ГОСТ 34.321—96

МЕЖГОСУДАРСТВЕННЫЙ СТАНДАРТ

Информационные технологии
Система стандартов по базам данных

ЭТАЛОННАЯ МОДЕЛЬ УПРАВЛЕНИЯ ДАННЫМИ

Издание официальное

БЗ 3—2001

МЕЖГОСУДАРСТВЕННЫЙ СОВЕТ
ПО СТАНДАРТИЗАЦИИ, МЕТРОЛОГИИ И СЕРТИФИКАЦИИ
Минск

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН Институтом программных систем НАН Украины

ВНЕСЕН Государственным комитетом Украины по стандартизации, метрологии и сертификации

2 ПРИНЯТ Межгосударственным Советом по стандартизации, метрологии и сертификации (протокол № 10 от 3 октября 1996 г.)

За принятие проголосовали:

Наименование государства	Наименование национального органа по стандартизации
Азербайджанская Республика	Азгосстандарт
Республика Армения	Армгосстандарт
Республика Беларусь	Госстандарт Республики Беларусь
Республика Казахстан	Госстандарт Республики Казахстан
Кыргызская Республика	Кыргызстандарт
Республика Молдова	Молдовастандарт
Российская Федерация	Госстандарт России
Республика Таджикистан	Таджикгосстандарт
Туркменистан	Главгосинспекция «Туркменстандартлары»
Республика Узбекистан	Узгосстандарт
Украина	Госстандарт Украины

3 Настоящий стандарт соответствует международному стандарту ISO/IEC 10032:1995 «Information technology — Reference model of data management»

4 Постановлением Государственного комитета Российской Федерации по стандартизации и метрологии 22 февраля 2001 г. № 88-ст межгосударственный стандарт ГОСТ 34.321—96 введен в действие непосредственно в качестве государственного стандарта Российской Федерации с 1 июля 2001 г.

5 ВВЕДЕН ВПЕРВЫЕ

© ИПК Издательство стандартов, 2001

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания на территории Российской Федерации без разрешения Госстандарта России

Содержание

1 Область применения . . . 1
2 Термины и определения . . . 1
3 Графические представления . . . 3
4 Требования к управлению данными . . . 5
4.1 Информационные системы . . . 5
4.2 База данных и схема . . . 5
4.3 Средство моделирования данных . . . 5
4.4 Независимость данных . . . 5
4.5 Процессоры и интерфейсы . . . 5
4.6 Управление доступом . . . 5
4.7 Поддержка управления данными . . . 6
4.7.1 Управление конфигурацией, управление версиями и варианты . . . 6
4.7.2 Параллельная обработка . . . 7
4.7.3 Управление транзакцией базы данных . . . 7
4.7.4 Проектирование производительности . . . 7
4.7.5 Идентификация объектов . . . 7
4.7.6 Расширение средства моделирования данных . . . 7
4.7.7 Поддержка для различных средств моделирования данных в интерфейсе пользователя . . . 7
4.7.8 Контрольные журналы . . . 7
4.7.9 Восстановление . . . 8
4.7.10 Логическое реструктурирование данных. Реорганизация физической памяти . . . 8
4.8 Дополнительные эксплуатационные требования для поддержки управления данными в распределенной информационной системе . . . 8
4.8.1 Управление распределением . . . 8
4.8.2 Управление транзакцией базы данных . . . 9
4.8.3 Связи . . . 9
4.8.4 Экспорт—импорт . . . 9
4.8.5 Независимость распределения . . . 9
5 Пары уровней и связанных процессов . . . 9
5.1 Пары уровней . . . 9
5.1.1 Блокирование пар уровней . . . 10
5.2 Зависимость пар уровней от средства моделирования данных . . . 11
5.2.1 Пары уровней и правила структурирования данных . . . 11
5.2.2 Пары уровней и правила манипулирования данными . . . 12
5.3 Пары уровней и связанные с ними процессы . . . 12
5.4 Многократное связывание . . . 13
5.5 Управление доступом для пар уровней . . . 13
5.6 Модификация схемы . . . 13
6 Архитектурная модель . . . 14
6.1 Понятия моделирования . . . 14
6.2 Общая модель управления данными . . . 14
6.3 Специализация модели в различных средах . . . 15
6.4 Среда базы данных . . . 15
6.5 Управление распределенными данными . . . 15
6.6 Модель экспорта—импорта . . . 17
6.7 Модель управления доступом . . . 17
7 Стандартизация управления данными . . . 18
7.1 Цели стандартизации управления данными . . . 18
7.2 Средства достижения целей стандартизации управления данными . . . 19
7.3 Аспекты стандартизации управления данными . . . 20
Приложение А Алфавитный указатель терминов на английском языке . . . 22

МЕЖГОСУДАРСТВЕННЫЙ СТАНДАРТ

Информационные технологии
Система стандартов по базам данных

ЭТАЛОННАЯ МОДЕЛЬ УПРАВЛЕНИЯ ДАННЫМИ

Information technology.
Database standards system.
Reference model of data management

Дата введения 2001—07—01

1 Область применения

Настоящий стандарт устанавливает эталонную модель управления данными.

Эталонная модель определяет общую терминологию и понятия, относящиеся к данным информационных систем. Такие понятия используются для определения услуг, предоставляемых системами управления базами данных или системами словарей данных.

Эталонная модель не рассматривает протоколы для управления данными.

Область применения эталонной модели включает процессы, которые касаются управления постоянными данными и их взаимодействия с процессами, отличающимися от требований конкретной информационной системы, а также общие услуги управления данными, для определения, хранения, поиска, обновления, ввода, копирования, восстановления и передачи данных.

2 Термины и определения

В настоящем стандарте применяют следующие термины с соответствующими определениями, предназначенные для использования в эталонной модели.

В скобках в качестве справочных приведены эквиваленты терминов на английском языке. Алфавитный указатель используемых в стандарте терминов на английском языке приведен в приложении А.

2.1 **база данных** (database): Совокупность взаимосвязанных данных, организованных в соответствии со схемой базы данных таким образом, чтобы с ними мог работать пользователь.

2.2 **вариант** (variant): Конфигурация всей информационной системы или ее части, которая сосуществует с другой системой, имеющей другую конфигурацию, но обеспечивающей те же средства.

2.3 **версия** (version): Конфигурация всей информационной системы или ее части, существующая в конкретный момент времени.

2.4 **вертикальная фрагментация** (vertical fragmentation): Назначение экземпляров различных частей одного типа в две или более среды базы данных.

2.5 **временные данные** (transient data): Данные, которые поступают в информационную систему или исключаются из нее при выполнении одной или нескольких транзакций.

2.6 **горизонтальная фрагментация** (horizontal fragmentation): Назначение различных множеств или различных типов экземпляров данных в две или более среды баз данных.

2.7 **данные** (data): Информация, представленная в формализованном виде, пригодном для передачи, интерпретации или обработки с участием человека или автоматическими средствами.

2.8 **данные распределения** (distribution data): Данные, которые определяют информацию о размещении, дублировании и фрагментации объектов данных в распределенной системе баз данных.

2.9 **данные управления доступом** (access control data): Данные, связанные с определением или модификацией привилегий управления доступом.

Издание официальное

2.10 **домен управления** (management domain): Область, охватывающая множество двух или более информационных систем, каждая из которых может быть распределенной и которые спроектированы и сконструированы для обмена данными и процессами.

2.11 **дополнительное средство значения** (added value facility): Средство, обеспеченное процессором в дополнение к средствам, требуемым средством моделирования данных.

2.12 **интерфейс** (interface): Определенный набор услуг, предоставляемых процессором.

2.13 **информационная система** (information system): Система, которая организует хранение и манипулирование информацией о предметной области.

2.14 **исходная схема** (source schema): Определение данных или множество определений данных до их преобразования в схему.

2.15 **клиент** (client): Пользователь, запрашивающий услуги, обеспечиваемые интерфейсом сервера.

2.16 **коммутационное соединение** (communications linkage): Средства для обмена данными между компьютерными системами или между пользователем и компьютерными системами.

2.17 **контроллер базы данных** (database controller): Абстрактное представление для набора услуг, которые согласованы с конкретным средством моделирования данных и реализуют его.

2.18 **контрольный журнал** (audit trail): Журнал, в котором процессы функционирования фиксируются в информационной системе.

2.19 **конфигурация** (configuration): Совокупность процессов информационной системы и способ, которым эти процессы взаимосвязываются.

2.20 **механизм управления доступом** (access control mechanism): Механизм, который может использоваться для осуществления защиты информации от несанкционированного доступа.

2.21 **независимость данных** (data independence): Независимость объектов данных от процессов, состоящая в том, чтобы объекты данных могли быть изменены без нарушения процессов.

2.22 **объект данных** (data object): Любое понятие или предмет, связанное с данными.

2.23 **ограничение целостности** (constraint): Ограничения на значения определенного набора объектов данных.

2.24 **определение данных** (data definition): Описание правил, которым должны подчиняться один или более наборов экземпляров данных.

2.25 **пара уровней** (level pair): Два смежных уровня данных, более высокий из которых всегда содержит тип «информации», соответствующей «экземплярам» на более низком уровне.

2.26 **постоянные данные** (persistent data): Данные, которые постоянно сохраняются в информационной системе в течение всего процесса обработки данных.

2.27 **правило манипулирования данными** (data manipulation rule): Правило, которому необходимо следовать, когда процесс определяется, или которому автоматически следует система управления данными, когда процесс выполняется.

2.28 **правило ограничения целостности** (constraining rule): Правило, являющееся частью средства моделирования данных и контролирующее спецификацию ограничений целостности, которые могут быть наложены на определенный набор объектов данных.

2.29 **правило структурирования данных** (data structuring rule): Правило, определяющее, как может быть структурирован набор экземпляров данных.

2.30 **привилегия** (privilege): Разрешение на использование определенной услуги управления данными для доступа к объекту данных, предоставляемое идентифицированному пользователю.

2.31 **прикладная задача** (application task): Задача, инициированная пользователем и требующая для своего решения обработки информации.

2.32 **прикладная система** (application system): Система, предназначенная для решения прикладных задач.

2.33 **прикладная схема** (application schema): Описание экземпляров данных, которые в любое время могут находиться в прикладной базе данных.

2.34 **прикладной процесс** (application process): Процесс, который является определенным в соответствии с требованиями конкретной информационной системы.

2.35 **приложение** (application): Операции по управлению и обработке данных, которые относятся к конкретным требованиям информационной системы.

2.36 **процесс** (process): Активный компонент информационной системы.

2.37 **процесс манипулирования данными** (data manipulation process): Процесс, семантика которого предписывается правилами манипулирования данными средства моделирования данных.

2.38 **процессор** (processor): Объект, обеспечивающий конкретное применение определенной совокупности команд.

2.39 **распределенная база данных** (distributed database): База данных, которая физически распределяется на две или более компьютерные системы.

2.40 **распределенная информационная система** (distributed information system): Информационная система, объекты данных и/или процессы которой физически распределяются на две или более компьютерные системы.

2.41 **санкционирование** (authorization): Определение привилегий для конкретного идентифицированного пользователя.

2.42 **связывание** (binding): Установление отношений между конкретными определениями данных и процессом.

2.43 **связь клиент-сервер** (client-server relationship): Связь, устанавливаемая в момент, когда клиент запрашивает услугу, которая должна выполняться сервером.

2.44 **сеанс** (session): Определенный период времени, в течение которого клиент может много раз взаимодействовать с сервером, причем и клиент, и сервер поддерживают данные друг о друге.

2.45 **сервер** (server): Процессор, предоставляющий услуги другому процессору.

2.46 **система словарей** (dictionary system): Информационная система, содержащая информацию об одной или более прикладных системах.

2.47 **система управления базами данных** (database management system): Совокупность программных и языковых средств, обеспечивающих управление базами данных.

2.48 **среда базы данных** (database environment): Совокупность, состоящая из базы данных, связанной с ней схемы базы данных и контроллера базы данных.

2.49 **среда управления данными** (data management environment): Совокупность данных и связанных с ними элементов обработки, объединенных в компьютерной системе.

2.50 **средство моделирования данных** (data modelling facility): Совокупность правил для определения схемы и манипулирования данными, хранимыми в соответствии со схемой.

2.51 **стандарт интерфейса** (interface standard): Стандарт, который определяет виды услуг, доступных в интерфейсе для обработки.

2.52 **стандарт обмена** (interchange standard): Стандарт, который определяет множество объектов данных в соответствии с правилами структурирования данных.

2.53 **схема базы данных** (database schema): Формальное описание данных в соответствии с конкретной схемой данных.

2.54 **схема данных** (data schema): Логическое представление организации данных.

2.55 **тип данных** (data type): Поименованная совокупность данных с общими свойствами.

2.56 **транзакция** (transaction): Совокупность связанных между собой операций, характеризуемых четырьмя свойствами: атомарность, непротиворечивость, локализация и продолжительность.

2.57 **управление базами данных** (database management): Процесс определения, создания, ведения баз данных, а также манипулирования ими.

2.58 **услуга** (service): Предоставление функциональных возможностей одного процессора другим процессорам.

2.59 **фрагментация (**fragmentation): Назначение экземпляров данных базы данных в две или более среды базы данных.

2.60 **функциональный стандарт** (functional standard): Стандарт, который состоит из собрания других стандартов, согласованных между собой.

2.61 **целостность данных** (data integrity): Соответствие значений всех данных базы данных определенному непротиворечивому набору правил.

3 Графические представления

В разделе идентифицированы графические символы, используемые в эталонной модели.

Использование символа для указания постоянных данных (рисунок 1) предназначено, чтобы охватить все виды носителей, на которые могут быть записаны данные. Имя идентифицирует тип содержания, записанного как постоянные данные.

Символ коммутационного соединения (рисунок 2) используется в диаграммах как конкретная форма связи между компьютерными системами.

Символ соединения обработки (рисунок 3) используется в диаграммах между процессом и постоянными данными или между процессами, чтобы указать поток данных.



Рисунок 1



Рисунок 2

Рисунок 3

Символ класса процессов (рисунок 4) используется, чтобы показать процесс манипулирования данными. Соединение обработки на левом ребре указывает вход, на правом ребре — выход и на вершине — ограничение.

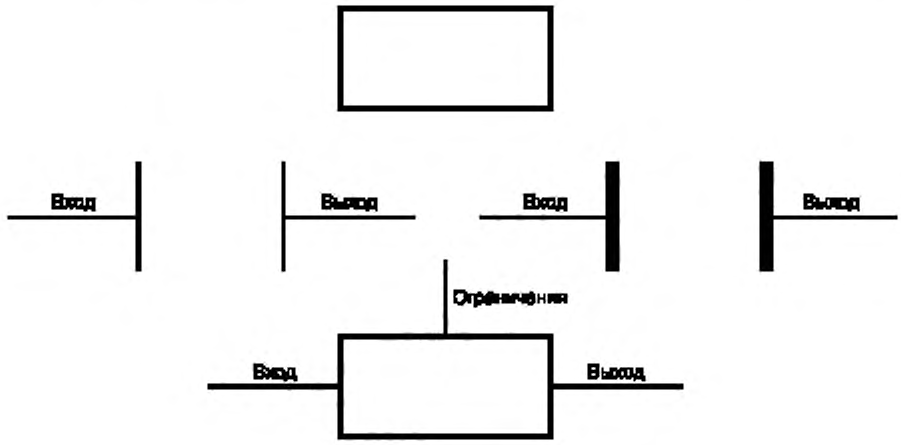


Рисунок 4

Символ класса процессоров приведен на рисунке 5.

Символ класса процессоров с интерфейсом услуг (рисунок 6) используется в диаграммах с символом соединения обработки, чтобы указывать взаимодействия, в которых он участвует как клиент и как сервер. Каждое соединение, обрабатываемое сервером, связывается с заштрихованным интерфейсом услуг.



Рисунок 5



Рисунок 6

4 Требования к управлению данными

4.1 Информационные системы

Информационная система — это система, которая организует процессы сбора, хранения и обработки информации о проблемной области. Она может быть размещена на одной или нескольких компьютерных системах. Если информационная система размещена на нескольких компьютерных системах, то она будет рассматриваться как распределенная информационная система.

Данные поступают в информационную систему и исключаются из нее, и эти взаимодействия могут осуществляться или людьми, или процессами.

Управление данными в настоящем стандарте будет касаться организации и управления постоянными данными. Постоянные данные — это данные, которые хранятся в информационной системе в течение определенного периода времени. Система, которая выполняет функцию организации и управления постоянными данными, называется системой управления данными.

4.2 База данных и схема

Постоянные данные в среде базы данных заключают в себе схему и базу данных. Схема — это описания содержания, структуры и ограничения целостности, используемые для создания базы данных. База данных — это набор постоянных данных, определенных с помощью схемы.

Система управления данными использует определения данных в схеме, чтобы предоставлять возможность доступа и управлять доступом к данным в базе данных.

4.3 Средство моделирования данных

Схему разрабатывают в соответствии с совокупностью правил структурирования данных. Каждая совокупность правил структурирования данных может иметь связанную с ней совокупность правил манипулирования данными, определяющую процессы, которые могут быть выполнены над структурированными данными.

Правила структурирования данных и правила манипулирования данными — это средства моделирования данных.

Язык баз данных используется, чтобы определить схему согласно правилам структурирования данных и процессы в соответствии со связанными с ними правилами манипулирования данных.

Примерами классов средства моделирования данных являются реляционный, сетевой и иерархический классы. Правила структурирования данных для двух средств моделирования данных в различных классах могут быть похожими, как, например, для сетевого и реляционного, но связанные с ними средства манипулирования данными могут отличаться.

4.4 Независимость данных

Независимость данных — это независимость процессов от объектов данных, которая состоит в том, что объекты данных могут быть изменены без нарушения процессов.

Независимость данных, как правило, достигается тремя способами.

Первый способ состоит в связывании процесса со схемой таким образом, что процесс знает только ту часть схемы, а именно прикладную схему, которая необходима процессу управления данными.

Второй способ — это обеспечение независимости прикладных процессов от физического представления данных.

Третий способ — это включение как можно большего количества ограничений целостности в схему, а не в прикладные процессы.

4.5 Процессоры и интерфейсы

Процесс управления данными может быть вызван пользователем, процессами управления данными или другими процессами. Процессы выполняются процессорами, каждый из которых имеет интерфейс. Интерфейс процессора должен быть точно определен. Такие интерфейсы могут быть независимыми от стандартного языка программирования, используемого для определения процесса с использованием интерфейса.

В любом интерфейсе существуют факторы, о которых пользователь должен знать, чтобы иметь возможность использовать основной процессор. Эти факторы должны быть сведены к минимуму, чтобы обеспечить как можно большую независимость в интерфейсе.

4.6 Управление доступом

Управление доступом — это предотвращение несанкционированного использования ресурса, включая предотвращение использования ресурса несанкционированным образом.

Для управления данными задание управления доступом состоит в разрешении санкционированного доступа к данным и предотвращении несанкционированного доступа. Такое управление доступом определяет процессы, которые может выполнять пользователь.

В любой организационной ситуации существуют требования к управлению доступом, которые могут быть выражены в терминах стратегии безопасности. Стратегия безопасности устанавливает, какую форму доступа требует каждый пользователь информационной системы. Информационная система должна иметь соответствующие механизмы управления доступом для проведения в жизнь стратегии безопасности.

Управление доступом должно основываться на принципе идентичности человека и процесса.

Требования управления доступом в контексте управления данными должны быть следующими:

- определять и впоследствии модифицировать привилегии управления доступом;
- проводить в жизнь в любое время ограничения управления доступом, которые применимы в том же месте в то же время.

Для определения привилегий требуются определенные средства. Процесс выделения привилегий пользователям называется санкционированием. Глобальные полномочия даются тому, кто должен создать или модифицировать другие привилегии управления доступом в среде управления данными.

Привилегии могут определяться в терминах идентификатора пользователя, ограничениями на использование информационной системы, баз данных, схем, типов данных, времени и размещения, а также используя их комбинации.

Может потребоваться дополнительная информация, такая, например, как идентификатор пользователя, который санкционирует привилегию.

Данные, описывающие привилегии, называют данными по управлению доступом. Эти данные должны храниться и управляться точно так же, как и любые другие данные в области управления данными.

Решение позволить любой конкретный доступ к данным основывается на привилегиях пользователя.

Проведение в жизнь управления доступом требует, чтобы пользователи и процессы, выполняющие роль пользователей, были идентифицированы и чтобы законность запроса на использование этого процесса доступа к требуемым данным могла быть проконтролирована в момент выполнения.

4.7 Поддержка управления данными

Требования, накладываемые информационными системами на управление данными, которые не зависят от конкретных требований информационной системы хранения и манипулирования данными, следующие:

- поддержка жизненного цикла информационных систем;
- управление конфигурацией информационных систем, управление версиями и варианты;
- параллельная обработка;
- управление транзакциями базы данных;
- проектирование производительности;
- идентификация объектов данных;
- расширение средства моделирования данных;
- поддержка для различных средств моделирования данных в интерфейсе пользователя;
- контрольные журналы;
- восстановление распределенной базы данных;
- реструктуризация логических данных;
- реорганизация физической памяти.

Управление данными обеспечивает обобщенные средства удовлетворения этих требований так, чтобы не было необходимости разрабатывать конкретные решения для каждой информационной системы.

4.7.1 Управление конфигурацией, управление версиями и варианты

Деятельность по управлению изменениями, осуществляемыми в конфигурации информационной системы за какой-то период времени, называется управлением конфигурацией. Следует идентифицировать дискретные версии системной конфигурации в конкретные моменты времени, а также продолжать следить за конфигурацией, которая принадлежит каждой конкретной версии.

Когда информационная система находится в некоторых фазах жизненного цикла, то для параллельного существования в различных формах могут потребоваться постоянные данные и процессы, которые являются частью информационной системы.

Две формы процесса могут считаться различными вариантами. Это означает, что каждый вариант удовлетворяет различным требованиям (таким, как различие внутренних представлений памяти) и ни один вариант не предназначен для того, чтобы изменять другой.

4.7.2 Параллельная обработка

Информационная система является ресурсом, который может быть распределен между несколькими пользователями одновременно. Пользователь может инициировать запрос на услуги системы управления данными, которыми можно управлять более целесообразно, если доступ к данным может быть сделан одновременно. Среда управления данными должна гарантировать выполнение отдельного намерения каждого пользователя таким образом, чтобы он согласовывался с его восприятием данных.

Параллельные взаимодействия не должны влиять друг на друга, а параллельная обработка не должна влиять на целостность данных.

4.7.3 Управление транзакцией базы данных

Транзакция базы данных определяется как ограниченная последовательность взаимодействий базы данных, которые вместе образовывают логическую единицу работы. В случаях обновления базы данных транзакция базы данных является последовательностью шагов обновления, которые изменяют содержание базы данных из одного непротиворечивого состояния в другое.

Требования к управлению транзакциями базы данных следующие:

- следствия всех изменений должны оставаться в базе данных после завершения транзакции базы данных или ни одни из них не остаются;
- после завершения работы транзакция базы данных оставляет базу данных в непротиворечивом состоянии;
- изменения, осуществленные транзакцией базы данных, должны быть невидимы для любой другой параллельной транзакции базы данных и наоборот;
- заблокированная один раз система должна гарантировать, что результаты транзакции базы данных переживают любые последующие отказы.

Параллельное выполнение нескольких транзакций базы данных должно быть эквивалентным в том смысле, что выполнение их параллельно является таким же самым, как если бы они выполнялись последовательно.

4.7.4 Проектирование производительности

Необходимо создавать возможности для улучшения производительности любой информационной системы: прикладной системы, системы словарей или системы, в которой интегрируются обе системы.

Основой получения таких улучшений является накопление статических данных о частоте использования процессов и частоте доступа и изменений в объектах данных.

4.7.5 Идентификация объектов

Каждый объект в среде базы данных должен быть уникальным. Это может быть достигнуто или путем присвоения каждому объекту уникального имени с использованием вложенной иерархии пространств имен, или при помощи другого механизма.

Для того чтобы дать уникальное имя в среде базы данных, может потребоваться, чтобы имена были определены с помощью имен во внешнем пространстве имен.

Именем может быть имя, назначенное пользователем или системой управления данными.

Требование именования существует для прикладных систем, систем словарей других типов информационной системы. Если имеется более чем одна среда базы данных в компьютерной системе, тогда требуется, чтобы одна среда была отличима от другой.

4.7.6 Расширение средства моделирования данных

Средство моделирования данных может быть типовым для систем управления данными. Одновременно может возникнуть требование добавлять типы данных и связанные с ними процессы.

Примером этого требования является полная текстовая обработка в соединении с обработкой структурированных данных типичным средством моделирования данных.

4.7.7 Поддержка для различных средств моделирования данных в интерфейсе пользователя

Следует иметь возможность отображать данные в формате, предпочитаемом системой управления данными, и формате, предпочитаемом пользователем. Это требование связано с тем, что пользователь может предпочесть манипулировать данными в соответствии со средством моделирования данных, отличным от средства, обеспеченного системой управления данными.

4.7.8 Контрольные журналы

Необходимо обеспечить возможность сохранять записи об успешных изменениях в данных в базе данных и в некоторых случаях — запись о транзакциях, которые запрашивают данные и генерируют отчеты. Эта запись может включать соответствующие значения данных, подробности

транзакции и идентификацию пользователя. Эти контрольные журналы могут быть определены, как требуемые для всех данных в базе данных, избранных типов данных или экземпляров определяемых данных.

4.7.9 Восстановление

База данных должна иметь возможность возвратиться к предшествующему непротиворечивому состоянию. Это требование может возникнуть из-за ошибочных транзакций, системного сбоя или потери хранимых данных. Чтобы удовлетворить эти требования, могут использоваться различные механизмы, такие как запись всех изменений, сделанных в базе данных, и сохранение резервных копий всей базы данных или ее части.

Модифицированные данные, которые распределяются в более чем одной базе данных, должны быть восстановлены таким образом, чтобы конечный результат имел непротиворечивое состояние и состояние базы данных было бы непротиворечивым.

4.7.10 Логическое реструктурирование данных. Реорганизация физической памяти

Логическое реструктурирование данных определяется как процесс изменения определения данных после того, как информационная система использовалась в течение некоторого времени. Изменение может быть дополнением к существующему определению данных или может заключать в себе модификацию части существующего определения данных.

Реорганизация физической памяти определяется как процесс изменения представления постоянных данных на носителе данных.

4.8 Дополнительные эксплуатационные требования для поддержки управления данными в распределенной информационной системе

В распределенной информационной системе объекты, принадлежащие одной информационной системе, распределяются на два или более компьютера. Когда распределяемые объекты являются объектами базы данных, система является распределенной системой баз данных.

Запрашиваемая услуга может быть доступна из множества вычислительных устройств, вмещающих дублированные данные.

Эксплуатационные требования, зависящие от распределяемых данных, следующие:

- управление распределением;
- управление транзакцией базы данных;
- связь;
- экспорт/импорт;
- независимость распределения.

Некоторые из этих требований также применимы к информационной системе, которая включает более чем одну среду базы данных в единственной компьютерной системе.

Необходимо поддерживать среду других возможностей:

а) распределенную систему базы данных, в которой составные среды базы данных проектируются таким образом, что возможно взаимодействие между любой парой;

б) систему баз данных, в которой две или более отдельно спроектированные системы баз данных объединяются, в определенном смысле, после периода раздельного использования и создаются для функционирования как одна распределенная система баз данных;

в) ситуацию, в которой каждая среда базы данных согласуется множеством стандартов и, следовательно, может взаимодействовать (возможно, на специальной основе) с другими средами баз данных, каждая из которых была спроектирована отдельно, но согласно тем же самым стандартам.

4.8.1 Управление распределением

Управление распределением включает управление фрагментацией, управление дублированием и автономию месторасположения.

Могут использоваться такие способы распределения данных:

а) назначить все экземпляры определенного типа на одну среду базы данных (нефрагментированный способ);

б) назначить множества экземпляров данных (возможно различных типов) на две или более среды баз данных (горизонтальная фрагментация);

в) назначить экземпляры различных частей того же самого типа на две или более среды баз данных (вертикальная фрагментация);

г) комбинация пунктов б) и в) (комбинированная горизонтальная и вертикальная фрагментация).

Горизонтальная фрагментация дает возможность записывать на вычислительном устройстве только экземпляры данных, которые относятся к этому вычислительному устройству.

Вертикальная фрагментация дает возможность записывать на конкретном вычислительном устройстве только экземпляры данных, которые относятся к нему.

Если фрагментация поддерживается в распределенной среде, то не требуется, чтобы пользователь информационной системы знал, как данные фрагментируются или распределяются между компьютерными системами.

По причинам производительности или защиты от сбоя компьютерной системы необходимо обеспечить копию всей базы данных или ее части. Такие дублированные данные могут храниться в компьютерной системе, отличной от той, в которой данные первоначально создаются и в дальнейшем управляются. Требование для фрагментации может быть объединено с требованием дублирования так, чтобы копии множества фрагментов назначались на две или более среды баз данных. Информация о том, какие объекты, в какой среде данных являются доступными, должна быть доступна (прямо или непрямо) в каждой среде.

Необходимо иметь возможность управлять содержанием точных копий, когда данные обновляются. Алгоритмы, которые обеспечивают контроль точных копий, должны также гарантировать обновления в транзакциях.

Требования для дублирования данных на различных компьютерных системах должны быть адресованы в связи с требованием для компьютерных систем быть автономными, насколько это возможно. Такие требования относятся к производительности, доступности данных в течение сбоя связи и к административным вопросам, таким как учет системных ресурсов и идентифицирование пользователей.

4.8.2 Управление транзакцией базы данных

Необходимо синхронизировать действия локальных систем управления транзакцией, чтобы гарантировать, что изменения в распределенных данных заканчиваются непротиворечивым состоянием для каждой базы данных, а также для всех баз данных.

Обработка в одной компьютерной системе может осуществляться параллельно с обработкой в другой компьютерной системе без влияния на целостность данных в каждой из компьютерных систем.

4.8.3 Связи

Необходимо обеспечить информационным системам возможность связываться друг с другом.

Для обмена объектов данных необходимо, чтобы средство моделирования данных, в соответствии с которым объекты данных структурируются, было использовано в каждой из компьютерных систем.

Необходимо иметь средства, которые предотвращают потерю целостности баз данных из-за таких видов сбоя связи:

- сообщение может быть потеряно во время передачи;
- сообщение не может поступить в надлежащем виде из-за ошибок трансляции и ретрансляции;
- при некоторых обстоятельствах сбой связи трудно отличить от сбоя на удаленном вычислительном устройстве.

Следует определить необходимую степень дублирования данных.

4.8.4 Экспорт—импорт

Данные экспортируются из одной среды и импортируются в другую. Для этого необходимо иметь копию части или всей базы данных, с определением данных или без него. Однажды экспортируемые данные могут быть импортированы во многие другие среды, если это требуется, а также сохраняться.

4.8.5 Независимость распределения

Прикладной процесс должен иметь доступ к данным в распределенной базе данных таким образом, чтобы он не зависел от того, как могут быть распределены данные.

5 Пары уровней и связанных процессов

5.1 Пары уровней

Конструкция «пара уровней» является способом объяснения связей между базой данных и схемой. Графическое представление на рисунке 7 применяется для иллюстрации соединения базы данных с ее определением.



Рисунок 7 — Конструкция «пара уровней»

Влияние конструкции «пара уровней» состоит в том, что каждая база данных соответствует структуре данных, определенной в связанной с ней схеме. Значения данных в базе данных могут обрабатываться только процессами манипулирования данными, связанными со схемой базы данных. Схема устанавливает точную форму разрешенной обработки. Поэтому конструкция «пара уровней» иллюстрирует средство для достижения непротиворечивых операций манипулирования данными.

Представление и интерпретация значений данных зависят от схемы. Обработка не может осуществляться до тех пор, пока схема не будет определена и активна. Когда требуются изменения в схеме, тогда связанная с ней база данных должна быть таким образом изменена, чтобы поддерживать непротиворечивость.

5.1.1 Блокирование пар уровней

Конкретная схема не только определяет данные, но и сама является набором сложных объектов данных, которые должны быть созданы и защищены и могут быть модифицированы. Средства управления данными являются пригодными для управления схемами. Схема в паре уровней может быть представлена в базе данных более высокого уровня, структура данных которой может быть определена схемой более высокого уровня. Эта база данных и схема составляют другую, более высокую пару уровней. Две пары уровней могут быть «блокированы», как показано на рисунке 8.

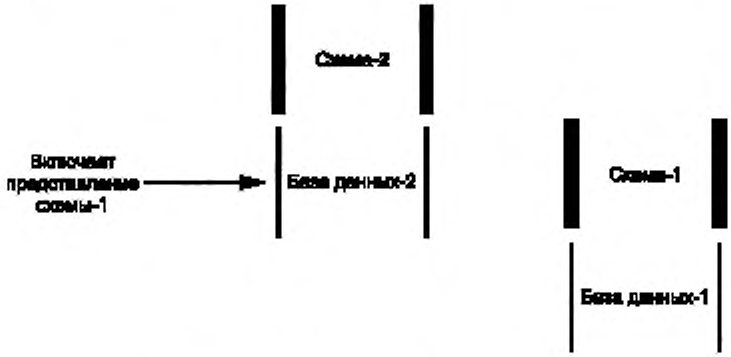


Рисунок 8 — Блокирование пары уровней

В соответствии с рисунком 8 база данных-1 согласована со схемой-1. Данные в базе данных-1 могут быть обработаны процессами манипулирования данными, которые соединены со схемой-1. База данных-2 соответствует схеме-2, и данные в базе данных-1 могут быть обработаны процессами манипулирования данными, которые соединены со схемой-2. Представление схемы-1 в базе данных-2 является исходной схемой. Исходная схема может быть выбрана из базы данных или, наоборот, обрабатываться операторами манипулирования данными точно так же, как и любые данные в базе данных.

Две пары уровней находятся на разных уровнях определения данных. Если схема-2 (рисунок 8) может иметь представление в форме экземпляров данных, записанных в базе данных, то понятие блокирования пар уровней есть рекурсивное понятие и может использоваться двумя и более парами уровней. Рекурсия останавливается, когда определение данных больше не может модифицироваться.

Обобщенное блокирование пар уровней приведено на рисунке 9.

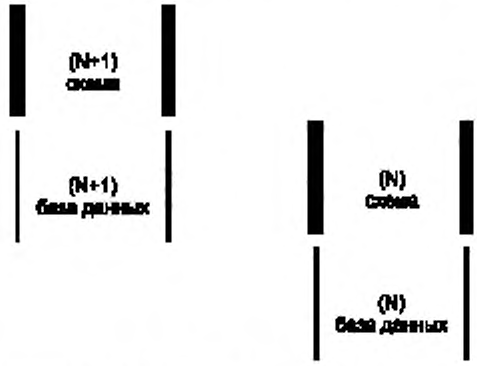


Рисунок 9 — Обобщенное блокирование пар уровней

Общие метки N и N+1 используются, чтобы показать более высокие уровни при рассмотрении общих свойств.

Блокирование пар уровней происходит с помощью связывания схемы пары уровней (N) с базой данных следующей пары уровней (N+1). Первая называется схемой (N), вторая — базой данных (N+1).

Реализация базы данных включает в себя процессы создания и поддержки определений данных. Эти определения становятся доступными для процессов манипулирования данными, затем выполняются операции выборки и модификации данных в базе данных.

Рисунок 9 иллюстрирует приведенные выше процессы следующим образом:

а) база данных (N) представляет данные, фактически предназначенные для манипулирования на уровне (N);

б) схема (N) представляет схему, способную управлять процессами для пары уровней (N). Эта схема содержит определения данных только для базы данных (N);

в) база данных (N+1) содержит определения данных, которые были созданы в течение процесса проектирования для базы данных (N) и поддерживались в течение системной операции. База данных (N+1) может также содержать другие данные, такие, например, как описания этих определений данных и проектов и описания процессов, которые используют их;

г) база данных (N+1) может содержать представления одной или более схем (N) в исходной форме. После того как одна из этих исходных схем (N) была выбрана, активизированный процесс может быть использован, чтобы конвертировать исходную схему (N) в форму, называемую объектной схемой, таким образом, чтобы могла быть заполнена связанная с ней база данных (N). Исходная схема (N) может быть активизирована более чем один раз, и каждая активизация создает отдельную объектную схему (N) со связанной с ней базой данных (N), которая может заполняться, используя процессы манипулирования данными.

Блокирование имеет пару самого низкого уровня, для которой данные на более низком уровне этой пары не содержат данные о схеме и ее компонентах и, следовательно, не могут быть активизированы. Эта пара уровней является тогда частью прикладной системы, и данные на более низком уровне этой пары уровней являются прикладной базой данных.

Блокирование также имеет пару самого высокого уровня, для которой схема на более высоком уровне пары уровней не записывается в базе данных более высокого уровня. Эта схема является тогда неявной в средстве моделирования данных, используемом системой управления данными.

5.2 Зависимость пар уровней от средства моделирования данных

Конструкция «пара уровней» и понятие средства моделирования данных тесно взаимосвязаны. Средство моделирования данных заключает в себя множество правил структурирования данных и связанное с ним множество правил манипулирования данными.

5.2.1 Пары уровней и правила структурирования данных

Средство моделирования данных включает совокупность правил структурирования данных, которые должны быть использованы для определения схемы. Эти правила включают правила для определения ограничений, которые могут быть частью схемы. Каждая схема должна быть полной и

непротиворечивой в соответствии с правилами структурирования данных и связанного с ними средства моделирования данных.

5.2.2 Пары уровней и правила манипулирования данными

Средство моделирования данных также включает правила для семантики процессов манипулирования данными. Для схемы (N) ограничения, которые основываются на правилах структурирования данных, также оказывают влияние на семантику обновления процессов манипулирования данными, выполняемых над базой данных.

5.3 Пары уровней и связанные с ними процессы

Данные в базе данных могут быть получены или модифицированы серией процессов манипулирования данными. Кроме того, если часть этих данных включает исходную схему, то можно выполнять активизированный процесс над данными.

Рисунки 10 и 11 иллюстрируют, как правила структурирования данных, используемые для определения схемы (N+1), влияют на процессы манипулирования данными в базе данных (N). Используются следующие пять процессов манипулирования данными: 1) — выбирать, 2) — активизировать, 3) — связывать с правилами манипулирования данными, 4) — связывать с правилами структурирования данных, 5) — выбирать или модифицировать. Перечисления от 1 до 5 обозначают шаги, которые объясняются на следующих рисунках. Рисунки 10 и 11 являются расширением рисунка 9 для того, чтобы показать активизацию, схемные соединения, промежуточные выборки данных, подготовку и действие других процессов манипулирования данными.

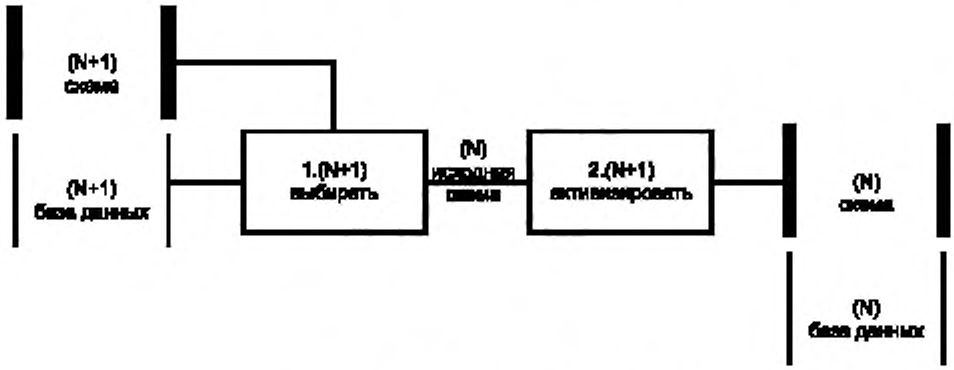


Рисунок 10 — Создание пустой базы данных

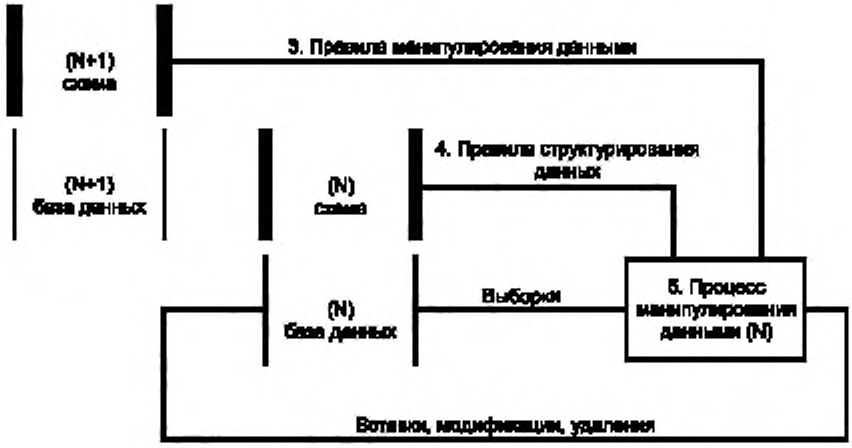


Рисунок 11 — Связывание и манипулирование данными

Следующие шаги описывают соответствующие процессы (рисунки 10 и 11) и показывают, как правила средства моделирования данных дают возможность процессу манипулирования данными (N) корректно управлять базой данных (N).

Шаг 1. Процесс манипулирования данными выбирает исходную схему (N) из базы данных (N+1), используя правила и структуры данных схемы (N+1). Представления исходной схемы (N) в базе данных (N+1) являются постоянными данными, которые могут быть модифицированы. Исходная схема (N) после выборки может иметь или не иметь форму представления постоянных данных. Например, операция выборки может просто устанавливать признак в базе данных (N+1) или может выбрать данные из базы данных (N+1) и хранить их в базе данных, отличающейся от той, в которой хранится база данных (N+1). Эта другая база данных должна также соответствовать схеме (N+1).

Шаг 2. Исходная схема активизируется, чтобы создать объектную схему и пустую базу данных (N). Осуществляется анализ, который предшествует активизации, для гарантии того, что исходная схема есть истинная схема. Такой анализ может быть выполнен в целом или частично с помощью процесса манипулирования данными (N+1), выполняющего выборку, с помощью другого процесса, который анализирует исходную схему, или в связи с активизацией.

Активизация может заканчиваться изменением признака, связанного с предварительно выбранной схемой (N) в базе данных (N+1). Альтернативно активизация может заканчиваться физическим движением данных (N+1) и изменением формы представления. В обоих случаях активизированная схема должна быть защищена от любого изменения, которое осуществлялось бы в базе данных, не соответствующей схеме.

Шаг 3. Правила манипулирования данными, которые управляют операцией процесса манипулирования данными (N), имея доступ к базе данных (N), являются теми же самыми или основанными на правилах, связанных со схемой (N+1).

Шаг 4. Каждый процесс манипулирования данными (N), который должен иметь доступ к базе данных (N), связывается с правилами средства моделирования данных в схеме (N+1). Когда уровень (N+1) является самым высоким уровнем, это связывание неявное.

Шаг 5. Для того чтобы процесс манипулирования мог иметь доступ к базе данных (N), схема (N) должна быть активной.

Шаги 1 и 2 обеспечивают создание пустой базы данных. Шаг 3 делает определенной связь между средством моделирования данных, определенным на уровне (N+1), и процессом манипулирования данными (N). Этот шаг делает явным важный атрибут блокирования пар уровней. Обе пары уровней требуются для того, чтобы процессы управления данными могли осуществляться корректно.

Связывание может быть выполнено различными способами (например, с помощью ссылки на схему или с помощью объединения схемы процесса) и в различные моменты времени (например, при выполнении или при компиляции). Сделанный выбор может влиять на время выполнения, пространство памяти и поддержку непротиворечивости процессов и схемы во время модификации последней. В противном случае выбор не влияет на результаты процессов.

5.4 Многократное связывание

Процесс манипулирования данными может быть многократно связан со схемами. С помощью этой способности конкретный объект базы данных может сделать уровень независимым, то есть к объекту может быть доступ из процессов на более чем одном уровне.

5.5 Управление доступом для пар уровней

Все данные являются субъектом для управления доступом, и в 4.6 описано, как это может выражаться в терминах привилегий, относящихся к типу данных. Идентификатор, который относится к пользователю или программе, действующей от имени пользователя, может быть независимым от уровня. Привилегия, которая относится к уровню данных (N), находится на уровне (N+1).

5.6 Модификация схемы

Необходимо иметь возможность изменять структуру базы данных. Это означает, что связанная с ней схема должна быть модифицирована. После того как схема модифицирована, данные из первоначальной базы данных необходимо корректно представить в соответствии с новой схемой.

Это можно выполнить с помощью модификации исходной схемы в базе данных (N+1), активизируя ее, чтобы создать новую схему (N) и пустую базу данных (рисунок 10), и передавая первоначальные данные в новую базу данных. Пошаговая модификация может также быть использована, чтобы удовлетворить это требование с тем же эффектом.

6 Архитектурная модель

6.1 Понятия моделирования

Архитектурная модель выражается в терминах процессоров, которые взаимодействуют как клиент и/или сервер. Эти термины используются для ссылок на процессоры, которые выполняют функции для конкретного взаимодействия.

Процессор обеспечивает услуги управления данными, которые используются другими процессорами, чтобы представить возможности информационной системы.

При взаимодействии клиент — сервер клиент делает запрос на услугу, включая любые значения данных, требуемые для этой услуги. Сервер обеспечивает один из следующих ответов:

- указание, что запрашиваемая услуга завершена;
- набор данных, который является результатом запрашиваемой услуги;
- сообщение, что услуга недоступна;
- сообщение, что запрашиваемые данные недоступны.

Каждый процессор должен определяться внешним интерфейсом, который он представляет как сервер. Этот сервер должен определять услуги и тип данных, к которым применяются услуги. Взаимодействие процессора с другими серверами относится к задачам моделирования.

Каждый процессор является экземпляром некоторого класса. Класс определяет услуги, общие для всех процессоров, которые являются экземплярами класса. Некоторые классы определяют тип данных, к которым применяются услуги, некоторые являются общецелевыми, когда экземпляр требует отдельной схемы, чтобы определить данные, к которым услуги применяются.

Процессор может быть клиентом многих серверов в любое время; несколько серверов могут поддерживать нескольких параллельных клиентов.

Для более детального описания архитектурной модели необходимо использовать:

- специализацию, где подкласс общего класса процессоров определяется как имеющий индивидуальные имя и услуги, которые являются дополнительными или модифицированными формами услуг общего процессора;
- декомпозицию, где услуги класса процессоров показаны как обеспечиваемые двумя или более классами процессоров с взаимодействиями между ними.

6.2 Общая модель управления данными

Общая модель управления данными представляет характеристики классов процессоров, общих для всего управления данными. Составляющими общей модели управления данными являются общая база данных, общая схема, контроллер общей базы данных, пользователь и процессор пользователя.

Эта модель основывается на характеристиках класса «общая база данных» и класса «общая схема».

Контроллер базы данных обеспечивает предоставление услуг управления данными для определения и доступа к классу баз данных.

Он связан с одной схемой и связанной с ней базой данных, которые вместе формируют общую среду базы данных. Каждая схема имеет имя, которое используется для идентификации среды базы данных.

Типичные услуги, обеспечиваемые контроллером базы данных, следующие:

- устанавливать сеанс управления данными для процессора-клиента, требуя явное или неявное связывание с поименованной средой базы данных;
- дополнять и модифицировать определения данных в схеме для базы данных;
- выбирать определения данных из схемы для базы данных;
- добавлять, модифицировать или удалять данные в базе данных;
- выбирать данные из базы данных;
- начинать транзакцию базы данных одного или более запросов на услугу;
- завершать транзакцию базы данных с помощью фиксирования или возврата транзакции;
- устанавливать процедуру копирования для базы данных;
- инициировать процедуры восстановления для базы данных;
- реорганизовать базу данных;
- заканчивать сеанс.

Запросы на эти услуги выражаются или операторами на языке баз данных для средства моделирования данных, поддерживаемого контроллером базы данных, или вызовами процедур, которые обеспечивают те же самые возможности.

Пользователь — это лицо или программа, которые заказывают услуги для управления данными.

Процессор пользователя — это процессор, который обеспечивает услуги управления данными для своих клиентов.

При обеспечении услугами процессор пользователя является клиентом услуг одного или более контроллеров базы данных; использование услуг любого одного контроллера базы данных требует, чтобы сеанс управления данными устанавливался между процессором пользователя и контроллером базы данных.

6.3 Специализация модели в различных средах

Общая модель может быть применена к различным видам информационных систем, описанных в разделе 4. На абстрактном уровне это применение может быть представлено замещением термина «общий» соответствующей меткой. Например, модель может быть применена к среде распределенной базы данных в терминах контроллера распределенной базы данных, распределенной базы данных и распределенной схемы. Подобным образом может быть идентифицирован контроллер словаря базы данных, а без любой метки контроллер базы данных применяется к среде базы данных компьютерной системы.

6.4 Среда базы данных

Контроллер базы данных должен поддерживать типичные услуги для среды базы данных.

Процессоры пользователя используют услуги управления данными в среде базы данных.

Услуги контроллера базы данных должны поддерживаться для многих пользовательских процессоров-клиентов параллельно, однако клиент должен иметь возможность выполнять транзакции базы данных без влияния со стороны других клиентов.

Несколько процессоров пользователя способны иметь доступ более чем к одной среде базы данных.

В этом случае процессор пользователя, который имеет доступ более чем к одной среде базы данных, должен иметь возможность управлять запросом на услугу к среде базы данных с данными, которые должны быть доступны. Любой запрос на услугу должен связываться с именем среды базы данных, для которой он предназначен. Услуги контроллера базы данных могут только поддерживаться в своей собственной среде базы данных. Любые транзакции базы данных или связи между данными, которые включают более чем одну среду базы данных, должны поддерживаться процессором пользователя.

Приведенные выше описания не рассматривают связь между средой базы данных и любой конкретной информационной системой. Разрешается процессору пользователя находиться в отличной от среды базы данных компьютерной системе. В этом случае требуются протоколы связи для соединения обработки. Эти протоколы могут быть или пассивными носителями взаимосвязи или обеспечивать специальные услуги, которые поддерживают удаленное использование услуг управления данными.

6.5 Управление распределенными данными

Декомпозиция абстрактной формы модели в среде распределенной базы данных может использоваться, чтобы показать услуги контроллера базы данных для управления данными, которые хранятся в ряде отдельных сред базы данных.

Клиент услуги на доступ к данным, которые хранятся во многих средах распределенной базы данных, не должен определять, в какой базе данных эти данные могут находиться. Таким образом, услуги контроллера базы данных для общей модели применяются также в распределенном контексте с дополнением, что услуги, относящиеся к определению распределенной базы данных, должны включать способность определять среду базы данных, в которой каждый экземпляр данных хранится. На это определение ссылаются, как на распределенные данные, которые являются данными, определяющими информацию размещения, копирования и фрагментации данных в среде распределенной базы данных.

Способ, которым данные распределяются, определяется в базе данных для распределенных данных. Подобным образом распределенные данные для распределенной базы данных могут распределяться каким-нибудь альтернативным способом.

В эталонной модели предполагается, что для одной распределенной базы данных имеется только одна среда базы данных для распределенных данных в каждой компьютерной системе.

Составляющими архитектурной модели управления распределенными данными являются администратор распределения, определитель распределения, пользователь, процессор пользователя, контроллер распределения, контроллер базы данных, схема для распределенных данных,

локальные распределенные данные, схема и база данных. Связь между составляющими архитектурной модели управления распределенными данными представлена на рисунке 12.

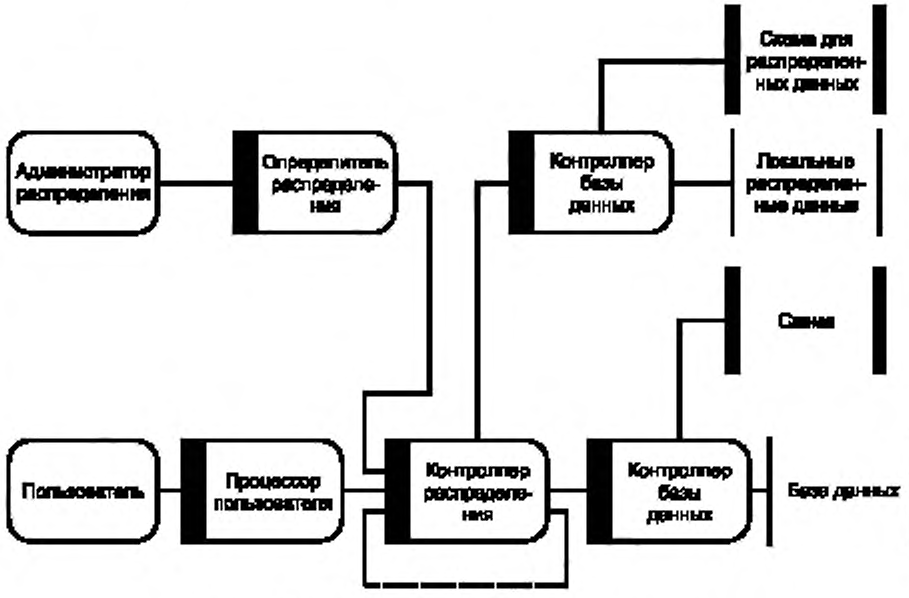


Рисунок 12 — Управление распределенными данными

Администратор распределения — это пользователь, относящийся к задаче определения распределенных данных. Определитель распределения — это процессор пользователя, обеспечивающий услуги для администрирования распределения.

Контроллер распределения — это процессор, который обеспечивает услуги управления данными для определения и доступа к распределенным базам данных вместе с услугами, относящимися к связанным с ними схемам и распределенным данным.

Контроллер распределения поддерживает только одно средство моделирования данных, которое устанавливает правила для определения и манипулирования данными в каждой среде базы данных.

Для распределенной базы данных допускается, что каждая компьютерная система имеет только один контроллер распределения.

Каждый контроллер распределения обеспечивает доступ к тем частям распределенной базы данных, которые хранятся в любой среде базы данных в той же самой компьютерной системе. Контроллер распределения поддерживает доступ к частям распределенной базы данных в удаленных средах базы данных через связь с другими контроллерами распределения в различных компьютерных системах.

Определитель распределения является только процессором пользователя для вызова услуг по доступу к распределенным данным. Другие виды процессоров пользователя не требуют никаких услуг, которые явно требуют доступа к распределенным данным, делая использование тех же самых услуг контроллера базы данных определенным для общей модели. Контроллер распределения предоставляет дополнительные услуги другим контроллерам распределения при поддержке распределенной обработки данных.

При получении первоначального запроса на услугу от процессора пользователя контроллер распределения сначала устанавливает среды базы данных, используя услуги контроллера распределенной базы данных для доступа к локальным данным распределения в той же самой компьютерной системе. Если требуемое определение недоступно как часть локальных данных распределения, то контроллер распределения связывается с другим контроллером распределения в другой компьютерной системе, чтобы найти требуемые распределенные данные.

Когда среды базы данных для ссылочных данных определены, тогда контроллер распределения может вызывать услуги любых контроллеров базы данных в той же самой компьютерной системе или связываться с удаленными контроллерами распределения, чтобы выполнить часть или всю запрашиваемую услугу.

6.6 Модель экспорта-импорта

Модель экспорта-импорта является специализированной общей моделью.

Использование услуги экспорта требует определения данных, которые должны экспортироваться из среды базы данных: экземпляры конкретного типа данных, число типов данных или данные вместе с соответствующими определениями схем. Файл должен быть поименованным. Должен быть выбран тип формата файла, соответствующий определенным данным.

Использование услуги импорта требует имени файла, в который данные были экспортированы.

Составляющими архитектурной модели экспорта-импорта являются пользователь, процессор пользователя, контроллер базы данных, процессор экспорта-импорта, схема базы данных, файл экспорта-импорта. Связь составляющих архитектурной модели экспорта-импорта представлена на рисунке 13.



Рисунок 13 — Модель экспорта-импорта

Архитектурная модель экспорта и импорта основывается на декомпозиции, которая разделяет контроллеры базы данных, обеспечивающие услуги для любых типов среды базы данных из процессора экспорта-импорта, который обеспечивает требуемые услуги. Использование услуг экспорта и импорта может быть параллельным с другими услугами контроллера распределенной базы данных.

6.7 Модель управления доступом

Модель управления доступом — это специализированная общая модель. Ее назначение — показать услуги контроллера базы для управления доступом к данным в среде базы данных и услуги для определения данных управления доступом, необходимых для этого управления доступом.

Управление доступом требует, чтобы каждый запрос на услугу контроллера базы данных был связан явно или неявно с аутентичным идентификатором. Этот идентификатор может быть неявно связан с запросом услуги с помощью декларирования при запуске или модификации в течение сеанса управления данными для клиента. Разрешение на конкретный запрос на услугу зависит от того, имел ли связанный с ним идентификатор заранее объявленные привилегии для процессов и вызванных данных.

Если соответствующие привилегии для запроса на услугу не существуют, то ответ на запрос указывает нарушение управления доступом, и это нарушение может быть записано для общего управления доступом.

Чтобы поддерживать управление доступом, услуги любого типа контроллера базы данных в общей модели должны быть соответственно модифицированы с учетом необходимости требуемых привилегий. Для большинства услуг не требуются изменения в форме запроса на услугу, но возможные ответы должны разрешать отказ из-за отсутствия привилегий.

Дополнительные услуги требуются для определения действительных идентификаторов для среды базы данных и привилегий, связанных с этими идентификаторами.

Составляющими архитектурной модели управления доступом в распределенной среде являются администратор управления доступом, определитель управления доступом, пользователь, процессор пользователя, процессор управления доступом, контроллер распределенной базы данных, схема данных управления доступом, данные управления доступом, распределенная схема, распределенная база данных. Связь между составляющими представлена на рисунке 14.

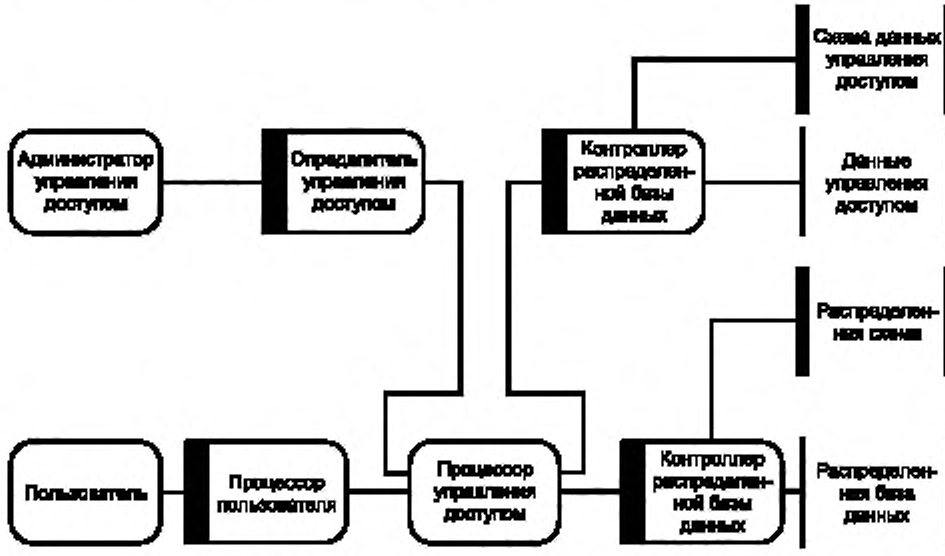


Рисунок 14 — Управление доступом в распределенной среде

Архитектурная модель управления доступом основывается на декомпозиции, которая разделяет контроллеры базы данных, предоставляющие услуги для среды базы данных без управления доступом, и процессоры, связанные с управлением доступом.

Администратор управления доступом — это пользователь, касающийся задачи определения данных управления доступом. Определитель управления доступом — это процессор пользователя, который обеспечивает услуги администраторам управления доступом.

Процессор управления доступом принимает любой запрос от идентифицированного пользователя на услугу. Это или запрос от процессора пользователя на общую услугу, или запрос от определителя управления доступом на услугу, связанную с определением данных управления доступом. Процессор управления доступом сначала использует данные управления доступом для определения, разрешен ли запрос. Тогда запрос или отклоняется, если пользователь не имеет соответствующих привилегий, или процессор управления доступом вызывает услуги контроллера базы данных, чтобы выполнить запрос. Это может заканчиваться доступом к базе данных или данным управления доступом.

7 Стандартизация управления данными

7.1 Цели стандартизации управления данными

Основными целями стандартизации управления данными являются:

а) стандартизация поддержки для всех распределенных сценариев;
б) обеспечение независимости размещения;
в) стандартизация управления транзакциями баз данных;
г) стандартизация экспорта и импорта баз данных;
д) уменьшение сложности обработки данных;
е) повышение общей производительности в распределенных сценариях;
ж) обеспечение независимости данных;
и) обеспечение мобильности приложений;
к) стандартизация использования средств моделирования данных.

Стандарты по поддержке распределенных баз данных должны основываться на интерфейсах услуг, обеспечиваемых в каждой среде управления данными.

Процессоры, связанные с интерфейсами, должны отвечать и реагировать на конкретный запрос на услугу стандартным способом.

В стандартизации поддержки распределенных баз данных необходимо учитывать следующее:

- система управления данными в одном домене управления может иметь доступ к данным, управляемым другой системой управления данными в другом домене управления, при условии появления требований в домене управления сервером;
- услуги управления данными и интерфейсы, в которых эти услуги доступны, должны быть спроектированы так, что пользователю не требуется знать, где хранятся данные;
- удаленный доступ не должен требовать принципиально нового подхода к услугам, которые обеспечивают управление данными;
- требования как для удаленного, так и локального доступа к данным должны применяться одинаково ко всем парам уровней.

Обеспечение межсетевого обмена неоднородной распределенной базы данных требует, чтобы был выбран и стандартизован соответствующий механизм для управления транзакциями базы данных.

Если данные должны быть понимаемыми в более чем одной системе управления данными, то необходима стандартная форма представления данных для передачи данных.

Необходима также стандартная форма представления архивных баз данных.

При разработке стандартных форм представления данных для передачи данных необходимо учитывать следующее:

- средства экспорта-импорта, вместе со средствами разгрузки и загрузки базы данных, полностью независимы от операций поиска и обновления;
- форму представления данных уровня схемы, связанной с данными более низкого уровня, можно рассматривать как потенциальную часть файла экспорта-импорта;
- при определении порции базы данных, для которой желателен файл экспорта-импорта, используются операции выборки, связанные со стандартными языками баз данных.

Для всех уровней данных, для которых могут создаваться файлы экспорта-импорта, может быть использован одинаковый подход к определению стандартной формы представления данных.

Независимость данных необходимо использовать для того, чтобы уменьшить затраты на модификацию и модернизацию системы.

Управление данными может быть поддержано с помощью стандартизации средства моделирования данных при его использовании для обмена между различными средами управления данными.

Выбирая способ моделирования данных в качестве стандартного, следует учитывать, что:

- пользователь услуг, обеспеченных процессором пользователя, не должен обязательно знать, какое средство моделирования данных используется как обменное средство моделирования данных;
- обменное средство моделирования данных поддерживается в различных доменах управления, если между ними возможен обмен данными.

7.2 Средства достижения целей стандартизации управления данными

Представленные в 7.1 цели стандартизации управления данными могут быть достигнуты с помощью следующих средств:

а) общее средство моделирования данных для каждой пары уровней;

б) общий механизм обмена для всех пар уровней;

в) использование одних и тех же процессоров для всех пар уровней;

г) стандартизованный подход к управлению доступом;

д) стандартизованное представление данных, необходимых для содействия взаимодействию;

е) поддержка фрагментации данных;

ж) разделение логических и физических структур;

и) доступ к схеме во время выполнения.

Система управления данными должна использовать то же стандартное средство моделирования данных для всех пар уровней и обеспечивать те же процессы манипулирования данными над данными уровня схемы, как над базой данных, управляемой схемой.

Требование непротиворечивости должно предотвращать модификацию данных уровня схемы без адекватного значения для данных, содержащихся в базе данных, управляемых схемой.

Такие требования могут адресоваться с помощью упаковки услуг манипулирования схемой, если также эффективно будет осуществляться соответствующая модификация существующих экземпляров базы данных. Услуги могут включать спецификацию преобразований в данные, чтобы достичь непротиворечивости с обновленными данными схемы.

Такие услуги модификации схемы в принципе должны быть равно применимы к любой паре уровней в области действия стандартизации управления данными.

Средства манипулирования схемой стандартизируются со ссылкой на их влияние на стандартизованное представление данных уровня схемы.

Необходимо иметь возможность модифицировать данные схемы. Определение средств манипулирования схемой должно касаться не только влияния на стандартное представление данных схемы. Необходимо также принимать во внимание влияние на связанные с ней данные, существующие в базе данных, в которой хранится схема.

Необходимо точное представление схемы в форме, соответствующей стандартному средству моделирования данных.

Процессор пользователя должен иметь возможность знать о более чем одной паре уровней, а также представлять услуги таким образом, что пользователь услуг будет рассматривать все данные как связанные с одной парой уровней.

Цель стандартизованного подхода к управлению доступом состоит в представлении возможности взаимодействия неоднородных распределенных баз данных и отдельно разработанных систем.

Для управления распределенной базой данных необходима база данных с информацией размещения фрагментов данных, которая может быть сама распределенной.

В случае централизованной системы управления данными информация размещения относится к подробностям конкретной программной среды.

Информация размещения данных должна быть глобально доступной в сети. Это требует стандартизации используемого формата для представления такой информации размещения.

Если должны поддерживаться неоднородные распределенные базы данных и их способность к взаимодействию, то требуется стандартизация распределенных данных и способа доступа к их фрагментам.

Стандартное средство моделирования данных должно обеспечивать средства структурирования такой информации. Эти средства следует использовать для обеспечения и локального, и удаленного доступов к такой информации.

Процессор распределения должен иметь возможность направлять горизонтальную и вертикальную фрагментацию в распределенных базах данных.

Система управления данными должна использовать такую информацию размещения для того, чтобы обеспечить некоторую степень независимости для прикладных процессов. Результаты запроса на информацию, которая требует данные из многих установок, должны объединяться для представления данных к требуемому приложению.

7.3 Аспекты стандартизации управления данными

Стандартизация управления данными, которые являются общими ко многим информационным системам, должна создавать удобства как для людей, использующих эти системы, так и для программных средств, которыми эти системы обеспечиваются.

Стандарты управления данными могут быть разделены на четыре группы.

Первая группа определяет вид услуг, которые обеспечиваются в интерфейсе. На стандарты этой группы ссылаются как на стандарты интерфейса. Такие стандарты не должны накладывать лишние ограничения на то, как конструировать элемент обработки, обеспечивающий услуги.

Вторая группа стандартов управления данными использует правила и соглашения средства моделирования данных, чтобы представить данные для конкретной цели. На такие стандарты ссылаются как на стандарты содержания данных. Стандарты содержания данных определяют содержание части схемы для данной пары уровней.

Третья группа стандартов управления данными — это стандарты обмена. Стандарты этой группы используются в распределенных системах и для связи между информационными системами в различных доменах управления, чтобы передавать данные из одной среды управления данными в другую. Стандарты обмена определяют физическое представление данных.

Четвертая группа — это функциональные стандарты, которые являются набором других стандартов. Функциональные стандарты могут принимать много форм, в зависимости от категорий отдельных стандартов и от зависимостей каждого от средств моделирования данных.

Понятие средства моделирования данных является фундаментальным в концепции стандартов управления данными. Оно может само быть объектом стандартизации или может быть использовано явно как средство определения в описании другого стандарта. Кроме того, правила структурирования данных и правила манипулирования данными средствами моделирования данных могут быть неявными в другом стандарте.

ПРИЛОЖЕНИЕ А

АЛФАВИТНЫЙ УКАЗАТЕЛЬ ТЕРМИНОВ НА АНГЛИЙСКОМ ЯЗЫКЕ

Access control data	2.9
Access control mechanism	2.20
Added value facility	2.11
Application	2.35
Application process	2.34
Application schema	2.33
Application system	2.32
Application task	2.31
Audit trail	2.18
Authorization	2.41
Binding	2.42
Client	2.15
Client-server relationship	2.43
Communications linkage	2.16
Constraining rule	2.28
Constraint	2.23
Configuration	2.19
Data	2.7
Data definition	2.24
Data independence	2.21
Data integrity	2.61
Data management environment	2.49
Data manipulation process	2.37
Data manipulation rule	2.27
Data modelling facility	2.50
Data object	2.22
Data schema	2.54
Data structuring rule	2.29
Data type	2.55
Database	2.1
Database controller	2.17
Database environment	2.48
Database management	2.57
Database management system	2.47
Database schema	2.53
Dictionary system	2.46
Distributed database	2.39
Distributed information system	2.40
Distribution data	2.8
Fragmentation	2.59
Functional standard	2.60
Horizontal fragmentation	2.6
Information system	2.13
Interchange standard	2.52
Interface	2.12
Interface standard	2.51
Level pair	2.25
Management domain	2.10
Persistent data	2.26
Privilege	2.30
Process	2.36
Processor	2.38

Server	2.45
Service	2.58
Session	2.44
Source schema	2.14
Transaction	2.56
Transient data	2.5
Variant	2.2
Version	2.3
Vertical fragmentation	2.4

УДК 681.324:006.354 МКС 35.100.70 35.240 П85 ОКСТУ 4002

Ключевые слова: эталонная модель, управление данными, база данных, информационная система, система словарей данных, данные, интерфейс, конфигурация, независимость данных, манипулирование данными, сервер, среда базы данных, схема данных, целостность данных

Редактор *В.П. Огурцов*
Технический редактор *О.Н. Власова*
Корректор *М.С. Кабашова*
Компьютерная верстка *О.В. Арсеевой*

Изд. лиц. № 02354 от 14.07.2000. Сдано в набор 12.04.2001. Подписано в печать 19.06.2001. Усл. печ. л. 3,26. Уч.-изд. л. 2,97.
Тираж экз. С 1296. Зак. 623.

ИПК Издательство стандартов, 107076, Москва, Колодезный пер., 14.
Набрано в Издательстве на ПЭВМ
Филиал ИПК Издательство стандартов — тип. «Московский печатник», 103062, Москва, Лялин пер., 6.
Плр № 080102

gost.e-ecolog.ru