ФЕДЕРАЛЬНОЕ АГЕНТСТВО
ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО/МЭК ТО 10032— 2007

ЭТАЛОННАЯ МОДЕЛЬ УПРАВЛЕНИЯ ДАННЫМИ

ISO/IEC TR 10032:2003
Information technology — Reference model of data management
(IDT)

Издание официальное

БЗ 1—2008/495

Москва
Стандартинформ
2009

gost.e-ecolog.ru

Предисловие

Цели и принципы стандартизации в Российской Федерации установлены Федеральным законом от 27 декабря 2002 г. № 184-ФЗ «О техническом регулировании», а правила применения национальных стандартов Российской Федерации — ГОСТ Р 1.0 — 2004 «Стандартизации в Российской Федерации. Основные положения»

Сведения о стандарте

1 ПОДГОТОВЛЕН Открытым акционерным обществом «Научно-исследовательский центр контроля и диагностики технических систем» (ОАО «НИЦ КД») на основе собственного аутентичного перевода стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК10 «Перспективные производственные технологии, менеджмент и оценка риска»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 27 декабря 2007 г. № 573-ст

4 Настоящий стандарт идентичен международному стандарту ИСО/МЭК ТО 10032:2003 «Информационная технология. Эталонная модель управления данными» (ISO/IEC TR 10032:2003 «Information technology — Reference model of data management»).

Наименование настоящего стандарта изменено относительно наименования указанного международного стандарта для приведения в соответствие с ГОСТ Р 1.5 — 2004 (подраздел 3.5)

5 ВВЕДЕН ВПЕРВЫЕ

Информация об изменениях к настоящему стандарту публикуется в ежегодно издаваемом информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячно издаваемых информационных указателях «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячно издаваемом информационном указателе «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет

© Стандартинформ, 2009

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Термины и определения 1
3 Обозначения и сокращения 5
4 Требования к управлению данными 6
5 Уровни данных и соответствующие процессы 14
6 Архитектурная модель 18
7 Цели и принципы стандартизации управления данными 26
Приложение А (справочное) Взаимосвязанные международные стандарты 32
Приложение В (справочное) Взаимосвязь действующих и разрабатываемых стандартов по базам данных с архитектурой эталонной модели управления данными 33
Библиография 39

Введение

Существует множество способов реализации систем управления данными[1)] для определения эталонной модели управления данными. Для определения функций управления данными или ссылок на них используют различные термины, для описания различных функций применяют одинаковые термины. Поэтому необходима стандартизация функций управления данными. Настоящий стандарт представляет эталонную модель управления данными и определяет области стандартизации этой модели.

Настоящий стандарт устанавливает эталонную модель управления данными, обеспечивает основу для координации разработки стандартов в области управления данными и позволяет в перспективе использовать действующие стандарты и проекты стандартов.

Термин «управление данными» включает в себя описание, создание, модификацию, использование данных и управление ими в информационных системах. Такие функции управления данными могут быть выполнены как общие услуги для прикладных информационных систем. Альтернативно каждое приложение может определять соответствующие данные и управлять ими. Если функции управления данными выполнены как общие услуги, то желательно обеспечить стандартизованные средства управления данными для совместного использования данных множеством пользователей. Такая стандартизация требует определения множества интерфейсов, для которых могут быть разработаны специальные стандарты.

Целью настоящего стандарта является обеспечение в пределах области его применения соответствующей структуры:

a) для идентификации интерфейсов;

b) для позиционирования всех таких интерфейсов относительно друг друга;

c) для идентификации средств обеспечения каждого интерфейса;

d) для идентификации процесса поддержки каждого интерфейса и, где это применимо, данных, необходимых для такой поддержки;

e) для распределения использования интерфейсов в соответствии со стадиями жизненного цикла информационных систем;

f) для идентификации связанных альтернатив, ассоциированных с соответствующим идентифицированным интерфейсом.

В настоящем стандарте выделены три главные цели стандартизации управления данными:

a) совместимость ресурсов;

b) минимизация стоимости поддержки информационной системы на всех этапах ее жизненного цикла;

c) оптимальное использование достижений стандартизации и оптимизация организации работ в этой области.

Совместимость ресурсов применима и к информационным ресурсам, например к представлению данных в базах данных, и к ресурсам различных процессоров. В настоящем стандарте особое внимание уделено совместимости информационных ресурсов, расположенных в различных местах и разработанных с использованием различных аппаратных средств и программного обеспечения. Система совместимости ресурсов – основа для управления доступом.

Снижение стоимости поддержки информационной системы относится ко всем фазам жизненного цикла информационной системы и охватывает затраты на ее проектирование, разработку, эксплуатацию и поддержку.

Оптимальное использование достижений стандартизации и оптимизация организации работ в этой области предусматривает сокращение числа необходимых стандартов и упрощение их содержания.

Настоящий стандарт идентифицирует области для разработки стандартов или постоянного улучшения их качества и обеспечивает общую структуру для комплексной поддержки и актуализации взаимосвязанных стандартов.

Настоящий стандарт обеспечивает структуру, позволяющую группам экспертов продуктивно и независимо разрабатывать стандарты для различных компонентов информационных систем.

Настоящий стандарт является основополагающим и позволяет разрабатывать новые стандарты по мере появления новых технологий.

[1)] *Данные — это информация, представленная в формализованном виде, пригодном для ее передачи, интерпретации и обработки с участием человека или автоматическими средствами (ГОСТ 34.320 — 96). Как правило, данные структурируют в виде таблицы.*

Описание эталонной модели управления данными представлено в настоящем стандарте следующим образом:

- раздел 4 содержит общие положения управления данными и требования, основанные на особенностях информационных систем;
- в разделе 5 рассмотрены основные понятия данных, которые требуются для эталонной модели, их связь друг с другом и понятием процесса;
- в разделе 6 приведена архитектурная модель, в рамках которой могут быть размещены различные данные и компоненты обработки, необходимые для управления данными;
- в разделе 7 описаны цели и принципы стандартизации управления данными;
- в приложении А приведен перечень взаимосвязанных международных стандартов;
- в приложении В показана связь действующих стандартов и проектов стандартов с архитектурной моделью.

Настоящий стандарт определяет классы услуг, ожидаемых и предоставляемых при управлении данными, и описывает структуру их взаимосвязи. Однако управление данными не является изолированным процессом и существует в средах, обеспечивающих другие услуги, такие как хранение данных и обмен ими.

Пояснения некоторых положений настоящего стандарта приведены в виде сносок, выделенных курсивом.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ЭТАЛОННАЯ МОДЕЛЬ УПРАВЛЕНИЯ ДАННЫМИ

Reference model of data management

Дата введения — 2008—09—01

1 Область применения

Настоящий стандарт устанавливает эталонную модель управления данными, разработанную ИСО, и структуру, необходимую для координации и разработки действующих и будущих стандартов в области управления постоянными данными в информационных системах. Перечень действующих стандартов в области управления данными представлен в приложении А.

Стандарт устанавливает общую терминологию и основные понятия, относящиеся ко всем данным информационных систем. Такие понятия используют для определения более специфических услуг, предоставляемых специальными компонентами управления данных, например системами управления базами данных или системами словарей данных. Определение подобных взаимосвязанных услуг идентифицирует интерфейсы, которые могут стать предметом будущих стандартов.

Настоящий стандарт не рассматривает услуги и протоколы управления данными. Стандарт не содержит требования для создания информационных систем и не служит основой для оценки соответствия.

Настоящий стандарт рассматривает процессы, которые касаются обработки постоянных данных и их взаимодействия с особыми процессами, характерными для специфических информационных систем, а также рассматривает общие услуги по управлению данными, такими как требования по определению (объявлению), хранению, извлечению, обновлению, поддержке в рабочем состоянии, копированию, восстановлению, передаче в приложениях и словарях данных.

В область применения настоящего стандарта входит рассмотрение стандартов в области управления данными, размещенными в одной или нескольких компьютерных системах, включая услуги по управлению распределенной базой данных.

Настоящий стандарт не рассматривает общие услуги, обычно предоставляемые операционной системой, т.е. процессы, связанные с конкретными типами физических устройств хранения данных, методами хранения данных, деталями коммуникаций и интерфейсом человек — компьютер.

Настоящий стандарт определяет услуги, предусмотренные в интерфейсе. Стандарт не налагает ограничений на реализацию процессов.

2 Термины и определения

В настоящем стандарте использованы следующие термины с соответствующими определениями.

Определения, установленные в настоящем разделе, предназначены для использования в настоящем стандарте. В пояснениях к термину даны более простые описания. Некоторые из нижеприведенных терминов определены в других стандартах, но они необходимы для использования в соответствующем контексте управления данными.

Издание официальное

2.1 **управление доступом** (access control): Процесс, заключающийся в предупреждении несанкционированного использования ресурсов.

Управление доступом связано с предоставлением возможности санкционированного и предупреждением несанкционированного доступа к данным. При управлении доступом должны быть определены процессы, которые может выполнять пользователь.

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

2.3 **механизм управления доступом** (access control mechanism): Механизм, предназначенный для реализации политики обеспечения и повышения безопасности.

2.4 **приложение** (application): Операции, связанные с управлением данными и их обработкой, выполняемые в соответствии с конкретными требованиями информационной системы.

2.5 **прикладной процесс** (application process): Процесс, определенный в соответствии с требованиями конкретной информационной системы.

2.6 **прикладная система** (application system): Система прикладных процессов, использующая услуги, предоставляемые интерфейсом человек — компьютер, средства коммуникаций и система управления для обработки данных, необходимые для соответствия требованиям информационной системы.

2.7 **контрольный журнал** (audit trail): Журнал, предназначенный для регистрации процессов функционирования в информационной системе.

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

2.9 **связывание** (binding): Установление отношений между конкретными определениями данных и процессами.

2.10 **клиент** (client): Субъект (пользователь или процесс), запрашивающий услуги, предоставляемые на другом процессоре (т.е. сервере).

2.11 **отношение клиент — сервер** (client — server relationship): Связь, устанавливаемая в момент, когда клиент запрашивает услугу, которая должна быть выполнена сервером.

2.12 **коммуникационное соединение** (communications linkage): Средства обмена данными между компьютерными системами или между пользователем и компьютерными системами.

2.13 **компьютерная система** (computer system): Совокупность аппаратных средств, управляемых программным обеспечением (операционной системой) как единый модуль. Компьютерная система может также предоставлять общие услуги, такие как управление доступом, взаимодействие процессоров и графический интерфейс пользователя.

2.14 **конфигурация** (configuration): Совокупность процессов информационной системы и способ взаимосвязи этих процессов.

2.15 **управление конфигурацией** (configuration management): Деятельность, связанная с управлением конфигурацией информационной системы на всех этапах жизненного цикла.

2.16 **правило ограничения целостности** (constraining rule): Правило, являющееся частью средства моделирования данных и управляющее выполнением требований, обеспечивающих целостность определенного набора данных.

2.17 **ограничения** (constraint): Ограничения на значения определенного набора данных.

2.18 **стандарт содержания данных** (data content standard): Логические требования к данным, имеющим общую область применения и используемым во многих прикладных системах.

2.19 **определение данных** (data definition): Описание правил, которым должен удовлетворять один или более набор данных.

2.20 **экспорт данных** (data export): Услуга по управлению данными, заключающаяся в извлечении данных из базы данных и создании копии этих данных, организованных в соответствии с форматом обмена данными.

2.21 **импорт данных** (data import): Услуга по управлению данными, заключающаяся во вставке данных в базу данных, организованную в соответствии с форматом обмена данными.

2.22 **независимость данных** (data independence): Независимость процессов от объектов данных, состоящая в том, что объекты данных могут быть изменены без нарушения процессов.

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

2.24 **формат обмена данными** (data interchange format): Правила структурирования определенных данных, устанавливающие формат данных, необходимый для экспорта данных из одной системы управления данными и их импорта в другую систему управления данными.

2.25 **стандарт обмена данными** (data interchange standard): Стандарт, определяющий совокупность данных в соответствии с правилами их структурирования, обеспечивающими обмен данными между двумя компьютерными системами.

2.26 **управление данными** (data management): Деятельность, направленная на определение, создание, хранение, поддержку данных, а также на обеспечение доступа к данным и процессам манипулирования в одной или более информационной системе.

2.27 **среда управления данными** (data management environment): Используемые в компьютерной системе формализованные принципы описания данных и соответствующие элементы их обработки.

2.28 **услуга по управлению данными** (data management service): Услуга, предоставляемая системой управления данными.

2.29 **сеанс управления данными** (data management session): Период времени, в течение которого клиент пользуется услугами процесса управления данными.

2.30 **система управления данными** (data management system): Система, предназначенная для организации данных и управления ими.

2.31 **процесс манипулирования данными** (data manipulation process): Процесс, семантика которого установлена правилами манипулирования данными в системе моделирования данных.

2.32 **правило манипулирования данными** (data manipulation rule): Правило, которому необходимо следовать при создании процесса или которому автоматически следует система управления данными при выполнении процесса.

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

2.34 **правило структурирования данных** (data structuring rule): Правило, определяющее способ структурирования набора данных.

2.35 **тип данных** (data type): Поименованная совокупность данных с общими статическими и динамическими свойствами, устанавливаемыми формализованными требованиями к данным рассматриваемого типа.

2.36 **база данных** (database): Совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных.

2.37 **контроллер базы данных** (database controller): Абстрактное представление для набора услуг, согласованных с конкретным средством моделирования данных.

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

2.39 **язык базы данных** (database language): Язык с использованием формального синтаксиса, предназначенный для определения, создания, организации доступа и поддержки базы данных.

2.40 **управление базой данных** (database management): Процесс создания, использования и поддержки базы данных.

2.41 **система управления базами данных;** СУБД (database management system; DBMS): Совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

2.42 **система словарей** (dictionary system): Информационная система, содержащая информацию о предприятии, его функциях, деятельности, процессах и данных, используемых в одной или более компьютерной системе.

2.43 **распределенная система базы данных** (distributed database): Совокупность данных, распределенных между двумя или более базами данных.

2.44 **распределенная информационная система** (distributed information system): Информационная система, объекты данных и/или процессы которой распределена на две или более базы данных.

2.45 **данные распределения** (distribution data): Сведения, определяющие информацию о размещении, дублировании и фрагментации объектов в распределенной системе базы данных.

2.46 **фрагментация** (fragmentation): Распределение данных между двумя или более средами управления базами данных.

2.47 **функциональный стандарт** (functional standard): Стандарт, состоящий из набора согласованных между собой стандартов.

2.48 **горизонтальная фрагментация**[1] (horizontal fragmentation): Фрагментация, при которой распределение данных сформировано из данных всех возможных типов одной записи.

2.49 **информационная система** (information system): Система, организующая обработку информации о предметной области и ее хранение.

2.50 **средства моделирования обмена данными** (interchange data modelling facility): Средства моделирования данных, поддерживающие обмен данными между системами управления данными.

2.51 **стандарт интерфейса** (interface standard): Стандарт, определяющий услуги, доступные в интерфейсе, для процессов.

2.52 **пара уровней** (level pair): Концепция моделирования, позволяющая связать схему с соответствующей базой данных. Для двух смежных уровней данных более высокий уровень всегда содержит определения данных, хранящихся на более низком уровне.

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

2.54 **постоянные данные** (persistent data): Данные, сохраняющиеся в информационной системе в течение более одного сеанса управления данными.

2.55 **привилегия** (privilege): Разрешение на использование отдельной услуги управления данными, обеспечивающей доступ к определенным данным, предоставляемое идентифицированному пользователю.

2.56 **процесс** (process): Компонент информационной системы, реализующий конкретный алгоритм обработки данных.

2.57 **процесс соединения** (processing linkage): Предоставление возможного взаимодействия между процессорами.

2.58 **процессор** (processor): Концепция моделирования, являющаяся комбинацией аппаратных средств и программного обеспечения, обеспечивающая предоставление услуги одному или более другому процессору или пользователю.

2.59 **схема** (schema): Описание содержания, структуры и ограничений, используемых для создания и поддержки базы данных.

2.60 **сервер** (server): Процессор, предоставляющий услуги одному или более другому процессору.

2.61 **услуга** (service): Предоставление функциональных возможностей одним процессором другим процессорам или одним процессом другим процессам.

2.62 **интерфейс услуг** (services interface): Определенный набор услуг, обеспечиваемых процессом или процессором.

2.63 **сеанс** (session): Промежуток времени, в течение которого клиент может активно взаимодействовать с сервером или клиент и сервер получают данные друг о друге.

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

2.65 **транзакция** (transaction): Совокупность связанных между собой операций, характеризуемых четырьмя свойствами: атомарностью, непротиворечивостью, локализацией и продолжительностью. Транзакция должна быть уникально идентифицирована пользователем.

2.66 **транзитные данные** (transient data): Данные, поступающие в информационную систему или исключаемые из нее, а в случае распределенной системы — данные, перемещаемые из одной компьютерной системы в другую при выполнении одной или более транзакции.

2.67 **процессор пользователя** (user processor): Процессор, предоставляющий услуги пользователю и являющийся клиентом (прямым или косвенным) контроллера базы данных.

2.68 **вариант** (variant): Конфигурация всей информационной системы или ее части, наряду с которой существует другая система, имеющая другую конфигурацию, обеспечивающая те же услуги.

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

2.70 **вертикальная фрагментация**[2] (vertical fragmentation): Фрагментация, при которой распределение данных сформировано из значений данных одного и того же типа.

---

[1] *Если данные представлены в виде таблицы, то горизонтальная фрагментация — это набор данных, сформированный из строк таблицы.*

[2] *Если данные представлены в виде таблицы, то вертикальная фрагментация — это набор данных, сформированный из столбцов таблицы.*

3 Обозначения и сокращения

3.1 В настоящем стандарте применены следующие обозначения (изображения) и сокращения.

3.1.1 Постоянные данные



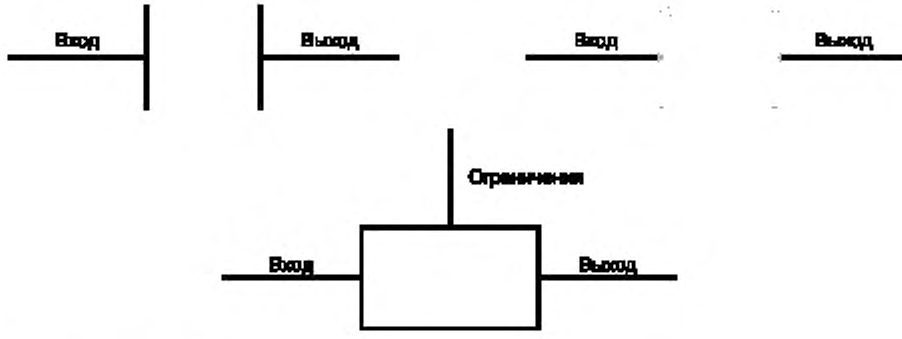
3.1.2 Коммуникационное соединение (каналы обмена информацией)

3.1.3 Технологическое соединение (связь между процессами обработки данных)

3.1.4 Класс процесса



Обозначение класса процесса используют для изображения процесса манипулирования данными. Для процесса соединения по левому краю указывают вход, по правому — выход, сверху — ограничения.



3.1.5 Класс процессора



3.1.6 Класс процессора с интерфейсом услуг



Обозначение класса процессора с интерфейсом услуг используют в диаграммах вместе с символом коммуникационного соединения для указания взаимодействий, в которых он участвует и как клиент, и как сервер. Каждое коммуникационное соединение, обрабатываемое сервером, связывают с интерфейсом услуг (затемнен).

3.1.7 Имена классов

Класс процессора указывают прописными буквами. В исключительных случаях используют строчные буквы.

3.2 Сокращения

ACID[1] — атомарность, непротиворечивость, изолированность, долговечность;
DBMS — система управления базами данных (см. 2.41);
IRDS[2] — системы словаря информационных ресурсов;
NDL[3] — язык базы данных сети;
OSI[4] — открытая взаимосвязь систем;
RDA[5] — удаленный доступ базы данных;
SQL[6] — язык структурированных запросов.

4 Требования к управлению данными

4.1 Цель

Целью раздела является описание:

a) основных понятий, связанных с информационными системами;
b) положений системы управления данными в информационной системе;
c) области применения управления данными.

4.2 Информационные системы

Информационная система — это система, которая организует процессы сбора, хранения и обработки информации о предметной области. Она может быть размещена на одной или нескольких компьютерных системах. Если информационная система размещена на нескольких компьютерных системах, то ее рассматривают как распределенную информационную систему.

Данные поступают в информационную систему и могут быть извлечены из нее, и эти взаимодействия могут быть осуществлены или людьми, или процессами, в том числе и другими информационными системами.

Настоящий стандарт различает два вида данных: временные данные и постоянные данные. Временные данные появляются или при поступлении данных в информационную систему, или при извлечении из нее, а также в случае функционирования распределенной информационной системы при перемещении данных из одной компьютерной системы в другую при выполнении одной или нескольких транзакций. Постоянные данные — это данные, которые сохраняются в информационной системе в течение определенного периода времени.

Управление данными в настоящем стандарте касается организации постоянных данных и управления ими. Система, выполняющая функцию организации постоянных данных и управления ими, называется системой управления данными.

4.2.1 Положение системы управления данными в информационной системе

На рисунке 1 показана связь системы управления данными с компьютерной системой и частями информационной системы, такими как процессы, средства коммуникации и интерфейс человек — компьютер.

В настоящем стандарте под компьютерной системой понимают набор аппаратных средств, управляемый как единый модуль с помощью программного обеспечения (например, операционной системы), которое может дополнительно обеспечивать общие услуги (например, связь между процессами и графический интерфейс пользователя).

---

[1] ACID — Atomicity, Consistency, Isolation and Durability properties.
[2] IRDS — Information Resource Dictionary System.
[3] NDL — Network Database Language.
[4] OSI — Open Systems Interconnection.
[5] RDA — Remote Database Access.
[6] SQL — Structured Query Language.

Все части информационной системы могут быть распределены между двумя или более компьютерными системами. Компьютерная система может включать в себя несколько частей информационной системы.

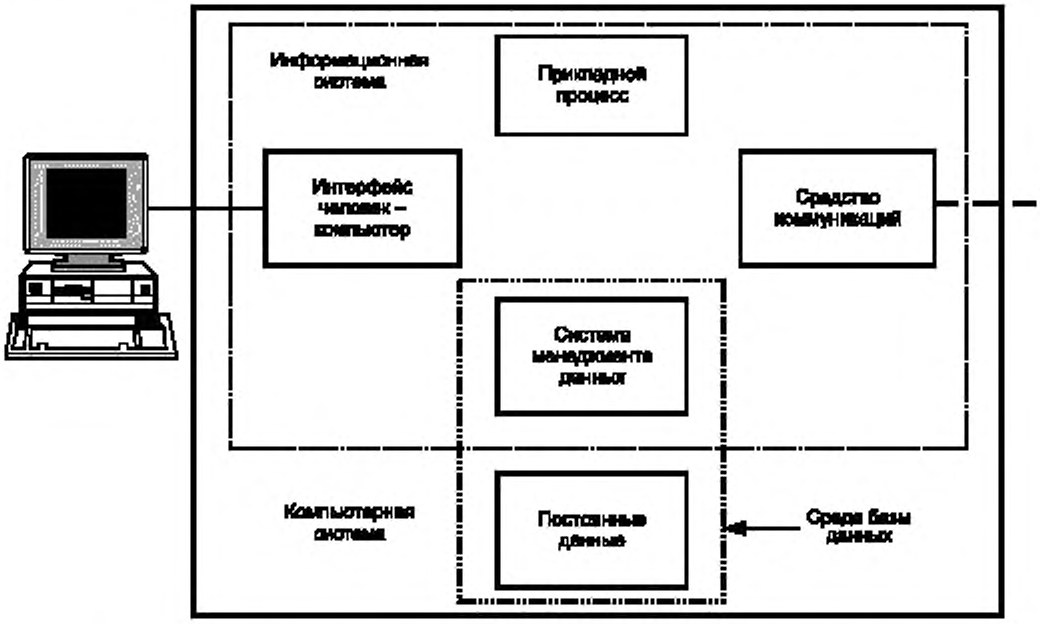


Рисунок 1 — Положение системы управления данными в информационной системе

На рисунке 1 показано следующее:

a) система управления данными предоставляет услуги, управляющие набором постоянных данных;

b) взаимодействие с пользователем обеспечивает интерфейс человек – компьютер,

c) прикладной процесс предоставляет возможности, необходимые для выполнения требований конкретной информационной системы;

d) взаимодействие с другими системам управления данными, информационными системами и компьютерными системами обеспечивается средствами обмена информацией;

e) услуги, предоставляемые интерфейсом человек – компьютер, могут быть использованы другими частями информационной системы;

f) прикладной процесс может использовать услуги, предоставляемые другими частями информационной системы;

g) каждая из частей информационной системы может использовать услуги, предоставляемые компьютерной системой;

h) комбинация системы управления данными и постоянных данных представляет собой среду базы данных.

4.3 База данных и схема

Постоянные данные в среде базы данных включают в себя схему и базу данных. Схема включает в себя описания содержания, структуры и ограничений целостности, используемые для создания и поддержки базы данных. База данных включает в себя набор постоянных данных, определенных с помощью схемы.

Система управления данными использует определения данных в схеме для обеспечения доступа и управления доступом к данным в базе данных.

4.4 Средства моделирования данных

Схему разрабатывают в соответствии с совокупностью правил структурирования данных. Каждая совокупность правил структурирования данных может иметь связанную с ней совокупность правил мани-

пулирования данными, определяющую процессы, которые могут быть выполнены над структурированными данными.

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

Важно различать правила, присущие средствам моделирования данных, используемые при подготовке схем, и правила, установленные для информационной системы, определенные в схеме. Последние правила представляют собой ограничения, которые использует система управления данными при заполнении базы данных в соответствии со схемой.

Средства моделирования данных могут быть определены или в терминах услуг, предоставляемых в интерфейсе услуг в системе управления данными, или с помощью языка базы данных.

Система управления базами данных объединяет в себе систему управления данными и другие процессы, которые поддерживают развитие и использование базы данных.

Язык базы данных используют для определения схемы в соответствии с правилами структурирования данных и процессов согласно правилам манипулирования данными.

Примерами трех классов средств моделирования данных являются реляционный, сетевой и иерархический классы. Правила структурирования данных для двух средств моделирования данных в различных классах могут быть очень похожими, например для сетевого и реляционного классов, а соответствующие им средства манипулирования данными могут различаться.

4.5 Независимость данных

Целью управления данными является достижение независимости данных, которая обеспечивает пополнение и модификацию данных без дополнительных изменений прикладных процессов.

Независимость данных, как правило, достигается тремя способами.

Первый способ состоит в построении прикладного процесса, который использует только необходимую для его работы часть схемы. Такую схему называют прикладной.

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

Третий способ — это включение как можно большего числа ограничений целостности в схему, а не в прикладные процессы. Степень внесения ограничений в схему зависит от способности средств моделирования данных определять ограничения, используемые при разработке схемы.

4.6 Услуги управления данными

Услуги управления данными обеспечиваются в интерфейсе услуг системы управления данными. Эти услуги поддерживают использование средств моделирования данных (определены ли они в терминах услуг или языка базы данных) и все другие средства обслуживания, необходимые для управления постоянными данными.

В любом процессе может потребоваться использование услуг управления данными, доступных в интерфейсе услуг. Существует требование независимости интерфейса услуг от способа выполнения услуг системой управления данными и физическим представлением постоянных данных.

Сеанс управления данными включает в себя последовательность запросов от одного процесса для услуг управления данными, соответствующих одной среде базы данных.

4.7 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 помощью интерфейса.

В любом интерфейсе существуют параметры, о которых пользователь должен знать, чтобы иметь возможность использования основного процессора. Для упрощения работы с интерфейсом число этих параметров должно быть сведено к минимуму.

4.8 Управление доступом

В любой ситуации должны быть определены требования к управлению доступом, которые могут быть отражены в политике обеспечения безопасности. Политика обеспечения безопасности устанавливает формы доступа каждого пользователя в информационную систему, а информационная система должна иметь соответствующие механизмы управления доступом, позволяющие выполнить политику обеспечения безопасности.

Для управления данными задание управления доступом состоит в разрешении санкционированного доступа к данным и предотвращении несанкционированного доступа идентифицированного пользователя (человека или процесса) к определенным данным. Такое управление доступом определяет процессы, кото-

рые может выполнять пользователь. Управление доступом должно быть основано на комбинации идентификатора пользователя, идентификатора процесса и ссылочных данных. Привилегии управления доступом устанавливаются пользователю для того, чтобы пользователь мог выполнять определенные процессы при работе с данными в конкретной ситуации.

Требования к управлению доступом применительно к управлению данными должны допускать:

- определение и последующую модификацию управления доступом;
- введение в действие в любое время ограничений управления доступом, которые необходимы в конкретном месте в конкретное время.

4.8.1 Определение и модификация привилегий управления доступом

Для определения привилегий требуются средства, которые включают в себя инициализацию, модификацию, временное прекращение и отмену. Процесс выделения привилегий пользователям называется санкционированием. Глобальные полномочия предоставляются ответственному лицу (администратору баз данных). Администратор управляет процессом создания и модификации управления доступом в среде управления данными.

Привилегии могут быть определены следующими данными: идентификатором пользователя, ограничениями на использование прикладных процессов баз данных, схем, данных, дат размещения и периодом валидации (действия) привилегий или комбинацией перечисленного.

Может потребоваться дополнительная информация, такая, например, как идентификатор пользователя, который санкционирует привилегию. Данные, описывающие привилегии, связаны с данными по управлению доступом. Эти данные необходимо хранить и обрабатывать точно так же, как и любые другие данные в области управления данными, включая применение средств управления доступом пользователей, которые устанавливают привилегии.

4.8.2 Введение в действие управления доступом

Решение о доступе к данным основано на привилегиях пользователя.

Введение в действие управления доступом требует, чтобы пользователи и процессы, выполняющие роль пользователей, были идентифицированы и чтобы законность запроса на услуги доступа к требуемым данным была проконтролирована в момент выполнения.

4.8.3 Внешняя безопасность управления данными

Следующие составляющие политики обеспечения безопасности связаны с управлением доступом к данным (вне области применения управления данными, определенной для целей настоящего стандарта):

a) аутентификация идентификации пользователя;

b) защита сохраняемых данных обеспечением доступа к ним только через систему управления данными;

c) защита коммуникационных данных обеспечением доступа к ним только через систему управления данными;

d) предпринимаемые меры (например, регистрация в отчетах и контрольных журналах для последующего анализа) в случае нарушений или попытки нарушений безопасности пользователями;

e) любые нарушения или попытки нарушения безопасности пользователями без установленных привилегий должны быть зарегистрированы для последующего анализа.

4.9 Требования к сопровождению управления данными

Требования информационных систем к управлению данными, не зависящие от конкретных требований информационной системы к хранению данных и манипулированию ими:

a) поддержание жизненного цикла информационных систем;

b) управление конфигурацией, управление версиями и вариантами;

c) параллельная обработка;

d) управление транзакциями базы данных;

e) поддержание и повышение производительности;

f) обращение к данным;

g) расширение средств моделирования данных;

h) поддержание различных средств моделирования данных в интерфейсе пользователя;

i) контрольные журналы;

j) восстановление;

k) логическое реструктурирование данных;

l) реорганизация физической памяти.

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

4.9.1 Поддержание жизненного цикла информационных систем

Каждая прикладная система проходит жизненный цикл, который обычно состоит из нескольких фаз. Жизненный цикл может начинаться со стадии планирования информационной системы, на которой должно быть определено, какие прикладные системы необходимы в организации, и заканчиваться анализом, проектированием, созданием, эксплуатацией, пересмотром[1)] и выводом из эксплуатации. В настоящем стандарте не дано точное определение термина «жизненный цикл».

Поддержание жизненного цикла информационной системы приводит к установлению трех типов требований. Во-первых, прикладная система, возможно, потребует пересмотра после перехода на стадию эксплуатации, и поэтому главными требованиями являются функциональные возможности системы управления данными, например способность изменить схему или процесс. Во-вторых, это требование регистрации информации о жизненном цикле и обеспечения средств управления деятельностью в пределах каждой стадии и между ними. В-третьих, это требование хранения, изменения и извлечения процесса и данных при управлении версиями.

4.9.2 Управление конфигурацией, управление версиями и вариантами

Управление конфигурацией включает в себя деятельность по управлению изменениями конфигурации информационной системы на всех этапах жизненного цикла за определенный период времени. Следует идентифицировать отдельные версии системной конфигурации в конкретные моменты времени, а также отслеживать конфигурации каждой конкретной версии.

На различных стадиях жизненного цикла информационной системы могут параллельно существовать в различных форматах постоянные данные и процессы. На стадии пересмотра в переходный период могут параллельно существовать старые и новые форматы данных, старые и новые процессы.

Две формы процесса можно считать различными вариантами. Это означает, что каждый вариант удовлетворяет своим требованиям (например, в представлении сохраняемых данных в памяти), но ни один вариант не предназначен для замены другого.

4.9.3 Параллельная обработка

Информационная система является ресурсом, который может быть распределен между несколькими пользователями одновременно. Это требование о параллельной обработке существенно, даже если только один пользователь вовлечен в использование информационной системы. Должна быть разработана более рациональная система управления одновременно поступающих запросов пользователей, имеющих доступ к общим данным. Среда управления данными должна гарантировать выполнение отдельного запроса каждого пользователя в соответствии с их представлением о данных.

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

4.9.4 Управление транзакциями базы данных

Транзакция базы данных определена как ограниченная последовательность взаимодействий базы данных, которые вместе образуют 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. Обновленное состояние базы данных должно быть оценено с точки зрения соответствия схеме базы данных и, возможно, правилам вложенных прикладных процессов. В соответствии с тестом ACID на качество системы установлены следующие требования к управлению транзакциями базы данных:

a) атомарность. Входящие в транзакцию операции (изменения) должны выступать вместе как неделимая единица работы, т.е. либо все операции успешно завершаются в базе данных, либо отменяются;

b) непротиворечивость. После завершения работы транзакция базы данных оставляет базу данных в непротиворечивом состоянии;

c) изолированность. Изменения, осуществленные транзакцией базы данных, должны быть невидимы для любой другой параллельной транзакции базы данных и наоборот;

d) долговечность. Сохранность изменений, совершенных транзакцией, независимо от сбоев и отказов системы.

Обеспечение выполнения таких свойств транзакций идентифицировано как концепция «упорядочиваемости». Параллельное выполнение нескольких транзакций базы данных должно быть эквивалентным в том смысле, что их параллельное выполнение должно быть таким же, как если бы они были выполнены последовательно.

[1)] *В данном случае пересмотр включает в себя доработку и совершенствование. Примеч. перев.*

4.9.5 Поддержание и повышение производительности

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

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

4.9.6 Обращение к данным

Все данные в среде базы данных должны быть уникально отличимыми от других данных в той же самой среде базы данных. При необходимости должны быть введены ограничения, действующие до тех пор, пока идентификация не станет уникальной.

Имя данных может быть назначено пользователем или системой управления данными. В последнем случае имя может не иметь какого-либо значения для пользователя.

Требование присвоения имени существует для прикладных систем, систем словарей и информационной системы других типов. Если имеется более чем одна среда базы данных в компьютерной системе, то требуется, чтобы одна среда отличалась от другой.

4.9.7 Расширение средств моделирования данных

Средство моделирования данных может быть типовым для систем управления данными. Одновременно может возникнуть требование увеличения типов данных и связанных с ними процессов. Примером этого требования является полная обработка текста совместно с обработкой структурированных данных.

4.9.8 Поддержание различных средств моделирования данных в интерфейсе пользователя

Пользователь может отдать предпочтение формату обработки данных, предусмотренному средствами моделирования данных, отличному от формата системы управления данными. Поэтому между указанными двумя форматами должно быть установлено взаимно-однозначное соответствие.

4.9.9 Контрольные журналы

Необходимо обеспечить сохранение записей об успешных изменениях данных в базе данных, а в некоторых случаях возможно ведение записи о транзакциях, которые запрашивают данные и генерируют отчеты. Эта запись может включать в себя соответствующие значения данных, подробности транзакции и идентификацию пользователя. Такие контрольные журналы могут быть оформлены для всех данных базы данных и/или для данных избранных типов или экземпляров определенных данных.

4.9.10 Восстановление распределенной базы данных

База данных должна иметь возможность возвращения к предшествующему непротиворечивому состоянию. Это требование может возникнуть из-за ошибочных транзакций, системного сбоя или потери хранимых данных. Для удовлетворения этих требований могут быть использованы различные механизмы, такие как запись всех изменений, проведенных в базе данных, и сохранение резервных копий всей базы данных или ее части.

4.9.11 Логическое реструктурирование данных

Логическое реструктурирование данных — это процесс изменения определения данных после использования информационной системы в течение некоторого времени. Изменение может быть дополнением к существующему определению данных или может заключать в себе модификацию части существующего определения данных.

Внесение дополнительной информации обычно затрагивает только те прикладные процессы, которые используют измененные данные. Однако определенные модификации могут затронуть существующие определения данных, и необходимо гарантировать в рамках реструктуризации, что данные и схема являются согласованными.

4.9.12 Реорганизация физической памяти

Реорганизация физической памяти — это процесс изменения представления постоянных данных на носителе данных.

4.10 Дополнительные эксплуатационные требования для поддержания управления данных в распределенной информационной системе

Кроме эксплуатационных требований информационной системы, существуют требования, предъявляемые распределенной информационной системой. В этих случаях управляемые данные сохраняются в более чем одной компьютерной системе.

В распределенной информационной системе данные, принадлежащие одной информационной системе, распределяются на две или более среды базы данных, каждая из которых находится на одном компьютере.

Возможна ситуация, когда запрашиваемой стороне неизвестна идентичность системы, запрашивающей услугу. Альтернативно запрашиваемая услуга может быть доступна из множества вычислительных устройств, хранящих дублированные данные.

Эксплуатационные требования, связанные с распределяемыми данными, следующие:

a) управление распределением;

b) управление транзакцией базы данных,

c) связь;

d) экспорт/импорт;

e) независимость распределения;

f) автономность системы;

g) восстановление распределенной базы данных.

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

Первое требование относится к степени управления распределением данных, которая варьируется между двумя крайностями:

a) с одной стороны, отсутствует управление распределением данных. Это означает, что каждый прикладной процесс несет ответственность за идентификацию имени (определение местонахождения не является обязательным) среды базы данных, в которой доступны все необходимые данные. Каждая среда базы данных автономна;

b) с другой стороны, распределение данных на две или более среды базы данных полностью скоординировано до степени, при которой прикладной процесс не знает, как данные фактически распределены. Совокупность всех данных рассматривается как распределенная база данных. Такая распределенная база данных должна управляться внутренне непротиворечиво и соответствовать определениям в одной схеме. Эта схема может быть до некоторой степени распределенной.

Вторая классификация относится к сценариям распределения, которые описывают следующие альтернативные пути разработки и развития распределенной информационной системы:

a) распределенную систему базы данных, в которой составные среды базы данных спроектированы таким образом, что возможно взаимодействие между любой их парой;

b) систему баз данных, в которой две или более отдельно спроектированные системы баз данных объединены в определенном смысле после периода раздельного использования и сформированы для функционирования одной распределенной системы баз данных;

c) ситуацию, в которой каждая среда базы данных соответствует множеству стандартов и, следовательно, может взаимодействовать (вероятно, на специальной основе) с другими средами баз данных, каждая из которых была спроектирована отдельно, но согласно одним и тем же стандартам.

4.10.1 Управление распределением данных

Управление распределением данных включает в себя управление фрагментацией и репликацией.

В зависимости от целей и возможностей распределения данных фрагментацию осуществляют в распределенной базе данных различными способами. Обычно выделяют два способа фрагментации — горизонтальную фрагментацию, в которой распределение формируют из всех типов данных одной записи, и вертикальную фрагментацию, в которой распределение формируют из данных одного типа всех записей.

При горизонтальной фрагментации каждая среда базы данных включает в себя данные всех типов, необходимые для прикладного процесса.

Вертикальная фрагментация позволяет записывать на конкретном вычислительном устройстве только данные конкретного типа, необходимые для прикладного процесса.

Могут быть применены такие способы распределения данных:

a) без фрагментации;

b) с использованием горизонтальной фрагментации;

c) с использованием вертикальной фрагментации;

d) с использованием комбинации горизонтальной и вертикальной фрагментации.

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

Для повышения производительности или защиты от сбоя компьютерной системы необходимо обеспечить копирование всей базы данных или ее частей. Такие репликационные данные могут быть сохранены в компьютерной системе, отличной от той, в которой данные первоначально были созданы, и в дальнейшем должны быть управляемы. Требование для фрагментации может быть объединено с требованием реплика-

ции так, чтобы копии множества фрагментов назначались в две или более среды баз данных. Информация о том, какие данные в какой среде данных требуются, должна быть доступна (прямо или косвенно) в каждой среде.

Необходимо иметь возможность управлять содержимым репликаций, когда данные обновляются. Алгоритмы, обеспечивающие контроль репликаций, должны также гарантировать обновления в транзакциях.

4.10.2 Управление транзакцией базы данных

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

Обработка в одной компьютерной системе может быть выполнена одновременно с обработкой в другой компьютерной системе без влияния на целостность данных в каждой из компьютерных систем.

4.10.3 Связь

Необходимо обеспечить информационным системам связь друг с другом.

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

Необходимо иметь средства, предотвращающие потерю целостности баз данных из-за сбоя связи следующих видов:

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

Для предупреждения потери целостности базы данных из-за подобных сбоев следует определить необходимую степень дублирования данных.

4.10.4 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 могут быть сохранены.

4.10.5 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.

Прикладные процессы должны быть независимыми от распределения данных. Существует несколько видов независимости распределения. В случае локальной независимости прикладной процесс должен знать о существовании различных фрагментов в распределенной базе данных, но не их местоположение. При прозрачности фрагментации прикладной процесс не нуждается в знании о существовании фрагментов.

4.10.6 Автономность системы

В некоторых распределенных информационных системах требуется автономность компьютерной системы, хранящей распределенные данные. Автономная компьютерная система должна быть способна функционировать независимо от других компьютерных систем. Такое требование относится к требованию иметь и использовать доступ к данным во время сбоев связи и возникновения административных проблем, например связанных с бухгалтерским учетом и/или установлением подлинности пользователей.

Требования, установленные для автономных компьютерных систем в распределенной информационной системе, предусматривают также установление существенных требований к управлению данными для такой информационной системы.

4.10.7 Восстановление распределенной базы данных

Модифицированные данные, распределенные в более чем одной базе данных, должны быть восстановлены таким способом, чтобы полученные результаты находились в непротиворечивом состоянии с базой данных.

4.11 Системы словарей

Относительно управления данными, связанными с данными об информационных системах, существуют многочисленные требования. Эти данные управляются специальной информационной системой, называемой системой словарей.

Чтобы отличать особую цель системы словарей от целей других информационных систем, последние называют прикладными системами. Система словарей — информационная система, содержащая данные об одной или более прикладной системе.

Примерами требований, относящихся к области применения системы словарей, являются требования к поддержке жизненных циклов, управлению версиями, управлению конфигурацией, аудиту и технологии разработки.

5 Уровни данных и соответствующие процессы

5.1 Цель

Целью настоящего раздела является описание структуры данных в соответствии с требованиями настоящего стандарта с точки зрения соотношения данных друг с другом и связи со структурой процесса.

5.2 Пара уровней

Конструкция «пара уровней» является способом описания связей между базой данных и схемой. Графическое представление (рисунок 2) применяют для иллюстрации связи базы данных с ее описанием.



Рисунок 2 — Структура «пары 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, соответствующими схеме базы данных. Схема устанавливает точную форму разрешенной обработки. Поэтому конструкция «пара уровней» иллюстрирует способ достижения непротиворечивых операций манипулирования данными.

Представление и интерпретация значений данных зависят от схемы. Обработка не может быть осуществлена до тех пор, пока схема не будет определена и активирована. Если требуются изменения в схеме, то соответствующая ей база данных должна быть изменена таким образом, чтобы сохранить непротиворечивость.

5.2.1 Объединение в блоки 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 пригодны 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 «объединенными в блоки», как показано на рисунке 3.

В соответствии с рисунком 3 база данных-1 согласована со схемой-1. Данные в базе данных-1 могут быть обработаны процессами манипулирования данными, которые соответствуют схеме-1. База данных-2 соответствует схеме-2, а данные в базе данных-1 могут быть обработаны процессами манипулирования данными, которые соответствуют схеме-2. Представление схемы-1 в базе данных-2 является исходной схемой. Исходная схема может быть выбрана из базы данных или, наоборот, получена путем манипулирования данными точно так же, как и любые данные в базе данных.

Две пары уровней находятся на разных уровнях определения данных. Если схема-2 (см. рисунок 3) может быть представлена в форме данных строки, записанных в базе данных, то принцип блокирования пар уровней является рекурсивным и может быть использован двумя и более парами уровней. Рекурсивное применение должно быть прекращено, когда определение данных больше не может быть модифицировано.

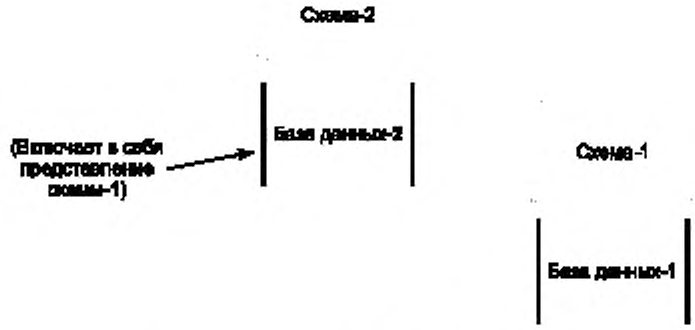


Рисунок 3 — Объединенные в блоки пары уровней

5.2.2 Рекурсивное использование пар уровней

Для потенциально сложных пар уровней требуется механизм ссылки на каждый уровень. Общие ярлыки (N), (N+1) и т.д. использованы в настоящем стандарте для отражения более высоких уровней при рассмотрении общих свойств.

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

Обобщенное блокирование пар уровней с использованием обозначений приведено на рисунке 4.

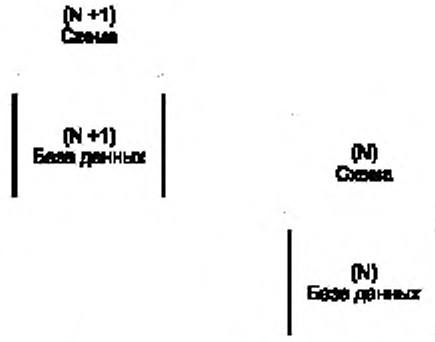


Рисунок 4 — Обобщенное блокирование пар уровней

5.2.3 Операции с использованием пар уровней

Реализация базы данных включает в себя процессы создания и сопровождения определений данных. Эти определения становятся доступными для процессов манипулирования данными. Затем должны быть выполнены операции выборки и модификации данных в базе данных.

Рисунок 4 иллюстрирует указанные выше процессы следующим образом:

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

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

c) база данных (N+1) содержит определения данных, которые были созданы в процессе проектирования базы данных (N) и поддерживались при функционировании системы. База данных (N+1) может также содержать другие данные, такие, например, как описания этих определений и проекты этих данных, и требования к процессам, которые их используют;

d) база данных (N+1) может содержать представления одной или более схемы (N) в исходной форме. После того как одна из этих исходных схем (N) была выбрана, может быть использован активированный процесс конвертирования исходной схемы (N) в форму, называемую объектной схемой, таким образом, чтобы могла быть заполнена соответствующая ей база данных (N). Исходная схема (N) может быть активирована более чем один раз, и каждая активация создает отдельную объектную схему (N) с соответствующей базой данных (N), которая может быть заполнена с использованием процессов манипулирования данными.

Сблокированная пара имеет самый низкий уровень, если данные на более низком уровне не содержат данных о схеме и ее компонентах и, следовательно, не могут быть активированы. Эта пара уровней является тогда частью прикладной системы, а данные на более низком уровне по отношению к этой паре уровней являются прикладной базой данных.

Сблокированная пара имеет самый высокий уровень, если схема на более высоком уровне не записана в базе данных более высокого уровня. Эта схема является тогда неявной в средстве моделирования данных, используемом системой управления данными.

5.3 Зависимость пар уровней от средства моделирования данных

Конструкция «пара уровней» и понятие средства моделирования данных взаимосвязаны. Средство моделирования данных заключает в себе множество правил структурирования данных и соответствующее множество правил манипулирования данными.

5.3.1 Пары уровней и правила структурирования данных

Средство моделирования данных представляет собой совокупность правил структурирования данных, которые должны быть использованы при определении схемы. Эти правила включают в себя правила определения ограничений, которые могут быть частью схемы. Каждая схема должна быть полной и непротиворечивой в соответствии с правилами структурирования данных соответствующего средства моделирования данных.

Представленная на рисунке 4 схема (N+1) ограничивает данные, которые могут быть созданы в базе данных (N+1). В результате этого схема (N+1) влияет на каждую исходную схему (N), содержащуюся в базе данных (N+1).

5.3.2 Пары уровней и правила манипулирования данными

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

5.4 Пары уровней и соответствующие процессы

Данные, содержащиеся в базе данных, могут быть извлечены или модифицированы серией процессов манипулирования данными. Кроме того, если часть этих данных включает в себя исходную схему, то над данными может быть выполнен процесс активации.

Рисунки 5 и 6 иллюстрируют, как правила структурирования данных, используемые для определения схемы (N+1), влияют на процессы манипулирования данными в базе данных (N).

Использованы следующие пять процессов манипулирования данными: 1 — выбирать; 2 — активировать; 3 — связывать с правилами манипулирования данными; 4 — связывать с правилами структурирования данных; 5 — выбирать или модифицировать.

Рисунки 5 и 6 представляют собой расширение рисунка 4 и показывают активацию соединения схем, промежуточный отбор данных, подготовку и функционирование других процессов манипулирования данными.

Следующие шаги описывают соответствующие процессы (рисунки 5 и 6) и показывают, как правила средства моделирования данных позволяют 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 выбора может состоять из установки признака (флага) в базе данных (N+1) или отбора данных из базы данных (N+1) и хранения их в другой базе данных, отличной от базы данных (N+1). Эта новая база данных должна также соответствовать схеме (N+1).

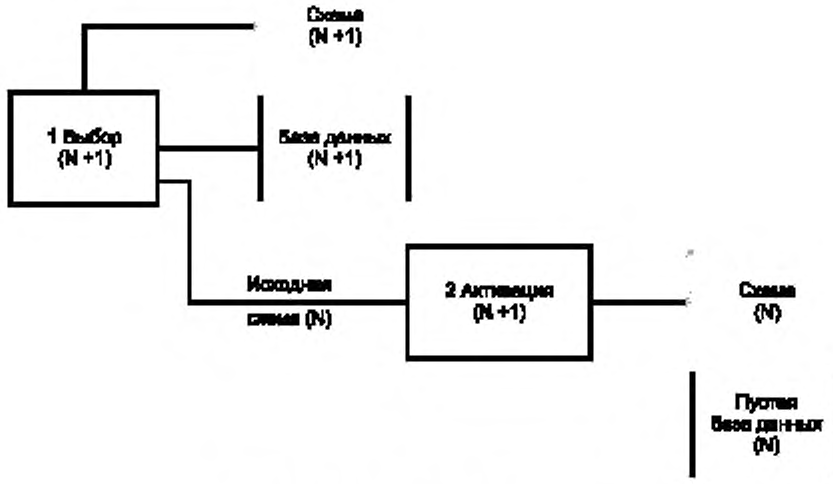


Рисунок 5 — Создание пустой базы данных

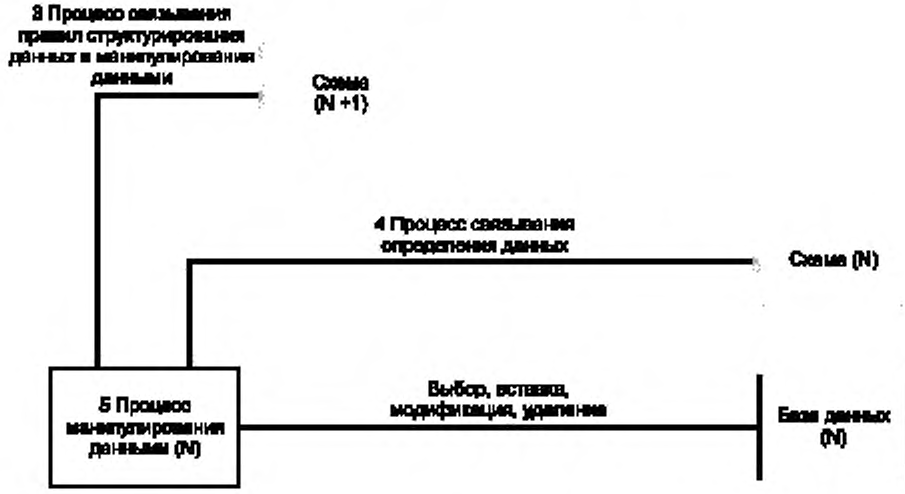


Рисунок 6 — Связывание данных и манипулирование данными

Шаг 2. Исходная схема активируется для создания объектной схемы (N) и пустой базы данных (N). Осуществляется анализ, предшествующий активации, для гарантии того, что исходная схема есть истинная схема. Такой анализ может быть проведен в целом или частично с помощью процесса манипулирования данными (N+1), выполняющего выбор с помощью другого процесса, анализирующего исходную схему, или одновременно с активацией.

Активация может заканчиваться изменением признака (флага), связанного с предварительно выбранной схемой (N) в базе данных (N+1). Кроме того, активация может заканчиваться физическим перемещением данных (N+1) и изменением формы представления. В обоих случаях активированная схема должна быть защищена от любых изменений в базе данных, не соответствующих схеме.

Шаг 3. Правила манипулирования данными, которые управляют выполнением процесса манипулирования данными (N), имеющего доступ к базе данных (N), являются теми же самыми или основанными на правилах, соответствующих схеме (N+1).

Шаг 4. Процесс манипулирования данными (N) для обеспечения его доступа к базе данных (N) должен соответствовать правилам средств моделирования данных в схеме (N+1). Если уровень (N+1) самый высокий, то это соответствие неявно.

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

Шаги 1 и 2 обеспечивают создание пустой базы данных. Шаг 3 устанавливает связь между средством моделирования данных, определенным на уровне (N+1), и процессом манипулирования данными (N). Этот шаг устанавливает сблокированность пар уровней. Оба элемента пары уровней необходимы для обеспечения корректного выполнения процессов управления данными.

Взаимосвязь может быть установлена различными способами (например, ссылкой на схему или присоединением схемы к процессу) и в различные моменты времени (например, при выполнении или компиляции). Сделанный выбор может влиять на время выполнения, объем памяти и легкость поддержания непротиворечивости процессов и схемы во время модификации последней. В противном случае выбор не влияет на результаты процессов.

5.5 Управление доступом для пар уровней

Все данные и процессы, проводимые с использованием данных, должны быть объектами управления доступом. Управление доступом одинаково применимо к любой паре уровней.

5.6 Модификация схемы

Необходимо иметь возможность изменять структуру базы данных. Это означает, что соответствующая 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, модифицируя исходную схему в базе данных (N+1), активируя ее для создания новой схемы (N) и пустой базы данных (рисунок 5) и передавая первоначальные данные в новую базу данных. Для удовлетворения этого требования также может быть применена пошаговая модификация. Механизмы передачи данных от первоначальной к новой версии базы данных не входят в область применения настоящего стандарта.

6 Архитектурная модель

6.1 Цель

Целью настоящего раздела является описание архитектурной модели, содержащей различные данные и компоненты обработки, необходимые для управления данными. Архитектурная модель учитывает требования к управлению данными (раздел 4), структуру данных и принципы обработки данных (раздел 5). Вначале должна быть описана общая модель, идентифицирующая обработку данных, общую для управления всеми данными, а затем разработана модель, учитывающая специальные требования (идентифицированные в разделе 4) к базе данных, распределенной среде базы данных, экспорту и импорту данных и средствам управления доступом.

Принципы, используемые для идентификации потенциальных стандартов управления данными в рамках этой архитектуры, описаны в разделе 7. Связь с действующими стандартами и проектами стандартов в области управления данными показана в приложении В.

6.2 Понятия моделирования

В соответствии с настоящим стандартом предполагают, что процессор является некоторой комбинацией аппаратных средств и программного обеспечения, предоставляющей услуги одному или более процессору и/или пользователю.

Архитектурную модель представляют в терминах различных классов процессоров, где каждый класс процессора ориентирован на специальные требования управления данными. Архитектурная модель не предназначена для внедрения в технических компьютерных системах.

6.2.1 Характеристики процессоров эталонной модели

a) Архитектурную модель представляют в терминах процессоров, которые взаимодействуют как клиент и/или сервер. Эти термины используют для ссылок на процессоры, которые функционируют при конкретном взаимодействии. Специальный процессор может играть различные роли при других взаимодействиях.

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

1) признак (индикатор), что запрашиваемая услуга завершена;

2) набор данных, который является результатом запрашиваемой услуги;

3) сообщение, что услуга недоступна (поскольку клиент не имеет необходимых полномочий или запрашиваемая услуга не может быть предоставлена процессором);

4) сообщение, что запрашиваемые данные недоступны;

5) сообщение, что запрос сформулирован некорректно.

c) Каждый процессор определяется внешним интерфейсом, который он представляет как сервер. Этот сервер должен определять услуги и тип данных, к которым применяют услуги. Для любого процессора внутреннее представление данных, способ выполнения обработки и любое использование услуг, предоставляемых другими серверами, не связаны с определением его интерфейса. Взаимодействие процессора с другими серверами относится к задачам моделирования.

d) Каждый процессор относится к некоторому классу. Класс определяет услуги, общие для всех процессоров класса. Некоторые классы определяют тип данных, к которым применяют услуги. Некоторые классы являются общецелевыми, когда процессору необходима отдельная схема для определения данных, к которым применяют услуги.

e) Процессор может быть клиентом нескольких серверов в любое время. Несколько серверов могут одновременно обслуживать нескольких клиентов.

f) Процессоры могут быть использованы как стандартные блоки, необходимые для обеспечения услуги управления данными. Использование некоторых процессоров может быть рекурсивным.

6.2.2 Уровни абстракции

Для процессоров эталонной модели подробное описание деталей управления данными (например, подробное описание функционирования процессора или его взаимосвязей с другими процессорами в компьютерной системе) не очень важно. Идентифицированные в архитектурной модели взаимодействия обычно позволяют процессору клиента и процессору сервера находиться в одной компьютерной системе, в этом случае их взаимодействие поддерживается средствами функционирования системы, или в различных компьютерных системах, в таком случае их взаимодействие поддерживается средствами функционирования системы или средствами коммуникаций.

Для более детального описания архитектурной модели необходимо использовать:

a) специализацию, в которой подкласс общего класса процессоров определен как имеющий индивидуальное имя, и услуги, являющиеся дополнительными или модифицированными формами услуг общего процессора;

b) декомпозицию, в которой услуги класса процессоров показаны как обеспечиваемые двумя или более классами процессоров с взаимодействием между ними.

6.2.3 Примечание для процессоров

В диаграммах модели и соответствующих описаниях необходимо идентифицировать, является ли класс или экземпляр процессора ссылочным. Для отличия этих двух понятий имя класса пишут с прописной буквы, а имя процессора — со строчной буквы. Например, один из процессоров архитектурной модели принадлежит классу «Процессор пользователя», содержащему экземпляры, имеющие ссылку на процессор пользователя.

Диаграммы используют для того, чтобы показать классы процессора, вовлеченные в специальный тип обработки, и взаимодействие в формате «клиент — сервер» между классами. Условные обозначения в общей диаграмме для класса процессора представлены на рисунке 7, где во внутреннем поле должно быть указано имя класса, а в затемненной части отображен интерфейс. Остальная часть текста диаграммы поясняет диаграмму.



Рисунок 7 – Пример диаграммы процессора

Для любого процессора предусматривают две формы взаимодействия с другими процессорами. Первая ситуация — когда процессор используют в качестве сервера, если существует соединительная линия, отображающая запросы на услугу от клиента на интерфейс. Вторая ситуация — когда процессор используют в качестве клиента услуг другого сервера, если существует соединительная линия от точки на границе (не интерфейс) на интерфейс другого сервера. Каждая линия представляет собой процесс соединения,

которое может поддерживать последовательно одно или более взаимодействие между процессорами сервера и клиента.

Архитектурная модель включает в себя множество диаграмм, построенных с использованием описанных правил. В каждом случае необходимо определить взаимодействие процессоров. В некоторых ситуациях могут быть установлены ограничения на допустимые взаимодействия между процессорами, в других — подобные ограничения могут отсутствовать. Эти требования (или их отсутствие) не всегда очевидны по диаграмме класса. В таких случаях приводят ряд примеров взаимодействующих процессоров для интерпретации диаграммы класса.

6.3 Общая модель управления данными

Общая модель управления данными отражает характеристики классов процессоров, общих для управления данными в целом, и классов пользователей, заинтересованных в определении и доступе к данным базы данных. На рисунке 8 приведены компоненты общей модели управления данными.

Эта модель основана на характеристиках общего класса базы данных и общей схемы класса, как описано в разделе 5. Каждый класс и его процессоры имеют следующие характеристики управления данными, термин «общий» необходимо использовать для классов управления данными.

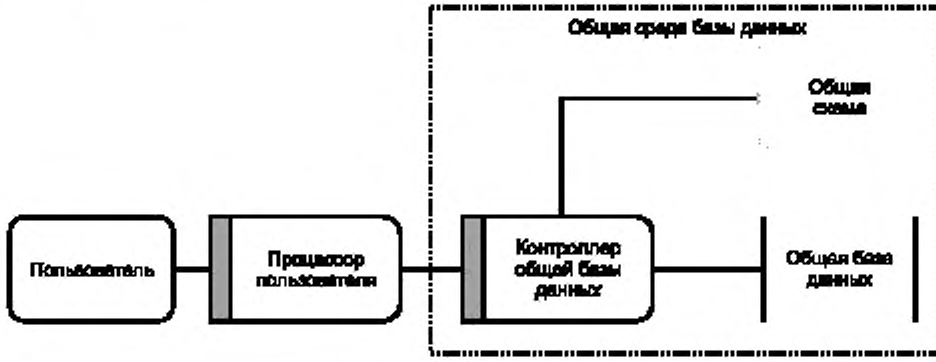


Рисунок 8 — Общая модель управления данными

6.3.1 Контроллер общей базы данных

Контроллер общей базы данных — имя класса процессоров общего назначения, которые обеспечивают предоставление услуг управления данными для определения класса баз данных и доступа к нему. Такие процессоры требуют доступа к схеме, а их услуги связаны с соответствующим классом схем.

Контроллер общей базы данных связан с одной схемой и соответствующей ей базой данных, которые вместе формируют среду общей базы данных. Каждая среда базы данных должна иметь уникальный идентификатор.

Данные, к которым получает доступ любой контроллер базы данных в любом классе контроллеров базы данных, должны быть структурированными и управляемыми в соответствии с правилами, установленными для одного определенного средства моделирования данных.

Данные, к которым получает доступ любой контроллер базы данных, могут включать в себя правила для одного или более определенного средства моделирования данных. Эти правила должны соответствовать правилам определения средства моделирования данных, на котором основывается контроллер базы данных.

Типичные услуги, обеспеченные контроллером базы данных:

a) установка сеанса управления данными для процессора клиента с требованием явного или неявного закрепления с названной средой базы данных (возможны запросы следующих услуг);

b) дополнение и изменение определения данных в схеме базы данных;

c) извлечение определений данных из схемы базы данных;

d) дополнение, изменение данных в базе данных или их удаление;

e) извлечение данных из базы данных;

f) начало транзакции базы данных после получения одного или более запроса на услуги;

g) завершение транзакции с сохранением или отменой изменений, внесенных транзакцией;

h) установление процедур резервного копирования для базы данных;

i) инициация процедур восстановления для базы данных;
j) реорганизация базы данных;
k) прекращение сеанса.
Запросы на эти услуги выражают контроллеру базы данных в виде:
a) операторов на языке базы данных;
b) сообщений, поступающих на интерфейс услуг управления данными;
c) вызовов процедур.

6.3.2 Процессор пользователя

Процессор пользователя — имя класса процессоров, которые являются клиентами для услуг управления данными, прямо или косвенно предоставляемых контроллерами базы данных. Каждый процессор пользователя предоставляет целый ряд услуг, многие из которых, при необходимости, пользуются услугами управления данными. Описание услуг процессора пользователя лежит вне области применения настоящего стандарта.

При предоставлении услуг в интерфейсе процессора пользователя могут быть использованы средства моделирования данных, отличные от средств моделирования данных, используемых общим контроллером базы данных. В этом случае правила, установленные для средств моделирования данных, используемые в интерфейсе процессора пользователя, должны быть определены с использованием средств моделирования данных контроллера общей базы данных.

При работе процессор пользователя является клиентом услуг одного или более контроллера базы данных. При использовании процессором услуг одного (любого) контроллера базы данных должен быть установлен сеанс управления данными между процессором пользователя и контроллером базы данных.

6.3.3 Пользователь

Пользователь — имя класса лиц или процессоров, которые заказывают услуги, предоставляемые процессором пользователя.

6.4 Требования к модели в различных средах

Общая модель, представленная в 6.3, может быть применена к информационным системам различных видов, описанных в разделе 4. На абстрактном уровне это применение осуществляют, заменяя термин «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.

Требования к модели более подробно изложены в последующих пунктах путем использования декомпозиции абстрактных классов и/или рассмотрения возможных экземпляров данных с целью показать составляющие классы.

6.5 Среда базы данных

Применительно к общей модели контроллер базы данных должен поддерживать типичные услуги для среды базы данных, идентифицированные в 6.3.1.

В 6.2 указано, что для полной интерпретации основных понятий моделирования требуется понимание возможных вариантов каждого класса. Следующие примеры приведены для демонстрации двух различных классификаций, которые возможны для среды базы данных, и последствий использования услуг контроллера базы данных.

Существенной особенностью примера, показанного на рисунке 9, является то, что несколько процессоров пользователей используют услуги управления данными одной среды базы данных. Для упрощения на рисунке только один пользователь связан с каждым процессором пользователя. Это не является установленным ограничением.

Этот пример иллюстрирует услуги контроллера базы данных, которые должны поддерживаться для многих пользовательских процессоров-клиентов параллельно, однако клиент должен иметь возможность выполнять транзакции базы данных, не подвергаясь воздействию со стороны других клиентов.

В примере, приведенном на рисунке 10, в отличие от примера, приведенного на рисунке 9, показано, что несколько процессоров пользователя способны иметь доступ более чем к одной среде базы данных.

В этом случае процессор пользователя, обладающий доступом более чем к одной среде базы данных, должен иметь возможность управлять запросом на услугу среды базы данных и обладать возможностью доступа к данным. Любой запрос на услугу должен включать в себя идентификатор среды базы данных, для которой он предназначен. Путь, которым такой идентификатор связан со средой базы данных, находится вне области применения настоящего стандарта. Услуги контроллера базы данных могут поддерживаться только в своей собственной среде базы данных. Любые транзакции базы данных или связи между данными, включающие в себя более чем одну среду базы данных, должны поддерживаться процессором пользователя, отличным от контроллера базы данных.

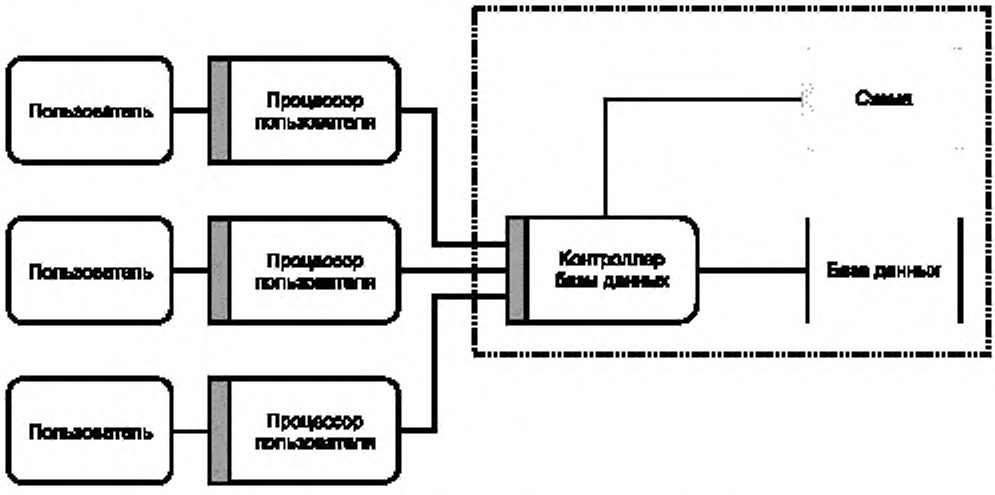


Рисунок 9 — Пример доступа к среде базы данных

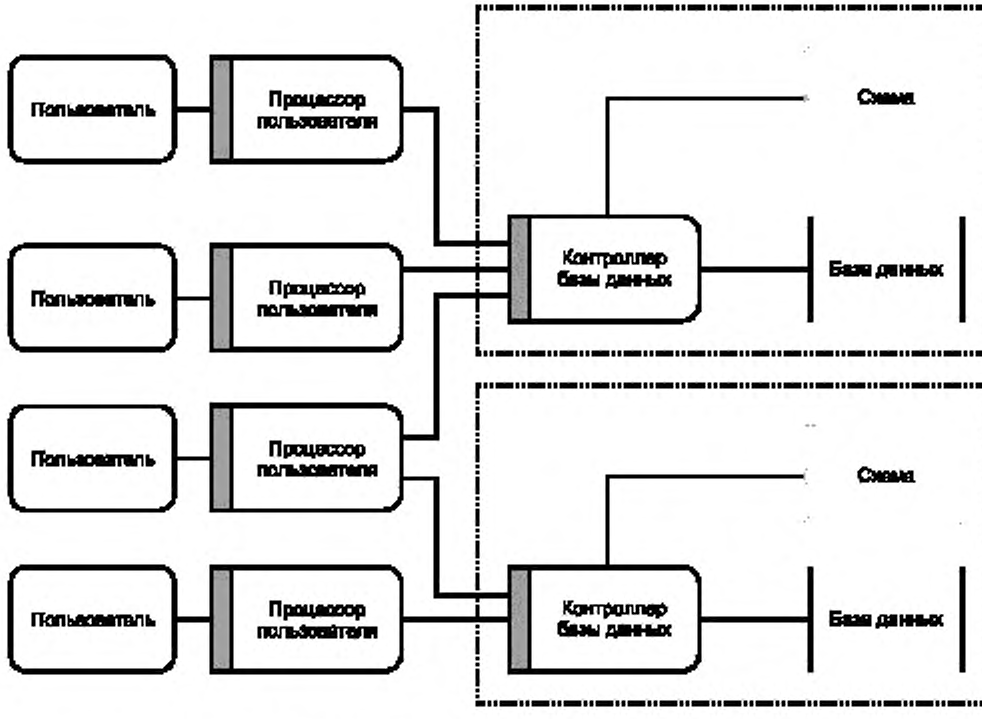


Рисунок 10 — Пример доступа ко многим средам базы данных

Контроллер базы данных может использовать интерфейс услуг для того, чтобы получить доступ к данным, которые не находятся под его собственным управлением.

Приведенный пример показывает, что услуги контроллера базы данных не поддерживают управление данными в мультиплексных средах базы данных, хотя процессоры пользователей в конкретных обстоятельствах могут эту услугу предоставить.

Приведенные выше описания не рассматривают связь между средой базы данных и любой конкретной информационной системой. На рисунке 10 две среды базы данных могут находиться в одной или различных компьютерных системах. Кроме того, оба примера учитывают вариант, когда процессор пользователя находится в отличной от среды базы данных компьютерной системе. В этом случае требуются протоколы связи для процесса соединения. Эти протоколы могут быть пассивными носителями взаимосвязи или обеспечивать специальные услуги, которые поддерживают удаленное использование услуг управления данными.

6.6 Управление распределенными данными

Разработка общей модели управления данными может быть использована для выполнения требований к распределенной информационной системе.

Управление распределенными данными используют при работе с несколькими различными комбинациями представленных классифицированных схем. Основным общим свойством является то, что услуги, предоставляемые процессором пользователя, могут быть связаны с данными, которые хранятся в одной или нескольких отдельных средах базы данных. Возможна также ситуация, когда запрашиваемая процессором пользователя услуга доступна только в некоторой компьютерной системе, отличной от той, в которой происходит запрос.

Все услуги контроллера базы данных для общей модели применимы в распределенной базе данных. Услуги, связанные с распределением, должны предоставляться, при необходимости, контроллером базы данных в интерфейсе различных процессоров.

Запрос данных из распределенной базы данных является типичной услугой, предоставляемой контроллером распределения. При получении запроса на услугу контроллер распределения устанавливает необходимость доступа к другой компьютерной системе. Контроллер распределения должен также установить перечень предоставляемых по запросу услуг.

Для каждой среды базы данных существует набор постоянных данных, которые использует контроллер базы данных, обрабатывая запросы на услуги. Этот набор данных называют локальными данными распределения.

Для распределенных данных способ распределения данных должен быть установлен в базе данных. Аналогично данные в распределенной базе данных могут быть распределены каким-либо альтернативным способом.

На рисунке 11 приведена архитектурная модель управления распределенными данными. Она основана на декомпозиции, позволяющей показать услуги контроллеров базы данных для локального управления данными и услуги контроллеров распределения данных для управления распределенными данными.

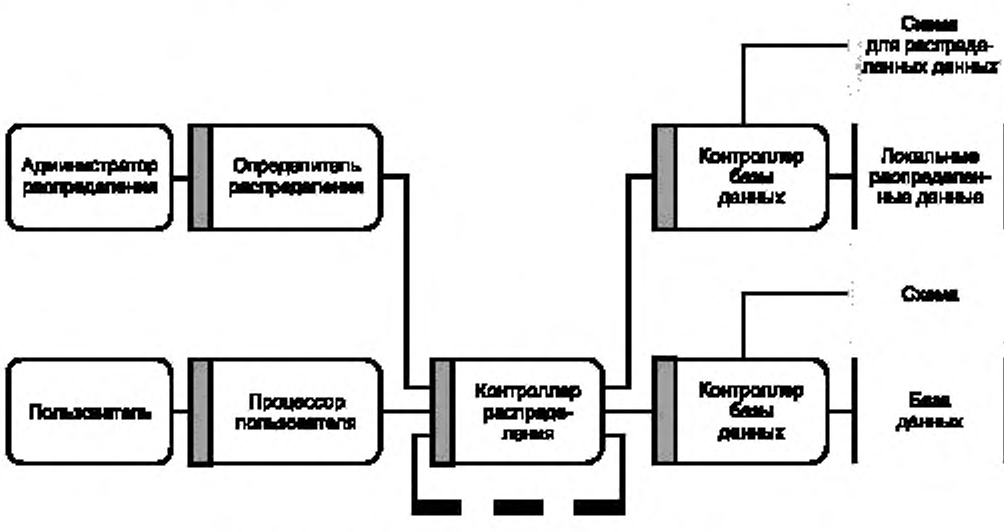


Рисунок 11 — Управление распределенными данными

К этой модели применимо приведенное ранее описание пользователя, процессора пользователя и контроллера базы. Администратор распределения — это обычный класс пользователя, имеющий отношение к задаче определения распределенных данных. Определитель распределения — это процессор пользователя, обеспечивающий услуги администрирования распределения.

6.6.1 Контроллер распределения

Контроллером распределения называется класс процессоров, обеспечивающих услуги определения данных и доступа к распределенным базам данных. Контроллер распределения должен иметь возможность обращения к частям схемы и частям распределенных данных, прямо или косвенно связанным с распределенной базой данных, а также к распределенным данным, находящимся в локальной среде базы данных и/или в удаленной среде базы данных.

Распределенные данные, используемые контроллером распределения, должны быть доступны с помощью тех же средств моделирования данных.

Для любой распределенной базы данных предполагается, что каждая компьютерная система имеет контроллер распределения. Каждый контроллер распределения обеспечивает доступ к тем частям распределенной базы данных, которые хранятся в любой среде базы данных одной и той же компьютерной системы. Контроллер распределения поддерживает доступ к частям распределенной базы данных в удаленных средах базы данных через связь (показано на рисунке 11 выделенной пунктиром линией) с другими контроллерами распределения в различных компьютерных системах.

Определитель распределения, показанный на рисунке 11, является только процессором пользователя для вызова услуг по доступу к распределенным данным. Другие процессоры пользователя не запрашивают никаких услуг по доступу к распределенным данным и могут воспользоваться теми же самыми услугами контроллера базы данных, определенными для общей модели. Контроллер распределения предоставляет дополнительные услуги другим контроллерам распределения при сопровождении распределения данных.

При получении первоначального запроса на услугу от процессора пользователя контроллер распределения сначала определяет среды базы данных, установленные для данных, на которые имеются ссылки, используя услуги контроллера базы данных, чтобы получить доступ к локальным распределенным данным в той же самой компьютерной системе. Если необходимые требования недоступны как часть локальных распределенных данных, контроллер распределения обращается к другим контроллерам распределения в другой компьютерной системе, чтобы найти необходимые 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 услуги или всю запрашиваемую услугу. Это достигается путем определения контроллером распределения оптимального использования ресурсов функционирования и коммуникации.

6.6.2 Роль контроллера распределения и пар уровней

Запрос от пользователя может быть связан с любой парой уровней, хотя пользователю, вводящему запрос, не требуются сведения о вовлеченной паре уровней. Если подобный запрос получен контроллером распределения, он должен определить вовлеченную пару уровней.

Контроллер распределения на основании запроса обеспечивает доступ к локальным распределенным данным с целью определить, в какой среде базы данных расположены необходимые данные и может ли запрашиваемая услуга быть выполнена в этой среде базы данных.

6.7 Модель экспорта/импорта

Модель экспорта/импорта является специализированной общей моделью, отражающей услуги, предоставляемые контроллером базы данных, включая возможности экспорта и импорта, как описано в 4.10.4.

Использование услуги экспорта требует определения данных, которые должны быть экспортированы из среды базы данных (экземпляры конкретного типа данных, количество 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 наличия имени и формата файла, в который данные были экспортированы.

Архитектурная модель экспорта и импорта основана на декомпозиции, отделяющей контроллеры базы данных, обеспечивающие услуги для любого из типов среды базы данных (6.4), от процессора экспорта-импорта, обеспечивающего необходимые услуги. На рисунке 12 показано использование услуг экспорта и импорта процессором пользователя. В распределенной среде могут быть использованы услуги контроллера распределения.

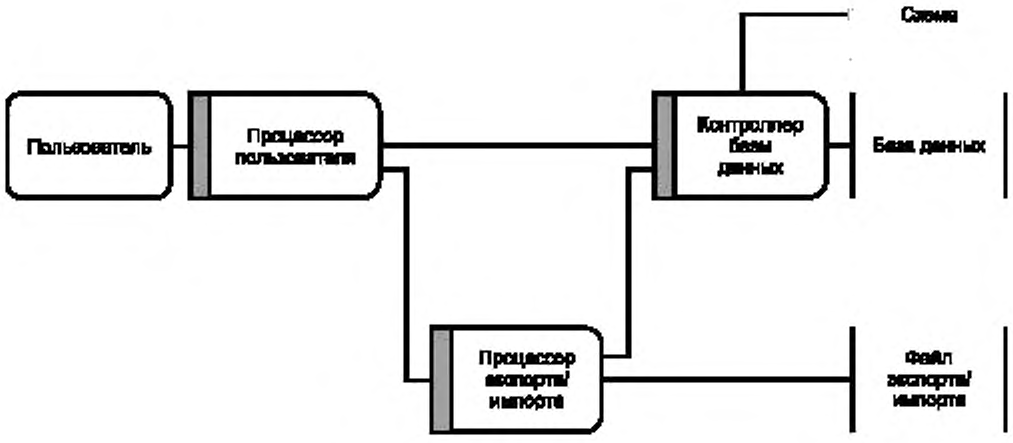


Рисунок 12 — Модель экспорта/импорта

6.8 Управление доступом для управления данными

Для услуг управления данными может потребоваться способность контролировать все формы управления доступом. Управление доступом может обеспечиваться процессором управления данными и может включать в себя услуги, необходимые для определения данных управления доступом. Управление доступом может быть применимо к нескольким процессорам.

Управление доступом требует, чтобы каждая услуга, обеспечиваемая любым процессором, могла быть специально авторизована, каждый запрос на услугу должен быть явно или неявно связан для пользователя с аутентичным идентификатором. Этот идентификатор может быть неявно связан с запросом услуги с помощью декларирования при запуске сеанса, который включает в себя услуги управления данными.

Процессор, проверяющий каждую авторизацию использования услуг, предоставляемых процессором, называют процессором управления доступом. Существуют несколько способов, которыми процессор управления доступом может быть связан с другим процессором, включая ситуации, когда:

a) процессор управления доступом используется управляющим процессором в качестве сервера;

b) процессор управления доступом требует услуг, предоставляемых управляющим процессором;

c) процессор управления доступом может быть встроен в управляющий процессор.

Независимо от используемого способа или санкционируют отдельный запрос на услугу, или связанный идентификатор устанавливает привилегии для процессов и вовлеченных данных. Если соответствующие привилегии для запроса об услуге отсутствуют, ответ на запрос указывает на нарушение управления доступом, и это нарушение может быть зарегистрировано для общего управления доступом.

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

Для определения действительных идентификаторов для среды базы данных и привилегий, связанных с этими идентификаторами, необходимы дополнительные услуги.

Один из способов управления доступом представлен на рисунке 13.

Администратор управления доступом — это частный класс пользователя, связанный с задачами установления данных управления доступом. Определитель управления доступом — это специализированный процессор пользователя, обеспечивающий услуги администраторам управления доступом.

Как показано на рисунке 13, использование контроллера базы данных может привести к неявному вызову взаимосвязанных функций управления доступом для подтверждения того, что процессор пользователя уполномочен выполнять необходимое действие. Такие функции управления доступом предусматривают извлечение данных управления доступом, соответствующих специальным требуемым операциям. Как это показано на рисунке 13, они однозначно сохраняются в качестве данных управления доступом.

Рисунок 13 иллюстрирует функциональные возможности управления доступом применительно к контроллеру базы данных. Однако управление доступом может быть уместным и для других процессоров, например для определителя управления доступом. Определитель управления доступом — это механизм установления уместных данных управления доступом, которые подтверждают наличие привилегий, существующих для операций, связанных со всеми такими процессорами.

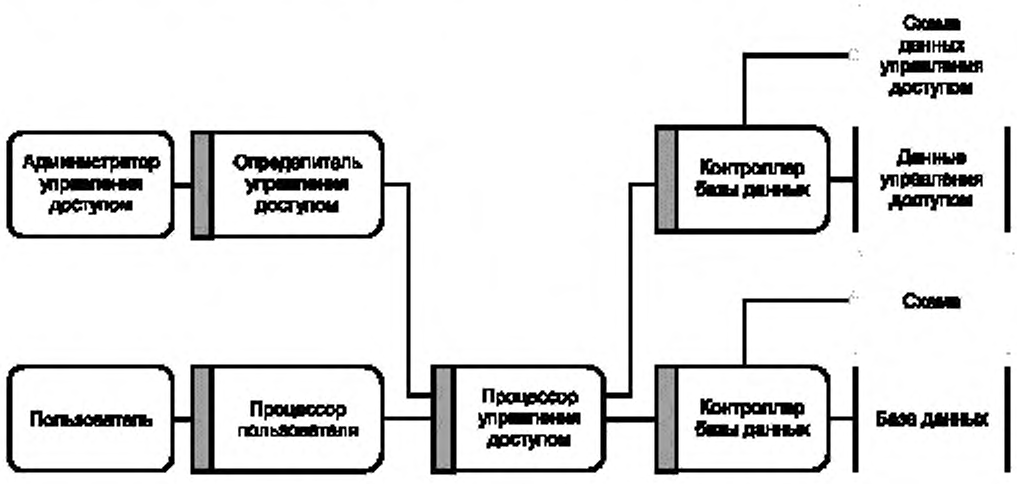


Рисунок 13 — Управление доступом в распределенной среде

7 Цели и принципы стандартизации управления данными

7.1 Цель

Настоящий раздел идентифицирует цели и принципы стандартизации управления данными.

Рассмотрены три составляющие стандартизации:

- технические цели, основанные на концепциях, приведенных в настоящем стандарте;
- средства достижения этих технических целей;
- разработка стандартов в области управления данными.

7.2 Технические цели стандартизации управления данными

Основные цели стандартизации управления данными представлены ниже. Каждая из них связана с одной или несколькими описанными главными целями:

a) поддержание сценариев распределения данных;
b) независимость размещения;
c) стандартизация управления транзакциями баз данных;
d) экспорт и импорт баз данных;
e) уменьшение сложности обработки данных;
f) обеспечение общей производительности в распределенных сценариях;
g) независимость данных;
h) мобильность приложений;
i) экстенсивное использование средств моделирования данных;
j) гибкое предоставление данных пользователю.

Последние две цели являются наиболее важными для стандартизации управления данными. В отличие от первых восьми целей они показывают дальнейшее развитие технологий. Средства достижения этих целей не могут быть идентифицированы в настоящем стандарте.

7.2.1 Поддержание сценариев распределения данных

При разработке распределенной информационной системы отсутствует необходимость использовать одного и того же поставщика для каждой компьютерной системы и системы управления данными. Такая система называется гетерогенной. База данных, доступ к которой и ее обновление могут быть обеспечены различными системами управления данными, называется гетерогенной распределенной базой данных.

Различные типы связей между процессами требуют различных средств коммуникации, которые можно обеспечить средствами операционных систем, услугами коммуникаций OSI или другими услугами коммуникаций.

Стандарты, относящиеся к обеспечению распределенных баз данных, должны быть основаны на выбранных интерфейсах услуг, предоставляемых в каждой среде управления данными. Процессоры, связанные с интерфейсами, должны реагировать стандартным (предопределенным) образом; если процессором получен запрос на услугу, он должен ответить стандартным образом.

В дополнение к предусмотренным в домене управления взаимодействиям процессоров система управления данными в одном домене управления должна быть в состоянии получить доступ к данным, управляемым другой системой управления данными в другом домене управления, в соответствии с требованиями управления доступом в функционирующем домене управления сервером. Процессоры двух доменов управления могут иметь идентичное или различное исполнение.

7.2.2 Независимость размещения

Для пользователя услуг управления данными через процессор пользователя не должно иметь значения, сохранены ли данные, к которым получен доступ, локально или удаленно.

Для обеспечения механизмов коммуникации распределенной системы, не требующей независимости размещения, услуги управления данными и интерфейсы, в которых эти услуги доступны, должны быть разработаны так, чтобы пользователь не должен был знать, где сохраняются данные.

Процессор пользователя не должен требовать понимания размещения обрабатываемых данных. Единственным исключением является процессор пользователя, используемый во время установки системы для определения размещения любой среды базы данных, с которой система может обмениваться данными.

Удаленный доступ может требовать дополнительной информации для того, чтобы определить местонахождение удаленных сохраняемых данных, но не должен требовать другого подхода к поставляемым услугам управления данными. Требования для удаленного и локального доступа к данным должны быть одинаково применимы ко всем парам уровней.

7.2.3 Стандартизация управления транзакциями баз данных

Концепция транзакции базы данных — это соответствующий механизм, направленный на решение проблем контроля целостности, восстановления и параллельности в пределах системы управления данными. Принятые свойства транзакции базы данных (ACID — атомарность, непротиворечивость, изолированность, долговечность) могут быть установлены с помощью таких концепций, как упорядочиваемость в локальной среде управления данными.

Для обеспечения взаимодействий распределенной базы данных должен быть выбран и стандартизован соответствующий механизм управления транзакциями базы данных.

7.2.4 Экспорт и импорт баз данных

Для разделения всей или части базы данных, включая данные прикладного уровня и данные любой пары уровней, должно быть обеспечено сопровождение обмена данными между системами управления данными.

Обеспечение различных форм применения системы управления данными требует представления данных в стандартной форме для их передачи, если такие данные должны быть поняты в более чем одной системе управления данными. Средства экспорта/импорта особенно важны при разделении информации анализа и проектирования между системами словарей и компьютером с установленными средствами системного проектирования.

Должны быть установлены требования к стандартной форме представления архивных баз данных. При этом также могут быть использованы стандартные форматы данных.

Средства экспорта/импорта вместе со средствами разгрузки и загрузки базы данных полностью независимы от операций поиска и обновления. Причины такого разветвления зависят от степени, до которой средства моделирования и форматы данных являются собственностью индивидуальных поставщиков.

В выбранных процессах требуется указать часть базы данных, необходимую для создания файла экспорта/импорта, которая не должна значительно отличаться по семантике от обычных поисковых утверждений (операторов).

Форму представления данных уровня схемы, связанных с данными более низкого уровня, можно рассматривать как потенциальную часть файла экспорта/импорта. Однако поскольку такие метаданные по сути неотличимы от данных более низкого уровня, например уровня прикладных данных, можно использовать те же самые средства моделирования данных и применять одинаковый подход для всех уровней данных, для которых могут быть созданы файлы экспорта/импорта.

Принципы эталонной модели состоят в том, что должны быть использованы выбранные операции и соответствующие им стандартные языки базы данных в определении части базы данных, для которой запрашивается файл экспорта/импорта. Может возникнуть необходимость расширить поисковую семантику языка манипулирования данными для того, чтобы учесть особые требования экспорта/импорта, такие как обеспечение сопровождения данных на уровне схемы прикладного уровня данных.

Для случая интерфейса услуг ответ на такой запрос будет означать, что результатом поисковых операций может быть сообщение (которое может быть длинным). Ответ может быть использован для различных целей (т.е. возможно хранение или коммуникация данных как файла без представления их пользователю).

7.2.5 Уменьшение сложности обработки данных

Большая сложность существующих распределенных систем управления данными может быть непосредственно приписана трудностям, с которыми сталкиваются при отображении данных в различных средствах моделирования данных.

Для преодоления подобных трудностей применяют стратегию, заключающуюся в использовании стандартного подхода к устранению различий между средствами моделирования данных различных систем управления данными в гетерогенной среде.

В соответствии с настоящим стандартом принят подход, позволяющий устранить потребность в учете потенциально различных средств моделирования данных, используемых локальными системами управления данными.

Несмотря на то, что обмен данными между средствами моделирования данных может представить интерес для поставщиков, желающих использовать интерфейс систем управления данными, не соответствующих выбранным стандартным средствам моделирования данных, нет необходимости отражать такие модели обмена данными в стандартах управления данными.

7.2.6 Общая производительность в распределенных сценариях

В каждом из распределенных сценариев, особенно в распределенной базе данных и объединенном сценарии базы данных, объектом стандартизации является повышение общей производительности.

7.2.7 Независимость данных

Независимость данных (см. 4.5) важна во всех информационных системах и ее необходимо использовать для того, чтобы уменьшить затраты на модификацию и модернизацию системы.

7.2.8 Обеспечение мобильности приложений

Целью стандартизации управления данными является повышение мобильности прикладных систем от одной компьютерной системы к другой.

7.2.9 Экстенсивное использование средств моделирования данных

Любые средства моделирования данных должны обладать свойством экстенсивности, необходимым для внедрения и использования новых требований и технологий.

7.2.10 Обеспечение гибкого представления данных пользователю

Один процессор любой информационной системы следует рассматривать как процессор пользователя, ответственный за координацию взаимодействия с пользователем.

Может возникнуть потребность в различии между структурами данных, видимых пользователю, и их соответствующим использованием при обмене средств моделирования данных.

Такие пользовательские интерфейсы должны быть всесторонне рассмотрены для обеспечения надежности средств управления основными данными независимо от связи этих средств с централизованными или распределенными системами.

В область применения стандартизации управления данными обычно не включают стандартизацию интерфейсов в различных типах процессоров пользователя.

7.3 Средства достижения целей

Первые восемь целей стандартизации управления данными могут быть достигнуты с помощью описанных ранее следующих средств:

a) общих средств моделирования данных для каждой пары уровней;

b) общего механизма обмена для всех пар уровней;

c) одинаковых процессоров, удобных для всех пар уровней;

d) стандартизованных услуг в интерфейсе контроллера базы данных;

e) стандартизованного подхода к управлению доступом;

f) стандартизованного представления данных, необходимых для средств взаимодействия;

g) обеспечения работы с фрагментацией данных;

h) разделения логических и физических структур;

i) доступа к схеме во время выполнения.

7.3.1 Общие средства моделирования данных для каждой пары уровней

Система управления данными должна использовать одно и то же стандартное средство моделирования данных для всех пар уровней, установленных в стандартах, и обеспечивать одинаковые процессы манипулирования данными для данных уровня схемы и базы данных, управляемой схемой.

Правила структурирования данных для одного средства моделирования данных могут быть определены с использованием других средств моделирования данных. В этом случае важно различать определяемые и ранее определенные средства моделирования данных. Это возможно в правилах структурирования данных для средств моделирования данных, которые должны быть определены и управляемы в соответствии с правилами для тех же средств моделирования данных.

Для представления способов позиционирования других средств моделирования данных необходимо идентифицировать определенные пары уровней. На более ранних стадиях жизненного цикла информационных систем обычно применяют метод разработки прикладной модели данных с использованием средств моделирования данных, отличных от использованных в соответствующей паре уровней.

7.3.2 Общий механизм обмена для всех пар уровней

Настоящий стандарт учитывает различные средства моделирования данных. Стандартизация средств моделирования данных может быть использована для поддержания обмена между различными средами управления данными, распределенными с помощью управления данных.

Средства моделирования данных, выбранные в качестве стандарта для распределенной системы управления данными, называют средствами моделирования обмена данных.

Пользователь услуг, обеспеченных процессором пользователя, не обязательно должен знать, какие средства моделирования данных используются в качестве средства моделирования обмена данных.

Важно, чтобы одинаковые средства моделирования обмена данных поддерживались в различных доменах управления, если между ними возможен обмен данными. Средства моделирования обмена данных и централизованные средства моделирования данных в паре уровней должны быть теми же средствами моделирования данных.

7.3.3 Использование одних и тех же процессоров для всех пар уровней

Должно быть установлено точное представление схемы в соответствующей форме для стандартных средств моделирования данных.

Если процессоры, например различные контроллеры базы данных и процессоры пользователя, должны совместно использовать такую информацию, то применяют стандартизованную форму представления данных.

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

7.3.4 Стандартизованные услуги в интерфейсе контроллера базы данных

Услуга интерфейса контроллера базы данных должна быть стандартизована для облегчения взаимодействия различных сред базы данных.

Эти услуги могут быть подмножеством задачи поддержания стандарта языка базы данных, но в качестве альтернативы и для соответствующего представления может потребоваться специальная форма.

7.3.5 Стандартизованный подход к управлению доступом

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

7.3.6 Стандартизованное представление данных, необходимых для средств взаимодействия

Для управления распределенной базой данных необходима база данных с информацией о размещении фрагментов данных, которая также может быть распределенной.

В случае централизованной системы управления данными информация о размещении данных относится к деталям конкретной программно-аппаратной среды.

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

7.3.7 Обеспечение работы с фрагментацией данных

При необходимости поддерживать гетерогенные распределенные базы данных и их способность к взаимодействию, 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 адресации горизонтальной и вертикальной фрагментации в распределенных базах данных.

Система управления данными должна использовать такую информацию о размещении для того, чтобы обеспечить некоторую степень независимости для прикладных процессов. Результаты запроса на информацию, которая требует получения данных из многих наборов данных, должны быть объединены для представления данных по запросу.

7.3.8 Разделение логических и физических структур

Способ структурирования данных в соответствии с правилами структурирования данных средств моделирования соответствует способу логического структурирования данных. Логическая структура данных должна быть отделена от способа, в соответствии с которым данная логическая структура должна быть представлена в качестве постоянных данных в устройстве хранения.

Часто требуется изменить физическую структуру данных. Потребность в изменении может быть вызвана логической реструктуризацией. В любом случае следует обеспечивать соответствие физического представления данных определению данных.

7.3.9 Доступ к схеме

Требование единственного источника данных, управляемого системой словарей информационных ресурсов, можно выполнить с помощью системы словарей данных, активной относительно других процессоров, используемых в информационной системе.

Управление данными, сохраняемыми в базе данных, обрабатывается системой словарей, в которой метаданные доступны таким же способом, как и схема базы данных.

Требования к форме данных допускают поиск данных схемы любым зарегистрированным пользователем соответствующего стандартного процессора.

Должна быть предусмотрена возможность модифицировать схемы данных. Определение средств манипулирования схемой должно касаться не только их влияния на стандартное представление схемы данных, но следует также принимать во внимание влияние на связанные с ней данные, существующие в базе данных, в которой схема сохраняется.

Требования непротиворечивости должны предотвращать модификацию данных уровня схемы без адекватного изменения данных, содержащихся в базе данных, управляемых схемой.

Такие требования могут быть выполнены с помощью преобразования базы данных средствами манипулирования схемой, если, кроме того, будет проведена соответствующая модификация существующей базы данных. Услуги могут включать в себя требования к преобразованиям данных, обеспечивающие непротиворечивость обновленных данных схемы.

Такие услуги модификации схемы должны быть применимы к любой паре уровней в области действия стандартизации управления данными. Вопрос о фактическом влиянии подобных изменений, в особенности о недоступности базы данных для изменений, находится вне области применения настоящего стандарта.

Важно учесть, что средства манипулирования схемой должны быть стандартизованы с точки зрения их влияния на стандартизованное представление данных уровня схемы и на уместные данные более низкого уровня.

7.3.10 Средства моделирования данных пользователя, отличные от средств моделирования обмена данных

В локальной системе управления данными может быть использовано более одного средства моделирования обмена данных. Должна быть предусмотрена система конвертирования средств моделирования данных в средства моделирования обмена данных, поддержание которых обеспечивается в специальной компьютерной системе. Эта система должна быть дополнительно разработана при определении архитектурной структуры системы управления данными для возможного использования в распределенной среде.

7.4 Аспекты стандартизации управления данными

Стандартизация управления данными, общими для нескольких информационных систем, должна создавать удобства как для персонала, использующего эти системы, так и для программных средств, которых эти системы обеспечивают.

Необходимо идентифицировать различные категории стандарта управления данными, а также понимать роль средств моделирования данных в стандартах управления данными.

7.4.1 Категории стандартов управления данными

Стандарты управления данными могут быть разделены на четыре категории, которые не обязательно являются взаимоисключающими. Внутри этих категорий необходимо различать стандартизацию структуры данных и экземпляров данных.

Первая категория стандартов управления данными определяет вид услуг, обеспечиваемых в интерфейсе. На стандарты этой категории ссылаются как на стандарты интерфейса. Такие стандарты не должны предусматривать излишние ограничения способов конструирования функционального элемента, обеспечивающего услуги.

Вторая категория стандартов управления данными использует правила и соглашения средств 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 категория стандартов управления данными — это стандарты обмена. Стандарты этой категории используются в распределенных системах, а также для связи между информационными системами в различных доменах управления при передаче данных из одной среды управления данными в другую. Стандарты обмена определяют физическое представление данных.

Четвертая категория — это функциональные стандарты, которые являются набором других стандартов. Функциональные стандарты могут иметь различные формы в зависимости от категорий отдельных стандартов и от особенностей средств моделирования данных.

7.4.2 Роль средств моделирования данных в стандартизации

Понятие средства моделирования данных является фундаментальным в концепции стандартов управления данными. Существуют три пути использования средств моделирования данных в стандартах управления данными:

a) средства моделирования данных могут быть предметом отдельного стандарта;

b) средства моделирования данных могут быть явно использованы как инструмент определения технических требований другого стандарта;

c) правила структурирования данных и правила манипулирования данными средств моделирования данных могут быть неявно использованы в другом стандарте (например, в качестве стандарта для языка базы данных).

7.4.3 Стили стандартизации

При разработке стандартов в области информационных технологий могут быть использованы различные стили, некоторые из них не зависят от категории стандарта (см. 7.4.1).

Идентифицированы следующие стили:

a) абстрактный синтаксис;

b) панели (абстрактный формат экрана);

c) конкретный синтаксис;

d) вызов процедуры;

e) условные обозначения (например, используемые в услугах OSI).

Стандарт абстрактного синтаксиса определяет семантику ряда услуг без предписания лингвистической формы, используемой при инициализации или вызове каждой услуги.

Интерфейс панели подобен абстрактному синтаксису, но предписывает упаковку услуг в качестве стандарта для пользователя-человека.

Конкретный синтаксис может быть применен для описания лингвистической формы, используемой при инициализации или вызове каждой услуги.

Интерфейс вызова процедуры определяет упорядоченный набор параметров и соответствующих обязательных правил для формулировки запроса стандартного языка программирования. Вызов процедуры может также быть неявным, т.е. формулировка запроса может быть использована для перевода в формулировку запроса до компиляции.

Вызов процедуры может инициировать процессор, включающий в себя данные допуска другой компьютерной системы. Этот вид вызова процедуры иногда называют «удаленным вызовом процедуры». Стандарт для вызова процедуры должен описывать услуги, предоставляемые в локальном масштабе и при удаленном доступе.

Условные обозначения — стандартный набор независимых условных обозначений языков программирования, необходимый для определения перечня параметров и базовых элементов (примитивов), используемых в среде открытых систем.

Любой из этих стилей может быть использован для определения стандарта интерфейса. Абстрактный или конкретный синтаксис может быть использован для того, чтобы определить стандарт контекста или стандарт обмена. Следует установить специальный стиль для определения функционального стандарта, а также стиль для каждого индивидуального стандарта в терминах, в которых он определен.

Приложение А
(справочное)

Взаимосвязанные международные стандарты

Одновременно с подготовкой настоящего стандарта в рамках ИСО/МЭК ТК1/ПК32 завершена или проводится разработка стандартов управления данными в различных областях, включенных в следующий список взаимосвязанных стандартов:

ИСО/МЭК 9075 (все части) Информационная технология. Языки базы данных. Язык структурированных запросов (SQL) (см. [1] — [9]);

ИСО/МЭК 9579:2000 Информационные технологии. Дистанционный доступ к базе данных для языка структурированных запросов (SQL) с расширением безопасности (см. [10]);

ИСО/МЭК 10027:1990 Информационная технология. Структура системы словарей информационных ресурсов (IRDS) (см. [11]);

ИСО/МЭК 10728:1993 Информационная технология. Интерфейс услуг системы словарей информационных ресурсов (IRDS) (см. [12]).

Приложение В
(справочное)

Взаимосвязь действующих и разрабатываемых стандартов по базам данных с архитектурой эталонной модели управления данными

В.1 Цель

Приложение содержит информацию о принятых и разрабатываемых стандартах JTC 1/SC21/WG3 (Подкомитет ИСО ТК1 по информационным технологиям), связанных с эталонной моделью управления данными, описанной в настоящем стандарте. Настоящее приложение включает в себя описание взаимосвязи упомянутых стандартов с моделью и приводит краткий обзор средств и ограничений, применяемых в эталонной модели управления данными.

В.2 Языки базы данных

В.2.1 Язык базы данных SQL

Язык базы данных SQL (см. ИСО/МЭК 9075 (все части) [1] — [9]) — стандартный язык, который устанавливает синтаксис и семантику утверждений для определения и манипулирования установленными типами данных, процессами транзакций и управления доступом, которые поддерживают использование баз данных SQL.

Язык базы данных используют для определения запросов об услугах в процессоре пользователя для вызова услуг контроллера базы данных, поддерживающего SQL.

Термины SQL — это термины прикладного использования эталонной модели управления данными, рассматриваемой в разделе 6. Термины, используемые в стандарте на SQL, связаны с терминами, используемыми в разделе 6 настоящего стандарта следующим образом (таблица В.1):

Таблица В.1 — Соотношение терминов: раздел 6 — SQL[1]/RMDM[2]

Термин SQL	Термин эталонной модели управления данными
Агент SQL	Процессор пользователя
Приложение SQL	Система управления базы данных
Среда SQL	Одна или несколько сред базы данных
Сервер SQL	Контроллер базы данных
Клиент SQL	Клиент контроллера базы данных (может быть процессором пользователя или контроллером распределения)
Агент не SQL	Процессор пользователя

Стандарт на SQL детализирует понятие процессора пользователя, различая понятия агента SQL и клиента SQL; клиент SQL может одновременно быть агентом SQL.

Другие термины стандарта на SQL, связанные с терминами эталонной модели управления данными, используемыми в разделе 5 настоящего стандарта, следующие (таблица В.2):

Таблица В.2 — Соотношение терминов: раздел 5 — SQL/RMDM

Термин SQL	Термин эталонной модели управления данными
Каталог[3]	Совокупность схем
База данных	База данных
Данные SQL	База данных плюс схема

---

[1] SQL — Structured Query Language — Язык структурированных запросов.

[2] RMDM — Reference Model of Data Management — Эталонная модель управления данными.

[3] *Большинство SQL баз данных используют набор SQL таблиц, хранящих служебную информацию для своих внутренних потребностей. Этот набор называется в различных публикациях как системный каталог, словарь данных или просто системные таблицы. Таблицы системного каталога напоминают обычные SQL таблицы: те же строки и столбцы данных. Например, одна таблица каталога обычно содержит информацию о таблицах, существующих в базе данных, по одной строке на каждую таблицу базы данных; другая содержит информацию о различных столбцах таблиц, по одной строке на столбец, и т. д. Таблицы каталога создаются и присваиваются с помощью самой базы данных, и идентифицируются с помощью специальных имен, таких, например, как SYSTEM. База данных создает эти таблицы и модифицирует их автоматически; таблицы каталога не могут быть непосредственно подвергнуты действию команды модификации. (Примеч. перев.).*

Среда пользователя — первоначально это автономная среда базы данных, однако она может быть расширена при ее соединении с другими средами базы данных.

В стандарте на SQL термин «пара уровней» явно не используется. Однако стандарт различает «схемы» и информационную схему. В каталоге существует только одна информационная схема. База данных, связанная с информационной схемой, содержит данные о всех схемах в каталоге, включая непосредственно информационную схему.

Применяя термины стандарта на SQL и используя условные обозначения раздела 3 настоящего стандарта, общую модель (см. рисунок 8) можно представить в следующем виде (рисунок В.1).

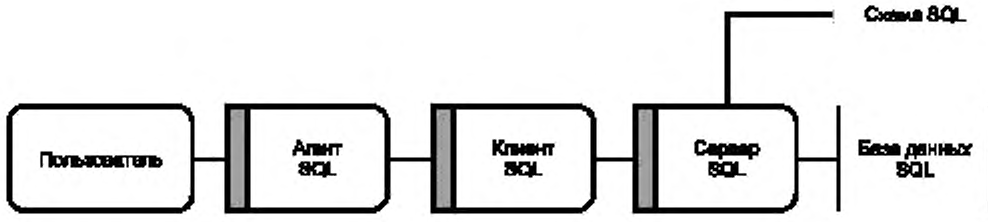


Рисунок В.1 — Общая модель управления данными применительно к SQL

Аналогично модель управления распределенными данными (рисунок 11) может быть представлена в следующем виде (рисунок В.2).

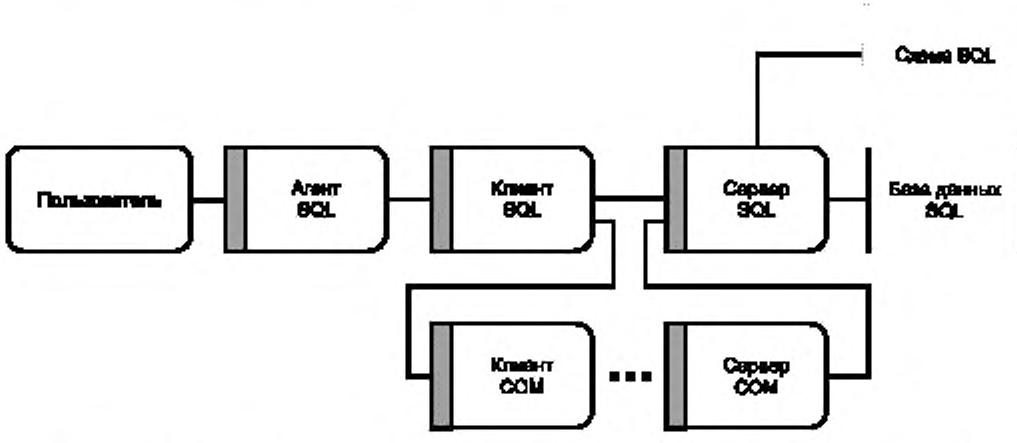


Рисунок В.2 — Модель управления распределенными данными применительно к терминам SQL

Различие между рисунками 11 и В.2 состоит в том, что согласно рисунку В.2 клиент SQL должен определить, с какой удаленной средой SQL необходимо соединяться. На рисунке 11 контроллер распределения является создателем этого определителя, а на рисунке В.2 процессор, названный «Сервер COM[1)]», расположен в среде SQL, отличной от процессора, названного «Клиент COM».

Язык базы данных SQL может быть использован для услуг, перечисленных для контроллера общей базы данных в 6.3, за исключением:

h) процедур создания резервных копий базы данных;

i) процедур восстановления базы данных;

j) физической реорганизации базы данных.

Каждую из этих услуг можно обеспечить путем применения контроллера базы данных SQL определенным прикладным способом. Язык базы данных SQL также может быть использован для определения данных управления доступом, как описано в 6.8.

[1)] COM (Component Object Model) — модель компонентных объектов.

B.2.2 NDL

Язык базы данных NDL[1] — стандартный язык базы данных, который устанавливает синтаксис и семантику утверждений для определения и манипулирования установленными типами данных, процессами транзакций и сопровождения баз данных NDL.

Язык базы данных применяют для определения запросов об услугах в процессоре пользователя. Этот язык может быть использован для вызова услуг контроллера базы данных, поддерживающего NDL. Стандарт NDL не содержит термина, аналогичного контроллеру базы данных, и поэтому использует термин «контроллер базы данных NDL». Использование такого контроллера базы данных NDL показано на рисунке B.3.

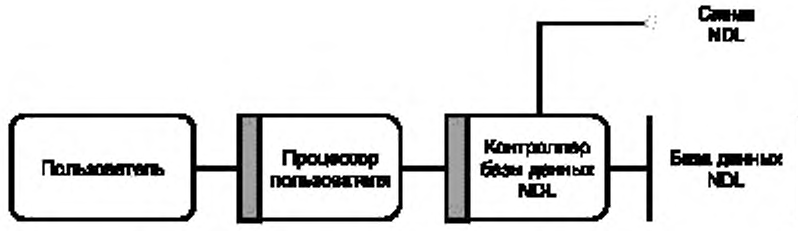


Рисунок B.3 — Размещение контроллера базы данных NDL

Язык базы данных NDL используют для услуг, перечисленных для контроллера общей базы данных в 6.3, за исключением:

b) модификации определений данных;

h) процедур создания резервных копий базы данных;

i) процедур восстановления базы данных;

j) физической реорганизации базы данных.

Каждую из этих услуг можно обеспечить путем применения контроллера базы данных NDL определенным прикладным способом.

B.3 Системы словарей информационных ресурсов (IRDS)

B.3.1 Структура IRDS

Стандарт структуры IRDS[2] ИСО/МЭК 10027 [11] определяет структуру, в которой могут быть разработаны стандарты серии IRDS. Структура идентифицирует три определенные пары уровней следующим образом:

- пара уровней определения IRD[3];
- пара уровней IRD;
- прикладная пара уровней.

Стандарты IRDS направлены на первые две из этих трех пар уровней. Пары уровней – это прикладное использование понятия пары уровней, установленное в разделе 5 настоящего стандарта.

Главный процессор, идентифицированный в структуре, — это процессор интерфейса услуг IRDS. Термины, используемые в стандарте структуры IRDS, связаны с терминами, используемыми в настоящем стандарте, следующим образом (таблица B.3):

Таблица B.3 — Соотношение терминов: IRDS/RMDM

Термин структуры IRDS	Термин эталонной модели управления данными
Прикладная программа или интерфейс человек — машина IRDS	Процессор пользователя
(Нет эквивалента)	Система управления базы данных
(Нет эквивалента)	Среда базы данных
Процессор услуг базы данных	Контроллер базы данных
Процессор интерфейса услуг IRDS	Клиент контроллера базы данных

[1] NDL — Network Database Language.

[2] IRDS — Information Resource Dictionary System.

[3] IRD — Information Resource Dictionary.

Другие термины стандарта структуры IRDS, связанные с терминами эталонной модели управления данными, используемыми в разделе 5 настоящего стандарта, приведены в таблице В.4.

Таблица В.4 — Соотношение терминов: раздел 5 — IRDS/RMDM

Термин IRDS	Термин эталонной модели управления данными
Схема определения IRD	Схема
Схема IRD	Схема
Определение IRD	База данных
IRD	База данных

В структуре IRDS и связанной серии стандартов используется термин «пара уровней». Структура IRDS требует применения одинаковых средств моделирования данных и для определения данных на уровне IRD, и для определения данных на уровне определения IRD.

Стандарт структуры IRDS подробно устанавливает понятие процессора пользователя при определении процессора синтаксиса командного языка, процессора интерфейса панели, процессора интерфейса командного языка человека и других процессоров IRDS в интерфейсе человек — машина.

При использовании условных обозначений раздела 3 настоящего стандарта и терминов стандарта структуры IRDS общая модель, представленная на рисунке 8, принимает следующий вид (рисунок В.4).

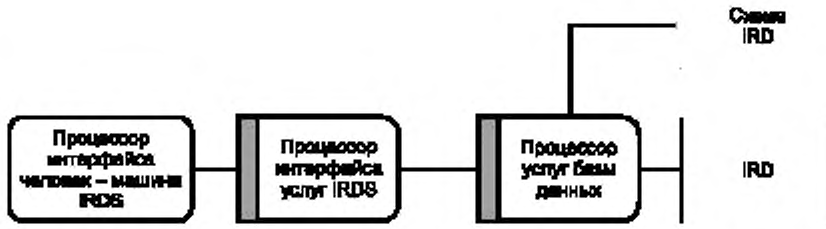


Рисунок В.4 — Общая модель управления данными применительно к IRDS

На рисунке В.4 процессор услуг базы данных — это прикладное использование контроллера базы данных, а процессор интерфейса человек — машина IRDS — это прикладное использование процессора пользователя. Существенным различием между рисунком 8 и рисунком В.4 является то, что процессор интерфейса услуг IRDS является клиентом процессора услуг базы данных, но не процессором пользователя.

В.3.2 Интерфейс услуг IRDS

Стандарт интерфейса услуг IRDS (ИСО/МЭК 10728 [12]) определяет ряд таблиц для схемы определителя IRD. Эти таблицы определителей IRD должны быть заполнены пользователем словаря и содержать описание одной или нескольких схем IRD. Таблицы определителей IRD установлены в стандарте [12], использующем ИСО/МЭК 9075 (все части) ([1] - [9]) для утверждений по описанию схемы. Средства моделирования данных, используемые для определения схемы IRD, основаны на этих же стандартах SQL ([1] - [9]) аналогично определению схемы определителя IRD. Необходимо также обеспечить средства управления версиями.

Стандарт также определяет ряд доступных для пользователя услуг процессора интерфейса услуг IRDS для манипулирования данными и относительно пары уровней определителя IRD, и относительно пары уровней IRD.

В.4 Удаленный доступ к базам данных RDA[1)]

Стандарт ИСО/МЭК 9679 [10] определяет способ использования протоколов OSI для обеспечения требований пользователей одной компьютерной системы при получении доступа к контроллеру базы данных в другой компьютерной системе.

Стандарт ИСО/МЭК 9679 [10] определяет способ использования протоколов OSI для обеспечения требований пользователей клиента SQL в одной компьютерной системе при получении доступа к серверу SQL в другой компьютерной системе.

Используя стандарт RDA SQL, модель распределенного управления данными (рисунок 11) можно представить в следующем виде (рисунок В.5).

[1)] RDA — Remote Database Access.

Различие между рисунками 11 и В.5 состоит в том, что согласно рисунку В.5 клиент SQL должен предоставить информацию, определяющую, с какой удаленной средой SQL необходимо соединяться, тогда как в соответствии с рисунком 11 контроллер распределения выполняет роль создания этого определения.

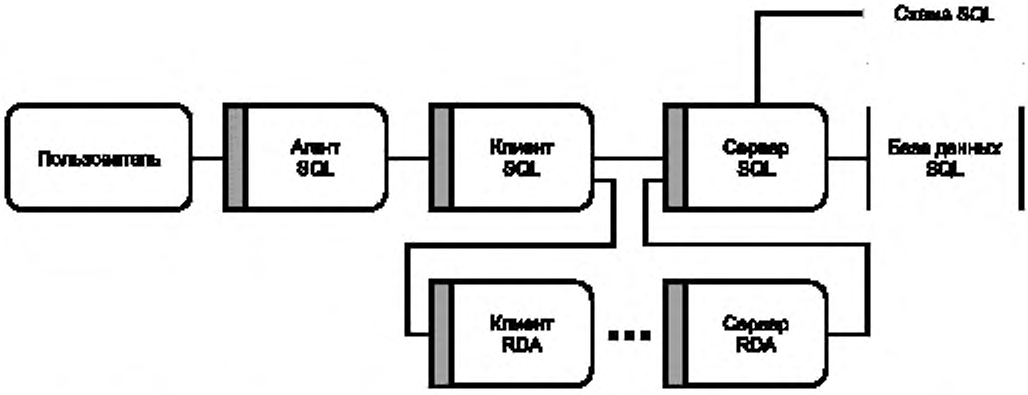


Рисунок В.5 — Модель управления распределенными данными применительно к RDA SQL

В.5 Экспорт/импорт

Стандарт разрабатывают для услуг, предоставляемых процессором экспорта/импорта, и для формата файла экспорта/импорта в соответствии с моделью по 6.7. Стандарт первоначально предназначен для экспорта/импорта базы данных SQL. Схематично это можно представить на рисунке В.6 в терминах экспорта/импорта, как было показано на рисунке 12.

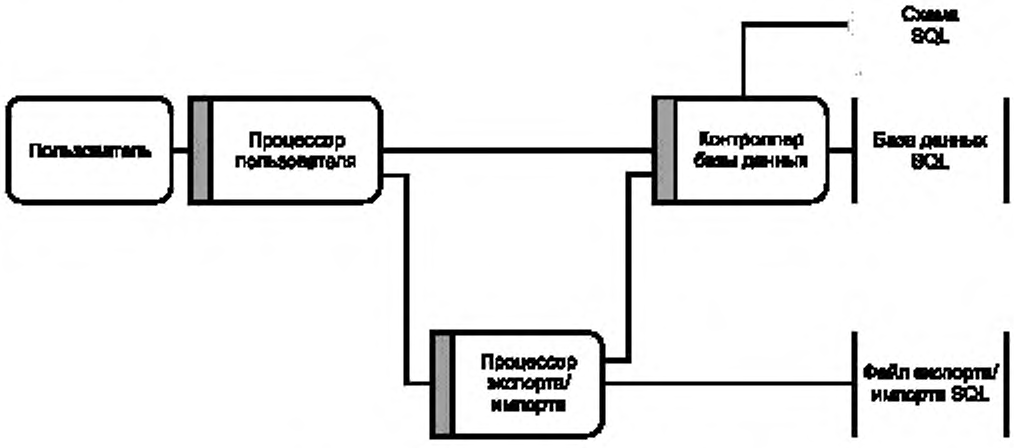


Рисунок В.6 — Экспорт/импорт в базе данных SQL

В.6 Направления стандартизации

В рамках архитектурной модели для распределенного управления данными, как показано на рисунке 11, можно выявить перспективные направления стандартизации, предусматривающие стандартную схему для распределенных данных и услуг, обеспеченных в интерфейсе контроллера распределения. Эти услуги включают в себя:

а) определение компьютерных систем, в которых должны быть сохранены данные и соответствующие процессы;

b) назначение компьютерной системы для хранения определяемых данных и связанных процессов;

c) определение компьютерной системы, в которой должны быть найдены требуемые данные и связанные процессы. Определение таких данных и связанных процессов является требованием процессора пользователя.

На рисунке В.7 показано, что связь между компьютерными системами существует только между контроллерами распределения. Для обеспечения взаимодействия услуги, необходимые для управления распределенными данными в нескольких средах базы данных, нуждаются в поддержке приложения RDA. Рисунок В.7 — развернутая версия рисунка 11, идентифицирующая специальную роль услуг RDA.

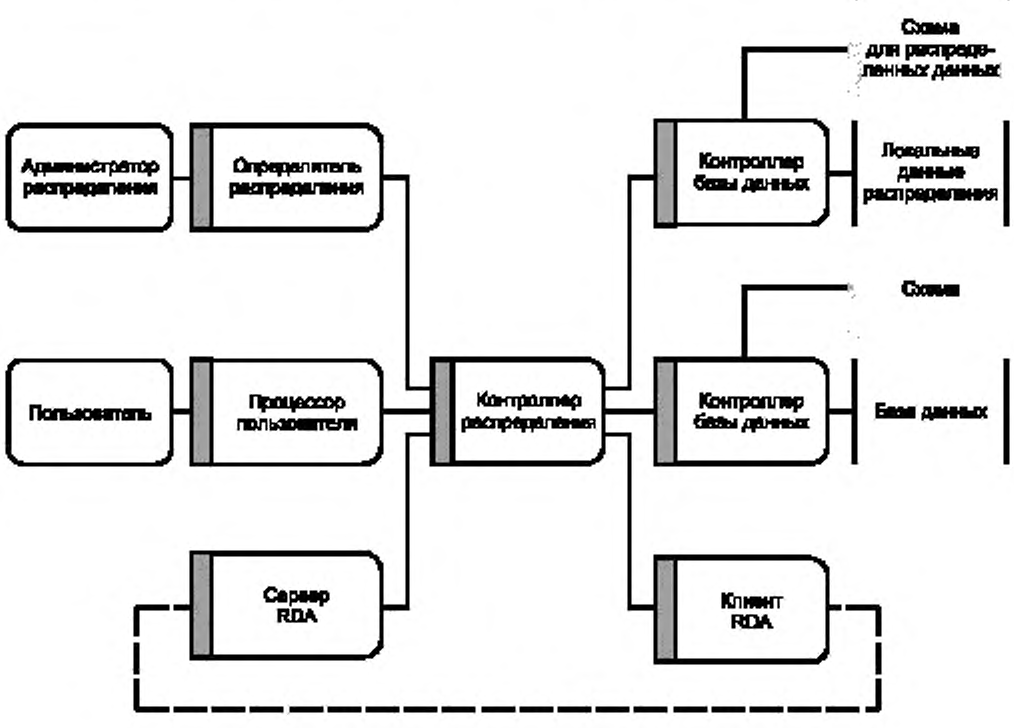


Рисунок В.7 — Положение RDA для распределенного управления данными

Библиография

[1] ИСО/МЭК 9075-1:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 1. Структура (SQL/Структура)

(ISO/IEC 9075-1:2003) (Information technology — Database languages — SQL — Part 1: Framework (SQL/Framework))

[2] ИСО/МЭК 9075-2:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 2. Основы (SQL/Основы)

(ISO/IEC 9075-2:2003) (Information technology — Database languages — SQL — Part 2: Foundation (SQL/Foundation))

[3] ИСО/МЭК 9075-3:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 3. Прикладной программный интерфейс уровня вызовов (SQL/CLI)

(ISO/IEC 9075-3:2003) (Information technology — Database languages — SQL — Part 3: Call-Level Interface (SQL/CLI))

[4] ИСО/МЭК 9075-4:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 4. Модули постоянного хранения (SQL/PSM)

(ISO/IEC 9075-4:2003) (Information technology — Database languages — SQL — Part 4: Persistent Stored Modules (SQL/PSM))

[5] ИСО/МЭК 9075-9:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 9. Управление данными от внешних источников (SQL/MED)

(ISO/IEC 9075-9:2003) (Information technology — Database languages — SQL — Part 9: Management of External Data (SQL/MED))

[6] ИСО/МЭК 9075-10:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 10. Привязка объектного языка (SQL/OLB)

(ISO/IEC 9075-10:2003) (Information technology — Database languages — SQL — Part 10: Object Language Bindings (SQL/OLB))

[7] ИСО/МЭК 9075-11:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 11. Схемы информации и определения (SQL/Схемы)

(ISO/IEC 9075-11:2003) (Information technology — Database languages — SQL — Part 11: Information and Definition Schemas (SQL/Schemata))

[8] ИСО/МЭК 9075-13:2003 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 13. Процедуры и типы SQL с использованием языка программирования Java TM (SQL/JRT)

(ISO/IEC 9075-13:2003) (Information technology — Database languages — SQL — Part 13: SQL Routines and Types Using the Java TM Programming Language (SQL/JRT))

[9] ИСО/МЭК 9075-14:2006 Информационная технология. Языки базы данных. Язык структурированных запросов (SQL). Часть 14. Спецификации, связанные с XML (SQL/XML)

(ISO/IEC 9075-14:2006) (Information technology — Database languages — SQL — Part 14: XML-Related Specifications (SQL/XML))

[10] ИСО/МЭК 9579:2000 Информационная технология. Дистанционный доступ к базе данных для языка структурированных запросов (SQL) с расширением безопасности

(ISO/IEC 9579:2000) (Information technology — Remote database access for SQL with security enhancement)

[11] ИСО/МЭК 10027:1990 Информационная технология. Структура системы словарей информационных ресурсов (IRDS)

(ISO/IEC 10027:1990) (Information technology — Information Resource Dictionary System (IRDS) framework)

[12] ИСО/МЭК 10728:1993 Информационная технология. Интерфейс сервисов системы словарей информационных ресурсов (IRDS)

(ISO/IEC 10728:1993) (Information technology — Information Resource Dictionary System (IRDS) Services Interface)

УДК 658:562.014:006.354 ОКС 35.180 Т59

Ключевые слова: управление доступом, данные управления доступом, прикладной процесс, компьютерная система, управление данными, среда управления данными, база данных, язык базы данных, управление базой данных, система управления базами данных

Редактор *Л. В. Афанасенко*
Технический редактор *В. Н. Прусакова*
Корректор *Н. И. Гаврищук*
Компьютерная верстка *Т. Ф. Кузнецовой*

Сдано в набор 23.10.2008. Подписано в печать 03.02.2009. Формат 60×84 1/8. Бумага офсетная. Гарнитура Ариал Печать офсетная. Усл. печ. л. 5,12. Уч.-изд. л. 4,90. Тираж 683 экз. Зак. 2372.

ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4.
www.gostinfo.ru gost.e-ecolog.ru
Набрано и отпечатано в Калужской типографии стандартов 248021 Калуга, ул. Московская, 256.
gost.e-ecolog.ru