



УДК 004.4

© 2012 г. **А.Н. Родионов**, д-р техн. наук
(ВЦ ДВО РАН, Хабаровск)

КАЧЕСТВО ДАТАЛОГИЧЕСКИХ СХЕМ. ПРИНЦИП КОМПАКТНОСТИ МОДЕЛЕЙ ДАННЫХ И ЕГО ПРИЛОЖЕНИЯ. I

К диаграммам баз данных предъявляется множество в чем-то дополняющих, в чем-то противоречащих друг друга требований. В то же время обеспечение максимальной компактности структур моделей данных считается фундаментальным требованием, поскольку формирует каркас всей проектируемой или модернизируемой даталогической конструкции. Проблема компактности – это проблема инвариантной альтернативности получаемых схем. В статье выделяются специфические классы категорий сущностей, непосредственно являющихся источником обозначенной проблемы. Разрабатываются методы и алгоритмы получения компактных схем, в основе которых лежат предварительно унифицированные и заведомо эффективные структурные фрагменты

Ключевые слова: компактность моделей данных, сложный сущностный тип, даталогическая модель, типы структур моделей данных, ключевые атрибуты отношений, ограничения целостности, абстракции обобщения, структурное ядро кластерного блока.

Введение

Если обратиться к основополагающим принципам организации баз данных, то принцип “компактности данных” в гораздо большей степени, чем все остальные: полноты, минимальной избыточности, адекватности и адаптивности, предопределяет устройство ее структурной части (модель данных).

При проектировании схем отношений – будущих таблиц базы данных – важно, чтобы сами отношения впоследствии, по мере заполнения их данными, не представляли собой сильно разреженные матрицы. Набор отношений модели данных должен быть запроектирован таким образом, чтобы сумма “пустых” атрибутов всех отношений была минимальной. К сожалению, ни один из используемых сегодня методов построения даталогических схем (нормализации отношений, семантического моделирования, абстрагирования) не только не решает поставленную задачу, но и, что может показаться несколько странным и удивительным, не предназначен для ее решения.

По существу, истоки обозначенной проблемы лежат в плоскости корректного начального распределения множества элементов предметной области между группами, именуемыми типами сущностей. Критерии, на которые следовало бы

ориентироваться в процессе проведения подобного рода распределения, до сих пор не проработаны. Существуют серьезные трудности их построения, одна из которых носит системный характер и связана с отсутствием математической модели, увязывающей взаимное влияния многочисленных переменных и параметров, отражающих этот процесс. Отсюда задание типов сущностей превращается в некоторую эвристическую процедуру, формализованную на сегодняшний день лишь частично.

Настоящая работа – это попытка заложить теоретические основы для процедур синтеза типов сущностей. Многие используемые здесь положения взяты из ранее написанных работ [1 – 4] и представляют собой своеобразный аксиоматический базис. Всякий раз, когда в этом возникает необходимость, в статье делаются ссылки на эти положения. Работа логически построена следующим образом.

Первая часть посвящена истокам и содержанию проблемы максимальной компактности. Демонстрируются многовариантные пути ее решения. Для такого типа сущности как “Оборудование для речных и морских судов”, рассматриваемого в качестве сквозного примера на протяжении всей работы, синтезируются абсолютно идентичные компактные даталогические подсхемы без заведомо явных преимуществ одной над другой. Выдвигается гипотеза о существовании сложных сущностных кластеров, сочетающих и объединяющих в себе подтипы, претендующие на роли самостоятельных и самодостаточных типов.

Во второй части вводятся и анализируются дополнительные условия сравнимости альтернативных даталогических схем. Для того, чтобы можно было воспользоваться расширенным набором критериев сравнения эффективности моделей данных, формальные рамки исследуемой системы расширяются. Вводятся критерии времени доступа к данным, размещаемым в альтернативных даталогических структурах, и средней продолжительности манипулирования данными в процессе ввода и корректировки их пользователями. Показывается, что обозначенных критериев оказывается достаточно для выбора наилучшей схемы из набора альтернативных схем.

Проблема максимальной компактности структур моделей данных

Любое отношение базы данных будет содержать элементы (ячейки), остающиеся всегда пустыми (незаполненными). Таких элементов при хорошо запроектированной схеме будет мало, в противном случае их может оказаться слишком много, что повлечет за собой необоснованный рост используемой памяти. На небольшом примере, касающемся даталогического представления характеристик такого типа сущности как “оборудование для речных и морских судов”, определимся с кругом вопросов, возникающих в процессе решения задачи обеспечения максимальной компактности сложных информационных типов. Подобного рода задачи встречаются и решаются во многих предметных областях, где возникает проблема даталогического представления переменного числа подтипов сущностей, интегрируемых в один обобщенный тип. Вот примерный и одновременно очень краткий перечень групп судового оборудования со списками общих и уникальных свойств каждой группы (табл. 1).

Таблица 1

Группы судового оборудования	Общие свойства	Уникальные свойства
Судовой двигатель	Наименование, единица измерения, вес, высота, ширина, длина	Число цилиндров, ход поршня, рабочий объем, максимальная мощность, максимальные обороты, мощность на винте и т.д.
Электродвигатели	Наименование, единица измерения, вес, высота, ширина, длина	Напряжение, сила тока, частота тока, мощность, количество оборотов в минуту и т.д.
Электронасосы	Наименование, единица измерения, вес, высота, ширина, длина	Потребляемая мощность, напор, максимальная подача, номинальный ток, напряжение, частота вращения и т.д.
Генераторы	Наименование, единица измерения, вес, высота, ширина, длина	Мощность, вид топлива, ЕДС в обмотке якоря, скорость вращения якоря, ток в якоре, ток возбуждения и т.д.
Компрессоры	Наименование, единица измерения, вес, высота, ширина, длина	Вид привода, давление, тип насоса, мощность, топливо, мощность двигателя и т.д.

Оставим в стороне проблему выбора типа сущности, предположив, что именно такой тип сущности определен, например, техническим заданием на проектирование. Требуется разработать структуру кластерного блока для типа сущности “оборудование”, удовлетворяющего одновременно двум условиям-требованиям: во-первых, условию оптимальности компактности структур, в общем виде сформулированному во введении, во-вторых, требованию обеспечения возможности одновременного отражения и точечного, и суммарного (интегрального) учета отдельных экземпляров сущности “оборудование” на всех стадиях их жизненного цикла.

В работе [1] можно найти детальное изложение процедуры конструирования кластерного блока, объекты которого “хранят” информацию об идентификаторах, постоянных свойствах и подмножествах сущностей, обобщенных в тип. Этими алгоритмами и будем в дальнейшем пользоваться.

Добиться компактности структур, “обслуживающих” отдельные экземпляры судового оборудования, можно несколькими способами. Самый простой – ограничиться использованием одной, так называемой универсальной структуры (рис. 1).

Идентификатор оборудования	Наименование оборудования	Единица измерения	Свойства				
			Мощность	Направление вращения

Рис.1. Универсальное отношение, представляющее тип сущности “оборудование”.

Очевидно, что таблица с такой структурой по мере ее заполнения будет содержать огромное количество пустых элементов.

Гораздо более рациональны две другие схемы. Одна включает множество справочных структур – по одной для каждого вида (подтипа) оборудования (рис. 2, 4). При построении другой (рис. 3) использовалась уже упомянутая про-

цедура синтеза обобщенного типа сущности. Экземплярами обобщенного типа могут быть экземпляры любых подтипов.

Остановимся далее на недостатках и достоинствах этих двух схем, для чего проведем сравнение реквизитного состава структур, представленных на упомянутых схемах (рис. 2, 3).

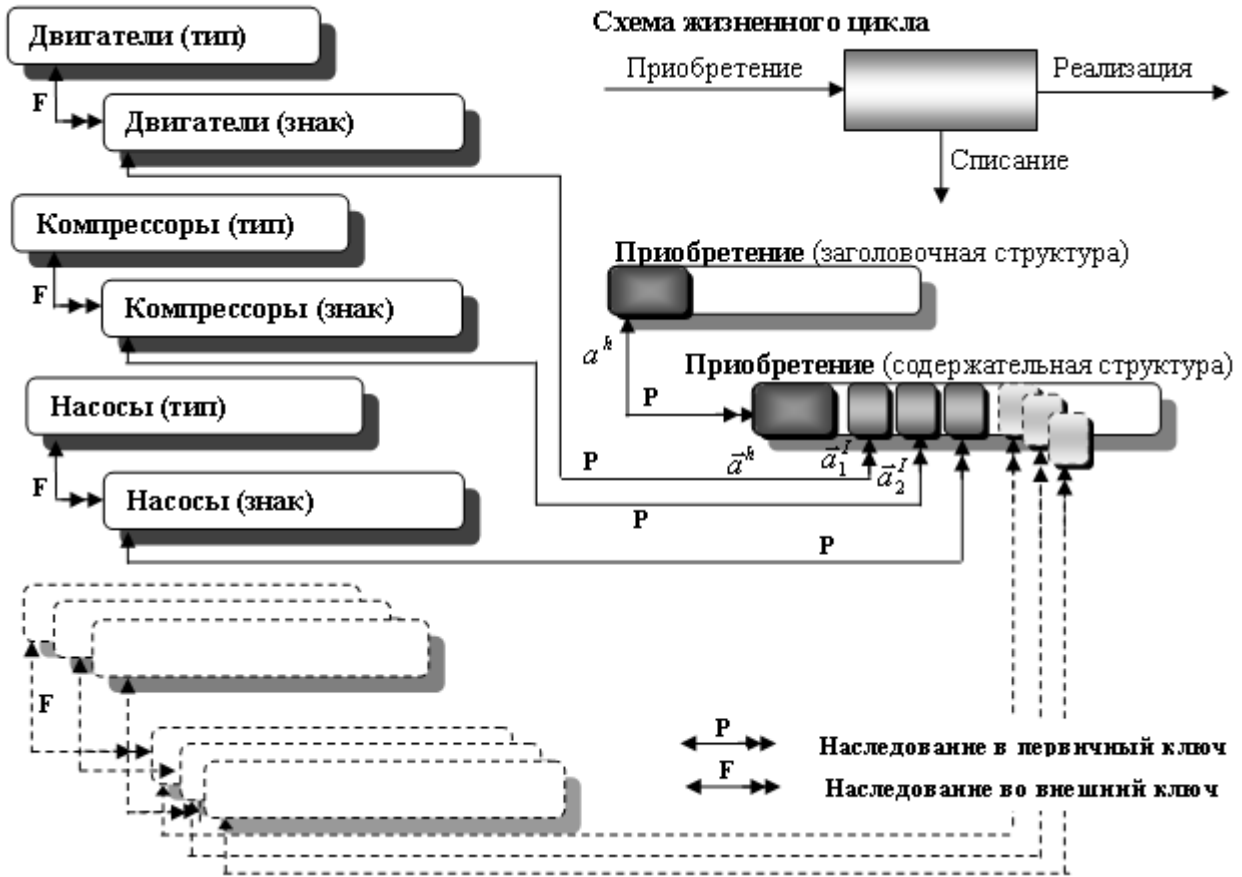


Рис. 2. “Условный” справочный кластер групп судового оборудования, не включающий мета-структуру.

Поскольку любое кластерное образование объединяет, помимо справочных, и некоторые другие разновидности структур, в том числе и крайне необходимые для представления отдельных фаз жизненного цикла документальные структуры, в рассмотрении будут задействованы все структурные компоненты кластерных образований.

Ограничимся простейшей моделью жизненного цикла, включающей стандартные фазы прихода (приобретение) и расхода (списание и реализация).

Подсхема, не задействующая мета-структуру (рис. 2), несмотря на свою простоту, не отличается компактностью. Каждая документальная содержательная структура, начиная со второго уровня и выше, сопровождающая такой кластер, всегда будет включать подмножество реквизитов, значения которых будут оставаться незаполненными (пустыми). Нетрудно догадаться, что произвольная запись, ассоциируемая только с одним видом оборудования: двигателем, компрессором, насосом или чем-то иным, вследствие показанных на схеме (рис. 2) особенностей наследования простых ключевых реквизитов других структур в процессе формирования собственного составного ключа, будет идентифицироваться

только по двум из них: \mathbf{a}^h – наследуемого ключевого атрибута заголовочной структуры и \mathbf{a}_c^l – одного из наследуемых ключевых атрибутов справочных структур: $\mathbf{a}_c^l \in \mathbf{A}^l = \{\mathbf{a}_1^l, \mathbf{a}_2^l, \dots, \mathbf{a}_c^l, \mathbf{a}_c^l\}$. Все остальные атрибуты множества \mathbf{A}^l , вошедшие в состав первичного ключа, оказываются при этом незадействованными.

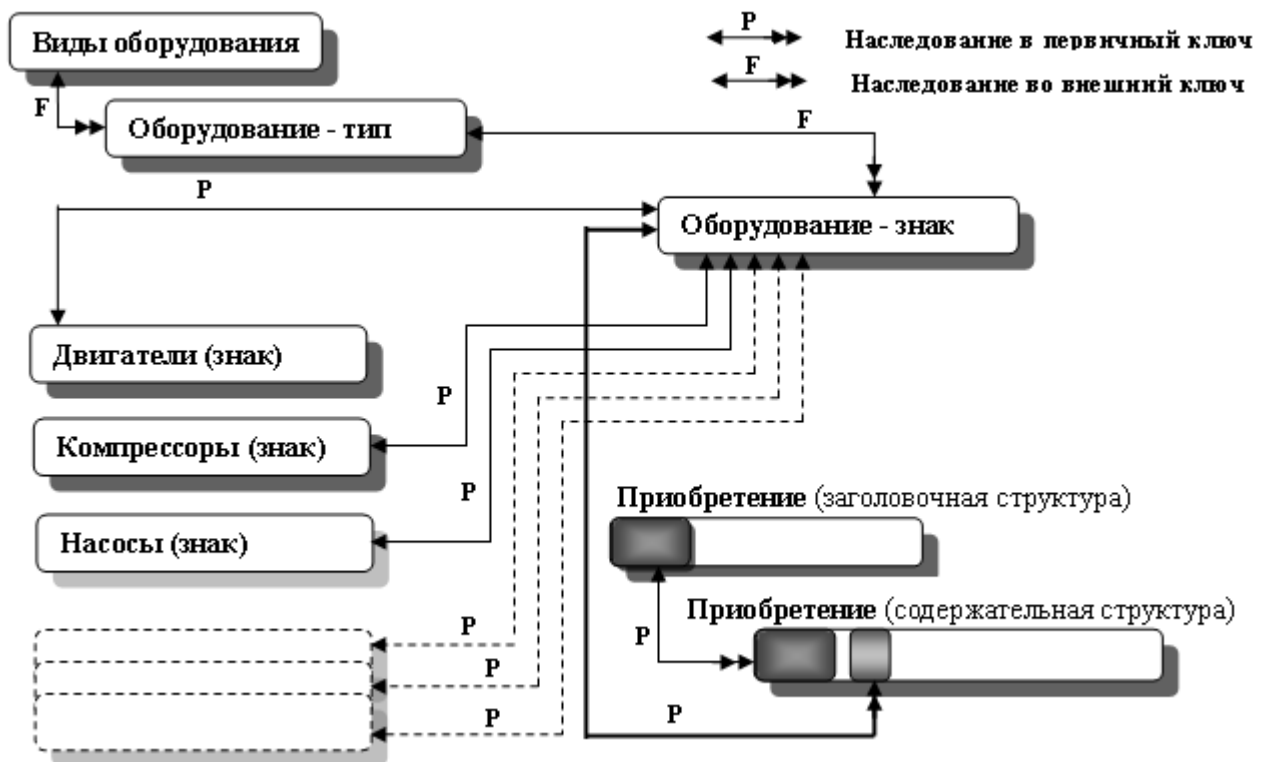


Рис. 3. Схема компактного кластерного блока для представления судового оборудования.

Такого недостатка лишена альтернативная подсьема, в которую включены “детализирующие” таблицы. “Детализирующие таблицы”, связанные с головной таблицей типом связей “1:1 – один к одному”, содержат наборы атрибутов, отражающих только уникальные свойства, присущие определенному виду оборудования. Очевидно, что ввиду наличия подмножества уникальных свойств у каждого из подвидов оборудования, детализирующих таблиц будет ровно столько, сколько групп оборудования из моделируемой предметной области требуется представить в схеме организации данных.

Остановимся далее на функциях, которые выполняют отдельные группы структур в таком кластере.

Два базовых объекта: оборудование-тип и оборудование-знак образуют ядро кластерного блока. В состав оборудования-типа входят: атрибут, идентифицирующий тип сущности, и атрибуты, соответствующие постоянным свойствам моделируемого типа сущности. Включение в кластер объекта-типа предоставляет также возможность организовать суммарный (интегральный) учет экземпляров сущностей, обладающих одинаковыми значениями постоянных свойств.

К стандартному набору идентифицирующих атрибутов другого объекта – “оборудование-знак” – добавлены атрибуты, соответствующие свойствам, которые являются одновременно и общими, и актуальными уже для отдельных экземпляров сущностей. К таким свойствам применительно к рассматриваемому типу

сущности могут быть отнесены: норма амортизационных отчислений, балансовая стоимость, месяц-год выпуска, инвентарный номер, вес. Понятно, что этот список не является завершенным и может быть продолжен.

Но кроме общих, как уже было замечено, свойств определенные подгруппы (подтипы) оборудования характеризуются наборами собственных, уникальных свойств. Эти уникальные свойства и представляются соответствующими атрибутными наборами, входящими в состав детализирующих таблиц.

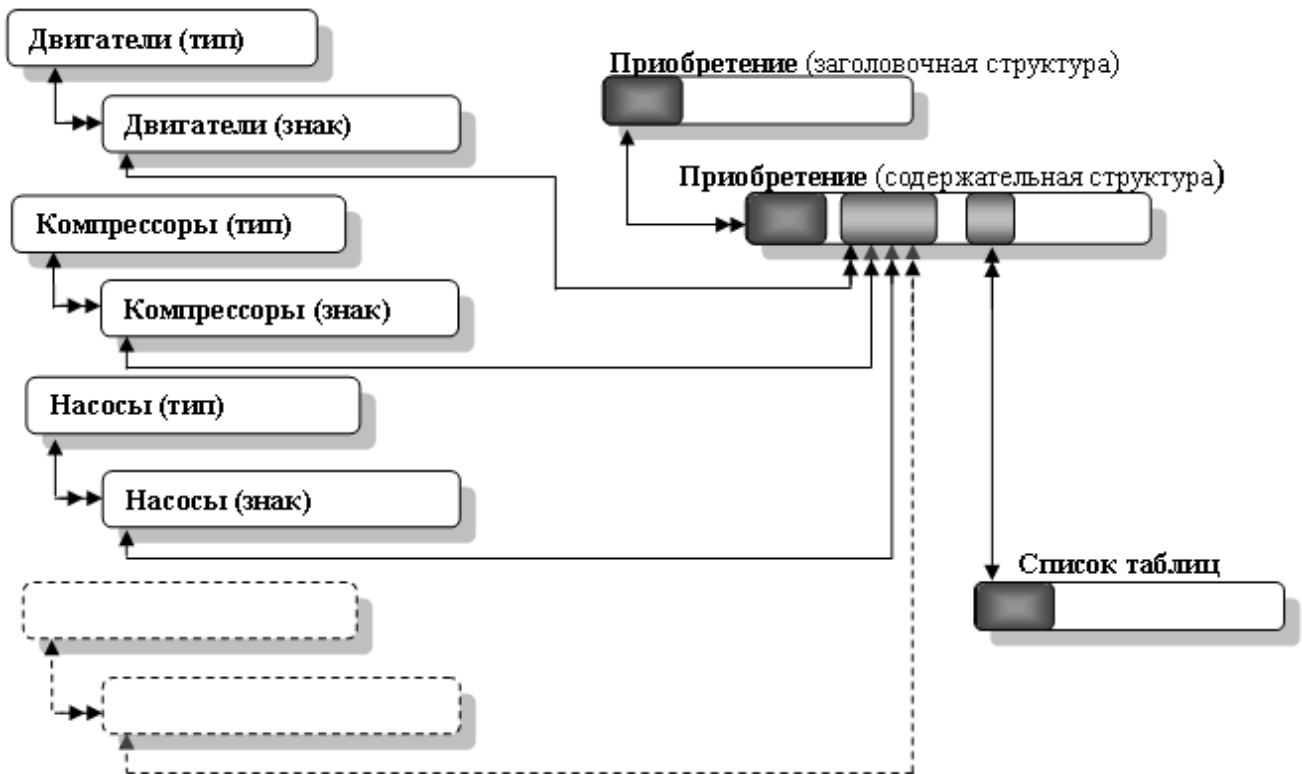


Рис. 4. “Условный” справочный кластер групп судового оборудования, содержащий мета-структуру “Список таблиц”.

Показанная на рис. 3 подсьема компактна уже хотя бы потому, что в ней устранена избыточность реквизитного состава первичных документальных содержательных структур. Ввиду того, что каждый экземпляр оборудования имеет уникальный идентификатор, являющийся одновременно первичным простым суррогатным ключом таблицы “Оборудование-знак”, достаточно только его единственного наследования с включением в состав составного первичного ключа документальной структуры для отражения, например, факта приобретения оборудования.

В состав подсьемы (рис. 3) включена также структура, поименованная “Виды оборудования”. Это в соответствии с классификацией, приведенной в [1], – объект-группировка, которая здесь, помимо своей главной классификационной функции – распределения экземпляров объектов между непересекающимися подмножествами, – призвана обеспечить автоматический доступ к требуемой детализирующей таблице в момент ввода данных. Отметим, что новое поведение (новая роль) группировочного объекта влечет за собой и необходимость в расширении набора стандартных методов каскадных модификаций, реализуемых со-

временными системами управления базами данных (СУБД).

Несмотря на очевидную выигрешность в плане компактности данной под-схемы, и предыдущее решение имеет серьезный потенциал для своего улучшения. Покажем, после каких модернизаций под-схема, показанная на рис. 2, может оказаться сопоставимой “по компактности” с только что рассмотренной.

Для этого добавим в схему-структуру “Список таблиц”, экземплярами которой станут идентификаторы и названия справочных таблиц, включенных в кластер “Оборудование”. Задействование подобной структуры (рис. 4) устраняет основной недостаток такой схемы – переменное число атрибутов составного первичного ключа содержательной документальной структуры и, как следствие, разреженность столбцов, соответствующих этим атрибутам, возникающая в процессе заполнения их данными.

При этом возникает ситуация, которая ранее в теории реляционного моделирования данных не рассматривалась, а именно – одновременное наследование нескольких атрибутов первичных ключей справочных таблиц в атрибут внешнего ключа содержательной документальной таблицы.

В общем-то очевидно, что связи между таблицами, интерпретируемыми как ссылочные целостности, формально представляют собой частный случай инъективных функций, заданных на одних и тех же множествах, когда каждый элемент множества может отображаться только сам на себя.

В нашем же случае речь идет о семействе функций j , заданных также на одном множестве, но выбор конкретной функций, которая будет представлять экземпляр связи во взаимодействии между справочной и документальной таблицами, ставится в зависимость от значения, принимаемого другой функцией – q -функцией, которая также задана на одном-единственном множестве. Только элементы этого множества являются идентификаторами записей структуры “Список таблиц”.

В этой связи для формализации сказанного уместно воспользоваться функционалом J , посредством которого можно будет осуществлять выбор соответствующей функции, принимающей участие в реализации связи в процессе поддержки процедуры множественного наследования: $J : q(X) \rightarrow f$.

Вследствие всего только что сказанного необходимо каким-то образом различать в дальнейшем однозначное наследование первичного ключа одной, справочной таблицы во внешний ключ другой, документальной таблицы и *множественное наследование*, в котором одновременно может принимать участие один из первичных ключей только одной из нескольких потенциальных таблиц – претендентов на наследование.

Несмотря на то, что подобная разновидность ссылочной целостности современными СУБД не обеспечивается, реализовать ее прикладными программными средствами не составляет большого труда.

Таким образом, как нетрудно заметить, следование принципу компактности в процессе проектирования модели данных может привести к появлению как минимум двух альтернативных даталогических схем: с однокластерной и многокластерной организацией для одного и того же типа сущности.

Дополнительные условия сравнимости альтернативных схем

Знакомство с вопросами организации компактных структур моделей данных позволяет более точно, чем это было сделано ранее, сформулировать проблему максимальной компактности. Она сводится к выбору из множества альтернативных компактных схем, наилучшим образом удовлетворяющей критерию компактности. Только что было показано, что таких базовых схем может существовать как минимум две. Одна синтезируется в предположении наличия множества типов сущностей и последующего их объединения в один кластерный блок. В другой, напротив, выделяется один тип сущности, который в последующем распадается на подтипы.

Если обратиться к истокам (мотивам) структуризации информации с целью ее размещения в базах данных автоматизированных информационных систем, то именно подразумеваемый принцип компактности лежал в основе всех предложенных в то время декомпозиционных алгоритмов. Достаточно сослаться на аргументацию, объясняющую необходимость в проведении декомпозиции так называемого универсального отношения, которую можно найти в работе [5]. Одноединственное отношение можно использовать для хранения информации обо всех сущностях моделируемой предметной области и их свойствах. Здесь, правда, следует сделать одно небольшое уточнение. Посредством универсального отношения можно отразить статическое состояние сущностей предметной области. Если нужно фиксировать еще и динамику взаимодействий сущностей, то потребуется как минимум еще одно универсальное отношение.

Но, как было показано в предыдущем разделе, строгое следование этому принципу приводит к появлению альтернативных компактных подсхем, что, естественно, ставит задачу выбора одной из них. В рамках формальной системы под названием “модель данных” сделать правильный выбор не представляется возможным ввиду очевидного отсутствия соответствующих критериев (по крайней мере нам таковые неизвестны). Отсюда вытекает необходимость в расширении формальных границ исследуемой системы, в которую модель данных входила бы на правах одного из компонентов. В этом случае можно попытаться задействовать набор каких-то других характеристик для проведения сравнения альтернативных даталогических конструкций. Речь здесь идет о системе базы данных или системе баз данных.

Ограничимся в качестве показателей сравнения двумя характеристиками:

продолжительностью выполнения запросов на считывание информации из табличных структур сущностного кластера с последующим размещением выходных данных во вспомогательных таблицах среды выполнения;

временем, необходимым для ручного ввода данных во вспомогательные таблицы, с возможной последующей корректировкой этих данных.

Справедливости ради стоит сказать, что в действительности таких характеристик несколько больше. Это, например, группа показателей, отражающих надежность функционирования системы баз данных, трудоемкость и стоимость разработки программного обеспечения и некоторые другие. Но все перечисленные характеристики актуальны на этапах проектирования информационных систем

(ИС) и теряют свою значимость в процессе эксплуатации последних, так как приближаются к своим предельным величинам к моменту начала эксплуатации ИС.

Руководствуясь сказанным, обратимся к стандартному алгоритму, реализующему последовательности запросов на считывание данных из документальных структур, организованных в соответствии с требованиями компактности (рис. 3 и 4) и оценим суммарное время выполнения этих запросов.

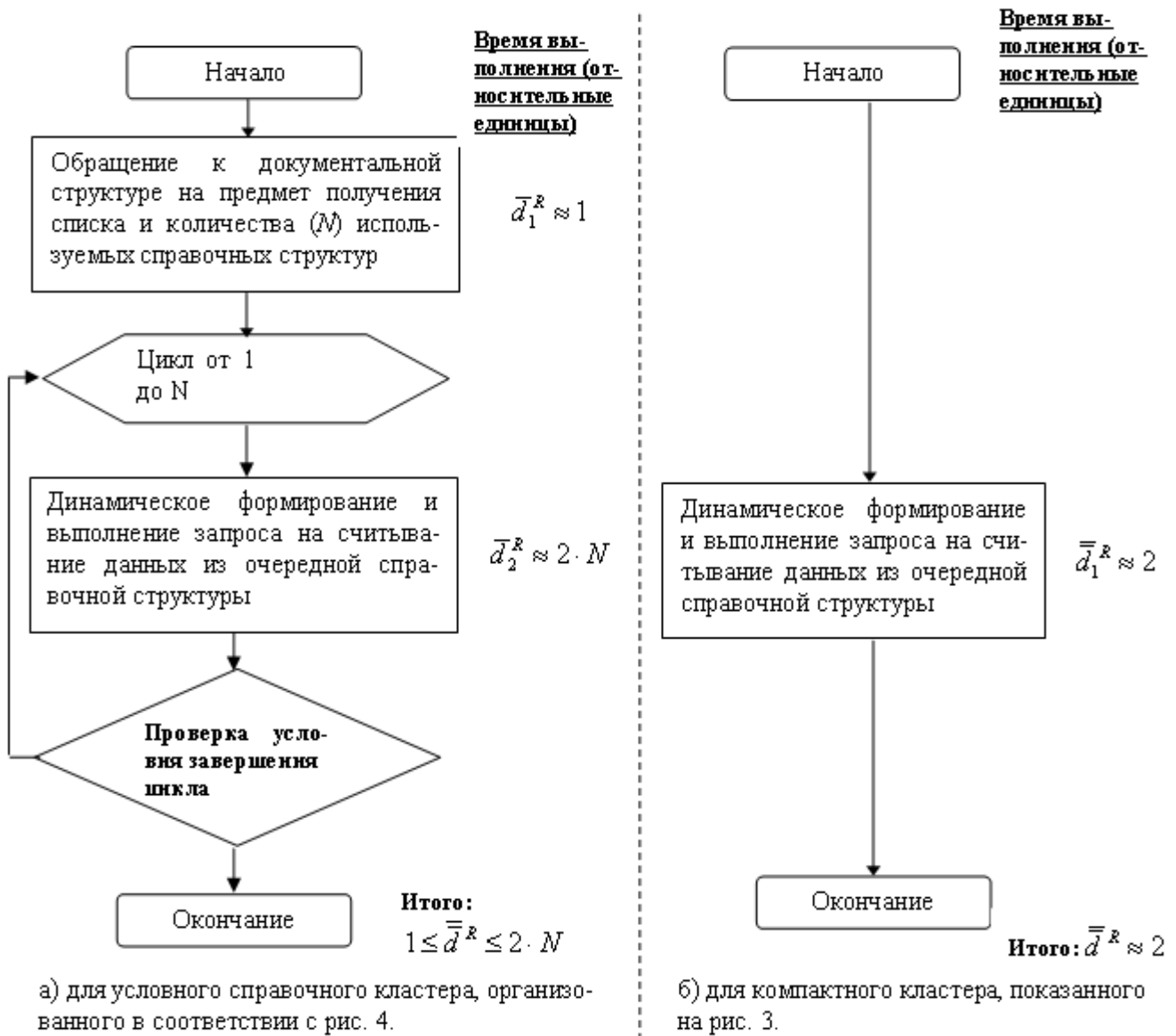


Рис. 5. Алгоритм считывания данных из документальных структур с примерной оценкой времени его выполнения.

Ввиду переменного числа “справочников”, которые будут принимать участие в запросе (рис. 5а), на первом шаге требуется определиться с их количеством и названиями, а затем, организовав соответствующий цикл, приступить к динамическому формированию запросов, каждый из которых будет производить считывание данных из “своего” справочника.

Альтернативный вариант (рис. 5б) намного проще, так как предусматривает обращение только к одной справочной структуре, что, естественно, заставляет считывающий алгоритм выполняться намного быстрее.

Что касается записи данных, то в этом случае времена \bar{d}^w и $\bar{\bar{d}}^w$ соответственно для первого и второго вариантов оказываются сопоставимыми: $\bar{d}^w \approx \bar{\bar{d}}^w$.

Намного сложнее, чем считывание данных, спроектировать удобную для пользователя интерфейсную форму, которая будет использована для ввода информации. Применительно к рассматриваемой предметной области пользователь при, например, вводе данных о поступлении оборудования, организованных в соответствии со схемой, показанной на рис. 4, должен предварительно решить, к какому виду принадлежит то или иное оборудование, и только после этого выбрать соответствующий справочник.

Вот, например, как может выглядеть интерфейсная форма (рис. 6), реализующая выполнение процедур ввода данных в документальные структуры, представленные на рис. 4.

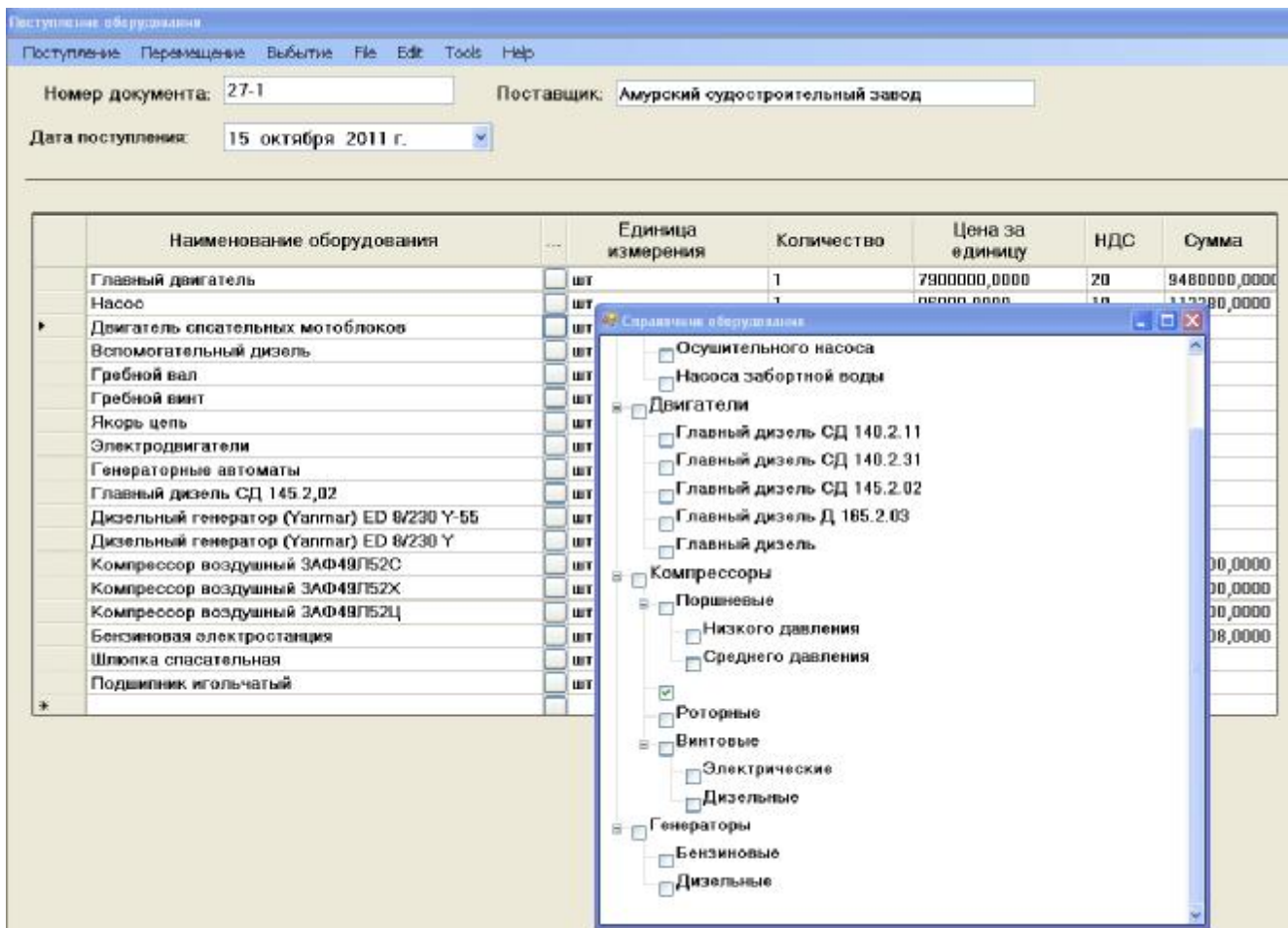


Рис. 6. Интерфейсная форма для ввода данных, извлекаемых из предварительно формируемого “иерархического” справочника.

Каждая строка содержательной таблицы снабжена кнопкой, запускающей на выполнение метод, сначала “формирующий” список справочных таблиц кластера, а затем выводящий его на экран. Далее пользователь должен выбрать нужный ему справочник и получить доступ к его записям для выбора соответствующей позиции оборудования.

Справедливости ради следует сказать, что современные программные классы, управляемые данными, в частности классы, ориентированные на иерархиче-

ское представление данных, несколько упрощают и унифицируют описанный процесс.

Очевидно, что необходимость в дополнительных манипуляциях, связанных с предварительным определением типа справочной структуры, как это было предложено делать в предыдущих случаях, отпадает, если информация в базе данных организована по другой альтернативной схеме (рис. 3). При этом трудоемкость выполнения операций по вводу данных в содержательную документальную структуру стремится к минимуму.

Все только что представленные соображения, касающиеся трудоемкости ввода данных и времени реализации запросов на получение “нужной” информации, хранящейся в базе данных, говорят в пользу существенных кластеров, обладающих сложной сетевой организацией (рис. 3). Попутно заметим, что такая организация имеет ряд преимуществ и при проведении адаптационных мероприятий, часто выполняемых над схемами организации данных. Но сложнейшие аспекты трансформации уже задействованных в моделях данных структур выходят за рамки настоящей статьи.

Заключение

Следование требованию компактности структур моделей данных в процессе проектирования последних может приводить к необходимости синтеза сложных даталогических конструкций. Такие конструкции представляют собой систему взаимодействующих таблиц, именуемую кластером и соответствующую одному из типов сущностей моделируемой предметной области.

В работе показано, что для одного и того же сложного типа сущности могут быть получены как минимум три подсхемы, две из которых оказываются альтернативными в плане их компактности.

Образующие кластер структуры, как и связи между ними, типизированы и соответствуют стандартным типам, перечисленным в [1, 2], за исключением “детализирующих” объектов и “множественного наследования”, ввод которых в рассмотрение, расширяя категоричный аппарат даталогического моделирования, способствует достижению абсолютной компактности.

Включение в состав кластеров “детализирующих” объектов позволяет также частично раскрыть природу и область применения такого типа связи как “*один к одному (1:1)*”. Этот тип связи крайне редко применяется в практике проектирования моделей данных из-за отсутствия рекомендаций по его использованию, что в целом является прямым следствием недостаточной теоретической проработки данного вопроса.

Наличие альтернативных даталогических конструкций заставляет расширять формальные границы моделируемой системы, чтобы сформулировать дополнительные условия сравнимости альтернативных схем. Расчет и сопоставление значений, принимаемых вводимыми критериями, отражающих соответствующие условия, свидетельствуют о приоритете схем с односущностной организацией (рис. 3).

ЛИТЕРАТУРА

1. Родионов А.Н. Моделирование данных: от сущностей к структурам данных. Сущности и объекты // Вестник ХГАЭП. – 2010. – №1 (46). – С. 43-61.
2. Родионов А.Н. Моделирование данных: от сущностей к структурам данных. Структуры концептуальной модели // Вестник ХГАЭП. – 2010. – №6 (51). – С. 49-68.
3. Родионов А. Н. “Мигрирующие” объекты моделей данных: ”слабые сущности” и документы // Вестник ХГАЭП. – 2010. – №1 (52). – С. 40-65.
4. Родионов А. Н. Критерии качества даталогических схем. Полнота моделей данных // Вестник ХГАЭП. – 2010. – №2 (53). – С. 28-51.
5. Джексон Г. Проектирование реляционных баз данных для использования с микроЭВМ. – М.: Мир, 1991

Статья представлена к публикации членом редколлегии А.Д. Плутенко.

E-mail:

Родионов Александр Николаевич. – ran@newmail.ru.

ХII Международная конференция

**«Устойчивость и колебания нелинейных систем управления» (конференция Пятницкого)
Москва, ИПУ РАН, 5 - 8 июня 2012 г.**

ТЕМАТИКА КОНФЕРЕНЦИИ

Общие вопросы теории устойчивости и стабилизации движения.
Общие вопросы и методы теории нелинейных колебаний.
Методы функций Ляпунова для нелинейных систем управления и метод Гамильтона-Якоби-Ляпунова-Беллмана в теории оптимального управления и в игровых задачах управления.
Гладкая и негладкая динамика.
Проблемы управляемости и наблюдаемости систем управления.
Проблемы робастного управления.
Управление механическими системами.
Устойчивость и управление гибридными системами и системами с переключениями.
Прикладные задачи управления и компьютерные методы.

ОРГАНИЗАТОРЫ

Российская академия наук.
Отделение энергетики, машиностроения, механики и процессов управления РАН.
Научный совет по теории управляемых процессов и автоматизации РАН.
Учреждение Российской академии наук Институт проблем управления им. В. А. Трапезникова РАН

ТЕКУЩАЯ ИНФОРМАЦИЯ

Подробная информация о конференции (онлайн регистрация, правила оформления тезисов докладов и т.п.) - <http://www.stab12.ru>

ЯЗЫКИ КОНФЕРЕНЦИИ: русский, английский

АДРЕС ОРГКОМИТЕТА

117997 Москва, ул. Профсоюзная, 65, Институт проблем управления РАН,
Оргкомитет ХII конференции по устойчивости.
Телефон: +7(495)334-93-69, +7(495)334-91-30,
факс: +7(495)334-93-69
E-mail: stab@stab12.ru,
Председатель оргкомитета: Валентин Николаевич Тхай, д.ф.-м.н., проф.
Ученый секретарь: Мария Александровна Муницына, к.ф.-м.н.