Защита данных в базах данных: обеспечение целостности и безопасности данных

"Стыдно не уметь защищать себя рукою, но ещё более стыдно не уметь защищать себя словом".

Аристотель, древнегреческий философ

Общие положения

Защита данных – это организационные, программные и технические методы и средства, направленные на удовлетворение ограничений, установленных для типов данных или экземпляров типов данных в системах обработки данных (ГОСТ 20886-85).

Защита данных включает предупреждение случайного или несанкционированного доступа к данным, их изменения или разрушения со стороны пользователей или при сбоях аппаратуры. Реализация защиты включает:

- контроль достоверности данных с помощью ограничений целостности;
- обеспечение безопасности данных (физической целостности данных);
- обеспечение секретности данных.

Обеспечение целостности данных

Типы ограничений целостности в языке SQL:

1. Уникальность значения первичного ключа (PRIMARY KEY).
2. Уникальность ключевого поля или комбинации значений ключевых полей:

   UNIQUE(A),

где A – один или несколько атрибутов, указанных через запятую.

(1,2 – явные структурные ограничения целостности.)

3. Обязательность/необязательность значения (NOT NULL / NULL).

4. Задание диапазона значений атрибута Field:

   CHECK(field BETWEEN min_value AND max_value)

5. Задание взаимоотношений между значениями атрибутов Field1 и Field2:

   CHECK (field1 @ field2),

где @ – оператор отношения (например, знак ">").

Обеспечение целостности данных

6. Задание списка возможных значений (констант) для атрибута Field:
   CHECK (field IN (value1, value2,…, valueN)).
7. Определение формата атрибута Field (даты, числа и др.). Например:
   CHECK (field LIKE '_ _ _-_ _-_ _') -- формат телефонного номера
8. Определение домена атрибута на основе значений другого атрибута: множество значений некоторого атрибута отношения является подмножеством значений другого атрибута этого или другого отношения (внешний ключ, FOREIGN KEY).

(3.-8. – явные ограничения целостности на значения данных.)

9. Ограничения на обновление данных (например, каждое следующее значение атрибута должно быть больше предыдущего). В SQL напрямую не реализуется, требует использования специальных возможностей СУБД (триггеров).
10. Ограничения на параллельное выполнение операций (механизм транзакций) и проверка ограничений целостности после окончания внесения взаимосвязанных изменений.

Обеспечение безопасности данных

Под функцией безопасности (или физической защиты) данных подразумевается предотвращение разрушения или искажения данных в результате программного или аппаратного сбоя.

Обеспечение безопасности является внутренней задачей СУБД, поскольку связано с её нормальным функционированием, и решается на уровне СУБД.

Цель **восстановления базы данных** после сбоя – обеспечить, чтобы результаты всех подтверждённых транзакций были отражены в восстановленной БД, и вернуться к нормальному продолжению работы как можно быстрее, изолируя пользователей от проблем, вызванных сбоем.

Типичные сбои и способы защиты от них

1. Сбой предложения.

СУБД автоматически откатывает результаты этого предложения, генерирует сообщение об ошибке и возвращает управление пользователю (приложению пользователя).

2. Сбой пользовательского процесса.

Система автоматически откатывает неподтверждённые транзакции сбившегося пользовательского процесса и освобождает все ресурсы, занятые этим процессом.

3. Сбой процесса сервера.

Восстановление после сбоя процесса сервера может потребовать перезагрузки БД, при этом автоматически происходит откат всех незавершённых транзакций.

Типичные сбои и способы защиты от них

4. Сбой процесса операционной системы.

В этой ситуации сервер БД не может продолжать работу, и для восстановления базы данных требуется участие человека (обычно, администратора базы данных, АБД).

5. Сбой носителя (диска).

В этой ситуации сервер БД не может продолжать работу, и для восстановления базы данных требуется участие человека (обычно, администратора базы данных, АБД).

6. Ошибка пользователя.

Ошибки пользователей могут потребовать участия человека (АБД) для восстановления базы данных в состояние на момент возникновения ошибки.

Средства физической защиты данных

Резервное копирование

Резервное копирование означает периодическое сохранение файлов БД на внешнем запоминающем устройстве. Оно выполняется тогда, когда состояние файлов БД является непротиворечивым. В случае сбоя (или аварии диска) БД восстанавливается на основе последней копии.

Полная резервная копия включает всю базу данных (все файлы БД, в том числе вспомогательные, состав которых зависит от СУБД).

Частичная резервная копия включает часть БД, определённую пользователем. Резервная копия может быть **инкрементной**: она состоит только из тех блоков (страниц памяти), которые изменились со времени последнего резервного копирования.

Создание частичной и инкрементной РК выполняется средствами СУБД, а создание полной РК – средствами СУБД или ОС (например, с помощью команды ***copy***). В резервную копию, созданную средствами СУБД, обычно включаются только те блоки памяти, которые реально содержат данные (т.е. пустые блоки, выделенные под объекты БД, в резервную копию не входят).

Резервное копирование

Периодичность резервного копирования определяется администратором системы и зависит от многих факторов: объём БД, интенсивность запросов к БД, интенсивность обновления данных и др.

Как правило, технология проведения резервного копирования такова:

- раз в неделю (день, месяц) осуществляется полное копирование;
- раз в день (час, неделю) – частичное или инкрементное копирование.

Все изменения, произведённые в данных после последнего резервного копирования, утрачиваются; но при наличии **архива журнала транзакций** их можно выполнить ещё раз, обеспечив полное восстановление БД на момент возникновения сбоя. Журнал транзакций содержит сведения только о текущих транзакциях. После завершения транзакции информация о ней может быть перезаписана. Для того чтобы в случае сбоя обеспечить возможность полного восстановления БД, необходимо вести архив журнала транзакций, т.е. сохранять копии файлов журнала транзакций вместе с резервной копией базы данных.

Восстановление базы данных

В том случае, если нельзя восстановить БД после сбоя автоматически, восстановление БД выполняется в два этапа:

1) перенос на рабочий диск резервной копии базы данных (или той её части, которая была повреждена);
2) перезапуск сервера БД с повторным проведением всех транзакций, зафиксированных после создания резервной копии и до момента возникновения сбоя.

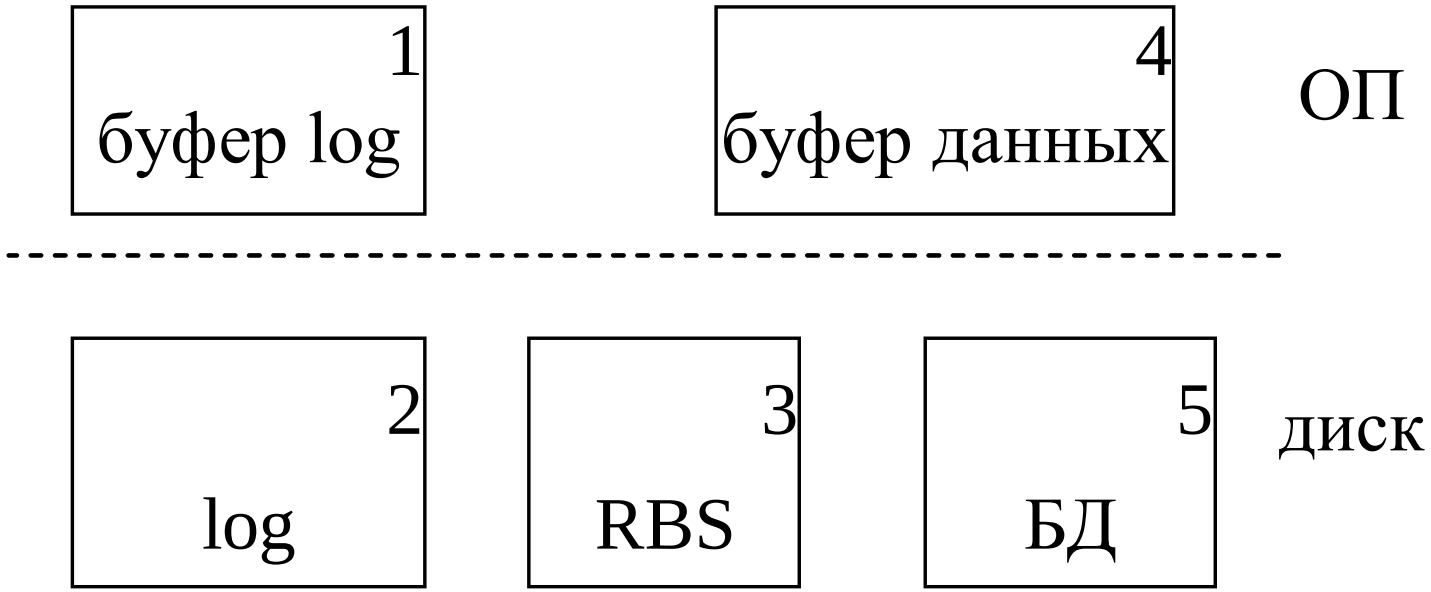
Если в системе есть архив транзакций, то повторное проведение транзакций может проходить автоматически или под управлением пользователя.

Если произошёл сбой процесса сервера, то требуется перезагрузка сервера для восстановления БД. При перезагрузке СУБД может по содержимому системных файлов узнать, что произошёл сбой, и выполнить восстановление автоматически (если это возможно). Восстановление БД в этой ситуации означает приведение всех данных в БД в согласованное состояние, т.е. откат незавершённых транзакций и проверку того, что все изменения, внесённые завершёнными транзакциями, попали на диск.

Восстановление базы данных

Прокрутка вперед заключается в применении к файлам данных всех изменений, зарегистрированных в журнале транзакций.

Прокрутка назад заключается в отмене всех изменений, которые не были подтверждены.



Восстановление базы данных

Последовательность восстановления БД:

1. Устранение проблем с диском (если они были).
2. Запись резервной копии на место основной базы данных.
3. Запуск сервера БД в режиме восстановления (recover).
4. Определение параметров восстановления при наличии архова журналов транзакций (синтаксис и возможности СУБД Oracle):

- до сигнала пользователя:

```
RECOVER  DATABASE  UNTIL  CANCEL
```

- до определенного момента времени:

```
RECOVER  DATABASE  UNTIL TIME  'YYYY-MM-DD:HH24:MI:SS'
```

- до определенной транзакции:

```
RECOVER  DATABASE  UNTIL  CHANGE  NNNNN
```

- до определенного журнала транзакций:

```
ALTER  DATABASE  RECOVER  LOGFILE  'log1.arc';
```