



Базы данных

***Гаврилов Александр Викторович
к.т.н., доцент***



Тема 13. Распределенные базы данных

Лекция 13

Вопросы лекции:

1. Понятие распределенной базы данных
2. Принципы организации распределенных баз данных
3. Двенадцать общих целей систем распределенных баз данных
4. Преимущества и недостатки распределенных СУБД
5. Примеры распределенных систем

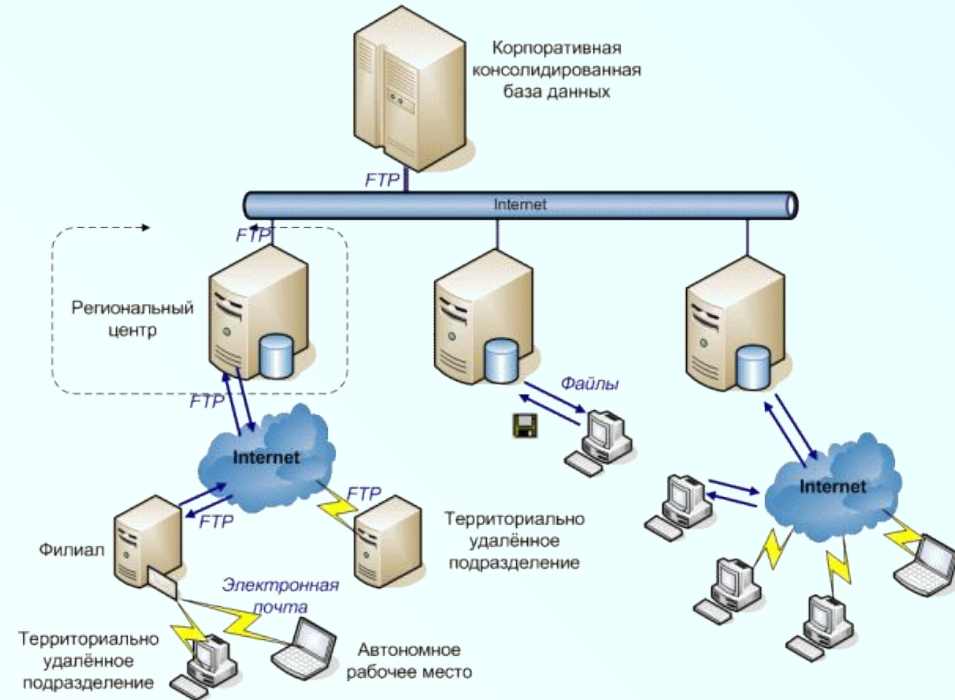
1. Понятие распределенной базы данных

Предпосылки появления распределенных баз данных

Основной предпосылкой разработки систем, использующих базы данных, является **стремление объединить все обрабатываемые в организации данные в единое целое и обеспечить к ним контролируемый доступ**. Хотя интеграция и предоставление контролируемого доступа могут способствовать централизации, последняя не является самоцелью. На практике создание компьютерных сетей приводит к децентрализации обработки данных. Децентрализованный подход, по сути, отражает организационную структуру многих компаний, логически состоящих из отдельных подразделений, отделов, проектных групп и т.п., которые физически распределены по разным офисам, отделениям, предприятиям или филиалам, причем каждая отдельная производственная единица имеет дело с собственным набором обрабатываемых данных. Разработка распределенных баз данных, отражающих организационные структуры предприятий, позволяет сделать общедоступными данные, поддерживаемые каждым из существующих подразделений, обеспечив при этом их хранение именно в тех местах, где они чаще всего используются. Подобный подход расширяет возможности совместного использования информации, одновременно повышая эффективность доступа к ней.

Доступ к информации

Распределенная база данных – набор логически связанных между собой совокупностей разделяемых данных (и их описаний), которые физически распределены в некоторой компьютерной сети.



Распределенная СУБД – программный комплекс, предназначенный для управления распределенными базами данных и обеспечивающий прозрачный доступ пользователей к распределенной информации.

2. Принципы организации распределенных баз данных

Фундаментальный принцип

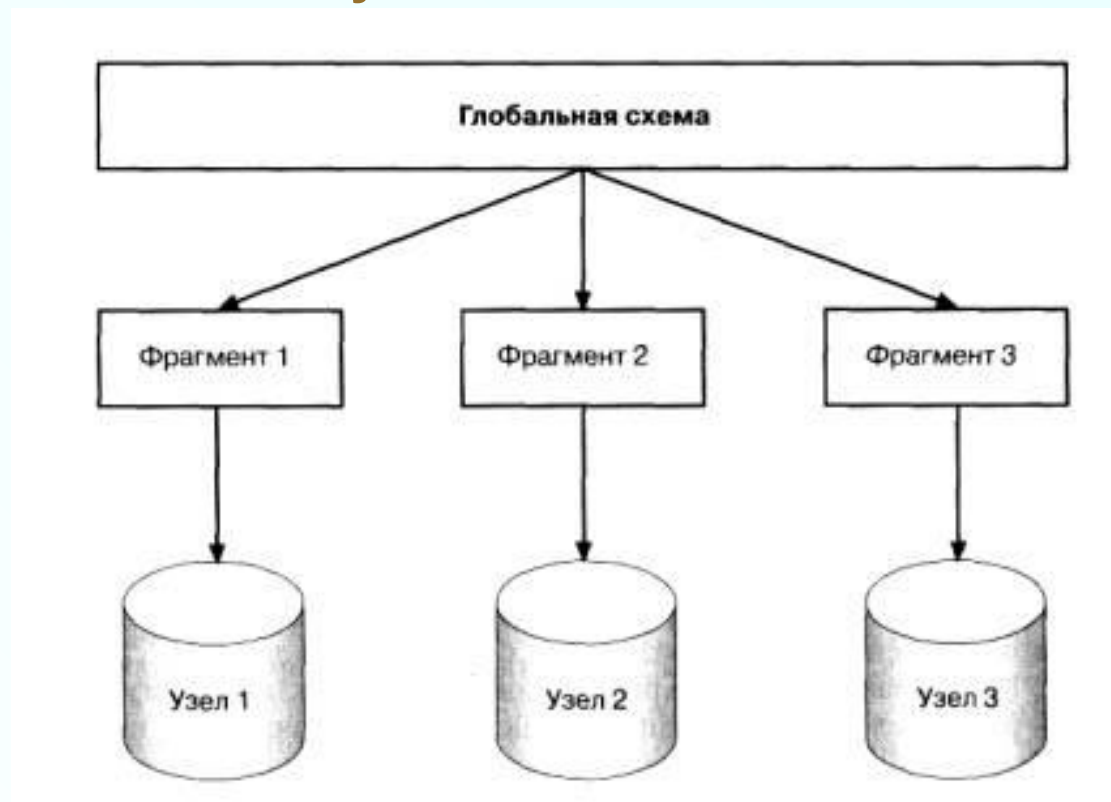
- Для пользователя распределенная система должна выглядеть так же, как нераспределенная система

Принципы организации распределенных баз данных

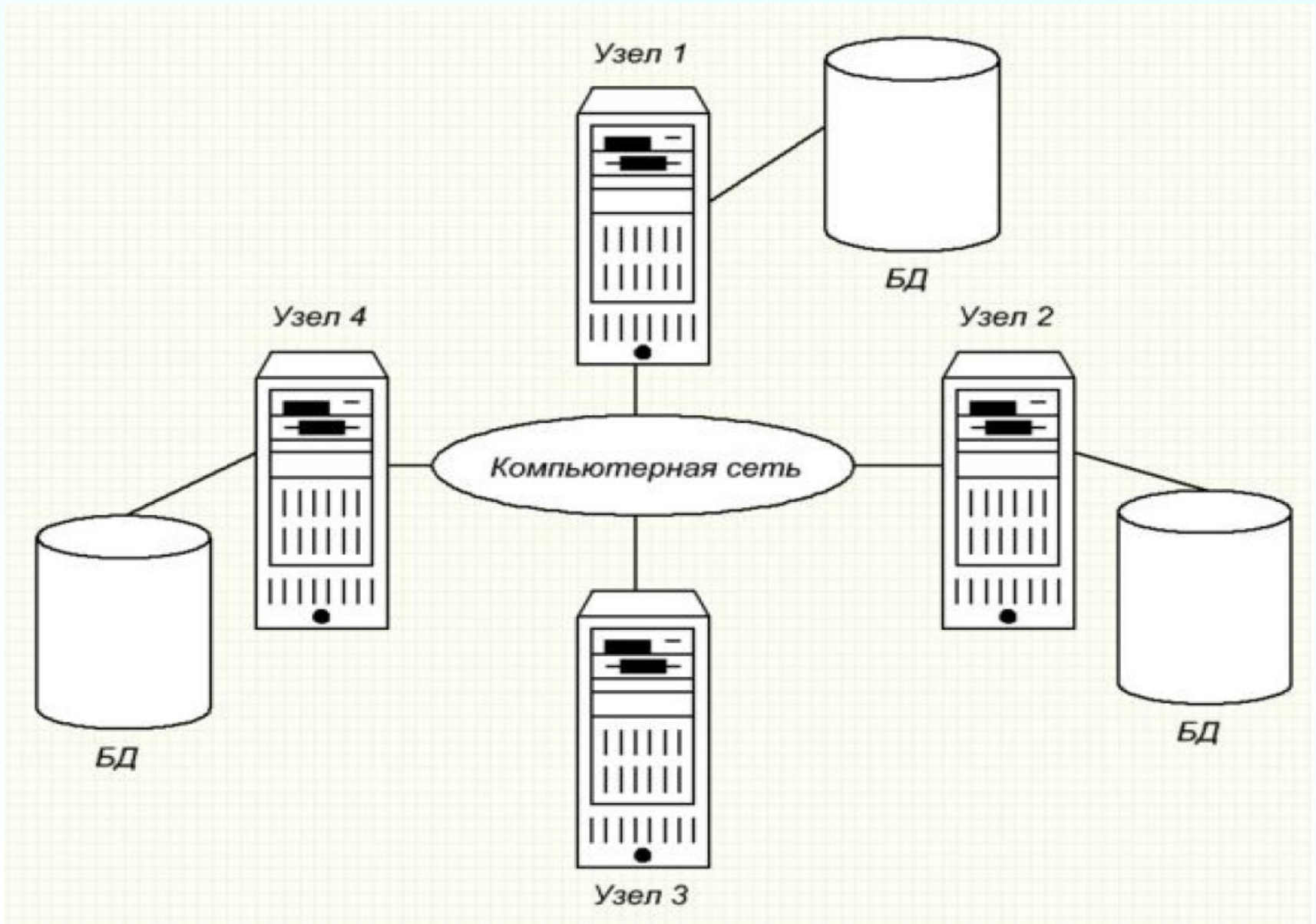
Распределенная система управления базой данных состоит из единой логической базы данных, разделенной на некоторое количество **фрагментов**.

Каждый фрагмент базы данных сохраняется на одном или нескольких компьютерах, работающих под управлением отдельных СУБД и соединенных между собой сетью связи.

Любой узел способен независимо обрабатывать запросы пользователей, требующие доступа к локально сохраняемым данным (т.е. каждый узел обладает определенной степенью автономности), а также способен обрабатывать данные, сохраняемые на других компьютерах сети.



Топология распределенной СУБД



Локальная база данных не обязательно должна существовать на каждом из узлов системы.

3. Двенадцать общих целей систем распределенных баз данных

Цели систем распределенных баз данных

1. Локальная независимость
2. Отсутствие зависимости от центрального узла
3. Непрерывное функционирование
4. Независимость от расположения
5. Независимость от фрагментации
6. Независимость от репликации
7. Обработка распределенных запросов
8. Управление распределенными транзакциями
9. Аппаратная независимость.

Двенадцать общих целей систем распределенных БД

1. Локальная независимость

- Узлы в распределенной системе должны быть независимы, или автономны. Локальная независимость означает, что все операции на узле контролируются этим узлом. Никакой узел X не должен зависеть от некоторого узла Y, чтобы успешно функционировать (иначе, если узел Y будет отключен, узел X не сможет функционировать, даже если на самом узле X будет все в порядке).
- Локальная независимость также означает, что локальные данные имеют локальную принадлежность, управление и учет. Все данные реально принадлежат одной и той же локальной базе данных, даже если доступ к ней осуществляется с других, удаленных узлов. Следовательно, такие вопросы, как безопасность, целостность, защита и представление локальных данных на физическом устройстве хранения, остаются под контролем и в пределах компетенции локального узла.

Двенадцать общих целей систем распределенных БД

2. Отсутствие зависимости от центрального узла

- Локальная независимость предполагает, что все узлы в распределенной системе должны рассматриваться как равные. Поэтому, в частности, не должно быть никаких обращений к центральному, или главному, узлу для получения некоторой централизованной услуги. Не должно быть, например, централизованной обработки запросов, централизованного управления транзакциями или централизованной службы присваивания имен, поскольку в таких случаях система в целом будет зависимой от центрального узла.
- Зависимость от центрального узла может оказаться нежелательной по крайней мере по двум следующим причинам. Во-первых, такой центральный узел, скорее всего, станет узким местом системы, и, во-вторых (что еще хуже), система станет уязвимой — если работа центрального узла будет нарушена, то вся система выйдет из строя (проблема единственного источника отказа).

3. Непрерывное функционирование

- В общем случае преимущество распределенных систем состоит в том, что они должны предоставлять более высокую степень надежности и доступности.
- Надежность понимается как высокая степень вероятности того, что система будет работоспособна и будет функционировать в любой заданный момент. Надежность распределенных систем повышается за счет того, что они не опираются на принцип "все или ничего": распределенные системы могут непрерывно функционировать (по меньшей мере, в сокращенном варианте) даже в случаях отказов части их компонентов, таких как отдельный узел.
- Доступность понимается как высокая степень вероятности того, что система окажется исправной и работоспособной и будет непрерывно функционировать в течение определенного времени.

Двенадцать общих целей систем распределенных БД

4. Независимость от расположения

- Пользователи не должны знать, где именно данные хранятся физически и должны поступать так (по крайней мере, с логической точки зрения), как если бы все данные хранились на их собственном локальном узле.
- Благодаря независимости от расположения упрощаются пользовательские программы и терминальные операции. В частности, данные могут быть перенесены с одного узла на другой, и это не должно требовать внесения каких-либо изменений в использующие их программы или действия пользователей. Такая переносимость желательна, поскольку она позволяет перемещать данные в сети в соответствии с изменяющимися требованиями к эффективности работы системы.
- Независимость от расположения представляет собой расширение концепции физической независимости от данных применительно к распределенным системам.

Двенадцать общих целей систем распределенных БД

5. Независимость от фрагментации

- Система поддерживает независимость данных от фрагментации, если некоторая переменная отношения может быть разделена на части, или фрагменты, при организации ее физического хранения, а различные фрагменты могут храниться на разных узлах. Фрагментация желательна для повышения производительности системы. В этом случае данные могут храниться в том месте, где они чаще всего используются, что позволяет достичь локализации большинства операций и уменьшения сетевого трафика.
- Система, которая поддерживает фрагментацию данных, должна поддерживать и независимость от фрагментации, т.е. пользователи должны иметь возможность работать точно так, по крайней мере, с логической точки зрения, как если бы данные в действительности были вовсе не фрагментированы. Независимость от фрагментации гарантирует, что в любой момент данные могут быть заново восстановлены (а фрагменты перераспределены) в ответ на изменение требований к эффективности работы системы, причем ни пользовательские программы, ни терминальные операции при этом не затрагиваются.

Двенадцать общих целей систем распределенных БД

6. Независимость от репликации

- Система поддерживает репликацию данных, если данная хранимая переменная отношения (или в общем случае данный фрагмент данной хранимой переменной отношения) может быть представлена несколькими отдельными копиями, или репликами, которые хранятся на нескольких отдельных узлах.
- Репликация желательна по крайней мере по двум причинам. Во-первых, она способна обеспечить более высокую производительность, поскольку приложения смогут обрабатывать локальные копии вместо того, чтобы устанавливать связь с удаленными узлами. Во-вторых, наличие репликации может также обеспечивать более высокую степень доступности, поскольку любой реплицируемый объект остается доступным для обработки
- Главным недостатком репликации является то, что если реплицируемый объект обновляется, то и все его копии должны быть обновлены.

Двенадцать общих целей систем распределенных БД

6. Независимость от репликации

- Репликация, как и фрагментация, теоретически должна быть "прозрачной для пользователя". Другими словами, система, которая поддерживает репликацию данных, должна также поддерживать независимость от репликации (иногда говорят "прозрачность репликации"). Для пользователей должна быть создана такая среда, чтобы они могли считать, что в действительности данные не дублируются.
- Независимость от репликации (как и независимость от расположения и независимость от фрагментации) является весьма желательной, поскольку она упрощает создание пользовательских программ и выполнение терминальных операций. В частности, независимость от репликации позволяет создавать и уничтожать дубликаты в любой момент в соответствии с изменяющимися требованиями, не затрагивая при этом никакие из пользовательских программ или терминальных операций.

7. Обработка распределенных запросов

- При обработке распределенных запросов важна оптимизация, поэтому распределенные системы всегда должны быть реляционными (реляционные системы позволяют оптимизировать обработку запросов, а нереляционные — нет).

8. Управление распределенными транзакциями

- Существует два главных аспекта управления транзакциями, а именно: управление восстановлением и управлением параллельностью обработки. Оба этих аспекта имеют расширенную трактовку в среде распределенных систем.
- В распределенной системе отдельная транзакция может потребовать выполнения кода на многих узлах, в частности, это могут быть операции обновления, выполняемые на нескольких узлах. Поэтому говорят, что каждая транзакция содержит несколько агентов, где под агентом подразумевается процесс, который выполняется для данной транзакции на отдельном узле. Система должна знать, что два агента являются элементами одной и той же транзакции, например, два агента, которые являются частями одной и той же транзакции, безусловно, не должны оказываться в состоянии взаимной блокировки.

8. Управление распределенными транзакциями

- Управление восстановлением. Чтобы обеспечить неразрывность транзакции (выполнение ее по принципу "все или ничего") в распределенной среде, система должна гарантировать, что все множество относящихся к данной транзакции агентов или зафиксировало свои результаты, или выполнило откат.
- Управление параллельностью. В большинстве распределенных систем управление параллельностью базируется на механизме блокировки, точно так, как и в нераспределенных системах. В более новых коммерческих продуктах используется управление параллельной работой на основе одновременной поддержки многих версий. Но на практике обычная блокировка все еще остается тем методом, который лучше всего подходит для большинства систем.

Двенадцать общих целей систем распределенных БД

9. Аппаратная независимость.

10. Независимость от операционной системы

11. Независимость от сети

12. Независимость от типа СУБД.

4. Преимущества и недостатки распределенных СУБД

Преимущества и недостатки распределенных СУБД

Преимущества:

- ☑ Отражение структуры организации
- ☑ Высокая степень разделяемости и локальной автономности, повышающая эффективность обработки
- Данные могут быть помещены на тот узел, где зарегистрированы пользователи, чаще всего работающие с этими данными. В результате заинтересованные пользователи получают локальный контроль над требуемыми им данными и могут устанавливать или регулировать локальные ограничения на их использование.
- Администратор глобальной базы данных (АБД) отвечает за систему в целом. Как правило, часть этой ответственности делегируется на локальный уровень, благодаря чему АБД локального уровня получает возможность управлять локальной СУБД

Преимущества и недостатки распределенных СУБД

Преимущества:

☑ **Повышение доступности данных**

□ В централизованных СУБД отказ центрального компьютера вызывает прекращение функционирования всей СУБД. Однако отказ одного из узлов распределенной СУБД или линии связи между узлами приводит к тому, что становятся недоступными лишь некоторые узлы, тогда как вся система в целом сохраняет свою работоспособность.

☑ **Повышение надежности**

□ Если организована репликация данных, в результате чего данные и их копии будут размещены на нескольких узлах, отказ отдельного узла или линии связи между узлами не приведет к прекращению доступа к данным в

Преимущества и недостатки распределенных СУБД

Преимущества:

☑ **Повышение производительности**

- Поскольку каждый узел работает только с частью базы данных, степень использования центрального процессора и служб ввода-вывода может оказаться ниже, чем в случае централизованной СУБД.

☑ **Экономические выгоды**

- Выгоднее устанавливать в подразделениях организации собственные маломощные компьютеры, кроме того, гораздо дешевле добавить в сеть новые рабочие станции, чем модернизировать систему с мэйнфреймом.
- Из-за относительно высокой стоимости передачи данных по сети (по сравнению со стоимостью их локальной обработки) может оказаться экономически выгодным разделить приложение на соответствующие части и выполнять необходимую обработку на каждом

Преимущества и недостатки распределенных СУБД

Преимущества:



Модульность системы

- В распределенной среде расширение существующей системы осуществляется намного проще. Добавление в сеть нового узла не оказывает влияния на функционирование уже существующих. Подобная гибкость позволяет организации легко расширяться. Перегрузки из-за увеличения размера базы данных обычно устраняются путем добавления в сеть новых вычислительных мощностей и устройств внешней памяти.
- В централизованных СУБД расширение базы данных может потребовать замены оборудования (более мощной системой) и используемого программного обеспечения (более мощной или более гибкой СУБД)

Преимущества и недостатки распределенных СУБД

Недостатки:

☑ **Повышение сложности**

- ☐ Распределенные СУБД, способные скрыть от конечных пользователей распределенную природу используемых ими данных и обеспечить необходимый уровень производительности, надежности и доступности, являются более сложными программными комплексами, чем централизованные СУБД.

☑ **Увеличение стоимости**

- ☐ Увеличение сложности означает и увеличение затрат на приобретение и сопровождение распределенной СУБД (по сравнению с обычными централизованными СУБД).
- ☐ Развертывание распределенной СУБД требует дополнительного оборудования, необходимого для установки сетевых соединений между узлами.
- ☐ Следует ожидать и увеличения расходов на оплату каналов связи, вызванных ростом сетевого трафика.
- ☐ Возрастут затраты на оплату труда персонала, который потребуются для обслуживания распределенных СУБД.

Преимущества и недостатки распределенных СУБД

Недостатки:

☑ Проблемы защиты

□ В централизованных системах доступ к данным легко контролируется. Однако в распределенных системах потребуется организовать контроль доступа не только к копируемым данным, расположенных на нескольких производственных площадках, но и защиту самих сетевых соединений.

☑ Усложнение контроля за целостностью данных

□ Целостность базы данных означает правильность и согласованность хранящихся в ней данных. Реализация ограничений поддержки целостности обычно требует доступа к большому количеству данных, используемых при выполнении проверок обеспечения требований целостности. В распределенных СУБД повышенная стоимость передачи и обработки данных может препятствовать организации эффективной защиты от нарушений

Преимущества и недостатки распределенных СУБД

Недостатки:



Отсутствие стандартов

- Функционирование распределенных СУБД зависит от эффективности используемых каналов связи, только в последнее время стали вырисовываться контуры стандартов на каналы связи и протоколы доступа к данным. Отсутствие стандартов существенно ограничивает потенциальные возможности распределенных СУБД.
- Кроме того, не существует инструментальных средств и методологий, способных помочь пользователям в преобразовании централизованных систем в распределенные.

Преимущества и недостатки распределенных СУБД

Недостатки:

- ☑ **Недостаток опыта**
 - В настоящее время еще не накоплен необходимый опыт промышленной эксплуатации распределенных систем, сравнимый с опытом эксплуатации централизованных систем
- ☑ **Усложнение процедуры разработки базы данных**
 - Разработка распределенных баз данных, помимо обычных трудностей, связанных с процессом проектирования централизованных баз данных, требует принятия решения о фрагментации данных, распределении фрагментов по отдельным узлам и репликации данных

5. Примеры распределенных систем

Примеры распределенных систем

АИПС "Оружие МВД"

<http://www.rdtex.ru>

система учета гражданского, служебного и боевого оружия, поступающего на вооружение в подразделения МВД России, производимого в Российской Федерации и ввозимого из-за рубежа.

- Локальные базы данных ведутся на каждом объекте автоматизации (информация по региону собирается в базе данных региона), а на верхнем уровне системы находится ссылочная база данных, содержащая сведения о месте регистрации оружия и месте хранения полной информации о нем.
- Данные в систему вносят подразделения, непосредственно занимающиеся учетом оружия (отделы по лицензионно-разрешительной работе, отделы материально-технического и хозяйственного обеспечения).

Примеры распределенных систем

АИПС "Оружие МВД"

- Распределенная промышленная транзакционная база данных и распределенная база данных нормативно-справочной информации реализованы в виде реляционных баз данных под управлением СУБД Oracle. Бизнес логика реализована с использованием процедур на языке PL/SQL.
- Клиентские места созданы на основе приложения Oracle Developer 6i. Для создания нерегламентированных запросов и отчетов использовался Oracle Discoverer 9.02, ориентированный на специалистов предметной области. Для обмена данными между различными объектами автоматизации разработан единый формат обмена, базирующийся на формате XML-файлов.

На основе материалов сайта <http://www.rdtex.ru>

Спасибо за внимание!