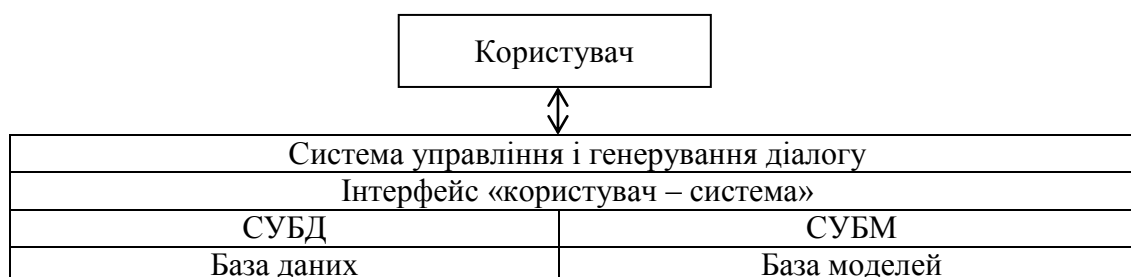


Рис. 4.1. Структурна схема СППР «Спрага»

Бази даних СППР включають як кількісну так і якісну інформацію, що надходить із різних джерел. Засоби створення і ведення БД повинні надавати такі можливості: об'єднувати різні джерела інформації,

використовуючи процедуру їх «добування» даних; представляти логічну структуру у термінах користувача; мати повний набір функцій управління даними.

База моделей повинна забезпечувати гнучкість моделювання, зокрема, за рахунок використання готових блоків моделей і підпрограм. Управління моделями дає такі можливості: каталогізувати та обслужити широкий спектр моделей, які підтримують всі рівні управління; легко івидко створювати нові моделі; пов'язувати моделі з відповідними базами даних.

Подальшим розвитком СППР «Спрага» є еволюціонуюча СППР. Крім інтерфейсу користувача, бази даних і бази моделей ця система включає базу текстів і базу правил, завдяки чому розширюються їх функціональні можливості. Інформаційна база СППР дає змогу використовувати як менш структуровані види інформації (тексти звичайною мовою), так і більш структуровану інформацію (правила представлення знань, евристичні процедури).

Структурна схема еволюціонуючої СППР зображена на рис. 4.2.

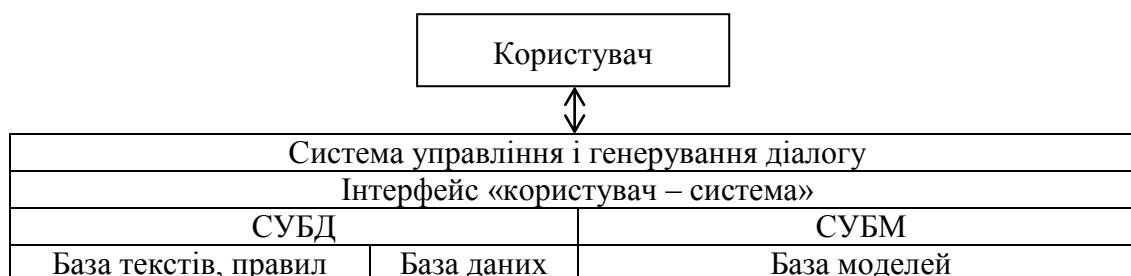


Рис. 4.2. Структурна схема еволюціонуючої СППР

Підхід, заснований на знаннях

Одним із перспективних напрямів розвитку систем підтримки прийняття рішень є об'єднання технологій підтримки рішень і технології штучного інтелекту [12, 13, 39]. Проте у контексті класифікації СППР доцільно розглянути модель СППР, яка ґрунтується на знаннях.

Елементи штучного інтелекту, зокрема використання звичайної мови для спілкування з системою, методологія експертних систем, інженерія знань і комп'ютерних мов штучного інтелекту знайшла застосування у трьох базових компонентах СППР: БД і СУБД, база моделей і система управління базою моделей (СУБМ), інтерфейсі користувача. Але є концепції створення СППР, в яких система знань в СППР виступає як один з визначальних чинників. Відмінною особ-

лівістю СППР, що ґрунтуються на знаннях, є явне виділення нового аспекту підтримки рішень – спроможність «розуміти» проблему, тобто здатність прийняти запит користувача, зібрати відповідну інформацію і підготувати звіт.

Структурну схему СППР, яка ґрунтується на знаннях, зображено на рис. 4.3. Ця система складається з трьох взаємодіючих частин: мовна система (МС); система знань (БД, СУБД, база знань (БЗ) і система управління базою знань (СУБЗ)), і системи обробки (розв’язання) проблеми (проблемний процесор).

Мовна система забезпечує зв’язок між користувачем і всіма компонентами комп’ютерної системи. За її допомогою користувач формулює проблему і керує процесом її рішення, використовуючи запропоновані мовною системою синтаксичні та семантичні засоби. Система знань вміщує інформацію стосовно предметної області. Типи цих систем відрізняються за характером представлення в них даних і використаними моделями формалізації знань (ієрархічні структури, графи, семантичні мережі, фрейми, обчислення предикатів тощо).

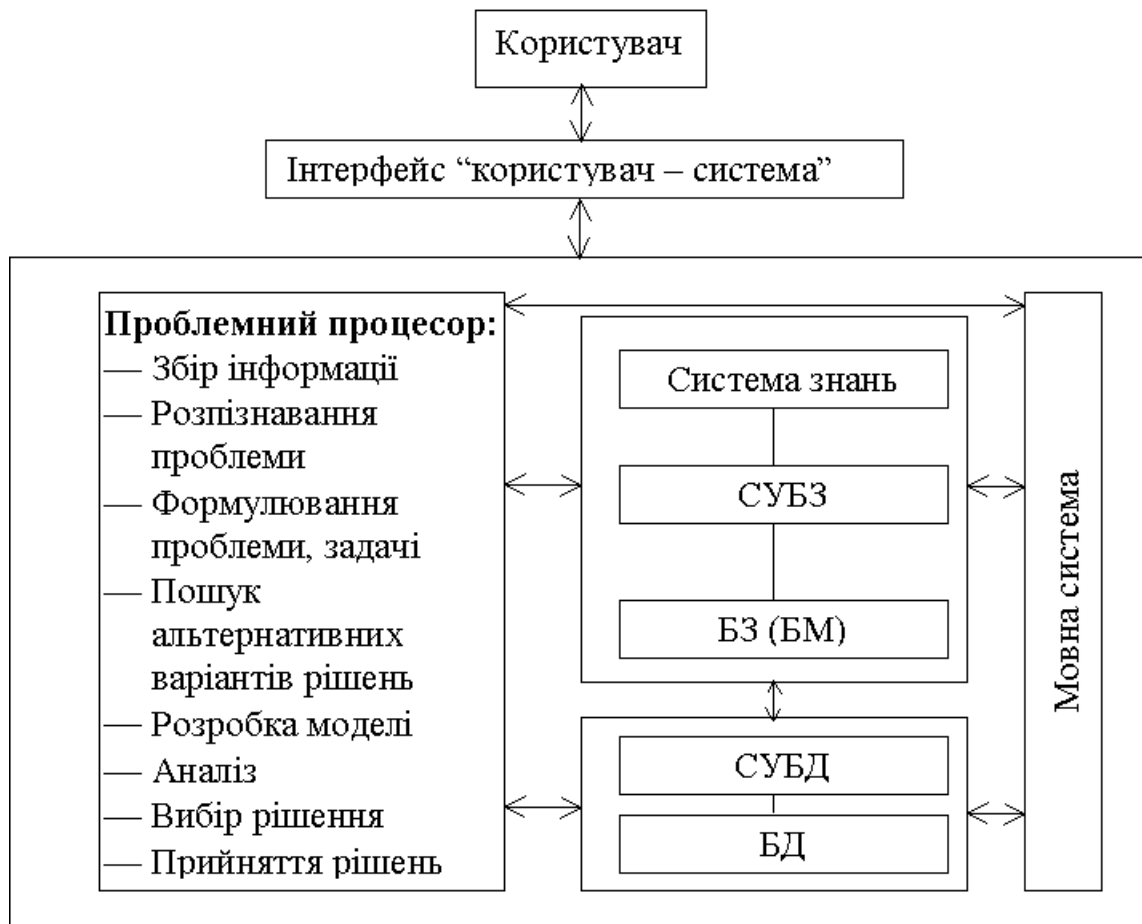


Рис. 4.3. Структурна схема СППР, яка ґрунтується на знаннях

Система обробки задач – це механізм, який пов'язує мовну систему і систему знань. Цей проблемний процесор забезпечує збір інформації, формулювання моделі, її аналіз, тощо. Він сприймає опис проблеми, виконаний відповідно до синтаксису мовної системи, і використовує знання згідно з прийнятими у системі знань правилами з метою створення інформації, необхідної для підтримки рішень. Проблемний процесор – це динамічна компонента СППР, що відображає (моделює) поведінку особи, яка вирішує проблему. Тому він повинен мати як мінімум, можливість інтегрувати інформацію, що надходить від користувача через мовну систему і систему знань, і, використовуючи математичні моделі, перетворювати формулювання проблеми у докладні процедури, виконання яких дає відповідь (розв'язок задачі). У складніших випадках проблемний процесор повинен вміти формулювати моделі, необхідні для вирішення поставленої проблеми.

Інструментальний підхід

Підвищена увага представників інформатики та економічної практики до методів розробки і впровадження СППР зумовила необхідність розробки програмних інструментів для створення СППР, що, в свою чергу, вплинуло на появу нової концепції класифікації СППР – інструментального підходу, розробленої Спрагом. В залежності від специфіки розв'язуваних задач і використовуваних технологічних засобів процесу створення систем можна виділити три рівні СППР:

- спеціалізовані (прикладні) СППР;
- генератори СППР (СППР-генератори);
- інструментарій СППР (СППР-інструментарій).

Спеціалізовані СППР призначені для використання окремим користувачем або групами користувачів. Вони дають змогу індивідуальному ОПР чи колективу ОПР вирішувати специфічні проблеми у конкретних ситуаціях.

СППР-генератор – це пакет взаємопов'язаних програмних засобів (пошуку, переробки й видачі даних, моделювання тощо), який дає змогу легко і швидко створювати спеціалізовану СППР. Прикладом може бути інформаційна керуюча система, яка складається з різноманітних елементів: пошуку інформації, підготовки звітів, мови моделювання, а також множини засобів для виконання фінансових та статистичних аналізів. Оскільки генератори СППР можуть використовуватись і не програмістами, для створення систем підтримки прийняття рішень в області планування й управління розроблено чимало СППР-генераторів: CUFFS88, EXPRESS, FAME та ін.

Концептуальна структура СППР-генератора, яка відображає точку зору користувача, включає п'ять компонент: управління інтерфейсом

користувача, управління представленнями даних і результатів, управління аналізом, системне управління, управління даними.

Управління інтерфейсом користувача повинне забезпечувати реалізацію трьох основних типів інтерфейсу: меню, мова команд, звичайна мова запитань і відповідей.

Управління представленнями повинне підтримувати різноаспектні образи користувача стосовно своєї проблеми, яку потрібно вирішувати. Ці представлення можуть виражатися у вигляді таблиць, графіків або командних процедур.

Управління аналізом даних зводиться до ведення бази моделей. У разі маніпулювання даними при математичному моделюванні множини інструкцій можна подати як підпрограму аналізу. СУБД повинна забезпечувати доповнення бази моделей за рахунок додаткових засобів аналізу. Системний адміністратор забезпечує координацію дій користувачів, а також системного тренажера, що використовується для підготовки користувачів.

Управління даними реалізується за допомогою СУБД, яка повинна містити засоби ведення словника даних, що дасть змогу створити на цій основі інші словники, наприклад, графічний словник чи словник моделей.

До прототипу описаного генератора можна зарахувати систему REGIMES, орієнтовану на персональні комп'ютери. Цей генератор складається з таких компонент: командний процесор, діалоговий процесор, процесор представлення результатів, підсистема управління регресійним аналізом, а також – три словники.

СППР-інструментарій надає в розпорядження проєктантів СППР потужні засоби, в тому числі нові мови спеціалізованої спрямованості, удосконалені операційні системи, засоби обміну інформацією, проєкції кольорових графічних образів та ін. Тому вони можуть використовуватись для створення як спеціалізованих СППР, так і для генераторів СППР.

4.2 Типи архітектур спеціалізованих СППР

Текстово-орієнтовані СППР

Такі системи створювались на початку 70-х років і містять в собі:

- мовну систему (МС), що забезпечує функції взаємодії користувача з системою;
- базу знань та даних (БЗД), яка складається з текстових файлів, що представляють собою інформацію для особи, яка приймає рішення (ОПР); це так звана електронна документація;
- систему обробки даних та генерування результатів (СОДГР), яка виконує різноманітні маніпуляції над текстовою документацією і

включає програмне забезпечення (ПЗ), що полегшує користувачеві складання запитів;

– систему представлення результатів (СПР), яка складає всі можливі формати представлення текстової, табличної і графічної інформації та повідомлення, які полегшують користувачеві спілкування з СППР.

Приклад. Текстова СППР (рис. 4.4.) для інженера технолога. Якщо необхідно розпочати виробництво технічно складної деталі або вузла, то при цьому виникає необхідність прийняти множину рішень стосовно таких факторів:

- які параметри повинен мати новий продукт?
- скільки буде коштувати його виробництво?
- які інструменти, станки і т. д. використовувати?
- яка площа необхідна для обладнання?
- яку, звідкіля і в якій кількості замовити сировину?
- які характеристики має подібний продукт конкурента?
- які проблеми необхідно врахувати стосовно охорони праці, набору додаткових робітників, обслуговування нової техніки?
- як організувати збут продукції?
- на який об'єм продажу і прибутку можна розраховувати.

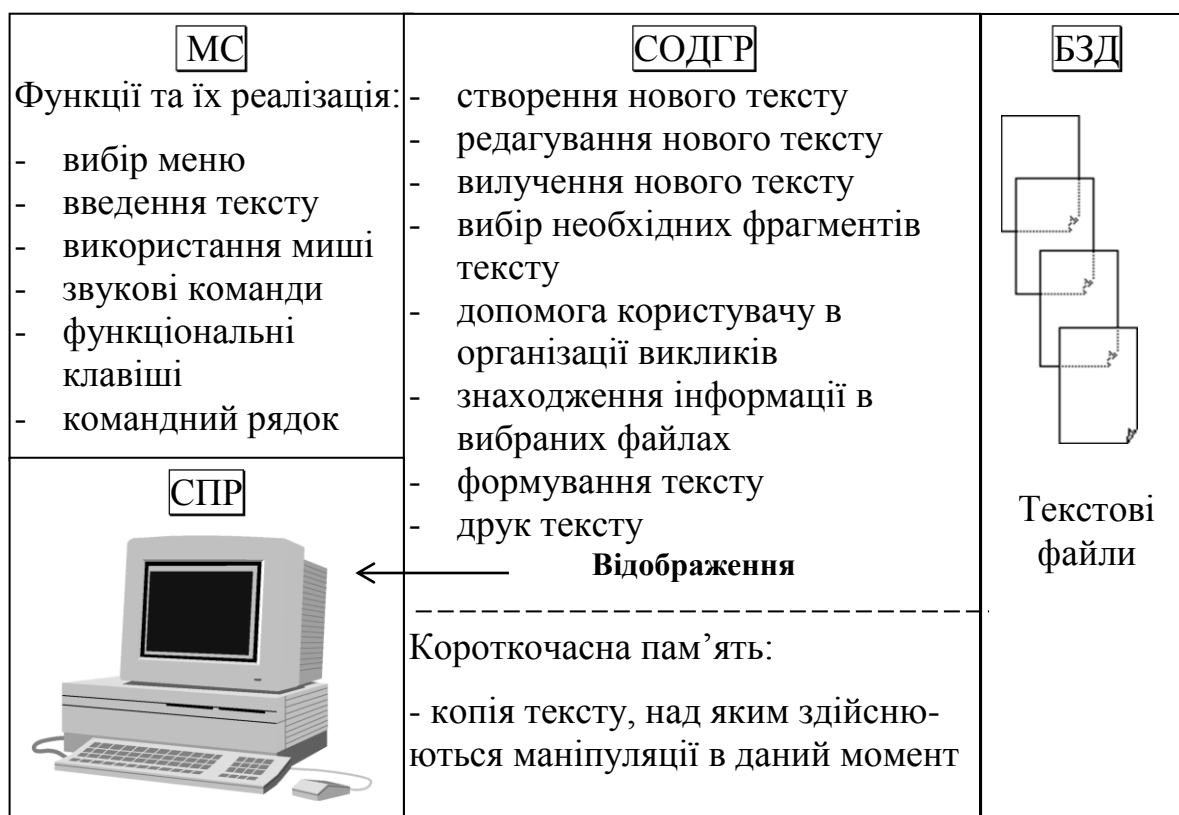


Рис. 4.4. Структура текстової СППР

Ще одна властивість текстової СППР – можливість гіпертекстової підтримки. Гіпертекст встановлює зв'язок між знаннями, які містяться в різних файлах тексту. При цьому кожний фрагмент тексту пов'язується з іншими фрагментами, які концептуально з ними пов'язані.

Наприклад, є фрагмент, де йде мова про потенційного конкурента. Цей фрагмент зв'язується з подібними фрагментами, де йде мова про інших конкурентів.

СППР, які орієнтовані на використання бази даних

Ще один окремий випадок СППР. Найбільш поширені – це реляційні бази даних, тобто в цьому випадку йде обробка строго структурованих знань у вигляді числових і описових даних (рис. 4.5).



Рис. 4.5. Структура СППР на основі БД

В такій системі СОДГР включає три типи програмного забезпечення (ПЗ):

- ПЗ для СУБД;
- інтерактивне ПЗ для обробки запитів;
- спеціальне ПЗ, яке створюється для задоволення потреб користувача (включає, як правило, деяку логіку стосовно аналізу даних і формування відповідей на запити, а також необхідні обчислення: статистичні розрахунки, оцінювання параметрів моделей і прогнозів, порівняння отриманих результатів).

СППР, які орієнтовані на використання електронних таблиць (ЕТ типу Excel)

При використанні технології на основі електронних таблиць для управління знаннями користувач СППР не тільки може створити, проглянути і модифікувати процедурні знання в БЗД, але може дати запит СОДГР виконати команди, які там містяться.

Такі СППР мають такі характеристики:

- БЗД містить файли з таблицями, які наповнені описовими і процедурними знаннями.
- СОДГР може виконувати алгоритмічні процедури, тобто аналіз типу «що буде, якщо ...».

Архітектура СППР на основі ЕТ зображена на рис. 4.6. Формули, які містяться в таблиці, представляють собою основу процедурних знань, які і є командами для СОДГР. Такі СППР, як правило, призначені для виконання аналізу «що буде, якщо ...» для того, щоб побачити і порівняти результати дій над вмістом таблиць. Це засіб для прискореного порівняння альтернатив.

Крім процедурних знань (формули у комірках) і описових знань (числа у комірках), таблиця може містити прості знання стосовно представлення результатів, а також лінгвістичні знання.

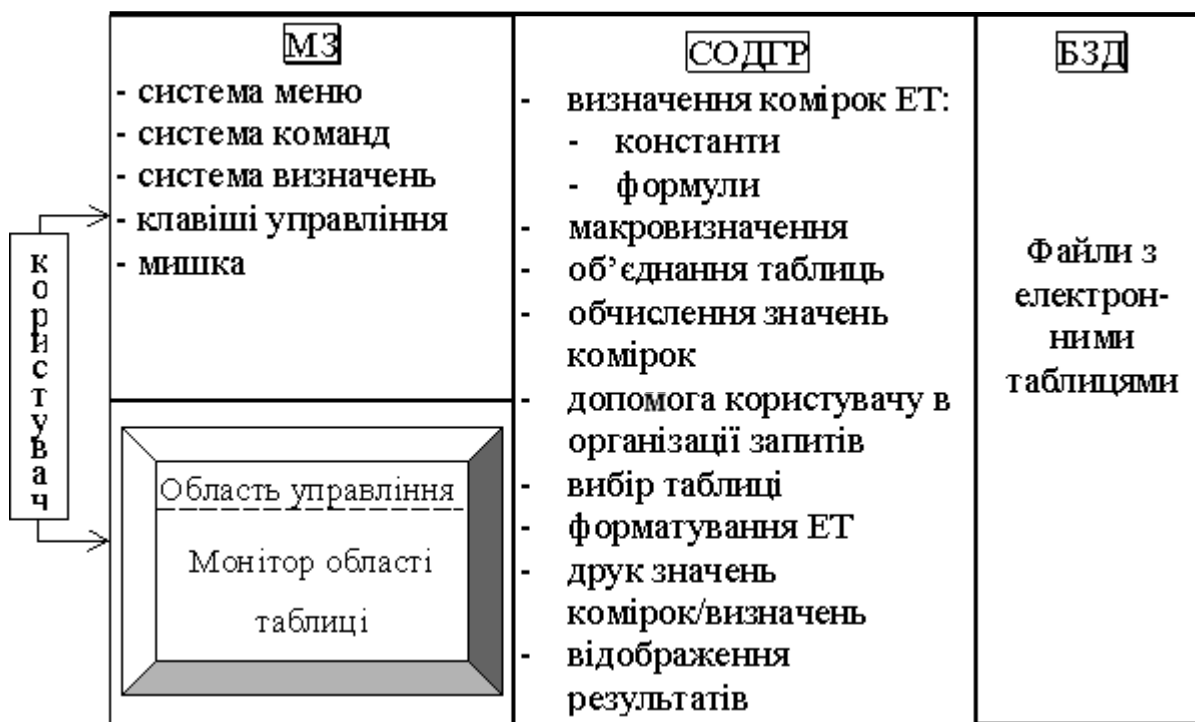


Рис. 4.6. Архітектура СППР на основі ЕТ

СППР на основі алгоритмічних процедур для розв’язання задач

Це може бути, наприклад, СППР для розв’язання задачі оптимізації інвестицій, або максимізації прибутку для конкретного виду виробництва, оптимального розміщення централізованих складів для торгової мережі і т. ін. Як правило, СППР такого типу містить множину алгоритмів для розв’язання вибраного класу задач. Набори алгоритмів для розв’язання конкретних задач фокусуються на задачах таких класів:

- аналіз фінансово-економічних процесів;
- прогнозування процесів довільної природи;
- планування (оперативне, тактичне і стратегічне);
- статистичний аналіз даних в різних галузях;
- розв’язування статистичних і динамічних оптимізаційних задач.

Існує два основних підходи до використання обчислювальних алгоритмів в СППР: (1) фіксований; (2) гнучкий.



Рис. 4.7. Структура СППР з фіксованим використанням алгоритмів

При фіксованому підході алгоритми є частиною СОДГР, а це означає, що до СППР не можна легко додати (видалити) нові алгоритми або модифікувати її в цілому. При гнучкому підході можна легко додавати, видаляти, модифікувати і об'єднувати множину алгоритмів у процесі експлуатації СППР завдяки модульній структурі системи.

При цьому БЗ може містити дані, постановки задач, формати звітів, правила прийняття рішень з метою координації виконання алгоритмічних модулів. Розглянемо більш докладно систему з фіксованим використанням алгоритмів. Структура такої системи наведена вище на рис. 4.7.

Структура гнучкої СППР на основі алгоритмічних процедур наведена на рис. 4.8.



Рис. 4.8. Структура гнучкої СППР на основі алгоритмічних процедур

Відмінність гнучкої системи від жорсткої полягає у тому, що БЗ містить алгоритмічні модулі, які можна комбінувати в необхідній послідовності та запускати на виконання. Крім того, гнучка система

передбачає можливість формування нових та модифікації існуючих критеріїв аналізу якості розв'язку задачі.

СППР на основі правил

В СППР на основі правил (рис. 4.9) в БЗД зберігаються правила та описи станів процесу, а СОДГР містить правила формування висновку для формування логічного висновку на основі правил типу [13]:

If < описання ситуації >;
 Then < які дії виконати >;
 Because < вказівка причин дій >.

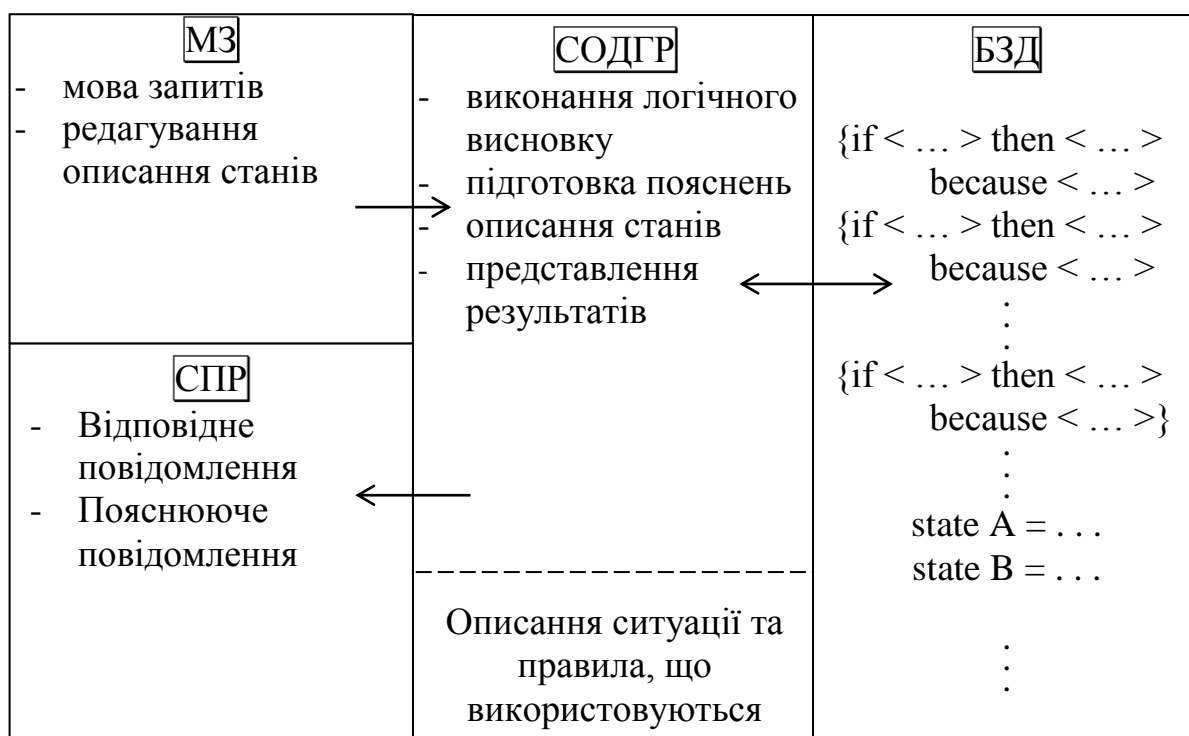


Рис. 4.9. Структура СППР на основі правил

Кожний набір правил в БЗ створюється для того, щоб отримати рекомендації стосовно розв'язання конкретної проблеми. Наприклад, один набір правил може стосуватися того, як уникнути випуску продукції з дефектами, а інший – надає пораду як краще організувати продаж готової продукції.

Описання поточного стану може стосуватися:

- поточного рівня технології на підприємстві;
- розміщення складів для розповсюдження товару на території України;

- кількості та кваліфікації спеціалістів на підприємстві.

Окрім запитів про допомогу (Help) та запитів щодо редагування опису станів, користувач СППР на основі правил може давати два типи запитів стосовно підтримки прийняття рішень:

- запити щодо рекомендацій (порад);
- запити щодо пояснення фактів.

Наприклад, користувач може дати запит відносно можливих причин відмови механічного вузла, а потім дати запит щодо докладного пояснення старіння або виробленості металевих деталей.

Гібридні СППР

Вище ми розглянули окремо такі СППР: текстові, на основі БД, на основі електронних таблиць, на основі алгоритмічних процедур та на основі правил. Якщо об'єднати кілька типів СППР в одну, то отримаємо гібридну систему. Часто об'єднують системи на основі БД та алгоритмічних процедур. Структура такої СППР наведена на рис. 4.10.

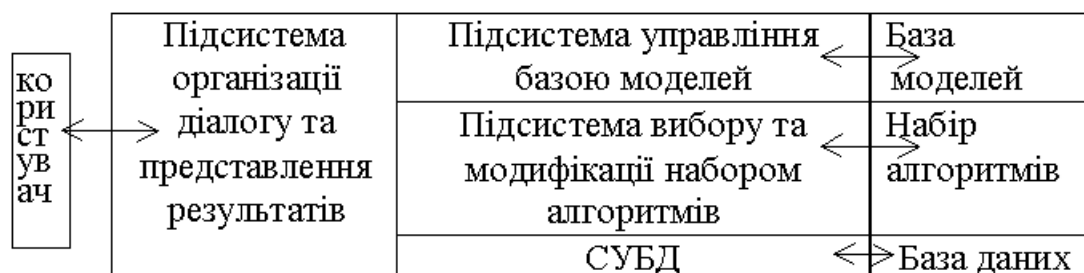


Рис. 4.10. Структура гібридної СППР

Очевидно, що такі системи є функціонально універсальнішими і дають більше можливостей з точки зору запитів і отримуваних результатів. При цьому СППР може сама приймати рішення про те, який метод обробки знань вибрати.

Можливий запит: «Показати середній приріст промислового виробництва для південних районів України».

Розглянемо загальну архітектуру гібридної СППР (рис. 4.11).

За таким типом побудована СППР при управлінні повітряним рухом у великих аеропортах. Вона має такі функції:

- чисто довідкові;
- розрахунок оптимальної траєкторії польоту в аварійних ситуаціях;
- автоматизований гнучкий розклад посадок і зльотів;
- СППР як радник в нештатних ситуаціях;
- стан парку літаків, приписаних до аеропорту;

- довідки стосовно кадрів (персоналу);
- довідки щодо матеріально-енергетичних ресурсів;
- інформація про нештатні ситуації в минулому;
- інформація щодо прогнозу погоди.

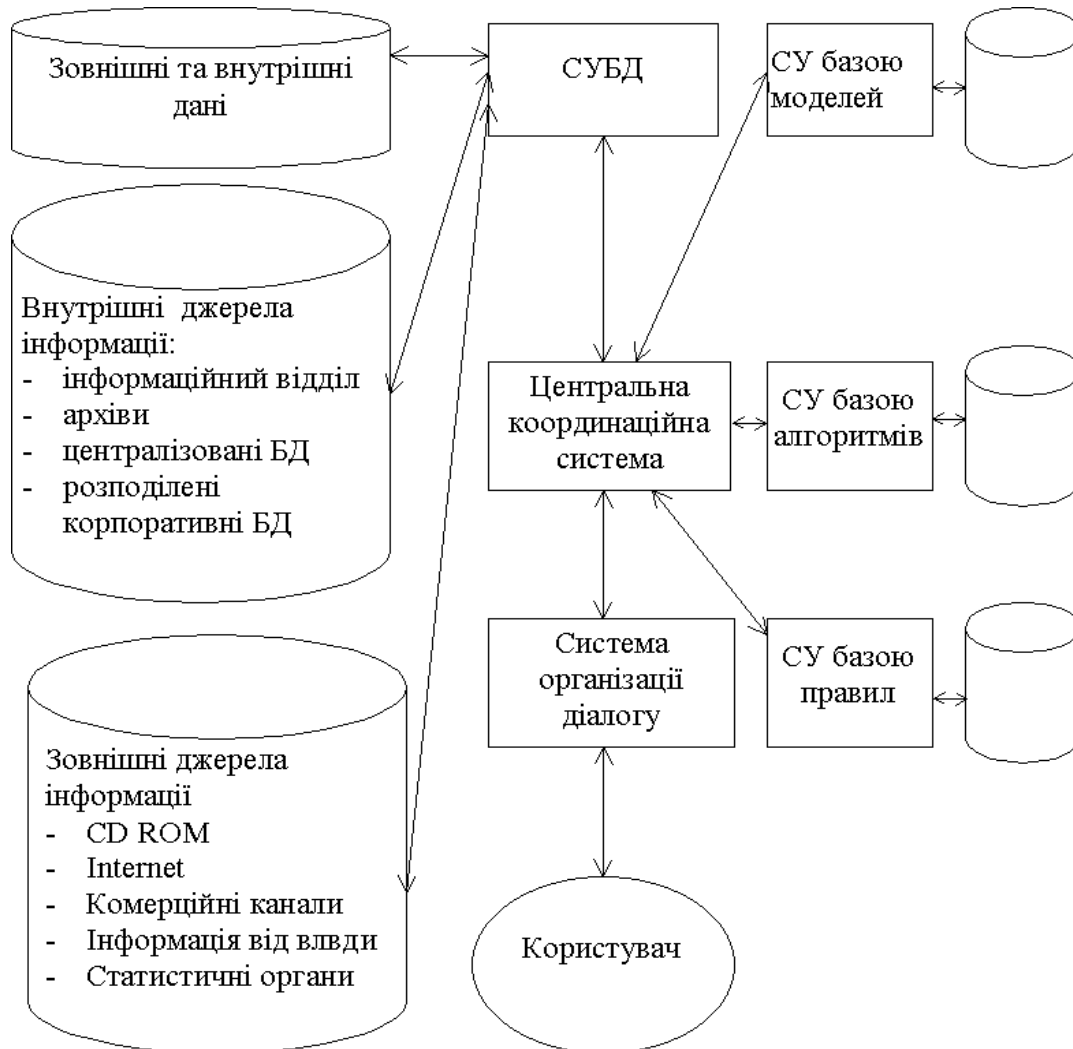


Рис. 4.11. Узагальнена архітектура гібридної СППР

4.3. Функції системи обробки даних та генерування результатів

Розробка системи обробки даних та генерування результатів (СОДГР) є головною системою (ядром) СППР з точки зору її основного призначення – підтримки прийняття рішень при розв’язанні задач в

обраній прикладній області. Розглянемо функції СОДГР на прикладі СППР при прогнозуванні динаміки часових рядів.

СОДГР приймає коректні запити від мовної системи (МС) і виконує такі дії (які задаються запитом):

- поповнює (при необхідності) базу даних і знань;
- вибирає алгоритм (алгоритми) обробки даних з бази даних і знань;
- застосовує вибраний алгоритм до даних, що містяться в базі даних, з метою побудови математичної моделі, обчислення прогнозу чи керуючих дій або виконання інших функцій СППР;
- використовує критерії адекватності моделі з метою визначення ступеня адекватності побудованої моделі;
- застосовує критерії якості прогнозу (або критерії якості керування) для визначення кращого прогнозованого значення або кращої траєкторії оптимального керування вибраним процесом;
- при необхідності реалізує функцію ретроспективного аналізу результатів з метою порівняння останнього отриманого результату (прогнозування, керування, планування і т.ін.) з результатами, які були отримані раніше; таке порівняння дає змогу визначити існування подібних ситуацій в минулому і теперішньому часі;
- передає отримані результати обчислень в систему представлення результатів, яка представляє отриманий результат у формі, заданій користувачем;
- зберігає, при необхідності, частину отриманих результатів в короткостроковій пам'яті для подальшого використання в поточній сесії використання СППР.

СППР, яка проектується для підтримки прийняття рішень в іншій прикладній області, буде мати інші функції, хоча досить часто СППР мають ряд подібних функцій. Наприклад, попередня обробка даних, побудова математичних та статистичних моделей, визначення їх адекватності, перетворення до стандартизованих форм (простір станів), знаходження розв'язків відповідних рівнянь, отримання функцій прогнозування і т. ін.

4.4. Вибір та описання алгоритмів, на яких базується СОДГР

При проектуванні СОДГР необхідно вибрати та описати всі обчислювальні алгоритми та алгоритми прийому/передачі даних, які будуть використовуватись в процесі підтримки прийняття рішень.

Якщо СППР *призначена для прогнозування*, то це будуть такі алгоритми:

- алгоритми попередньої обробки даних (логарифмування, нормування, фільтрація, заповнення пропусків, обробка великих імпульсних значень);

- оцінювання параметрів (коефіцієнтів) математичних та/або статистичних моделей;

- алгоритми обчислення статистичних параметрів, які характеризують якість (адекватність) побудованої моделі;

- алгоритми обчислення оцінок прогнозованих значень (на основі рівнянь АР, АРКС, АРІКС, АРУГ, УАРУГ множинної регресії; за допомогою МГВА; за допомогою фільтра Калмана; за методом подібних траєкторій; методами експоненціального згладжування; нечітких множин; нейронних мереж і т.ін.);

- алгоритми обчислення статистичних показників якості прогнозів;

- алгоритм(и) (або правила) вибору кращої моделі та кращого прогнозу на основі розрахованих показників адекватності моделі та якості прогнозу;

- алгоритми формування бази результатів для проведення ретроспективного аналізу;

- алгоритми поповнення бази даних і знань.

Якщо СППР *призначена для обчислення (прогнозування) оптимальних траєкторій* розвитку того чи іншого процесу (тобто, оптимального керування процесом), то вона повинна містити такі алгоритми:

- алгоритми попередньої обробки даних (логарифмування, нормування, фільтрація, заповнення пропусків, обробка великих імпульсних значень);

- оцінювання параметрів (коефіцієнтів) математичних та/або статистичних моделей;

- алгоритми обчислення статистичних параметрів, які характеризують якість (адекватність) побудованої моделі;

- алгоритм перетворення побудованої моделі в стандартизовану форму простору станів;

- алгоритми обчислення оптимальних траєкторій розвитку процесу, тобто траєкторій керуючих дій (сигналів) і траєкторій для вихідних змінних (змінних стану) процесу; при цьому постановка задачі оптимального керування вимагає таких знань: *математичну*

модель процесу, початкові умови, обмеження на змінні та критерій оптимальності;

– алгоритми обчислення показників якості процесу керування (прогнозування оптимальних траєкторій);

– алгоритм(и) (правила) вибору кращої траєкторії з множини знайдених.

Очевидно, що проектування СППР іншого типу потребує використання алгоритмів іншого функціонального призначення. Наприклад, якщо проектується СППР текстового типу, то СОДГР повинна містити алгоритми пошуку текстової інформації в базі знань, сортування у відповідності до запиту, забезпечення гіпертекстової підтримки, формування звітів і т. ін.

4.5. Дані і знання, які можуть використовуватись у СППР

При вивченні інформаційних СППР традиційно виникає питання: *що таке дані і знання, в чому полягає різниця між ними?* У спеціальній літературі можна зустріти різні означення, які принципово не відрізняються між собою. Одним із означень даних є таке [12, 41]:

Дані – це окремі факти, які характеризують об'єкти, процеси і явища предметної області та їх властивості.

В процесі комп'ютерної обробки дані трансформуються і умовно проходять наступні етапи:

D1 – дані як результат вимірів та спостережень;

D2 – дані на матеріальних носіях інформації (таблиці, протоколи, довідники);

D3 – моделі (структури) даних у вигляді діаграм, графіків, функцій;

D4 – дані у комп'ютері, представлені на мові опису даних;

D5 – бази даних на машинних носіях інформації.

Знання можна розділити на теоретичні і практичні (емпіричні). Емпіричні знання ґрунтуються на даних, отриманих емпіричним шляхом. Вони представляють собою результат розумової діяльності людини, спрямованої на узагальнення його досвіду, набутого в результаті практичної діяльності [12].

Знання – це закономірності предметної області (принципи, зв'язки, закони, закономірності), отримані в результаті теоретичної і практичної діяльності, а також професійного досвіду, які дозволяють робити постановки задач і розв'язувати їх в даній конкретній області.

В процесі створення та обробки знання трансформуються аналогічно до даних:

Z1 – знання в пам'яті людини як результат осмислення фактів, явищ, процесів та їх взаємодії;

Z2 – зберігання знань на матеріальних носіях: підручники, посібники, монографії, довідники;

Z3 – *поле знань* – умовне описання основних об'єктів предметної області, їх атрибутів та закономірностей функціонування, що зв'язують їх;

Z4 – кодування і зберігання знань на мовах представлення знань (продукційні мови, семантичні мережі, фрейми);

Z5 – база знань на машинних носіях інформації.

Часто використовують ще таке означення знань:

Знання – це хорошо структуровані дані, або дані щодо даних, або метадані.

4.6. Моделі представлення знань

Існують десятки моделей (або мов) представлення знань для різних предметних галузей. Однак більшість з них можна звести до таких класів:

- продукційні моделі;
- семантичні мережі;
- фрейми;
- формальні логічні моделі;
- процедурні моделі.

Продукційні моделі.

Продукційна модель – це модель, що ґрунтується на правилах і дозволяє представити знання у вигляді правил типу:

«**Якщо** < умова > **то** < дія > .

«Умова» – це деяке речення-зразок, за допомогою якого виконується пошук в базі знань, а «дія» – це той наслідок або дія, яку

необхідно виконати в результаті успішного результату пошуку (дії можуть бути проміжними і виступати надалі як умови), або ж дії можуть бути термінальними чи цільовими, що завершують роботу продукційної системи.

Висновок на основі такої бази знань може бути *прямим* (від даних до пошуку мети) або *зворотним* (від мети до її підтвердження – до даних).

Дані – це початкові факти, що зберігаються в базі фактів (даних), і використовуються машиною формування висновку в процесі перебору правил з продукційної бази знань.

Продукційна модель даних характеризується *наглядністю, високою модульністю, простотою внесення доповнень та змін, а також чіткістю механізму формування логічного висновку.*

Семантичні мережі

Термін *семантичний* означає *смисловий*, а *семантика* – це наука, яка встановлює відношення між символами і об'єктами, які вони позначають; тобто, це наука, яка визначає смисл знаків (символів).

Семантична мережа – це орієнтований граф, вершинами якого є поняття, а дуги – відношення між ними.

Поняття – це, як правило, абстрактні або конкретні об'єкти, а *відношення* – це зв'язки таких типів: «це» («a kind of» = АКО, «is»), «складає частину» («has a part»), «належить», «любить».

Характерною особливістю семантичних мереж є обов'язкова наявність таких трьох типів відношень:

- клас – *елемент класу* (дерево – береза);
- властивість – значення (колір – синій);
- приклад елемента класу (роза – чайна).

Можна назвати декілька напрямів класифікації семантичних мереж, зв'язаних з типами відношень між поняттями.

За числом типів відношень:

- однорідні (з єдиним типом відношень);
- неоднорідні (з різними типами відношень);

За типами відношень:

- бінарні (в яких відношення зв'язують два об'єкти);
- *N*-арні (в яких є спеціальні відношення, що зв'язують більше двох понять);

Найчастіше у семантичних мережах використовують такі відношення:

- зв'язок типу «частина – ціле» («клас – підклас», «елемент – множина» і т.ін.);
- функціональні зв'язки (визначаються, як правило, дієсловами «впливає», «створює», ...);
- кількісні зв'язки (більше, менше, дорівнює, ...);
- просторові зв'язки (далеко, близько, за, під, над, ...);
- часові зв'язки (раніше, пізніше, на протязі, ...);
- атрибутивні зв'язки (мати властивість, мати значення);
- логічні зв'язки (І, АБО, НІ);
- лінгвістичні зв'язки та інші.

Проблема пошуку розв'язку у базі знань типу семантичної мережі зводиться до задачі пошуку фрагмента мережі, який відповідає деякій підмережі, що відображає поставлений запит до бази. Приклад семантичної мережі показано на рис. 4.12.

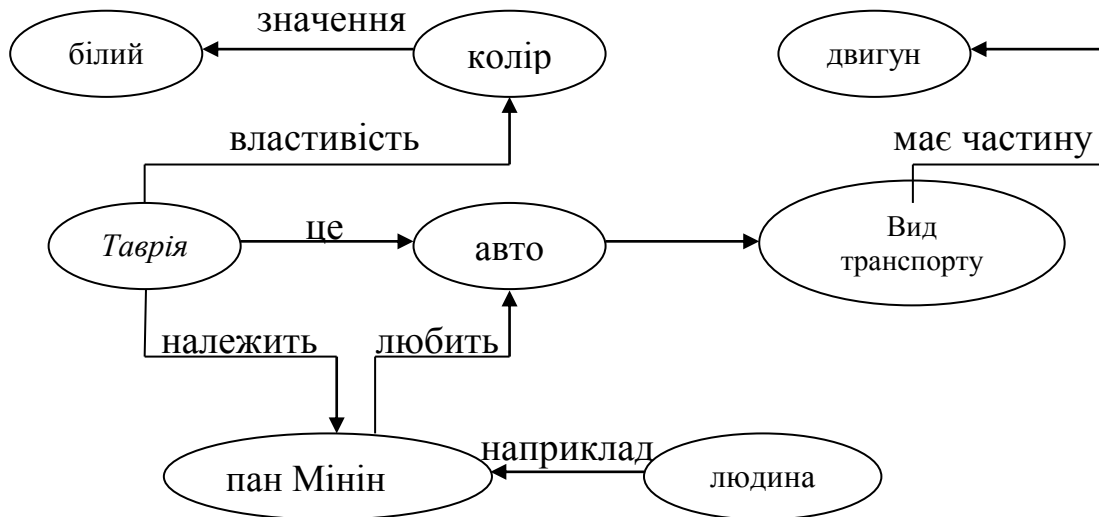


Рис. 4.12. Приклад семантичної мережі

Вершинами тут виступають поняття «людина», «пан Мінін», «Таврія», «автомобіль», «вид транспорту» і «двигун». Така модель запропонована американським психологом Квінлліаном. Перевагою даної моделі є те, що вона краще інших відповідає сучасним уявленням про організацію довгострокової пам'яті людини. Недолік моделі полягає у складності організації процедури пошуку висновку на семантичній мережі. Для реалізації семантичних мереж створено спеціальні мови: NET, SIMER+MIR та інші.

Широко відомі експертні системи, які використовують семантичні мережі як мову представлення знань – PROSPECTOR, CASNET, TORUS.

Фрейми

Термін *фрейм* (від англійського *frame* – каркас, рамка) запропонований в 70-х роках 20-го століття для позначення структури знань, пов'язаних із просторовими сценами. Так само як і семантична мережа, ця модель має психологічне обґрунтування.

Фрейм – це абстрактний образ для представлення деякого стереотипу мислення.

В психології і філософії відоме поняття абстрактного образу. Наприклад, коли промовляють вголос слово «кімната», воно породжує образ кімнати – «приміщення з чотирьома стінами, підлогою, стелею, вікнами та дверями, площею 12-20 квадратних метрів». З цього опису не можна нічого викинути. Так, якщо викинути вікна, то вже буде не кімната, а комора. Однак, в цьому опису є «отвори» (або «слоти» – *slots*) – це незаповнені значення деяких атрибутів, наприклад, кількість вікон, колір стін, висота стелі, покриття підлоги та інші.

В теорії фреймів такий образ кімнати називають фреймом кімнати. Фреймом називають, також, формалізовану модель для відображення образу.

Розрізняють *фрейми-зразки* або *прототипи*, які зберігаються у базі знань, і *фрейми-екземпляри*, які створюють для відображення реальних ситуацій на основі фактичних даних. Модель фрейма є досить універсальною, оскільки вона дає можливість відобразити все розмаїття знань про світ через такі елементи:

- *фрейми-структури*, що використовуються для позначення об'єктів та понять (вексель, позика, застава);
- *фрейми-ролі* (менеджер, касир, клієнт);
- *фрейми-сценарії* (банкротство, збори акціонерів, святкування іменин);
- *фрейми-ситуації* (тривога, аварія, робочий режим машини) та інші.

Традиційно структура фрейма може бути представлена у вигляді переліку його властивостей таким чином:

(ІМ'Я ФРЕЙМА:

(ім'я 1-го слота: значення 1-го слота),

(ім'я 2-го слота: значення 2-го слота),

...

(ім'я N -го слота: значення N -го слота)).

Таку ж структуру можна представити у вигляді таблиці з двома додатковими стовпчиками:

Таблиця 4.1

Структура фрейма

Ім'я фрейма			
Ім'я слота	Значення слота	Спосіб отримання значення	Приєднана процедура

Додаткові стовпчики таблиці призначені для опису способу отримання слотом його значення та можливого приєднання до слота спеціальних процедур, що допускається теорією фреймів. Значенням слота може бути ім'я іншого фрейма, так утворюються мережі фреймів.

Є декілька способів отримання слотом значень у *фреймі-екземплярі*:

- стандартизоване значення (default);
- через успадкування властивостей від фрейма, вказаного в слоті ЦЕ (a kind of = АКО = ЦЕ);
- за формулою, вказаною у слоті;
- через приєднану процедуру;
- явно через діалог з користувачем;
- з бази даних.

Важливою властивістю фреймів є переймання з теорії семантичних мереж так званого *успадкування властивостей*. В фреймах і в семантичних мережах успадкування реалізується через зв'язки типу АКО. Слот АКО вказує на фрейм більш високого рівня ієрархії, звідки неявно успадковуються, тобто переносяться, значення аналогічних слотів.

Формальні логічні моделі

Традиційно у представленні знань виділяють формальні логічні моделі, які ґрунтуються на класичному численні предикатів 1-го порядку, коли предметна область або задача описується у вигляді аксіом. Однак, числення предикатів 1-го порядку на практиці застосовується надзви-

чайно рідко внаслідок того, що воно висуває занадто високі вимоги до предметної області і накладає на неї сильні обмеження.

Процедурні моделі

Процедурні моделі знань – це *алгоритми, функції, формули та алгебраїчні вирази*, які використовуються у процесі побудови математичних моделей, визначення адекватності, обчислення прогнозів та критеріїв, задавання обмежень різного типу і т. ін. Наприклад, база знань може містити десятки різних алгоритмів обчислення параметрів моделей, які використовуються в залежності від конкретного запиту користувача.

4.6. Функції системи представлення результатів, форми представлення

Система представлення результатів (СПР) приймає результати обробки даних та знань від системи обробки даних та генерації результатів і надає їх користувачу в зручній для сприйняття формі.

Можливі форми представлення результатів роботи СОДГР залежать від конкретної прикладної галузі та характеру задач, що розв'язуються за допомогою СПР. Однак можна виділити декілька відносно універсальних форм представлення результатів, до яких відносяться такі:

- гістограми, графічне зображення функцій розподілу ймовірностей отриманих величин;
- графічне представлення результатів у вигляді двовимірних та тривимірних графіків;
- застосування ліній різних типів для різних змінних на графіках;
- одночасне зображення поточних результатів роботи СПР та результатів, отриманих раніше – *ретроспективне порівняння* результатів;
- представлення результатів у вигляді таблиць зручного формату;
- поєднання тексту з цифровим матеріалом, таблицями і графіками;
- кругові та стовпчикові діаграми;
- тривимірні стовпчикові діаграми;
- використання можливостей гортання сторінок (*paging = пейджинг*) та переміщення змісту екрану на один рядок за допомогою функції *скролінг (scrolling)*.

В проекті СППР необхідно описати окремо спеціальні способи представлення інформації, які сприяють прискоренню та поглибленню її сприйняття. Деякі з них розглянемо нижче.

Контрольні задачі і запитання

1. Опишіть існуючі підходи до проектування концептуальної моделі СППР.
2. Поясніть особливості інформаційного підходу та підходу ґрунтованого на знаннях до проектування СППР.
3. Надайте класифікацію типів архітектури спеціалізованих СППР.
4. Побудуйте узагальнену архітектуру гібридної СППР; наведіть приклад такої системи.
5. Сформулюйте функції системи обробки даних та генерування результатів.
6. Вкажіть варіанти вибору обчислювальних процедур для різних видів СППР.
7. Дайте визначення поняттям *дані* та *знання*. Поясніть, у чому полягають відмінності між ними; наведіть приклади знань та способи їх зберігання у базі даних і знань.
8. Наведіть типи моделей представлення знань.
9. Сформулюйте функції системи представлення результатів.
10. Надайте опис основних можливих форм представлення результатів функціонування інформаційної СППР.