

# ПРОБЛЕМЫ ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ ПРИМЕНИТЕЛЬНО К ИНФОРМАЦИОННО-УПРАВЛЯЮЩИМ СИСТЕМАМ ДЛЯ НАУЧНО-ПРОИЗВОДСТВЕННОГО ОБЪЕДИНЕНИЯ

Ф.А. Попов

*На основе опыта создания системы управления базы данных (СУБД) и автоматизированных систем автор уточняет методологию проектирования баз данных для интегрированных информационно-управляющих систем научно-производственных объединений, ориентированных на разработку сложных наукоемких промышленных изделий. Особое внимание уделяется проблемам такого рода проектирования, предлагаются пути и механизмы их разрешения.*

Эффективность организации информационной поддержки при разработке автоматизированных систем непосредственно связана с наличием развитых средств моделирования данных, образующих информационную модель предметной области — модель данных (МД). В развитых МД достигается высокая степень абстракции представления информации за счет ориентации на обобщенное (интенциональное) описание.

Сегодня для большинства исследователей и разработчиков становится очевидным, что реляционные системы, получившие распространение с момента опубликования в 1970 году Э.Ф. Коддом статьи о реляционной модели данных (РМД) [1], не удовлетворяют требованиям приложений и входят в противоречие с объектными методами разработки информационных систем (ИС) [2, 3]. Разрешается это противоречие двумя путями: создаются объектно-ориентированные СУБД, реляционные системы дорабатываются до объектно-реляционных [2, 3].

Кроме того, к современным СУБД предъявляется требование структурированности с использованием клиент-серверной модели, являющейся удобным концептуальным средством ясного представления функций различных программных элементов систем, а также обеспечивающей эффективную обработку данных в рамках информационно-вычислительных сетей [2, 3, 4].

При этом необходимо отметить, что, рассматривая историю СУБД, исследователи обычно ориентируются на зарубежные разработки (IMS - Information Management System, IDS - Integrated Data Store, System R, др.), а также на системы, созданные в нашей стране в академических центрах (CODASYL-система ВЦ АН СССР, иерархическая СУБД ВЕГА Новосибирского филиала ИТМ и ВТ АН

СССР), и упускают из внимания разработки, выполненные в отраслевых НИИ в рамках закрытых работ и слабо отраженные в открытой печати [5].

В этой связи необходимо отметить, что еще в середине 70-х годов в ряде оборонных предприятий была доказана ограниченность реляционных подходов к проектированию БД для многих приложений, в первую очередь для САПР специзделий, и в НПО «Алтай» была создана первая отечественная объектно-ориентированная СУБД, использовавшаяся до начала 90-х годов [6, 7]. Доступ к данным в первых версиях системы осуществлялся с интеллектуальных терминалов и спутников, разработанных здесь же и являющихся эволюционными звеньями на пути создания клиент-серверных моделей распределенной обработки информации [7, 8]. В конце 70-х годов данная система получила законченное выражение как объектно-ориентированная клиент-серверная СУБД, имеющая развитые интеллектуальные человеко-машинные интерфейсы, языки определения данных и манипулирования данными, возможность доступа к информации как с клиентских рабочих мест, так и из прикладных программ, написанных на языках высокого уровня.

Далее необходимо отметить, что разработка ИС в кратчайшие сроки всегда была сопряжена с рядом неудач, многие из которых были вызваны: отсутствием полной спецификации всех требований; отсутствием приемлемой методологии разработки; недостаточной степенью разделения проекта на отдельные компоненты, поддающиеся эффективному контролю и управлению.

В соответствии с этим ниже рассмотрен ряд наиболее важных проблем, возникающих при проектировании ИС с использованием

реляционных и объектных МД и показаны пути их разрешения.

**Поддержка целостности реляционной модели данных.** Организация информационной поддержки принятия решения при проектировании и функционировании информационно-управляющих систем (ИУС) сопряжена с решением взаимосвязанных задач определения данных и манипулирования ими. Использование тех или иных методических подходов при разработке системы определяется главным образом требованиями, которые предъявляются к ним со стороны программных и пользовательских запросов, основным из которых являются обеспечение целостности и актуальности БД.

Рассматриваемый подход основывается на расширении понятия целостности РМД предметной области, которое охватывает целостность сущностей и целостность связей, предложенные ранее Коддом, целостность свойств (Кулинич А.С. [9]), а также целостность понятий [4, 6], повышающем семантику модели данных. Рассматриваемый механизм поддержания целостности реализуется при проектировании ИУС путем создания средств описания и манипулирования логическими единицами РМД — локальными МД, отражающими представления отдельных пользователей о предметной области.

Информационная интеграция этапов проектирования ИУС осуществляется за счет поддержки целостности глобальной МД, формируемой из частных проектных решений, определяемых на различных этапах.

Поддержка целостности модели данных проектных решений должна обеспечивать:

- целостность сущностей, обусловленную ограничениями на значения ключевых атрибутов схем отношений базы данных;
- целостность связей, обусловленную необходимостью контроля значений атрибутов отношений, на которые осуществляется ссылка в РМД объектов предметной области;
- целостность свойств, обусловленную требованиями согласованности значений выделенных атрибутов разных локальных моделей, входящих в РМД одного проектного решения;
- целостность понятий, обусловленную необходимостью использования в различных локальных моделях одинаковых имен для идентичных сущностей.

Первые два аспекта проблемы целостности в РМД были впервые сформулированы Коддом в RM/T при определении реляцион-

но-полной СУБД и являются общими принципами организации РМД. Третий и четвертый аспекты проблемы целостности данных исследованы в работах [2, 4, 6, 9], отражают особенности свойств экземпляров классов объектов предметной области и представления о ней различных категорий пользователей, могут быть формализованы в виде правил создания, синтеза объектов проектирования из структур, являющихся прототипами элементов проектных решений этого объекта. При этом необходимо отметить, что в работе [9] данная проблема исследована только для неперекрывающихся моделей данных (фрагментов проектных решений в терминологии автора), что значительно сужает область применения предлагаемых автором методов ее решения.

В основе рассматриваемого подхода лежит процесс декомпозиции информационной модели проблемной области на части на основе нисходящего метода проектирования архитектуры информационного обеспечения [2,4,9,10]. Элементами сформированной таким образом структуры являются единицы манипулирования данными на уровне представлений отдельных пользователей, лежащих в основе понятий, интеграция которых осложняется тем обстоятельством, что различные группы пользователей могут иметь различные представления об одной и той же предметной области, часто не только дополняющие друг друга, но и частично или полностью совпадают. Описываемые процессы декомпозиции должны осуществляться на концептуальном уровне с использованием, например, ER-моделей [11].

В основе предлагаемой концепции согласования разных локальных моделей одного объекта проектирования лежит механизм их повторной декомпозиции с целью определения общих для различных РМД структур и уточнения используемых понятий, а также получения множества неперекрывающихся структур данных. В результате мы получаем множество  $M$  моделей, отражающих собственно представления пользователей о предметной области, а также множество  $B$  базовых объектов, используемых при построении моделей из  $M$ . Рассмотрим подробнее процесс выделения базовых объектов, использованных в различных представлениях пользователей. Пусть:  $P = \{P_i\}$ ,  $i \in \{1, 2, \dots, n\}$ , где  $n$  - количество представлений;  $P_i = \{O_i, P_{ij}\}$ , где  $O_i$  - множество объектов  $i$ -го представления,  $P_{ij}$  - множество связей между объектами;  $O_i = \{O_{i1}, O_{i2}, \dots, O_{im}\}$ ,  $i \in \{1, 2, \dots, n\}$ ,

$m \in \{1, 2, \dots, l\}$ , где  $m$  – количество объектов в представлении  $i$ ,  $l$  – суммарное количество объектов по всем представлениям.

Пусть далее:

$$R_{11} = O_1 \cap O_2, R_{12} = O_1 \cap O_3, \dots, R_{1,n-1} = O_1 \cap O_n;$$

$$R_{21} = O_2 \cap O_3, R_{22} = O_2 \cap O_4, \dots, R_{2,n-2} = O_2 \cap O_n;$$

$$\dots$$

$$R_{n-1,1} = O_{n-1} \cap O_n.$$

При этом для каждого непустого  $R_{kj}$ ,  $k \in \overline{1, n-1}$ ,  $j = n-k$  его элементы принадлежат представлениям  $O_k$  и  $O_{j+k}$  одновременно.

Тогда множество  $B$  базовых объектов можно сформировать за счет выполнения следующей итеративной процедуры:

$$B = R_{11};$$

$C = B \cap R_{12}$  – множество общих для  $B$  и  $R_{12}$  объектов, определение которых требует уточнения;

$$B = B \cup (R_{12} \setminus B);$$

$C = B \cap R_{13}$  – множество общих для  $B$  и  $R_{13}$  объектов, определение которых требует уточнения;

$$B = B \cup (R_{13} \setminus B);$$

$$\dots$$

$$C = B \cap R_{n-1,1} \text{ – множество общих для } B \text{ и}$$

$R_{n-1,1}$  объектов, определение которых требует уточнения;

$$B = B \cup (R_{n-1,1} \setminus B).$$

Дальнейшее согласование моделей может быть выполнено путем наложения условий на элементы области допустимых значений общих для этих моделей атрибутов. Эти условия позволяют выделить подмножество домена, которое становится множеством возможных значений рассматриваемого атрибута. При этом в процессе проектирования множество возможных значений атрибута образуется как результат пересечения подмножеств, получаемых в результате применения одного правила согласования, в которое входит выделенный атрибут.

Данный подход к решению задачи поддержки целостности РМД является основой методологической базы ведения проектных работ коллективами специалистов и дает возможность уже на этапе анализа информации сократить множество вариантов исходных данных, формируемых для проектной процедуры в соответствии с ограничениями целостности. Модели, построенные таким образом, служат основой для формирования понятийного базиса информационного представления объектов предметной области и взаимосвязей между ними.

**ER-подход к спецификации модели данных.** Следуя [9], под информационно-логической МД ИС будем в дальнейшем понимать четверку

$$M = \langle D, R, P, Q \rangle,$$

где  $D$  — информационное пространство модели организации данных целевых СУБД;  $R$  — множество допустимых классов объектов и взаимосвязей между ними, включающее семантически инвариантные классы объектов, описывающие основные понятия проблемной области;  $P$  — средства манипулирования данными;  $Q$  — семантические правила совместимости элементов ИС, определяющих множество допустимых проектных решений.

В МД  $D$  и  $R$  определяют язык описания данных объектов предметной области ( $R$  описывает структурные свойства классов объектов и взаимосвязи между ними,  $D$  определяет множество допустимых состояний отношений БД), средства  $P$  задают синтаксис языка манипулирования данными, а средства  $Q$  описывают виды соответствий между элементами объектов предметной области.

Рассматриваемая МД охватывает:

- семантически инвариантную часть, в которую входят модели данных классов объектов и взаимосвязей между ними в предметной области, отражающие их свойства, структуру, ограничения на данные и т.д.;

- локальные модели данных, возникающие в связи с особенностями процесса проектирования и отражающие требования конкретной категории пользователей;

- временные модели данных, включающие отношения и домены, сохраняемые в течение одного сеанса процесса проектирования или одной транзакции и уничтожаемые автоматически по окончании сеанса работы пользователя с системой.

Учитывая сложность решения задачи спецификации информационно-логической МД [9, 10], в качестве основного средства ее построения, как правило, выбирается ER-модель [11], в которой ограничения целостности описываются явно с помощью языка предикатов первого порядка. Формулы исчисления предикатов достаточно просто отображаются в виде условий языка запросов при формировании сложных конструкций операций средств манипулирования  $P$ .

Развитие объектно-ориентированных подходов к созданию автоматизированных систем потребовало совершенствования средств ER-моделирования, в результате чего была разработана расширенная ER-

модель (EER – модель), включающая дополнительные концепции специализации, генерализации и категоризации [12]. При этом вводятся понятия подтипа и супертипа сущности, определяется возможность наследования атрибутов. Специализация в EER-модели представляет собой нисходящий подход к определению множества супертипов и связанных с ним подтипов, генерализация же позволяет создавать обобщенный супертип на основе различных исходных подтипов. Категоризация обеспечивает создание связей одного подтипа с несколькими супертипами.

В заключение отметим, что рассмотренные особенности ER-модели позволяют без особых проблем отобразить ее в любую модель данных целевой СУБД (реляционную, объектно-ориентированную, объектно-реляционную, др.).

#### **Нормализация схем отношений**

Процесс проектирования БД включает в себя процессы нормализации схем отношений в рамках шести известных на сегодняшний день нормальных форм [1,13-15], общими свойствами которых являются следующие:

- каждая следующая нормальная форма в некотором смысле лучше предыдущей;
- при переходе к следующей нормальной форме свойства предыдущих нормальных форм сохраняются;
- характерный набор ограничений нормальной формы основан на понятии функциональных зависимостей.

Процесс нормализации обычно состоит из этапов, на каждом из которых отношения, полученные на предыдущем этапе, разбиваются на проекции с целью преобразования их в очередную нормальную форму. При этом в результате выполнения нормализации проектировщик получает гибкий проект БД, позволяющий легко вносить в нее необходимые изменения. В целом же нормализация позволяет исключить избыточные данные, в результате ее последовательного применения МД не будет содержать противоречий, приводящих к известным аномалиям вставки, удаления и обновления [2, 3]. Являясь формальным методом анализа отношений на основе их первичных или потенциальных ключей и функциональных зависимостей, она может быть значительно упрощена в случае предварительного построения и анализа моделей данных на основе использования ER-диаграмм. При этом большинство про-

блем, приводящих к различным аномалиям, могут быть исключены уже на этом этапе без применения формальных методов.

#### **Объектные модели данных**

Поясним кратко сформулированный во вступительной части к статье тезис о несостоятельности реляционных СУБД при решении ряда информационных задач. Проявляется эта несостоятельность в первую очередь в области автоматизированного проектирования, автоматизированного производства, автоматизированной разработки программного обеспечения, гипермедиа- и геоинформационных систем, составляющих основу автоматизированных ИУС для производства сложных промышленных изделий [2, 3, 6, 7, 16].

В частности, в БД для систем автоматизированного проектирования хранятся сведения, относящиеся к проектам различных технических конструкций. Такие проекты имеют следующие общие характеристики.

- данные по создаваемому изделию характеризуются разнообразием типов, каждый из которых обладает небольшим количеством экземпляров;
- проект эволюционирует со временем, при этом последствия любых изменений должны быть отражены во всех его версиях и представлениях, что может потребовать внесения корректировок в большое количество объектов данного проекта;
- часто для каждого компонента проекта необходимо организовать учет его версий и управление выбором конфигурации;
- в работе над проектом участвуют коллективы специалистов, что требует особых подходов к координации их деятельности.

В базах данных для автоматизированного производства хранятся сведения, аналогичные данным САПР, а также данные о производствах. При производстве изделий используются приложения, отслеживающие информацию о состоянии технологического процесса, контролирующие различные физические процессы и т.п. Такие приложения имеют иерархическую структуру, в которой приложения высокого уровня отслеживают работу всего предприятия, а приложения низкого уровня — отдельные производственные процессы. Эти приложения должны работать в режиме реального времени и быть способны эффективно управлять процессами для поддержания оптимальной производительности в рамках заданных допусков. Для реагирования на разные условия эксплуатации в

этих приложениях используется набор стандартных алгоритмов и настраиваемых правил. Операторы время от времени могут изменять эти правила с целью оптимизации производительности, принимая решения на основе архивных сведений, которые должны сохраняться в подобной системе. В данном случае вычислительные системы должны хранить огромный объем данных с иерархической природой, а также сложные связи между данными. Кроме того, в ней должны быть предусмотрены инструменты для перемещения к нужным данным, их просмотра и реагирования на их изменения.

Вся совокупность особенностей объектного подхода позволяет сделать вывод о его согласованности с методологией структурного системного анализа и об эффективности использования для представления данных, обусловленных процессами создания изделий.

#### **Фазы проектирования БД**

Весь процесс проектирования БД состоит из трех основных фаз: концептуального, логического и физического проектирования. Каждая фаза заключается в создании соответствующей МД, являющейся источником информации для очередной фазы. Особо важное значение в этом процессе имеет создание концептуальной МД как первого этапа построения описанной выше информационно-логической модели, разрабатываемой на основе информации, записанной в спецификациях требований пользователей.

При этом под концептуальной моделью данных понимается систематизированное информационное представление объектов предметной области и взаимосвязей между ними, отражающее формальные и эмпирические структурные зависимости между понятиями (классами объектов) и обеспечивающее методологическое единство описания состояния объектов и процессов.

На этом этапе определяется и формализуется совокупность требований к модели данных со стороны всех категорий пользователей системы в виде ряда условий, таких как полнота и адекватность отражения специфики проблемной области, простота и выразительность применяемых понятий и средств, а также возможность наращивания модели данных и ее уточнение по мере расширения знаний о предметной области. В процессе построения концептуальной МД автоматизированных систем необходимо также учитывать особенности, которые обу-

словлены высокой степенью многовариантности и сложностью формализации проектных решений. Кроме того, необходимо учитывать отсутствие точной алгоритмизации процесса проектирования и быстрое развитие научных исследований в области создания автоматизированных систем. Концептуальное проектирование БД не зависит от подробностей ее реализации, но качество концептуальной модели при этом является решающим фактором, определяющим трудозатраты на создание системы, ее эффективность и успех разработки в целом. Опыт разработки и эксплуатации ИС показывает, что ошибки, допущенные на этом этапе, являются наиболее трудно выявляемыми и устраняемыми, т.к. они обнаруживаются в основном на последующих этапах создания систем, таких, как реализация и сопровождение.

На фазе логического проектирования концептуальная МД уточняется и преобразуется в логическую модель, создаваемую на основе выбранной модели организации данных целевой СУБД. Иначе говоря, на этом этапе уже должно быть известно, какая СУБД будет использоваться в качестве целевой - реляционная, сетевая, иерархическая или объектно-ориентированная. Однако на этом этапе игнорируются остальные аспекты выбранной СУБД - например, особенности физической организации ее структур хранения данных. Логическая модель, отражающая особенности представления о создаваемой системе одновременно многих типов пользователей, является глобальной логической моделью данных. Существует два основных подхода к созданию глобальной логической модели данных - централизованный подход и подход на основе интеграции представлений [2]. Для создания крупных ИС наиболее предпочтительным и эффективным является второй подход, при котором глобальная логическая МД получается путем слияния отдельных моделей, отражающих представления разных групп пользователей [4, 6].

На фазе физического проектирования проектировщик принимает решения о способах реализации разрабатываемой БД. Поэтому физическое проектирование неразрывно связано с конкретной СУБД. Между логическим и физическим проектированием существует постоянная обратная связь, так как решения, принимаемые на этапе физического проектирования с целью повышения производительности системы, способны повлиять на структуру логической модели данных.

### **Особенности пользовательского интерфейса информационных систем**

Особенности построения таких интерфейсов детально рассмотрены в [17, 18], здесь же мы только отметим возможности среды WEB как платформы, используемой в целях предоставления пользователям интерфейса для работы с одной или несколькими базами данных [2, 17, 18]. При этом можно видеть, что известная трехуровневая архитектура клиент-серверных СУБД естественным образом отображается на среду Web, где Web-браузер выполняет роль «тонкого» клиента, а Web-сервер — сервера приложений. Среда Web, используемая в качестве платформы для систем с базами данных, становится сегодня основой для решения многих задач в области обработки данных.

### **Выбор СУБД**

Исходя из опыта создания СУБД и информационных систем на их основе, а также учитывая высказанные в свое время предложения Э. Кодда [1], можно утверждать, что любая полномасштабная СУБД должна обеспечивать следующие возможности:

- предоставление пользователям средств сохранения, извлечения и обновления информации в базе данных, скрывая при этом от конечного пользователя внутренние детали физической реализации системы;
- обеспечение разработчика средствами проектирования и быстрой разработки приложений, с автоматизированным получением необходимой проектной и рабочей документации;
- СУБД должна иметь доступный как конечным пользователям, так и функциям системы каталог, в котором хранится описание элементов данных. Системный каталог, или словарь данных, является основным средством обеспечения независимости приложений от структур данных [2, 3];
- обеспечение эффективной поддержки транзакций, наличие механизма, гарантирующего корректное обновление базы данных при параллельном выполнении операций обновления многими пользователями;
- СУБД должна иметь механизм, гарантирующий возможность доступа к базе данных только санкционированных пользователей, т.е. должна обеспечивать защиту базы данных от преднамеренного или случайного несанкционированного доступа;
- способность к интеграции с коммуникационным программным обеспечением с целью организации доступа удаленных поль-

зователей к централизованной БД (в рамках системы распределенной обработки);

- обеспечение разработчика инструментами для создания эффективных и простых пользовательских интерфейсов, являющихся интеллектуальными посредниками между пользователем и системой баз данных;
- СУБД должна обладать инструментами контроля за тем, чтобы данные и их изменения соответствовали заданным правилам, т.е. должна обеспечивать поддержку целостности данных;
- СУБД должна предоставлять набор различных вспомогательных утилит, предназначенных в первую очередь для оказания помощи администраторам БД в их работе.

Цель выбора СУБД заключается в выборе системы, в максимальной степени удовлетворяющей перечисленным выше требованиям с учетом развития возможностей ИС, при оптимальном уровне затрат, включающих расходы на приобретение СУБД и дополнительного аппаратного и программного обеспечения, а также расходы, связанные с необходимостью обучения персонала.

### **Адаптация информационно-управляющей системы к потребностям пользователей**

Одним из эффективных средств адаптации ИУС к потребностям пользователей являются специальные структуры данных, основанные на использовании динамических словарей, в которые помещаются значения атрибутов объектов базы данных, употребляемые в пользовательских запросах [6, 7]. Для каждого значения из словаря создается инверсный список, обеспечивающий прямой доступ к записям базы данных, содержащим данное значение. Динамический словарь организуется таким образом, что в процессе работы информационной системы значения, редко используемые в запросах, постепенно вытесняются из словаря и заменяются на более употребительные. Размер словаря определяет глубину опыта, накопленного системой в процессе ее работы при поиске ответов на запросы.

Результаты имитационного моделирования, подтвержденные результатами эксплуатации информационных систем, основанных на использовании фиксированных и динамических словарей показывают очевидные преимущества последних в части оперативности выполнения пользовательских запросов.

## ПРОБЛЕМЫ ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ ПРИМЕНИТЕЛЬНО К ИНФОРМАЦИОННО-УПРАВЛЯЮЩИМ СИСТЕМАМ ДЛЯ НАУЧНО-ПРОИЗВОДСТВЕННОГО ОБЪЕДИНЕНИЯ

### Выводы

В середине 70-х годов в НПО «Алтай» разработана универсальная объектная система управления базами данных, нашедшая широкое применение при решении задач проектирования и управления процессами изготовления изделий машиностроения, превосхитившая многие более поздние разработки такого рода.

Наиболее перспективным на сегодняшний день является использование для этих целей объектно-реляционных систем управления базами данных, сочетающих возможности объектного и реляционного моделирования.

Опыт разработки систем баз данных и информационных систем позволяет утверждать, что наиболее важными и ответственными моментами при их создании являются этапы их проектирования и выбора системы управления базами данных.

1. При проектировании информационных систем особую важность представляют концептуальный и логический этапы создания информационных моделей, а также процессы проектирования пользовательского интерфейса, направленные на удовлетворение информационных потребностей и удобство работы с данными для конечных пользователей.

2. Использование средств адаптации к потребностям пользователей в информационных системах обеспечивает сокращение временных затрат на поиск сведений, обусловленных пользовательскими запросами.

3. Перспективным направлением работ в области систем баз данных является использование среды Web как платформы приложений баз данных.

### ЛИТЕРАТУРА

1. Codd E.F. A relational model of data for large shared data banks. // Comm. ACM.-1970. - №13(6). - P.377-387.
2. Конноли Т., Бегг К., Страчан А. Базы данных: проектирование, реализация и сопровождение. Теория и практика, 2-е изд.: Пер с англ.-М.: Издательский дом «Вильямс», 2000. - 1120 с.
3. Дейт К. Введение в системы баз данных, 6-е издание: Пер. с англ. - К.;М.; СПб.: Вильямс, 2000.-848 с.
4. Попов Ф.А., Максимов А.В. Подходы к проектированию баз данных для автоматизиро-

ванных систем // Изв. АГУ. Серия Математика, информатика, физика. - 2003. - №1(27).-С.50-53.

5. Коголовский М.Р. Энциклопедия технологий баз данных.- М.: Финансы и статистика, 2002. - 800 с.

6. Попов Ф.А., Груздев Г.П., Галигузов С.Н. Информационно-поисковая система в системе автоматизированного проектирования изделий машиностроения // Автоматизация проектирования.-1977. - N 1(3). - С.51-56.

7. Попов Ф.А. От систем с интеллектуальными терминалами и спутниками к клиент-серверным системам// Наука. Культура. Образование. - 2001. - №6. - С.17-19.

8. Попов Ф.А. Клиент-серверные системы в историческом аспекте // Информационные технологии в экономике, науке и образовании: материалы 2-ой Всероссийской научно-практической конф. (19-20 апреля 2001 г.) - Бийск: Изд-во АлтГТУ, 2001. - С. 117-122.

9. Кулинич А.С., Лескин А.С., Мальцев П.А., Хоботов Е.Н. Системы поддержки решений для проектирования гибких производственных систем. - Санкт-Петербург: Наука, 1995. - 248 с.

10. Чери С., Пернички Б., Видерхольд Д. Методологии проектирования распределенных баз данных /Пер. с англ. // ТИИЭР. - 1987. - Т.75, №5. - С.7-22.

11. Chen P. The Entity-Relationship Model - Toward a Unified View of Data.// ACM Transactions on Database Systems. - 1976. - V.1, №1.

12. Teorey T.J., Yang D., Fry J.P. A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model // ACM Comp. Surv. - 1986. - №2.

13. Codd E.F. Further Normalization of the Data Base Relational Model// Data Base Systems, Courant Computer Science Symposia Series 6.- Englewood Cliffs, N.J.: Prentice-Hall, 1972.

14. Maier D. The Theory of Relational Databases// Computer Science Press. - New York, NY., 1983.

15. Fagin R. Multivalued Dependencies and a New Normal Form for Relational Databases // ACM TODS. - 1977. - №3.

16. Попов Ф.А., Данилов Д.В. О подходах к созданию геоинформационной системы масштаба предприятия // Ползуновский вестник. - Барнаул: Изд-во АлтГТУ, 2004. - №3. - С.39-41.

17. Попов Ф.А. Проблемы интеллектуализации пользовательских интерфейсов информационных систем // Ползуновский вестник. - Барнаул: Изд-во АлтГТУ, 2004.-№3. - С.99-103.

18. Попов Ф.А., Овечкин Б.П., Максимов А.В. Проблемы и принципы построения пользовательских интерфейсов информационных систем //Известия АГУ. - N1/2000. - С.57-60.