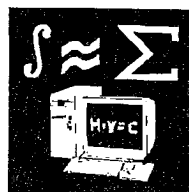


Математика и информатика



УДК 612.821:007

А.Н.МОРОЗЕВИЧ, Б.А.ЖЕЛЕЗКО, С.А.САМАЛЬ

ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННО-АНАЛИТИЧЕСКИХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ ОБЪЕКТАХ

Examined are general principles of the developing adapting analytical data decision support systems' (AD DSS) design process modeling. As the methodological simulation base the UPD-requirements' conception is used. For all that the AD DSS design is considered as the process of the application domain's knowledge models' transformation and of the User, Producer and Designer requirements' co-ordinative changes satisfaction. As the universal data-modeling method the generalized objects conception is proposed being used.

Информационно-аналитические системы поддержки принятия решения (ИА СППР) как класс человеко-машинных систем, предназначенных для оказания помощи лицу, принимающему решение (ЛПР), в его профессиональной деятельности по использованию данных, знаний и моделей, являются в социально-экономических системах «интеллектуальным» подклассом СППР [1].

ИА СППР весьма сложны в реализации. Поэтому необходимо, чтобы срок службы, а главное – время морального старения были как можно больше. Данную цель можно достичь, если при проектировании ИА СППР обеспечить возрастание их возможностей со временем и с усложнением решаемых проблем.

Общие принципы решения таких задач определены так называемой концепцией ПИР-требований, в основе которой – удовлетворение растущих соподчиненных требований Потребителя, Изготовителя и Разработчика [2]. Концепция распространяется и на ИА СППР (см.схему).

Необходимость в проектировании ИА СППР, как правило, возникает в процессе разработки или модернизации некоторой системы управления. При этом ИА СППР рассматривают как компонент системы, связанный не только с объектом управления, но и с внешней средой и обязанный удовлетворять требованиям, вытекающим из особенностей использования ИА СППР в социально-экономической системе.

Согласно определению*, проектированию ИА СППР должна предшествовать выработка требований, предъявляемых к проектируемой ИА СППР, и критериев качества ее работы, которые затем обеспечиваются соответствующими характеристиками ИА СППР. В общем же виде задача проектирования ИА СППР схожа с задачей проектирования управляющих вычис-

* Проектирование ИА СППР – процесс разработки комплекта документации, на основании которой можно изготовить ИА СППР, обладающую характеристиками, обеспечивающими выполнение возложенных на нее функций при заданном качестве функционирования.

лительных машин [3]. Однако в рамках концепции ПИР-требований объект проектирования приобретает нестационарность. Поэтому задача проектирования формулируется следующим образом.

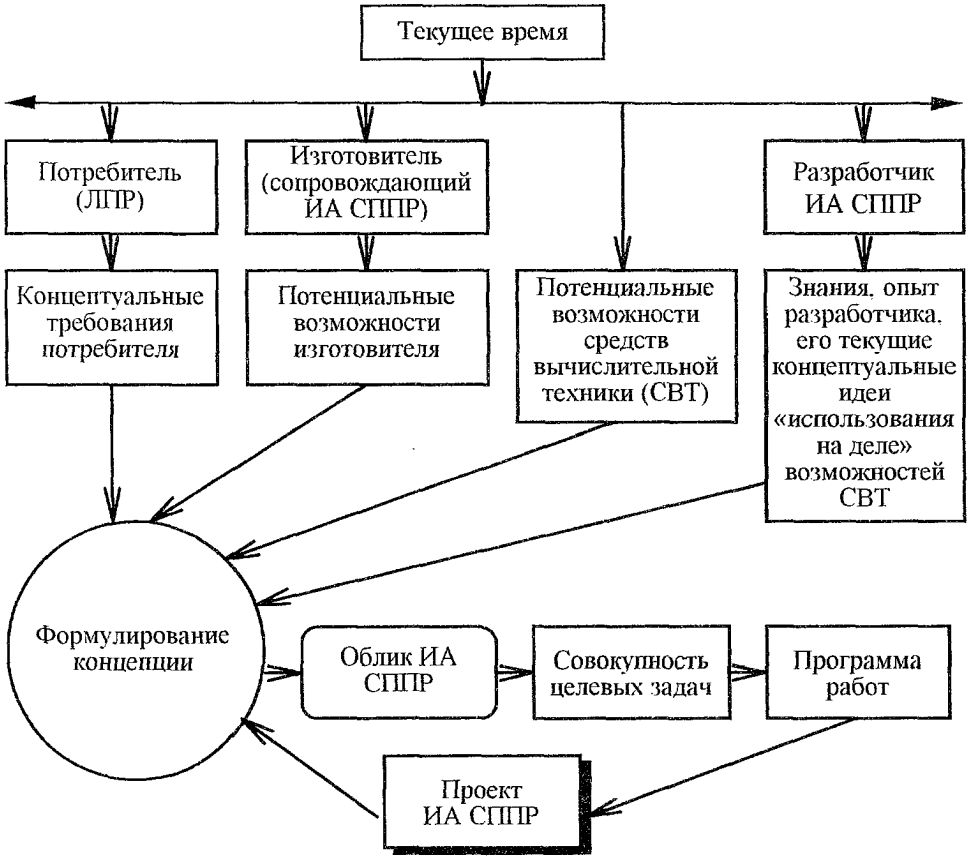


Схема формирования концепции ПИР-требований

Пусть $R(t) = \{r_1(t), r_2(t), \dots, r_m(t)\}$ – требования, предъявляемые к проектируемой ИА СППР; $P(t) = \{p_1(t), p_2(t), \dots, p_n(t)\}$ – характеристики, существенно влияющие на архитектуру ИА СППР; $L(t) = \{l_1(t), l_2(t), \dots, l_k(t)\}$ – система ограничений, накладываемых на основные характеристики, связанные с возможными пределами их изменения; $K(t) = F(R(t), P(t), L(t))$ – критерий оценки качества ИА СППР. Необходимо для заданных векторов $R(t)$ и $L(t)$ отыскать такой вектор $P(t)$:

$$\begin{aligned}
 p_1(t) &= f_1(R(t), P(t), L(t)); \\
 p_2(t) &= f_2(R(t), P(t), L(t)); \\
 &\dots \\
 p_n(t) &= f_n(R(t), P(t), L(t)),
 \end{aligned}$$

при котором величина критерия приняла бы экстремальное значение.

Решение, очевидно, лежит в области открытых архитектур, обеспечивающих в первую очередь расширение входной информационной базы, баз данных и баз знаний за счет «внешних поступлений» или самообучения (в том числе экспертов).

Анализ задачи проектирования показывает, что создание ИА СППР является видом деятельности с интенсивным использованием знаний. Необходимы знания о функциональных и операционных требованиях, а также

конфигурации строящейся системы; о том, как эти требования ограничивают структуру, и другие характеристики данной системы; о ее реализации и возможном описании языков, используемых в процессе разработки программного обеспечения ИА СППР; о структуре алгоритмов и данных, пакетах программ, конфигурации технических средств и т.д.

Разработка ИА СППР может рассматриваться как процесс трансформации моделей знаний о предметной области и удовлетворения соподчиненному росту требований (ограничений) [4]. Ограничения могут быть представлены с нескольких точек зрения [5–7]:

- семантики прикладных подсистем;
- организации системы;
- вычислительных аспектов;
- разработки.

На любой стадии проектирования модель ИА СППР является многоуровневым описанием, состоящим из моделей требований с соответствующими конструкциями и реализациями. Среда проектирования должна содержать инструментальные средства для создания, редактирования и проверки достоверности описания на каждом из этих иерархических уровней. Кроме того, необходимы инструментальные средства для трансформации модельных отображений, которые помогают специалистам в продвижении от исходных моделей требований к проектированию и от проектирования к реализации. В перспективе такие инструментальные средства могут сами рассматриваться как основанная на знаниях ИА СППР, которая управляет проектными решениями и предлагает возможности их развития. Проанализируем подробнее с этой точки зрения современное представление о том, что же является содержанием процесса разработки ИА СППР.

Построение ИА СППР является дорогостоящим видом деятельности и предполагает как разработку и построение баз данных, моделей и знаний, так и разработку прикладных программ и пользовательских интерфейсов. Таким образом, методология проектирования ИА СППР опирается на достижения в области технологий моделирования, проектирования баз данных, знаний и программирования.

Поэтому особое значение приобретает создание методологии, охватывающей все фазы разработки ИА СППР, начиная от накопления и анализа требований. Традиционно [5] основой для проектирования подобных систем считается построение глоссария для каждого проблемно-ориентированного приложения. Построенные глоссарии (по одному для каждой прикладной задачи) разделяются в дальнейшем на данные, операции и события, и для них конструируется концептуальная схема. Так, каждая прикладная задача определяет представление, а все представления должны быть объединены в единственную (глобальную) концептуальную схему. После компоновки концептуальная схема преобразуется в логическую, затем производится «физическое проектирование». Обычно используется расширенная модель типа «сущность–связь» и сети Петри для концептуального и динамического моделирования данных, операций и событий.

Программное обеспечение ИА СППР является средством, требующим эффективного управления на всех этапах от концепции до ее осуществления, проверки, эксплуатации и развития. Как и развитие баз данных, моделей и знаний, развитие программных средств является многостадийным процессом, который начинается с моделирования требований и продолжается разработкой, реализацией и тестированием.

Большой прогресс достигнут в сфере программирования [8]. Однако другие стадии процесса развития ИА СППР остаются относительно неисследованными. Более того, взаимоотношения между различными задачами и уровнями представления отражаются в лучшем случае на бумаге, что усложняет поиск исчерпывающих решений в области проектирования программного обеспечения ИА СППР.

Метод инженерии знаний рассматривает разработку и эксплуатацию ИА СППР как создание и последовательное ведение взаимосвязанных баз знаний. Существует два различных, но взаимосвязанных направления в развитии этого метода. Одно касается вопроса о том, какого рода знания должны быть охвачены, как эти знания могли бы быть представлены и какова адекватная методология их инженерии. Другое связано с вопросом о контроле, поддержке и документировании пошагового процесса инженерии знаний с помощью соответствующей базы знаний.

Процесс проектирования ИА СППР на основе знаний можно представить в виде некоторой схемы с трехуровневой архитектурой. Для моделирования требований необходим язык представления знаний, который может быть гибко приспособлен к любой из большого разнообразия возможных областей применения. Для концептуального проектирования и объектно-ориентированной спецификации необходима формальная рабочая среда, в которой компоненты и взаимосвязи компонентов ИА СППР могут быть единообразно описаны. Для программной реализации компонентов ИА СППР требуются средства с мощными моделями данных и абстрактными конструкциями предикатно-ориентированного доступа, позволяющие создавать масштабируемые прикладные программы.

Разработка в пределах каждого из этих трех уровней может обеспечиваться средами спецификаций, конструирования и программирования, содержащими такие инструментальные средства, как специализированные редакторы, средства прототипирования, визуализации и т.д. Преобразование между уровнями может поддерживаться специальными мастерами отображения, в качестве которых могут выступать, например, предпроцессоры [9], библиотеки классов [10] или CASE-средства [9, 11, 12].

Разработка ИА СППР представляет собой процесс проб и ошибок с возвратами к предыдущему состоянию и накоплением опыта. Формализация знаний может помочь избежать повторения одних ошибок дважды.

После разработки ИА СППР должна быть пригодна к эксплуатации в течение длительного периода времени. Эксплуатация охватывает коррекцию ошибок и учет изменяющихся требований, что требует многократного использования опыта, накопленного на более ранних стадиях разработки. Кроме того, потребители могут уже иметь свои методологии для разработки компонентов ИА СППР. Например, разработка программного обеспечения подразумевает использование библиотеки многократно применяемых компонентов, которые должны быть описаны с позиций формализации требований, проектирования и реализации.

Эти дополнительные условия требуют наличия уровня представления знаний об объектах и процессах разработки как таковых. Модель знаний, разрабатываемая в процессе создания программного средства, становится частью реализации ИА СППР. Это еще один довод в пользу рассмотрения разработки программного обеспечения ИА СППР как прикладной системы с интенсивным использованием баз моделей, данных и знаний. Анализ опубликованных материалов [5, 13] и собственный опыт [10] показывают,

что в принципе можно на 70% идентифицировать и сохранить те концепты и связи, которые, как предполагается, должны многократно использоваться в более поздних фазах процесса создания и эксплуатации ИА СППР. К основным из них можно отнести следующие.

Объекты: фрагменты требований, спецификаций, проектов, реализаций; процессы проектирования: проектные решения и их обоснования, результирующие зависимости; роли, которые играют персонал и компоненты компьютерных систем, их обязанности, задачи и последовательности информационных обменов.

Механизмы представления используемых при проектировании ИА СППР знаний могут рассматриваться как различные варианты представления некоторой основной формализации. На ранних стадиях процесса разработки они могут быть проблемно ориентированными и подчеркивать последовательное введение ПИР-требований. На более поздних стадиях они сосредоточиваются главным образом на решениях, принятых для удовлетворения этих ограничений в конкретной среде. Таким образом, процесс удовлетворения ПИР-требований частично управляется (ограничивается) преобразованиями между базовыми представлениями рассматриваемых уровней. Следовательно, основным вопросом при разработке (выборе) способа и языка моделирования требований является его адаптируемость и способность к расширению на множество областей применения. К числу наиболее важных особенностей, которые необходимо охватить, относятся описание объектов, правила и временное изменение процесса конструирования.

Одним из решений может быть использование в качестве способа моделирования знаний на различных уровнях концепции *K*-систем [14], а в качестве средства реализации этого способа моделирования – концепции ГО-объектов [14], используемых для представления знаний. По сравнению с многими довольно причудливыми языками моделирования требований использование концепции ГО-объектов позволяет достичь необходимых функциональных возможностей с минимальным количеством базовых конструкций, объединенных с технологиями расширения возможностей. Каждый объект предметной области представляет собой звено в цепи ГО, известное системе в течение одного интервала времени, и считается достоверным в течение другого интервала времени.

Классификационная иерархия образует в формируемой базе знаний слой: обозначения, простые классы, классы классов и т. д. Каждый класс определяет некоторый подязык для более низкого класса. Тем самым достигается ряд желательных свойств благодаря способности к расширению за счет неограниченной иерархии классов.

Для практической реализации предложенного подхода среда разработки должна обеспечивать взаимосвязанный набор графических инструментальных средств, которые, сохраняя преимущества формальной модели требований, позволят организовать процесс формулирования и верификации требований следующим образом.

1. Разработать общую структуру модели, используя хорошо известные графические формализации, такие, например, как IDEF0-диаграммы или диаграммы “сущность–связь” [9, 15].

2. Тщательно детализировать эту исходную модель, используя одну из общепринятых объектно-ориентированных методологий [13, 14] и инструментальных средств [8, 14, 15, 17].

3. Проверить достоверность модели, используя прототипирование компонентов ИА СППР и контрольные примеры, выполнить на них дедуктивные запросы и проверки целостности.

4. Расширить модель, путем агрегации/декомпозиции, обобщения/специализации или классификации/обработки [18].

Конкретные методологические и общие абстрактные модели могут рассматриваться как элементы модели программного процесса типа решение – объект–инструментальное средство (РОС) [19].

Модель РОС соответствует уже сложившейся традиции представлять программный процесс как хронологическую последовательность проектных решений. Объекты проектирования могут иметь семантические описания; эти описания состоят из произвольно выбранного комплекса ГО-объектов [14]. Проектные решения соотносят описания объектов проектирования своего входа и выхода с помощью зависимостей. Однако, как показали результаты проекта DAIDA [5], простая модель “объект–решение” адекватна только для представления неавтоматизированных процессов разработки программных средств. В эру частично автоматизированных CASE-сред программный процесс становится историей выполнения проектных решений с помощью инструментальных средств.

Каждый объект в модели среды создается не только со своим структурным описанием, но также с дополнительным набором определяемых пользователем правил преобразования и ограничения целостности, причем последним придают значение при обработке модели среды действительными проектами программных средств. Поэтому реальные объекты проекта, проектные решения и обращения к инструментальному средству проектирования записываются как простые классы модели. В этом отношении проект направляется (дедуктивные правила) или принуждается (ограничения целостности) к использованию только тех языков, методологий и инструментальных средств, которые признаются имеющимися в распоряжении в данный момент времени в среде разработки ИА СППР. Поскольку семантическое описание любой среды может быть ГО-объектом любого класса, то дополнительное множество концептов может быть относительно легко приспособлено для интегрирования новых методов и инструментальных средств, а также для того, чтобы снабжать соответствующие новые классы объектов, решений и инструментальных средств подходящими правилами и ограничениями.

Таким образом, разработка ИА СППР рассматривается как создание и постоянная эксплуатация взаимосвязанных баз знаний. Эти базы знаний должны охватывать весьма различные аспекты, чтобы экспертам было удобно выражать их наиболее понятным для них языком, основанным на использовании концепции ГО-объектов.

1. Морозевич А.Н. // Новые информационные технологии в образовании: Третьей международ. конф. Мн., 12–13 нояб. 1998 г. Мн., 1998. С.28.

2. Он же. // Проблемы качества и надежности изделий электронной техники, радиоэлектронной аппаратуры и средств управления: Тез. докл. науч.-технич. конф. Мн., 1–2 дек. 1988 г. Мн., 1998. С.28.

3. Применение управляющих вычислительных машин: Учеб. пособие / А.Н.Морозевич, А.В.Николаев, А.П.Папкевич, А.А.Петровский. Мн., 1988.

4. Morozovich A., Zhelezko B. // Proc. of Int. Conf. on Application of Computer Systems. Szczecin. 1997. P.434.

5. Jarke M., Mylopoulos J., Schmidt J. et al. // Программирование. 1991. №1. С.3.

6. Davis A // Communication of the ACM. Vol. 31. №9. P.1098.
7. Железко Б.А. // Проблемы переходной экономики и механизм ее функционирования в Республике Беларусь. Мн., 1997. С.414.
8. Калянов Г. // СУБД. 1997. №2. С.61.
9. Чен П. // Там же. 1995. №3. С.137.
10. Morozovich A.N., Zhelezko B.A. Computer system for the analysis of infrared spectra: Тез. докл. VII Белорусской математической конф. Мн., 18–22 ноябр. 1996. Мн., 1996. Ч.3. С.210.
11. Штрик А.А. // CASE-технология: Мат. семинара. М., 1993. С.10.
12. Горин С.В., Тандоев А.Ю. // СУБД. 1995. №3. С.20.
13. Аджиев В. // Открытые системы. 1996. №6. С.40.
14. Железко Б.А., Морозевич А.Н. // Новые информационные технологии в образовании. Мн., 12–13 нояб. 1996. Мн., 1996. С.84.
15. Функциональное моделирование. Методология IDEF0. М., 1993.
16. Козлинский А. // Компьютерное обозрение. 1993. №1. С.29.
17. Железко Б.А., Панасик Г.А., Пихун В.Н., Шелешкевич В.И. // УСИМ. 1990. №4. С.96.
18. Макетирование, проектирование и реализация диалоговых информационных систем / Л.И. Гуков, У.И. Ломако, А.В. Морозова и др. М., 1993.
19. Jarke M., Jeusfeld M., Rose T. // Information systems. 1990. Vol.15. №1. P.86.
20. Железко Б.А. Системы поддержки принятия решений: вопросы создания и примеры использования / Под ред. А.Н.Морозевича. Мн., 1998.

Поступила в редакцию 04.02.99.

УДК 519.10

В.А.ЕМЕЛИЧЕВ, В.Г.ПОХИЛЬКО

РАДИУС КВАЗИУСТОЙЧИВОСТИ МНОЖЕСТВА ПАРЕТО ВЕКТОРНОЙ ЗАДАЧИ МИНИМИЗАЦИИ ЛИНЕЙНЫХ ФОРМ НА МНОЖЕСТВЕ ПОДСТАНОВОК

For a linear vector optimization problem over a subset of the permutation group, we give a formula for the quasistability radius.

Пусть $A=\{a_{ij}\}_{n \times m}$ и $B=\{b_{ij}\}_{n \times m}$ – пара вещественных матриц, $n \geq 1$, $m \geq 2$; S_m – симметрическая группа подстановок, действующая на множестве $N_m=\{1,2,\dots,m\}$; $t \in S_m$. На множестве подстановок $T \subseteq S_m$, $|T| > 1$, зададим векторную целевую функцию

$$f(t, A, B) = (f_1(t, A, B), f_2(t, A, B), \dots, f_n(t, A, B))$$

с частными критериями

$$f_i(t, A, B) = \sum_{j=1}^m a_{ij} b_{it(j)} \rightarrow \min_{t \in T}, i \in N_n.$$

В этом контексте под векторной (n -критериальной) задачей $Z^n(A, B)$ будем понимать задачу поиска множества эффективных подстановок (множества Парето) $P(A, B)$, которое определим традиционным образом [1]:

$$t^\circ \in P(A, B) \Leftrightarrow t^\circ \in T \ \& \ \bar{\exists} t \in T \ (\tau(t, t^\circ, A, B) \leq 0 \ \& \ \tau(t, t^\circ, A, B) \neq 0),$$

где $\tau(t, t^\circ, A, B) = (\tau_1, \tau_2, \dots, \tau_n)$, $\tau_i = \tau_i(t, t^\circ, A, B) = f_i(t, A, B) - f_i(t^\circ, A, B)$, $i \in N_n$, $0 = (0, 0, \dots, 0) \in \mathbb{R}^n$.

Очевидно, что частным случаем нашей задачи (при $n=1$) является широко известная задача минимизации линейной формы на множестве подстановок (см., например, обзор [2]).