БЕЗОПАСНОСТЬ СИСТЕМ БАЗ ДАННЫХ

8. Обеспечение доступности

Обеспечение высокой доступности ИС и БД

- Понятия *доступности, готовности, работоспособности, надежности, отказоустойчивости* тесно взаимосвязаны. В контексте ИТ-безопасности преобладающим является термин ***доступности***, который отражает состояние ИС с точки зрения пользователя.
- Основные причины, по которым информация может быть утрачена:
  - отказ оборудования и системного ПО;
  - ошибка в прикладном ПО;
  - ошибка персонала, преднамеренное уничтожение информации.

Обеспечение высокой доступности ИС и БД

- Единственным принципом, на котором может базироваться решение проблемы обеспечения сохранности информации и, как следствие, гарантии ее доступности – это тот или иной способ ***дублирования данных***. Дублирование данных может быть:
  - постоянным, когда данные постоянно копируются в некоторую резервную среду хранения, и резервные данные соответствуют текущему состоянию данных (возможно, с некоторой задержкой);
  - реализовано путем архивации, т.е. создания копии данных в их текущем состоянии и консервации этой копии.

Достоинства и недостатки способов дублирования

- ***Постоянное дублирование*** данных позволяет в высокой степени обезопасить систему от отказа оборудования и системного ПО (1-я причина). Ошибки в прикладном ПО, а также ошибки персонала и умышленное уничтожение данных постоянным дублированием не устраняются.
- ***Архивация*** позволяет восстановить информацию после потери информации из-за любой из рассмотренных выше причин.
- Т.о. постоянное дублирование и архивация данных взаимно дополняют друг друга и должны применяться совместно.

Технологии постоянного дублирования

- Ядром ИС является аппаратная часть, ОС, СУБД и прикладное ПО. Соответственно, и постоянное дублирование может производиться на всех этих уровнях:
  - на уровне аппаратуры,
  - на уровне системного ПО (ОС),
  - на уровне СУБД
  - на уровне прикладного ПО.

Дисковые RAID-массивы

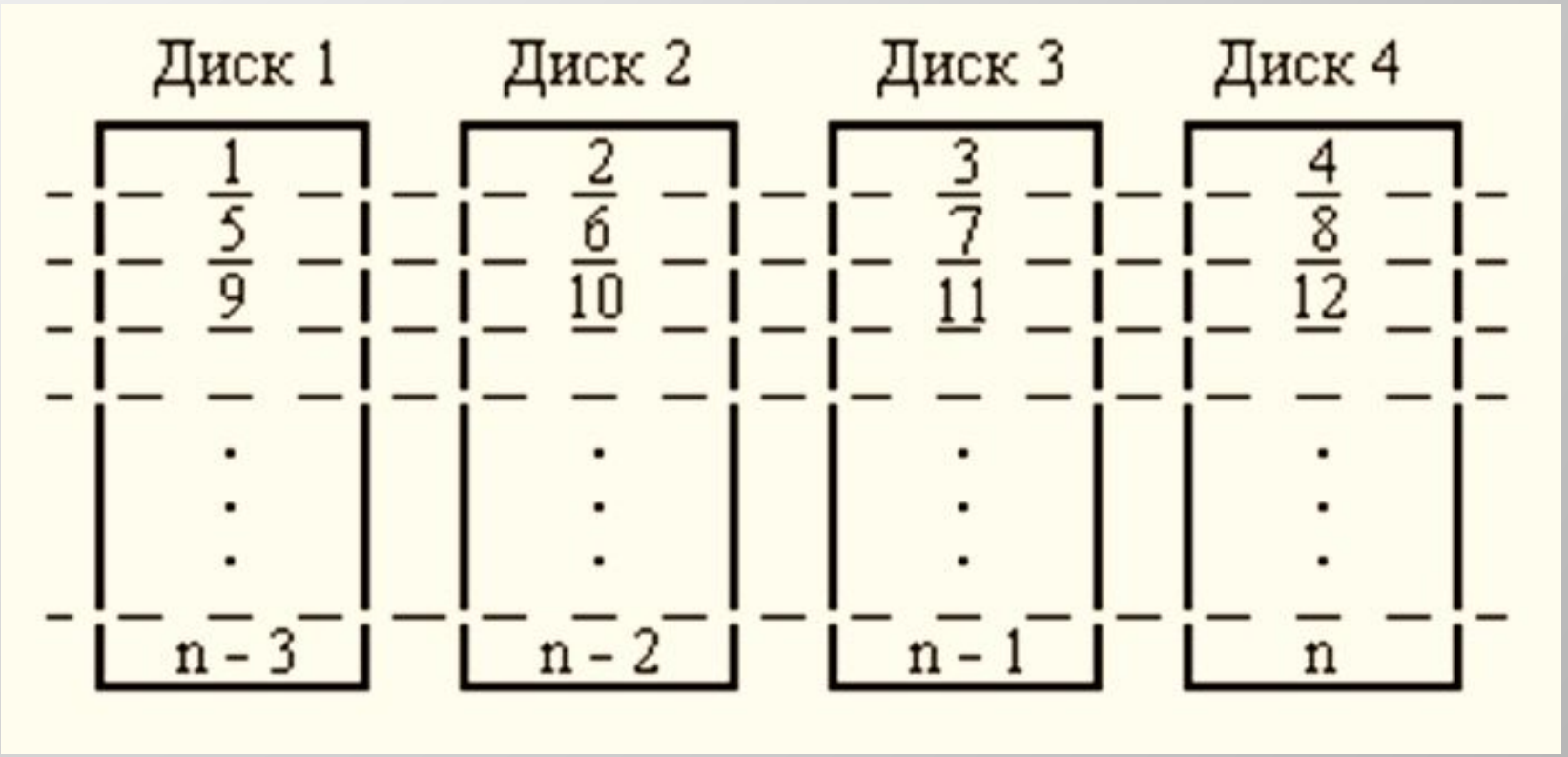
- Аппаратная избыточность может включать платформы с полным резервированием, поддерживающие процессоры, диски с двойным интерфейсом, дисковые массивы и пр.
- Постоянное дублирование информации может быть реализовано, в частности, с помощью дисковых RAID-массивов, которые представляют собой набор жестких дисков и спец. аппаратуру управления этими дисками.
- Если просто объединить несколько дисков в не избыточный массив, то среднее время между отказами (СВМО) будет равно СВМО одного диска, деленному на количество дисков.
- Существует несколько типов избыточных RAID-массивов с более высоким показателем СВМО.

Дисковые RAID-массивы

- В RAID системах для повышения надежности и производительности используются комбинации трех основных известных механизмов:
  - организация «зеркальных» дисков (полное дублирование);
  - подсчет контрольных кодов (четность, коды Хэмминга), позволяющих восстановить информацию при сбое;
  - распределение информации по различным дискам массива, что повышает возможности параллельной работы дисков. При описании RAID этот прием называют «stripped disks», что буквально означает «разделенные на полоски диски», или просто «полосатые диски».

Дисковые RAID-массивы

- Полозки в дисках (далее называемые блоками) данных чередуются, образуя единый логический диск.



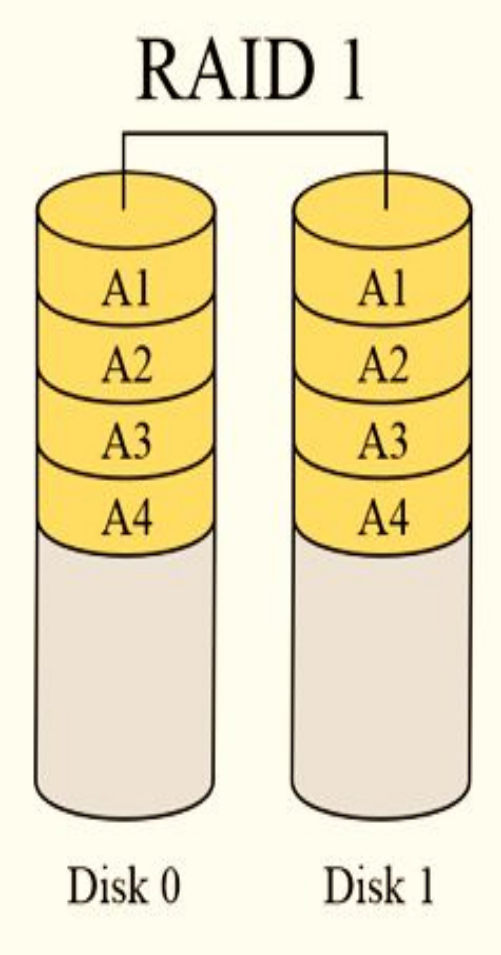
Дисковые RAID-массивы

- Определено шесть типов дисковых массивов, обозначаемых RAID 1 – RAID 6, различающихся по своим особенностям и производительности.
- Каждый из этих типов за счет определенной избыточности записываемой информации обеспечивает повышенную отказоустойчивость по сравнению с одиночным дисководом.
- Также существует массив дисков, не обладающих избыточностью, но позволяющий повысить производительность (за счет расслоения обращений). Его часто называют RAID 0.

Дисковый массив RAID 0

- Массив RAID 0 содержит группу дисков с чередованием блоков данных без контроля четности и без избыточности.
- Размеры чередующихся блоков могут быть большими или малыми, для много- и одно-пользовательских систем. соответственно
- Организация RAID 0 соответствует той, что показана на рис. выше. Операции записи и чтения могут выполняться одновременно на каждом дисководе. Минимальное количество дисководов для RAID 0 – два.
- RAID 0 обладают высокой производительностью и наиболее эффективным использованием дискового пространства.
- Надёжность меньше надежности самого ненадёжного диска

Дисковый массив RAID 1



- RAID 1. (mirroring «зеркалирование») – массив из 2-х дисков, являющихся полными копиями друг друга.
- Хранимые данные дублируются, но представляются КС как один диск. В рамках одной пары зеркальных дисков разбиение на полоски не производится, но чередование блоков может быть организовано для нескольких массивов RAID 1, образующих вместе один большой массив из нескольких зеркальных пар дисков

Дисковый массив RAID 1

- RAID 1 обеспечивает приемлемую скорость записи и чтения при распараллеливании запросов.
- Имеет высокую надёжность – работает до тех пор, пока функционирует хотя бы один диск в массиве.
- Недостаток RAID 1 в том, что по цене двух жестких дисков пользователь фактически получает лишь один.
- Все операции записи производятся одновременно в оба диска зеркальной пары. Но при чтении каждый из дисков пары может работать независимо, удваивая тем самым производительность при чтении. В этом смысле RAID 1 обеспечивает наилучшую производительность среди всех вариантов дисковых массивов.

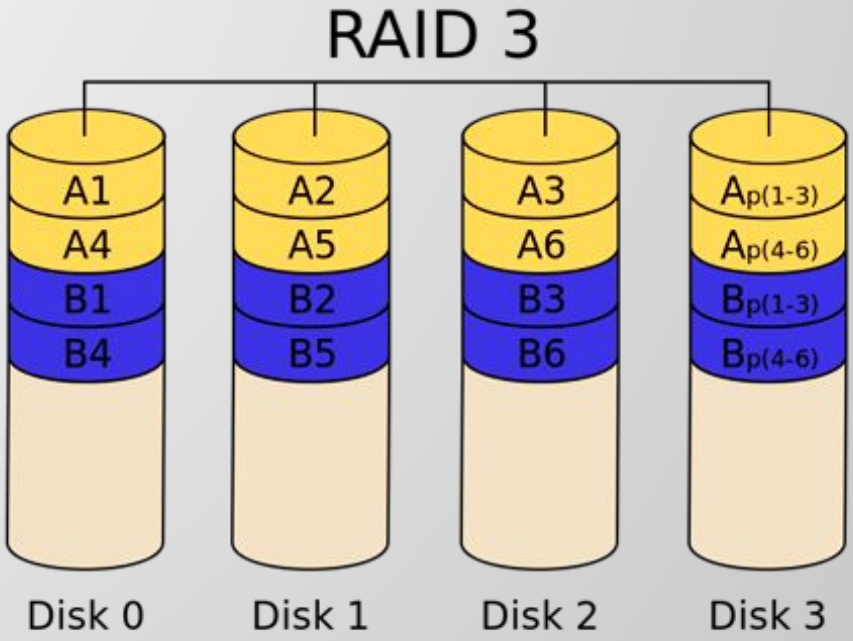
Дисковый массив RAID 2

- Массив RAID 2 основан на использовании кода Хемминга. Диски делятся на две группы: для данных (2^n-n-1 дисков) и для кодов коррекции ошибок (n дисков).
- Распределение данных по дискам такое же, как и в RAID 0. т.е. они разбиваются на небольшие блоки по числу дисков.
- Достоинство массива RAID 2: повышение скорости дисковых операций по сравнению с одним диском.
- Недостаток RAID 2: min число дисков, при котором имеет смысл его использовать, – 7. Поскольку во всех современных дисках имеется встроенный контроль, этот массив используется редко.

Дисковый массив RAID 3

- В RAID 3, как и в RAID 2 блоки чередуются по группе дисков, но один из дисков группы отведен для хранения информации о четности. При отказе дисковода восстановление данных осуществляется на основе вычисления значений функции XOR от данных, записанных на оставшихся дисках.
- Записи обычно занимают все диски (т.к. блоки короткие), что повышает общую скорость, но в каждый момент времени массив может обслужить только один запрос.

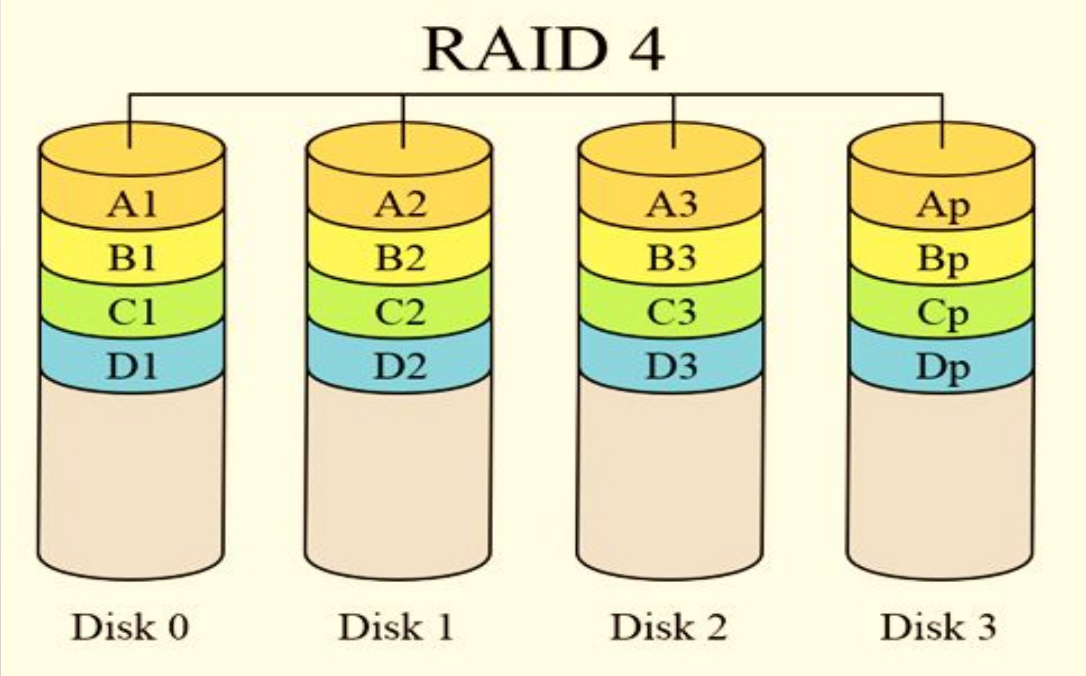
Дисковый массив RAID 3



- Достоинства: высокая скорость чтения и записи данных; min количество дисков для создания массива равно трём.
- Недостатки: массив хорош только для однозадачной работы с большими файлами; большая нагрузка на контрольный диск, как следствие, его надёжность ниже дисков с данными.

Дисковый массив RAID 4

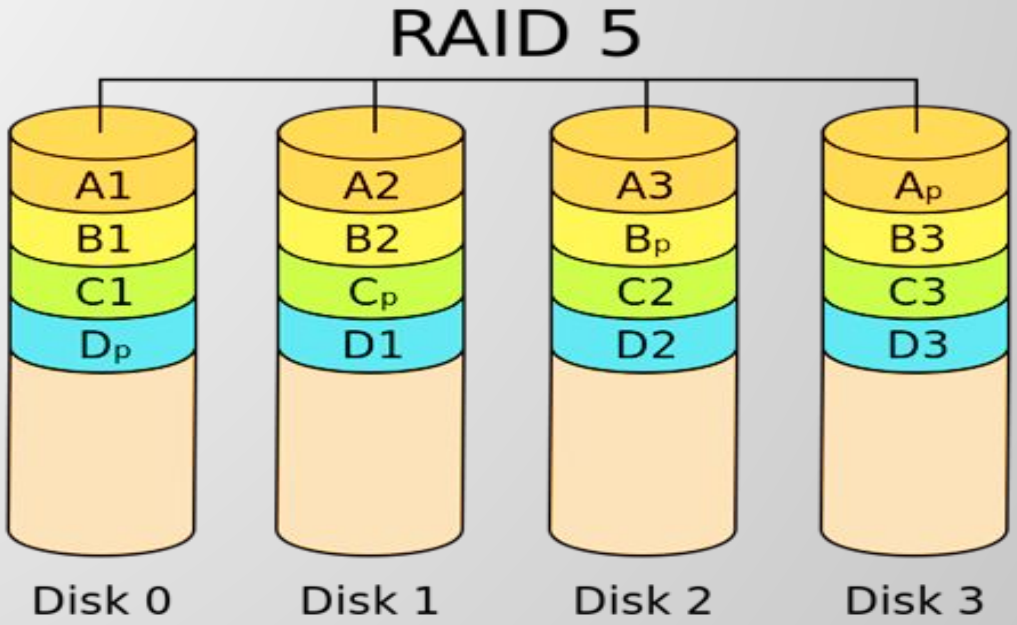
Организация RAID 4 похожа на RAID 3 с той разницей, что в нем используются блоки большого размера, поэтому записи можно читать с любого диска массива. При операциях записи всегда происходит обновление диска четности, поэтому их совмещение невозможно.



Дисковый массив RAID 5

- RAID 5 похож на RAID 4, но хранение кодов четности в нем осуществляется не на отдельном диске, а поочередно на всех дисках.
- Блоки данных и контрольные суммы циклически записываются на все диски массива, нет асимметричности в конфигурации дисков.
- Операции чтения также могут выполняться параллельно для всех дисков.
- Исчезает ограничения по одновременной записи. Операции записи, требующие участия двух дисководов (для данных и для четности) обычно также могут совмещаться.

Дисковый массив RAID 5

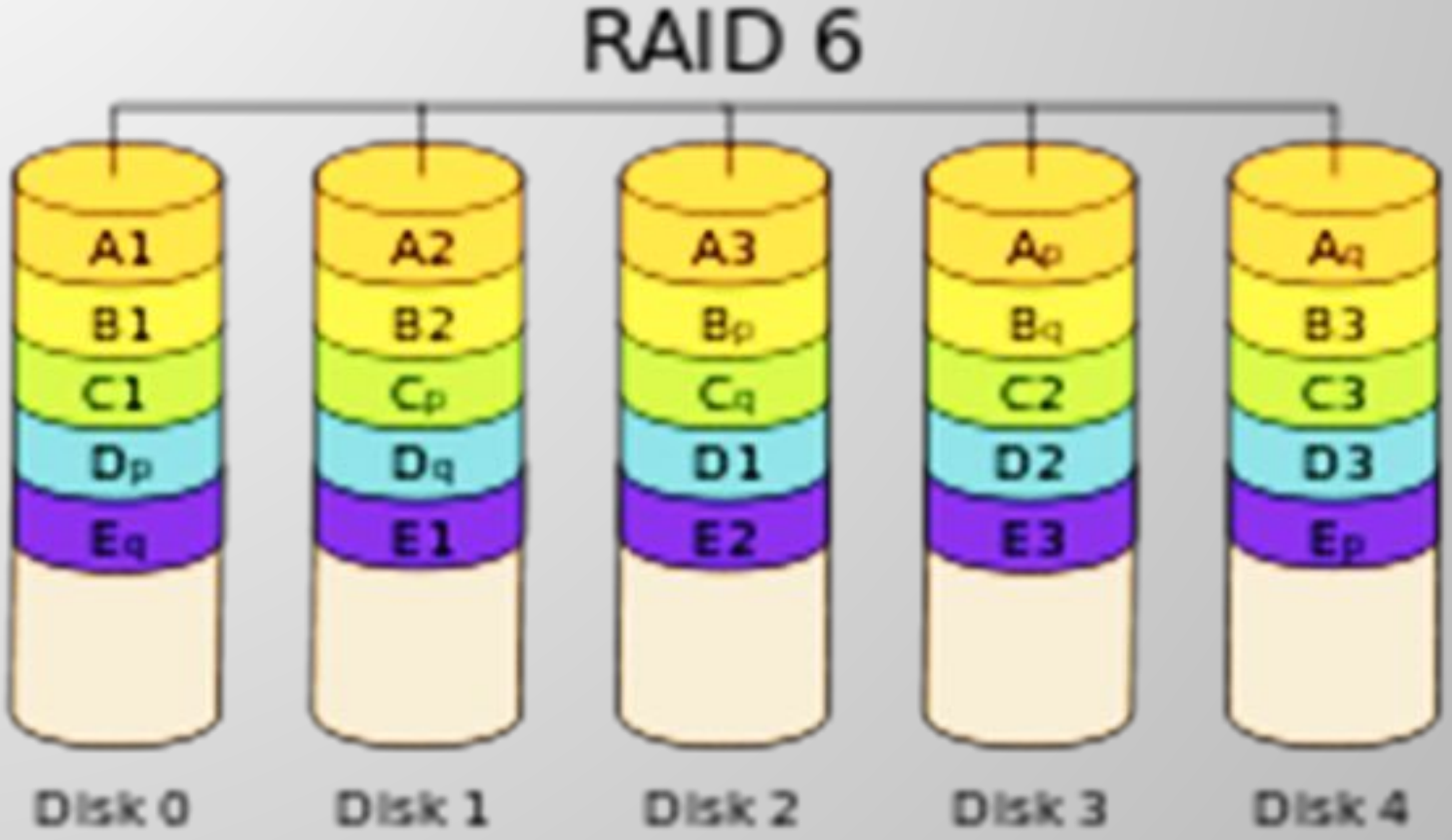


- RAID5 получил широкое распространение, в первую очередь, благодаря своей экономичности.
- Но при выходе из строя одного из дисков – весь том переходит в критический режим (degrade) и производительность резко падает.
- Минимальное количество используемых дисков равно трём.

Дисковый массив RAID 6

- RAID 6 – похож на RAID 5, но имеет более высокую степень надёжности - под контрольные суммы выделяется ёмкость 2-х дисков, 2 суммы рассчитываются по разным алгоритмам.
- Требует более мощный RAID-контроллер.
- Обеспечивает работоспособность после одновременного выхода из строя двух дисков – защита от кратного отказа. Для организации массива требуется минимум 4 диска. Обычно использование RAID-6 вызывает примерно 10-15% падение производительности дисковой группы, по сравнению с аналогичными показателями RAID-5, что вызвано большим объёмом обработки для контроллера.

Дисковый массив RAID 6



Двухуровневые RAID-массивы

- На основе наиболее распространенных вариантов RAID: 0, 1 и 5 и 6 могут формироваться т.н. двухуровневые архитектуры. Например:
  - RAID 10 – массив 0, построенный из массивов 1;
  - RAID 50 – массив 0, построенный из массивов 5;
  - RAID 60 – массив 0, построенный из массивов 6.
- За счет такой двухуровневой организации можно достичь требуемого баланса между увеличением надежности хранения данных, характерным для массивов RAID 1, RAID 5, и высокой скоростью чтения, присущей чередованию блоков на дисках в массиве RAID 0.

Использование дисковых RAID-массивов

- RAID массивы используются очень широко, так как обеспечивают надежный механизм хранения при невысоких накладных расходах.
- Недостатком такого подхода является то, что дублируется, по существу, только одна часть аппаратуры – диски, а сама КС, использующая RAID-массив, не задублирована.
- Полное дублирование компьютера (например, кластеры) обеспечивает существенно большую надежность. Однако такое решение требует дополнительной дорогостоящей аппаратуры, а также специальных расширений ОС.

Отказоустойчивые кластеры и системы

- строятся по трем основным принципам:

• ***с холодным резервом или активный / пассивный***. Активный работает, а пассивный включается в работу когда произойдет отказ. Пример – связка DRBD и HeartBeat.

• ***с горячим резервом или активный / активный***. Работают все узлы, при отказе одного узла нагрузка перераспределяется между оставшимися. Примеры – Microsoft Cluster Server, OpenSource проект OpenMosix.

• ***с модульной избыточностью.*** Применяется в случае, когда простой системы совершенно недопустим. Все узлы нработают одновременно, результат достижим и при отказе любого узла. Примеры – RAID и Triple modular redundancy.

Кластеры распределения нагрузки

- **Network Load Balancing, NLB**. Принцип их действия строится на распределении запросов через один или несколько входных узлов, которые перенаправляют их на обработку в остальные, вычислительные узлы.
- Главная цель такого кластера – производительность, однако, в них часто используются также и методы, повышающие надёжность.
- Подобные конструкции называются серверными фермами.

Вычислительные кластеры

- Существенными показателями являются высокая производительность в операциях над числами с плавающей точкой (flops) и низкая задержка пересылаемых в сети данных.
- Менее существенные показатели – скорость операций ввода-вывода, которая в большей степени важна для БД и web-сервисов.
- Вычислительные кластеры позволяют уменьшить время расчетов, по сравнению с одиночным компьютером, разбивая задание на параллельно выполняющиеся ветки, которые обмениваются данными по связывающей сети.

Системы распределенных вычислений (grid)

- ***Grid-системы*** не принято считать кластерами, но их принципы в значительной степени сходны с кластерной технологией.
- Главное отличие – низкая доступность каждого узла в заданный момент времени (узлы подключаются и отключаются в процессе работы), поэтому задача должна быть разбита на ряд независимых друг от друга процессов.
- В отличие от кластеров, grid-система не похожа на единый компьютер, а служит упрощённым средством распределения вычислений.
- Нестабильность конфигурации компенсируется большим числом узлов.

Кластер серверов, организуемых программно

- Кластер серверов (в информационных технологиях) - это группа серверов, объединённых логически, способных обрабатывать идентичные запросы и использующиеся как единый ресурс.
- Чаще всего серверы группируются посредством локальной сети.
- Группа серверов обладает большей надежностью и большей производительностью, чем один сервер.
- Объединение серверов в один ресурс происходит на уровне программных протоколов.

Кластер серверов, организуемых программно

- В отличие от аппаратного кластера, программно-организуемые кластеры, требуют:
- наличия специального программного модуля (Cluster Manager) , основной функцией которого является поддержание взаимодействия между членами кластера:
  - синхронизации данных между серверами;
  - распределение нагрузки между серверами;
- умения клиентского программного обеспечения распознавать кластер серверов и соответствующим образом обрабатывать команды от Cluster Manager.

Кластерные системы высокой доступности (готовности)

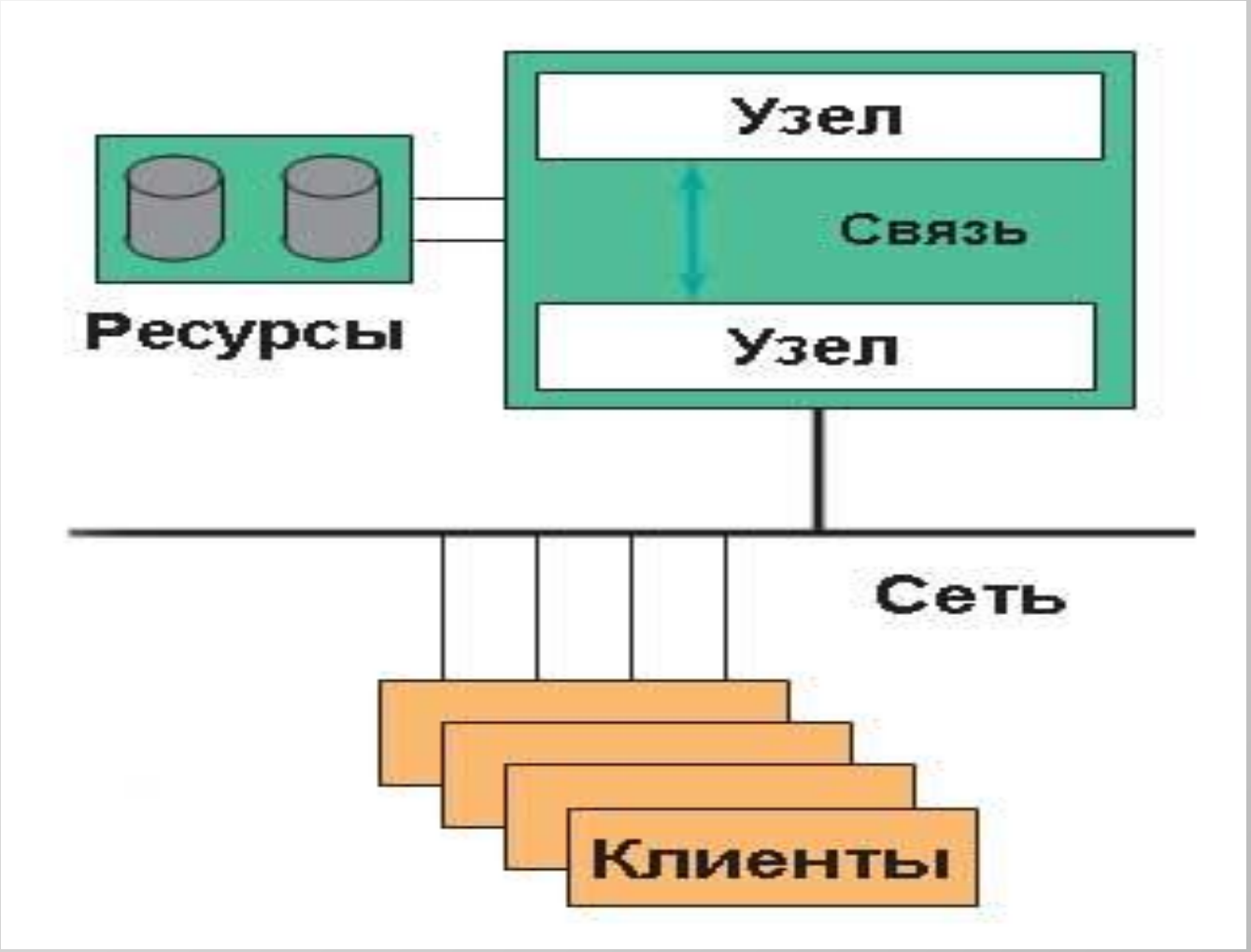
- Одним из основных механизмов обеспечения необходимого уровня надежности и отказоустойчивости ИС, а, в конечном итоге, высокого уровня их доступности и готовности (high availability) является ***кластеризация*** серверных систем.
- Термин *«кластеризация»* имеет много различных значений. Строгое определение могло бы звучать так:

«Реализация объединения вычислительных установок, представляющегося единым целым для ОС, системного, прикладного ПО и пользователей».

Кластеры высокой доступности

- Кластеры высокой доступности (готовности) (КВД) обозначаются аббревиатурой HA (High Availability).
- Создаются они для обеспечения высокой доступности.
- Избыточное число узлов, входящих в кластер, гарантирует предоставление сервиса в случае отказа одного или нескольких серверов. Типичное число узлов 2.
- КВД это группа независимых серверов, работающих как единая система, выступающая для клиентов в виде виртуального сервера.
- Кластер предусматривает единое администрирование, единую систему хранения данных и совместное (или нет) использование данных.

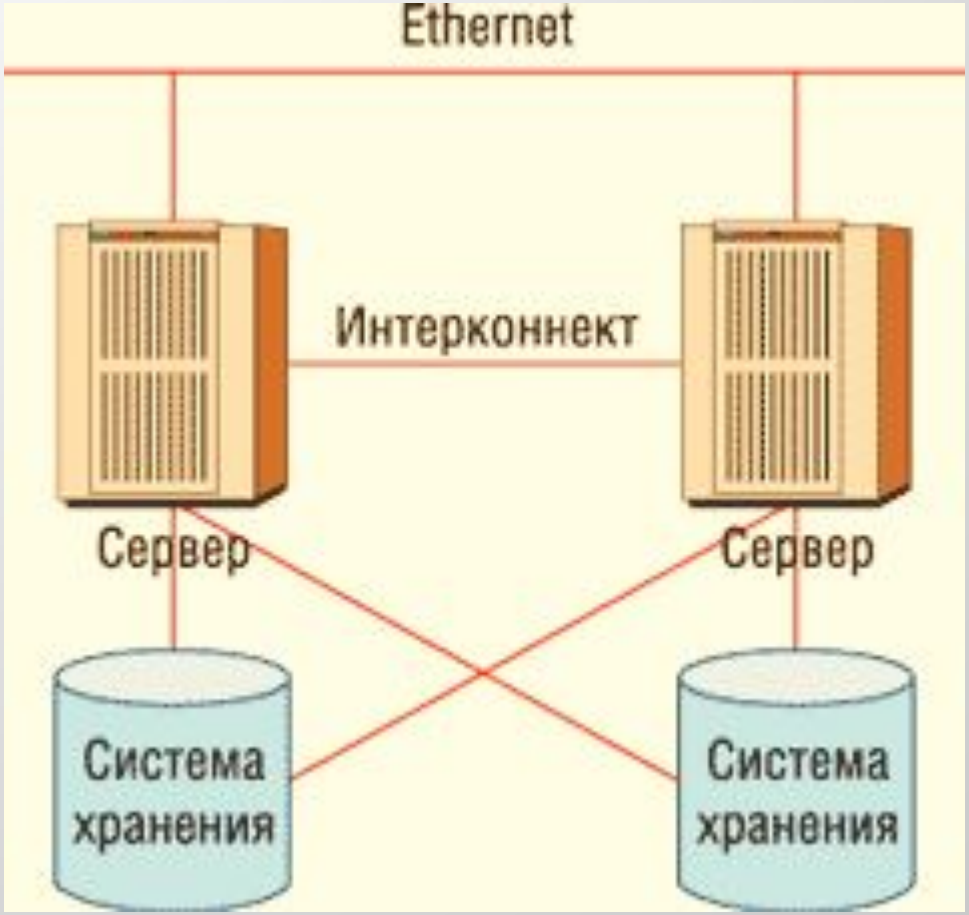
Кластер высокой доступности



Кластеры высокой доступности

- Построение отказоустойчивых кластеров предполагает использование в качестве узлов высоконадежных серверов с двукратным, а иногда и многократным, дублированием всех основных блоков и компонентов.
- Типовая КВД подразумевает объединение двух или более серверов в целостную систему, снаружи видимую пользователями как единый виртуальный сервер, исполняющий то или иное корпоративное приложение, например, СУБД.
- Для связи и передачи данных от потребителей и к ним обычно используется стандартное соединение Fast Ethernet или стандарт Gigabit Ethernet.

Кластеры высокой доступности



Между собой серверы общаются посредством ***интерконнекта*** – специальной выделенной сети для синхронизации состояний и посылки быстрых команд перезапуска заданий на резервном узле в случае обнаружения сбоев или отказов. Кроме служебных команд, между узлами практически нет трафика.

Кластеры высокой доступности

- Реальные кластерные системы высокой готовности строятся в одной из двух базовых архитектур:
  - с разделением ресурсов (share something);
  - без разделения ресурсов (share nothing).
- **В первом случае** (пример − Oracle Real Application Clusters) задача клиента «размазывается» по серверам кластера. Такой подход удачно сочетает высокую отказоустойчивость и производительность на относительно недорогих аппаратных средствах.
- **Во втором случае** задача клиента целиком и полностью решается на головном узле системы, а второй сервер играет роль горячего резерва, способного в любой момент времени подхватить «упавшее» приложение.

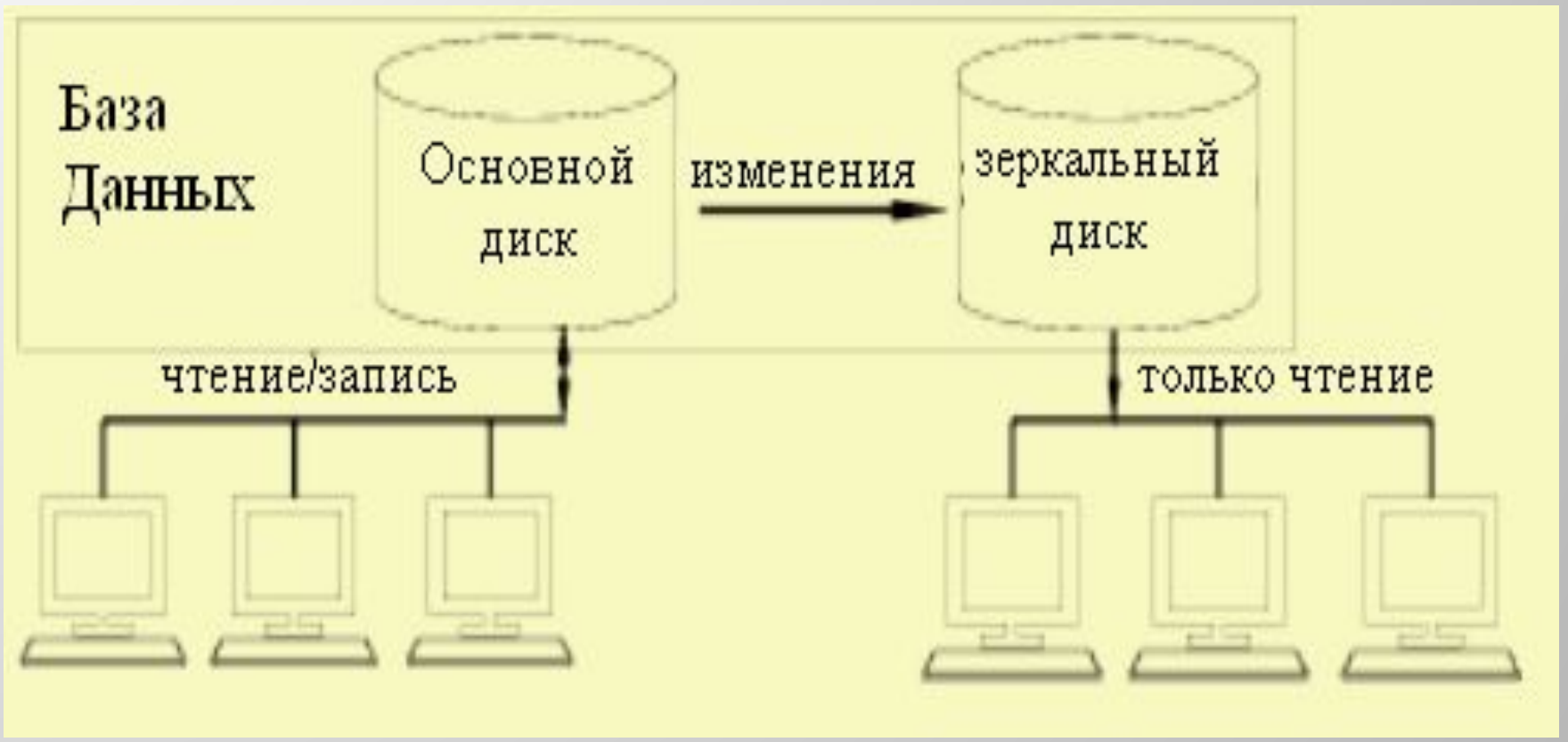
Технологии постоянного дублирования данных

- Технологии аппаратного дублирования важны, с точки зрения повышения общей надежности КС, но они, как правило, не ориентированы на обработку транзакций СУБД и на связанные с этим специфические ограничения, например, обеспечение атомарности транзакций.
- В результате СУБД не может воспользоваться преимуществами чисто аппаратных решений резервирования для повышения своей производительности.
- Для этих целей используются ***программные технологии постоянного дублирования данных***.

Управляемая избыточность данных

- Представлена в 2-х формах:
  - программное зеркалирование дисков (software mirroring данных);
  - тиражирование (репликация replication) данных.
- ***Программное зеркалирование*** (дуплексирование – duplexing, мультиплексирование - multiplexing), может не только защитить от аппаратных сбоев, но и улучшить производительность.
- Поскольку зеркалирование БД производится на другом физическом устройстве, то операции чтения данных можно распределить между двумя устройствами.
- Позволяет исключить отказы только одной части аппаратуры – дисков.

Программное зеркалирование дисков



Технологии постоянного дублирования данных

- Постоянное дублирование позволяют строить защиту данных на уровне серверов.
- ***Первая методика постоянного дублирования*** предполагает наличие двух идентичных серверов БД. Один из них (первичный сервер) работает в режиме чтение/запись, второй (вторичный) - только в режиме чтения. Первичный сервер передает на вторичный сервер все изменения. За счет того, что сервера идентичны, обеспечивается высокая скорость передачи изменений, легкость настройки и высокая надежность. При выходе из строя первичного сервера, вторичный может взять на себя все его функции.
- Возможность работы вторичного сервера в режиме чтения позволяет повысить общую производительность системы.

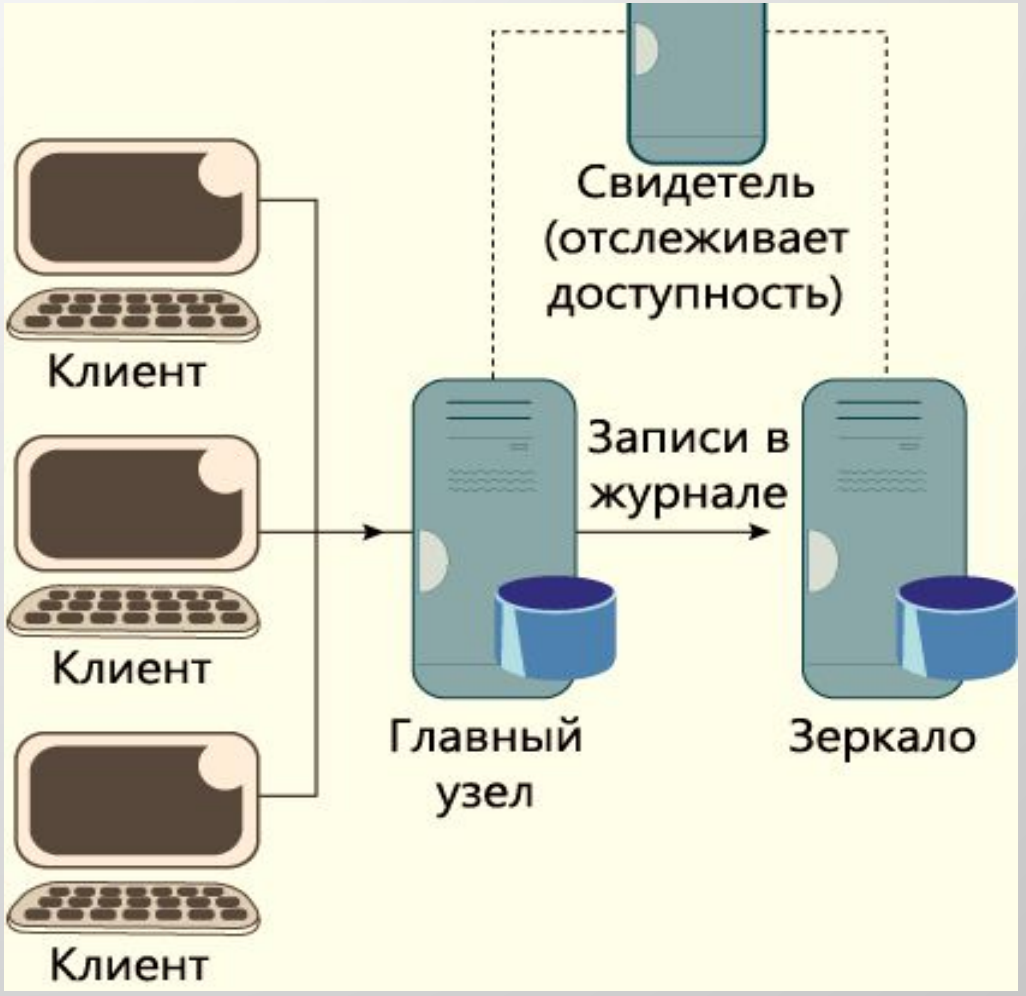
Технологии постоянного дублирования

- ***Вторая методика постоянного дублирования*** основывается на репликации важных данных.
- По сравнению с сервером горячего резерва, такая конфигурация более гибкая, так как отсутствует требование идентичности двух серверов.
- При реплицировании возможно использование более двух серверов. Однако конфигурирование и настройка подобной схемы более трудоемки.
- Реплицирование позволяет решить задачу сохранения данных, но оно все же ориентировано на задачи распределенной обработки. Другими словами, использование первой методики более привлекательно.

Зеркальное отображение баз данных

- Еще одним средством обеспечения высокой готовности и доступности ИС является ***зеркальное отображение баз данных*** (database mirroring).
- Принципы зеркального отображения баз данных рассмотрим на примере СУБД SQL Server.
- В этой СУБД существует три режима зеркалирования:
  - режим высокого уровня безопасности;
  - режим высокого уровня доступности;
  - режим высокой производительности.

Зеркальное отображение баз данных



Принципы зеркального отображения баз данных в режиме ***высокого уровня безопасности***

Режимы зеркального отображения баз данных

- Для зеркального отображения можно использовать два режима: синхронный и асинхронный.
  - ***синхронный режим*** (synchronous mode): транзакция не будет завершена, если она не прошла на обоих серверах. Идентичность данных на двух серверах гарантируется. Однако скорость работы транзакций при этом может существенно замедлиться. Этот режим работы подразделяется еще на два:
    - **ориентированный на отказоустойчивость** (high-availability);
    - **ориентированный на защиту данных** (high-protection).

Асинхронный режим зеркального отображения баз данных

- ***Асинхронный режим*** (asynchronous mode, другое называние − high-performance mode (режим высокой производительности) − в этом случае транзакция вначале завершается на первом сервере, а затем информация о ней немедленно передается на второй сервер. Задержек при работе транзакций не будет, но данные между серверами могут синхронизироваться с небольшим отставанием.

Преимущества зеркалирования

Преимущества зеркалирования перед использованием кластера:

- зеркальное отображение баз данных не требует применения специального оборудования;
- серверы, которые участвуют в зеркальном отображении баз данных, не обязательно должны находиться рядом друг с другом;
- в кластере серверы работают с одной физической БД, которая находится на внешнем накопителе. Выход из строя этого накопителя приведет к отказу всего кластера.
- В зеркальном отображении используются две отдельные копии БД, что повышает надежность работы.

Преимущества зеркалирования

Преимущества зеркалирования по сравнению с доставкой журналов:

- переключение ролей в случае отказа основного сервера может производиться автоматически (при наличии следящего сервера (witness server));
- не потребуется вносить какие-либо изменения в сетевую инфраструктуру или в настройки клиентов. Клиенты при необходимости автоматически переключаются на использование зеркальной копии;
- Устраняется задержка синхронизации серверов и гарантируется идентичность копий данных на обоих серверах.

Тиражирование данных

- ***Тиражирование*** − еще одна технология высокой отказоустойчивости, которая широко используется в современных АИС и БД. Используется, в частности в системе SQL Server, где на трактуется как ***перемещение журналов*** (log shipping). Ее суть состоит в автоматизации резервного копирования БД и ее восстановления на другом сервере.
- Перемещение журналов предоставляет дополнительную гибкую возможность для защиты против аппаратных сбоев. При перемещении журналов файлы первичного сервера копируются на вторичный, затем файлы восстанавливаются на вторичном сервере.

Недостатки технологии перемещения журналов

- ***Перемещение журналов*** не имеет механизма обнаружения сбоя и инициализации переключения на вторичный сервер.
- Имеется промежуток запаздывания во время переключения, когда все резервные копии восстанавливаются на вторичном сервере.
- Существует риск потери некоторых уже проведенных транзакций, если на первичном сервере происходит полный сбой аппаратного обеспечения.
- Перемещение журналов не обладает прозрачностью; клиенты должны подключаться к серверу с именем, отличным от имени первичного сервера.

Принципы тиражирования данных в Informix OnLine-DS

- В конфигурации серверов **Informix OnLine-DS** с тиражированием выделяется один основной сервер и ряд вторичных серверов. На основном сервере выполняется и чтение, и обновление данных, а все изменения передаются на вторичные серверы, доступные только на чтение.
- В случае отказа основного сервера вторичный автоматически или вручную переводится в режим доступа на чтение и запись.
- Прозрачное перенаправление клиентов при отказе основного сервера не поддерживается, но оно может быть реализовано в рамках приложений.

Принцип тиражирования данных в Informix OnLine-DS

Основной сервер доступен на чтение и на запись, вторичный – только на чтение

Принцип тиражирования данных в Informix OnLine-DS

Когда основной сервер выходит из строя, вторичный переводится в режим доступа и на чтение и на запись

Принципы тиражирования данных в Informix OnLine-DS

- ***Тиражирование*** осуществляется путем передачи информации из журнала транзакций (логического журнала) в буфер тиражирования основного сервера, откуда она пересылается в буфер тиражирования вторичного сервера. Такая пересылка может происходить либо в синхронном, либо в асинхронном режиме.
- ***Синхронный режим*** гарантирует полную согласованность БД − ни одна транзакция, зафиксированная на основном сервере, не останется незафиксированной на вторичном, даже в случае сбоя основного сервера.
- ***Асинхронный режим*** не обеспечивает полной согласованности, но улучшает характеристики системы.

Технологии архивации

- Так же как и постоянное дублирование, ***архивация*** может производиться на уровне аппаратуры, на уровне ОС, на уровне СУБД и на уровне ПП.
- ***Архивация на уровне аппаратуры*** (например, получение «слепка» жесткого диска) требует специальной аппаратуры. Более того, информация, имеющаяся на диске в тот или иной момент времени может быть недостаточной, так как часть данных находится в ОЗУ.
- Серверы БД при проведении подобной архивации должны быть переведены в нерабочий режим. Следовательно, уровень готовности ИС понижается.

Технологии архивации

- ***Создание архива на уровне ОС*** обычно подразумевает архивацию того или иного жесткого диска (или их набора), что является только архивацией данных, а не всей системы в целом.
- Использование архивации средствами ОС обычно приостанавливает обработку запросов и, так же как и аппаратное архивирование, требует выключение серверов БД, что понижает уровень готовности АИС.
- Возможно еще ***создание архивов путем архивации отдельных файлов*** на диске. Однако этот способ не гарантирует целостность данных и неудобен с точки зрения восстановления.

Технологии архивации

- ***Полное архивирование.*** Современные сервера БД предоставляют развитые и удобные средства создания архивных копий данных без прекращения доступа к ИС, хотя, некоторое замедление возможно.
- Сервер СУБД сам обеспечивает целостность и корректность данных в архиве.
- Архивация данных на уровне сервера СУБД, совместно с созданием архива ОС и ПП в сейчас является наиболее оптимальной стратегией архивации БД.

Резервное копирование БД

- Термины «Архивация», «Архивирование», «Резервное копирование» являются, по существу, синонимами, однако в различных ИС и СУБД реализуются по-разному и имеют свои отличительные особенности.
- Собирательный термин для процессов архивирования и резервного копирования – «сохранение».
- Необходимый функциональный охват ПО для резервного копирования сильно зависит от конфигурации, назначения ИС и предъявляемых к ним требований. Поэтому применяется весьма широкий спектр решений резервного копирования (архивирования).

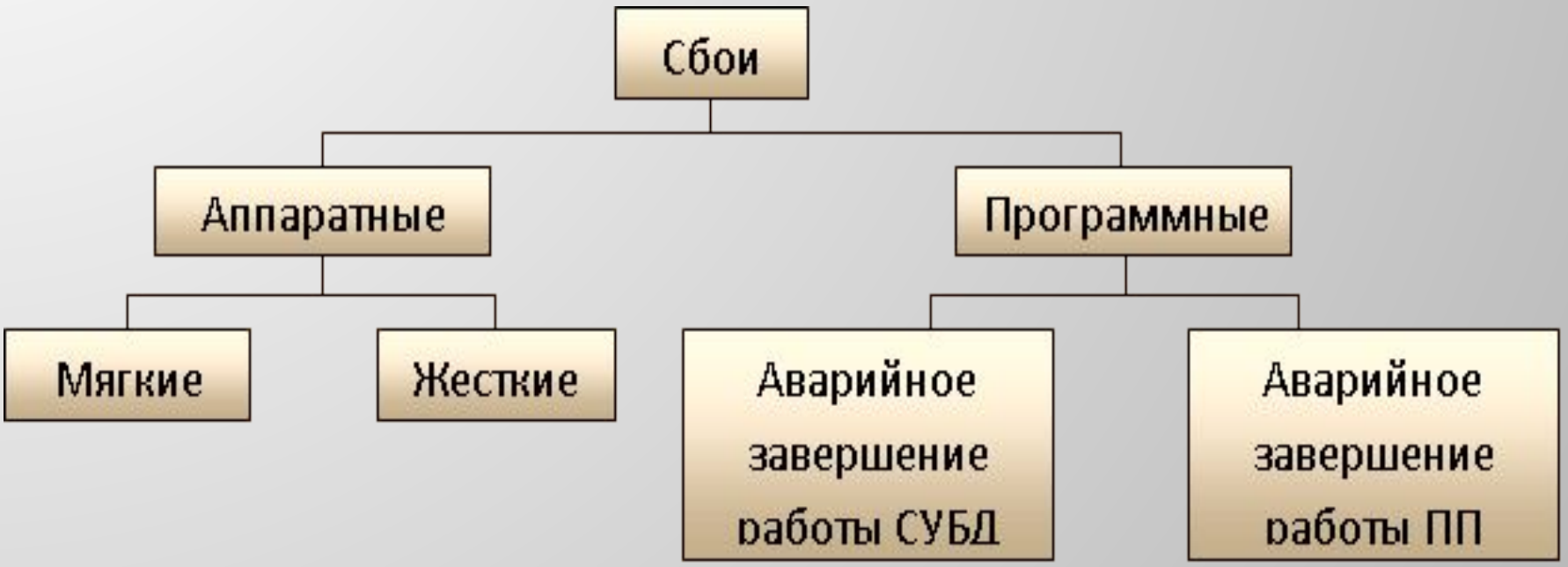
Архивирование баз данных

- В других СУБД используется термин «архивирование», под которым понимается процесс сохранения БД на ленточных и дисковых томах. Архив отражает состояние БД на момент начала архивирования.
- В случае непредвиденной утраты данных, БД восстанавливается из сохраненной архивной копии. Архивная копия БД позволяет восстановить БД только по ее состоянию на момент последнего архивирования.
- Логические журналы транзакций и их резервные копии хранят сведения о действиях сервера БД, произведенных после момента архивирования. Интерпретация этих журналов позволяет восстановить БД до состояния, более позднего, чем момент последнего архивирования.

Восстановление баз данных

- Одним из основных требований к СУБД является надежность хранения данных во внешней памяти. Под надежностью хранения понимается то, что СУБД должна быть в состоянии восстановить последнее согласованное состояние БД после любого аппаратного или программного сбоя.
- Обычно рассматриваются два возможных вида аппаратных сбоев:
  - ***мягкие сбои***, трактуются как внезапная остановка КС (например, аварийное выключение питания);
  - ***жесткие сбои***, характеризуемые потерей информации на носителях внешней памяти.

Виды сбоев в работе КС



Восстановление баз данных

- В общем случае восстановление – это процесс, позволяющий воссоздать БД на основе ранее сохраненных данных. Различают:
  - ***физическое восстановление;***
  - ***логическое восстановление.***
- При физическом восстановлении используется только архивная информация.
- Логическое восстановление дополнительно опирается на интерпретацию сохраненных журналов транзакций, что позволяет получить более свежее состояние БД (термин ***«накатить вперед»*** заархивированное состояние).

Восстановление баз данных

- Для восстановления БД нужно помимо архива БД нужна дополнительная информация, которую получают путем ведения журнала изменений БД.
- Журнал - это особая часть БД, недоступная пользователям БД и поддерживаемая с особой тщательностью.
- В журнал поступают записи обо всех изменениях основной части БД. В разных СУБД изменения БД журнализируются на разных уровнях:
  - на уровне логической операции;
  - на уровне страницы внешней памяти;
  - с одновременной поддержкой обоих подходов.

Восстановление баз данных

- Во всех случаях придерживаются стратегии «упреждающей» записи в журнал (так называемого протокола Write Ahead Log WAL).
- Эта стратегия заключается в том, что запись об изменении любого объекта БД должна попасть во внешнюю память журнала раньше, чем измененный объект попадет во внешнюю память основной части БД.
- Если в СУБД корректно соблюдается протокол WAL, то с помощью журнала можно решить все проблемы восстановления БД после любого вида сбоя.

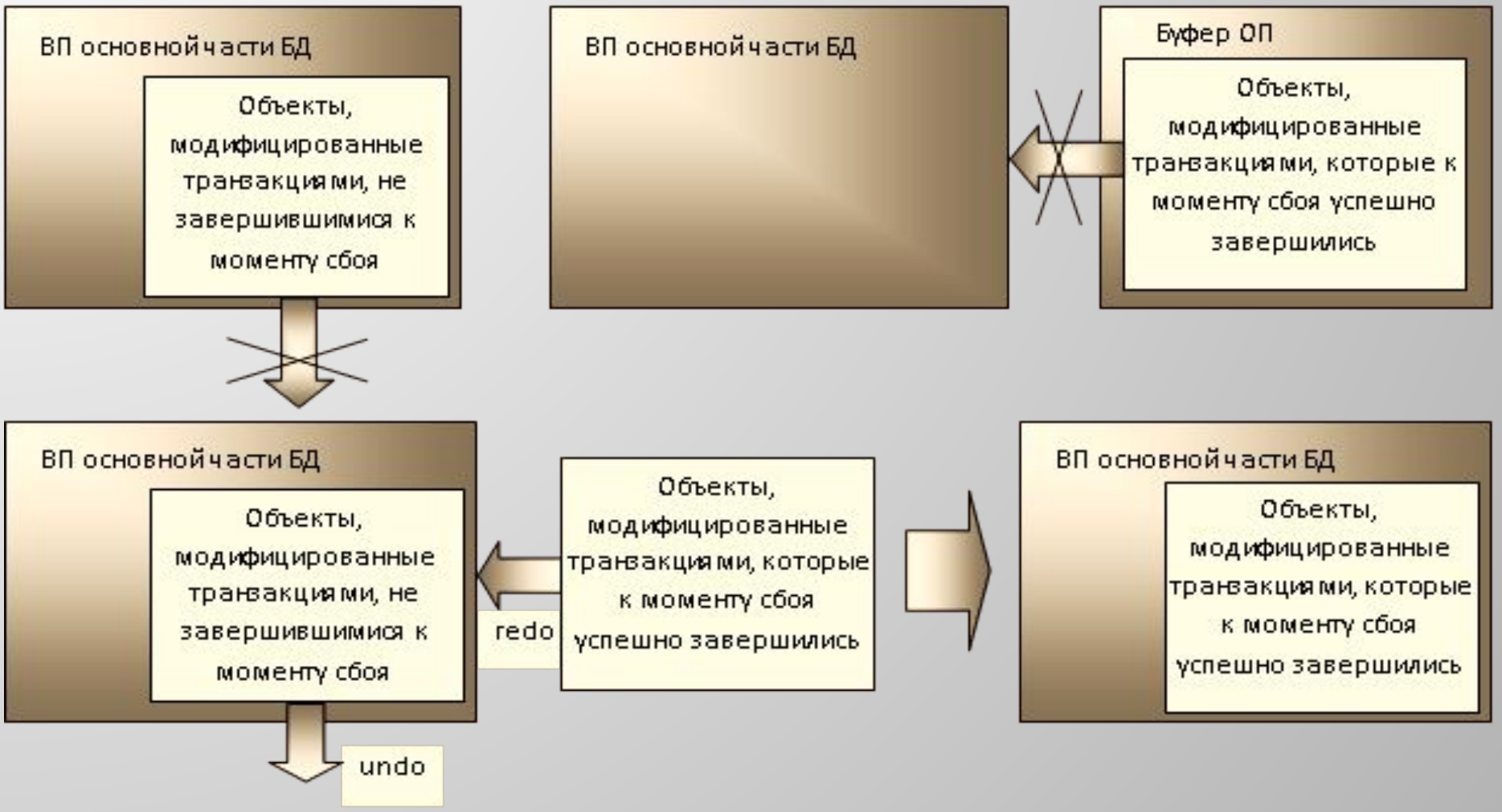
Восстановление баз данных

- Общими принципами восстановления являются следующие:
  - результаты зафиксированных транзакций должны быть сохранены в восстановленном состоянии БД (т. е. должно поддерживаться свойство долговечности (durability) транзакций);
  - результаты незафиксированных транзакций должны отсутствовать в восстановленном состоянии БД (в противном случае состояние БД может оказаться не целостным).

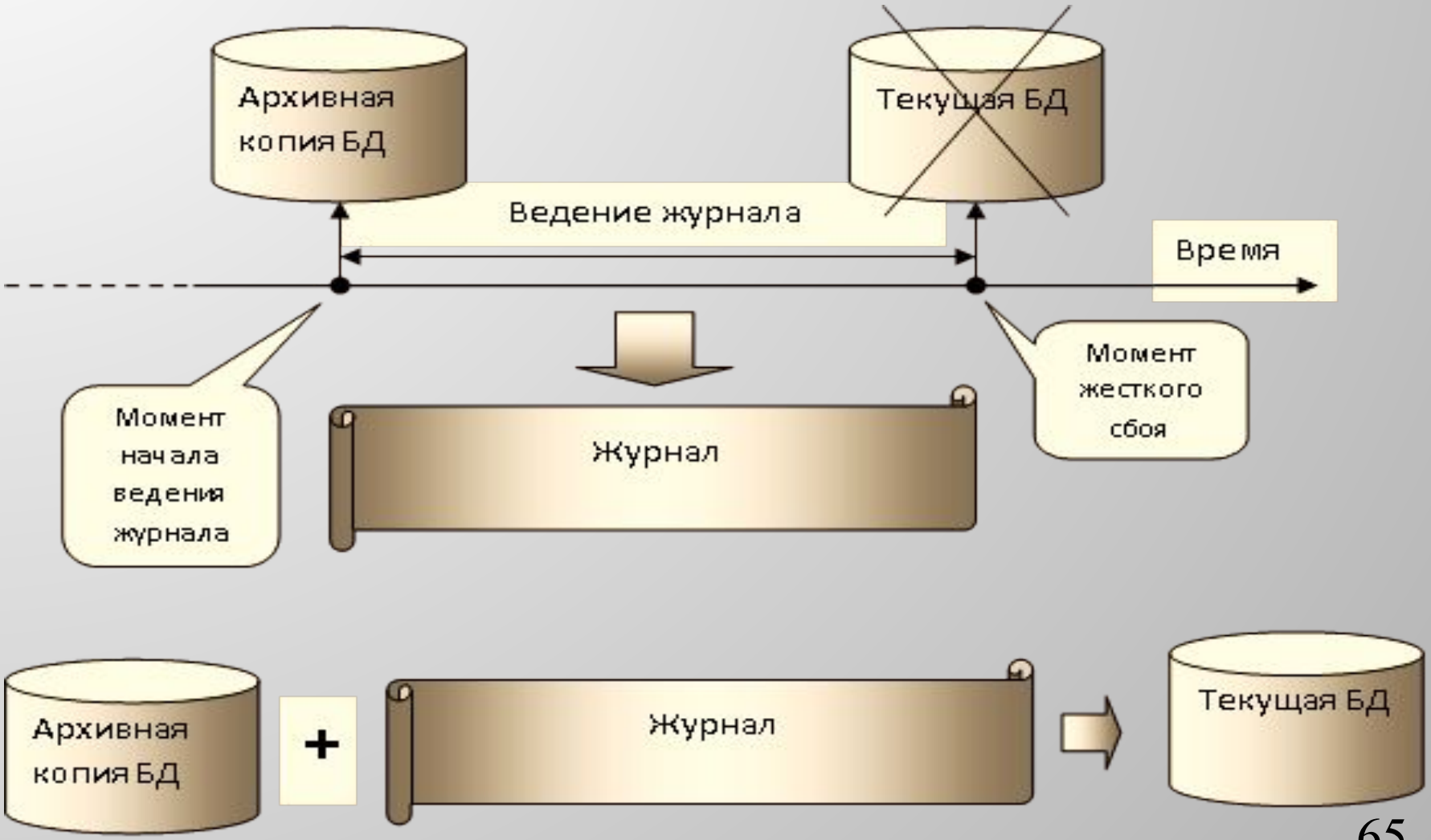
Восстановление баз данных

- Возможны следующие три ситуации, при которых требуется производить восстановление состояния БД:
  - Индивидуальный откат транзакции. Для восстановления согласованного состояния БД в этом случае нужно устранить последствия операторов модификации БД из этой транзакции.
  - Восстановление после внезапной потери содержимого ОП (мягкий сбой). Это потеря той части БД, которая к моменту сбоя содержалась в буферах ОП СУБД.
  - Восстановление после поломки основного внешнего носителя БД (жесткий сбой). Основой восстановления является архивная копия и журнал изменений БД.

Восстановление базы данных при мягком сбое



Восстановление базы данных при жестком сбое



Точки сохранения

- Точки сохранения (save points). Это особые точки, которые, с точки зрения логики процесса обработки данных фиксируют некоторые его логически завершенные части.
- Точки сохранения приняты в стандарте и многие СУБД используют их на практике. Во многом они похожи на подтранзакцию и предполагают, что внутри транзакции может происходить откат не к началу, а к некоторому промежуточному значению (точке сохранения), которое может отметить разработчик:
- `Rollback to <имя точки>`

Контрольные точки

- Контрольные точки (Check-points) необходимы для того, чтобы периодически фиксировать состояние системы.
- Т. е. через определенный период времени система принимает контрольную точку, в этот момент содержимое всех буферов ОП, в т. ч. и выполняемые в этот момент транзакции, записываются на диск.
- Если происходит отказ системы, то все транзакции, которые были начаты до или после записи контрольной точки, но успели закончиться до отказа, заново повторяются за счет информации, которая хранится на диске журнала, если они не закончились до отказа, то они автоматически отменяются.

Двухфазная фиксация.

- При работе с несколькими БД, имеющими раздельные журналы транзакций, необходим специальный системный компонент – координатор. Его назначение – гарантировать восстановление данных, если произошло нарушение. Фактически он берет на себя функции принятия решения, фиксировать или отменять транзакцию в системе вообще.
- Данные могут подтверждаться в каждой БД в разное время и в соответствии со своим журналом транзакций.
- Координатор принимает положительные отзывы от каждой из БД, и если все они получены, производит фиксацию, а если какой-либо отзыв не получен, происходит общий откат глобальной транзакции.

Репликация баз данных

- ***Репликация*** (replication) − это процесс автоматического распределения копий данных и объектов БД между узлами БД с одновременной синхронизацией реплик. В контексте ИТ-безопасности репликацию можно расценивать как технологию повышения отказоустойчивости.
- Отличие механизма репликации в том, что переключение узлов кластера работает на уровне серверов, гарантируя, что ИС будет доступна целиком. Перемещение журналов работает на более тонком уровне, гарантируя отказоустойчивость одной БД в пределах сервера. Репликация соответствует самому мелкому уровню: она обеспечивает отказоустойчивость по транзакциям.

Репликация баз данных

- В случае применения репликации для целей отказоустойчивости обычно задействуют не меньше трех серверов.
- Publisher – первичный сервер, куда приложения пересылают все транзакции.
- Subscriber – вторичный сервер, который действует как двойник первичного, доступный в случае сбоя на первичном сервере.
- Задача третьего сервера – Distributor, – гарантировать доставку транзакций на вторичный сервер. Как только транзакции копируются с Publisher на Distributor, Distributor доставляет их на Subscriber.

Заключение к дисциплине

- Обратной стороной процесса стремительного развития средств СУБД является вынужденное предоставление злоумышленникам все более изощренных средств для реализации угроз безопасности информации.
- Основным средством взаимодействия пользователей и приложений с СУБД является язык SQL – мощный непроцедурный инструмент определения и манипулирования данными.
- Хранимые процедуры добавляют к этому репертуару управляющие конструкции, которые могут быть использованы злоумышленниками для нарушения защиты БД.

Заключение к дисциплине

- Механизм правил (триггеры) дает возможность выстраивать сложные, трудные для анализа цепочки действий, позволяя неявным образом передавать право на выполнение процедур, даже не имея на это полномочий.
- В результате потенциальный злоумышленник получает в свои руки мощный и удобный инструментарий, а все развитие СУБД направлено на то, чтобы сделать этот инструментарий еще мощнее и еще удобнее.
- Тем самым в области ИТ, как и в любых других вооруженных противостояниях, наблюдается непрерывное противоборство средств нападения и защиты.
- При этом средства нападения во времени, практически всегда, опережают средства защиты.