



Резервное копирование (англ. backup copy) — процесс создания копии данных на носителе (жёстком диске, дискете и т. д.), предназначенном для восстановления данных в оригинальном или новом месте их расположения в случае их повреждения или разрушения.

Международный день резервного копирования (англ. World Backup Day) или **День бэкапа** (от *англ. backup*) проводится ежегодно 31 марта.





Зачем выполнять резервное копирование

Создание резервных копий баз данных SQL Server , выполнение проверочных процедур восстановления резервных копий

При правильном создании резервных копий баз данных можно будет восстановить данные после многих видов аварий

- сбой носителя;
- ошибки пользователей (например, удаление таблицы по ошибке);
- сбои оборудования (например, поврежденный дисковый накопитель или безвозвратная потеря данных);
- стихийные бедствия. Используя SQL Server Резервное копирование для Хранилище BLOB-объектов Azure

Кроме того, резервные копии баз данных полезны и при выполнении повседневных административных задач

Термины

Создание резервных копий

Копирование данных или записей журнала из базы данных SQL Server или журнала ее транзакций на устройство для резервного копирования, например на диск, на котором создается резервная копия данных или журнала.

Резервная копия

Копия данных SQL Server , используемая для восстановления данных после возникновения ошибки. Резервная копия данных SQL Server создается на уровне базы данных для одного или нескольких файлов или групп файлов. Нельзя создать резервные копии на уровне таблиц. Кроме резервной копии данных модель полного восстановления требует создания резервной копии журнала транзакций.

Модель восстановления

Свойство базы данных, с помощью которого выполняется управление обслуживанием журналов транзакций в базе данных. Существует три модели восстановления: простая модель восстановления, модель полного восстановления и модель восстановления с неполным протоколированием. Модель восстановления базы данных определяет требования к резервному копированию и восстановлению.

Восстановление

Многоэтапный процесс, в ходе которого все данные и страницы журнала копируются из указанной резервной копии SQL Server в определенную базу данных, а затем выполняется накат всех фиксированных транзакций, записанных в резервной копии журнала, путем внесения новых данных на основе зарегистрированных изменений.

Модель восстановления SQL Server

Модель восстановления — это свойство базы данных.

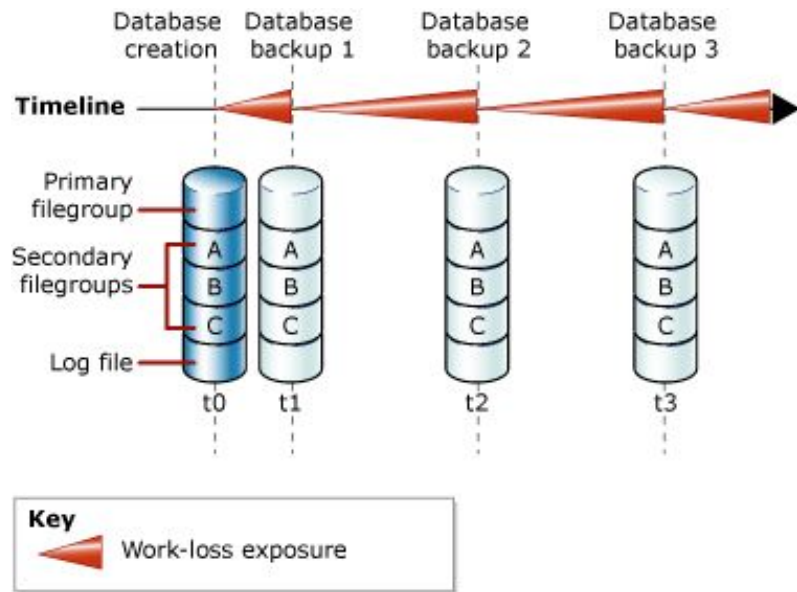
Модель восстановления контролирует следующее:

- Как SQL Server регистрирует транзакции для базы данных.
- Требуется ли резервное копирование журнала транзакций базы данных.
- Какие операции восстановления доступны для восстановления базы данных.

SQL Server предлагает три модели восстановления:

- простая модель восстановления (**SIMPLE**)
- модель полного восстановления (**FULL**)
- модель восстановления с неполным протоколированием (**BULK_LOGGED**)

Простая модель восстановления SQL Server



В SIMPLE модели восстановления SQL Server удаляет журналы транзакций из файлов журналов транзакций в каждой контрольной точке.

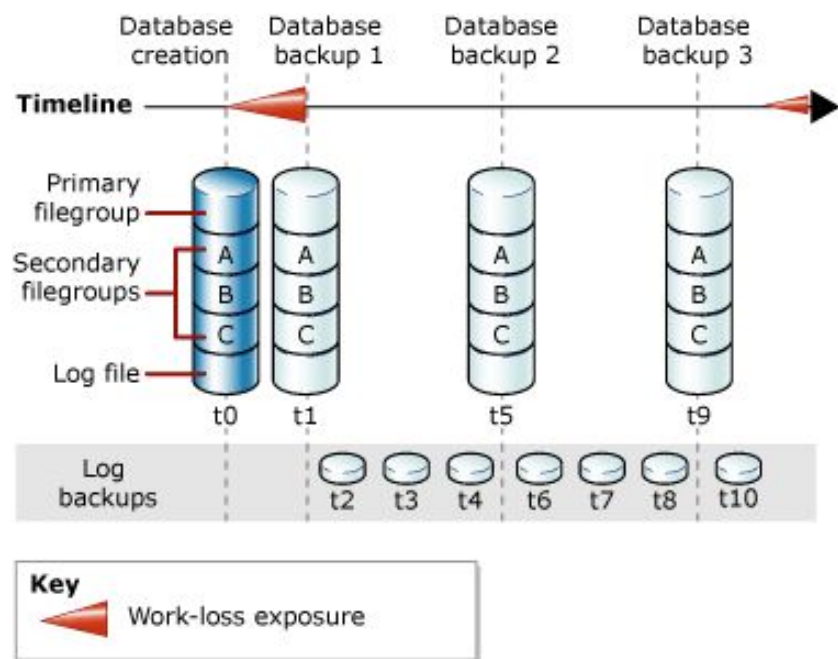
Это приводит к относительно небольшим файлам журнала транзакций.

Изменения с момента создания последней резервной копии не защищены. В случае аварийной ситуации эти изменения придется вносить повторно.

Операции, требующие резервного копирования журнала транзакций, не поддерживаются.

Восстановление на определенный момент времени не поддерживается

Полная модель и модель с неполным протоколированием



Для баз данных, в которых используются полная модель восстановления и модель восстановления с неполным протоколированием, создание резервных копий баз данных необходимо, но недостаточно для нормальной работы. Требуется также создание резервных копий журнала транзакций.

FULL модель восстановления позволяет восстановить базу данных в любой момент времени.

В FULL модели восстановления SQL Server хранит журналы транзакций в файлах журналов транзакций до тех пор, пока BACKUP LOG оператор не будет выполнен. Возможно переполнение журнала.

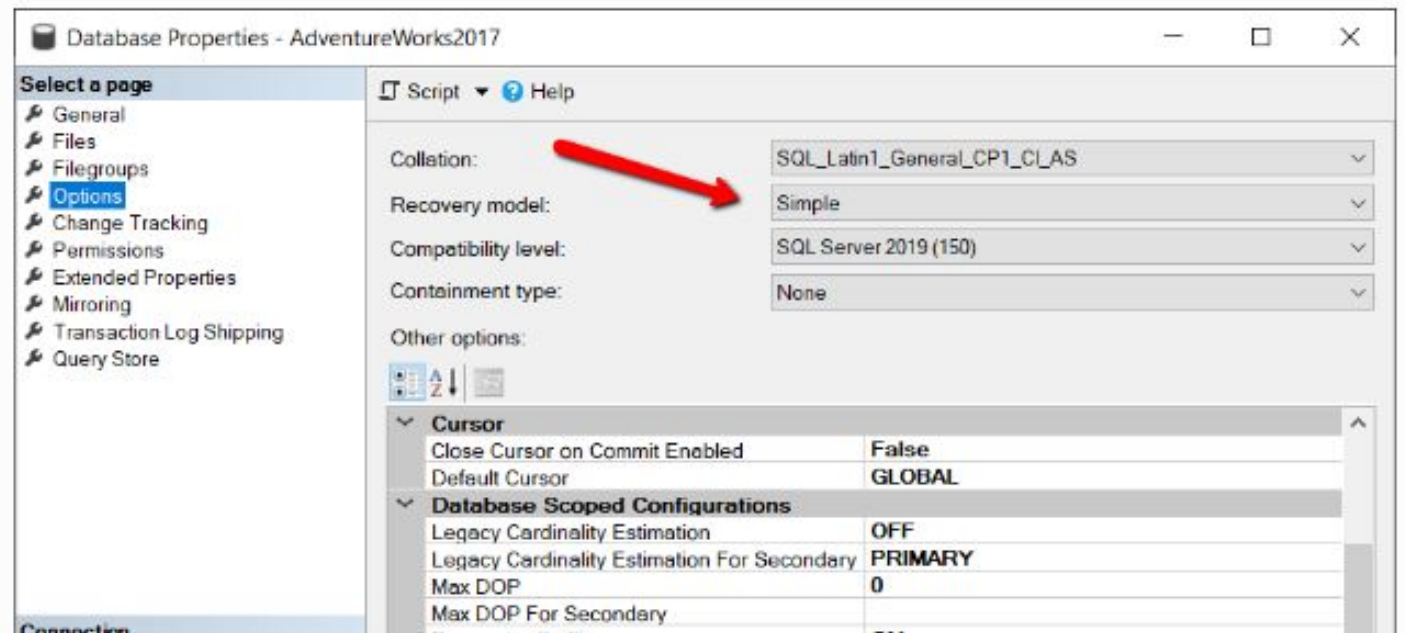
BULK_LOGGED восстановления работает почти так же, как и FULL модель восстановления, за исключением операций с неполным протоколированием (BULK INSERT). Не позволяет восстановить базу данных в любой момент времени.

Модель восстановления	Описание	Потеря данных	Восстановление на момент времени
SIMPLE	Нет резервных копий журналов	Изменения, внесенные с момента последней резервной копии, теряются.	Может восстановить только до конца резервной копии.
FULL	Требовать резервные копии журналов	Обычно нет	Может восстанавливаться до определенного момента времени при условии, что резервное копирование завершено к этому моменту времени.
BULK_LOGGED	Требовать резервные копии журналов	<p>Если журнал поврежден или с момента последней резервной копии журнала произошли операции с неполным протоколированием, изменения, внесенные с момента последней резервной копии, необходимо выполнить повторно.</p> <p>В противном случае работа не будет потеряна.</p>	Можно восстановить до конца любой резервной копии. Восстановление на момент времени не поддерживается.

Просмотр модели восстановления базы данных

```
SELECT name, recovery_model_desc  
FROM master.sys.databases  
WHERE name = 'HR';
```

name	recovery_model_desc
HR	FULL



Изменение модели восстановления

```
USE [master];  
GO  
ALTER DATABASE [model]  
SET RECOVERY FULL;  
GO
```

```
ALTER DATABASE database_name  
SET RECOVERY recovery_model;
```


Системные базы данных

Database	Recovery model	Backup
master	Simple	Yes
msdb	Simple	Yes
model	Full*	Yes
tempdb	Simple	No
resource	n/a	No

TYPES OF BACKUP: FULL, DIFFERENTIAL, AND INCREMENTAL

Full Backups: Entire data set, regardless of any previous backups or circumstances.



Differential Backups: Additions and alterations since the most recent full backup.



Incremental Backups: Additions and alterations since the most recent incremental backup.



Initial Full Backup • 1st Backup 2nd Backup 3rd Backup 4th Backup 5th Backup

 Data subject to backup

Основные типы резервного копирования

1. Полное
 2. Дифференциальное
 3. Инкрементное
- (резервное копирование журналов)

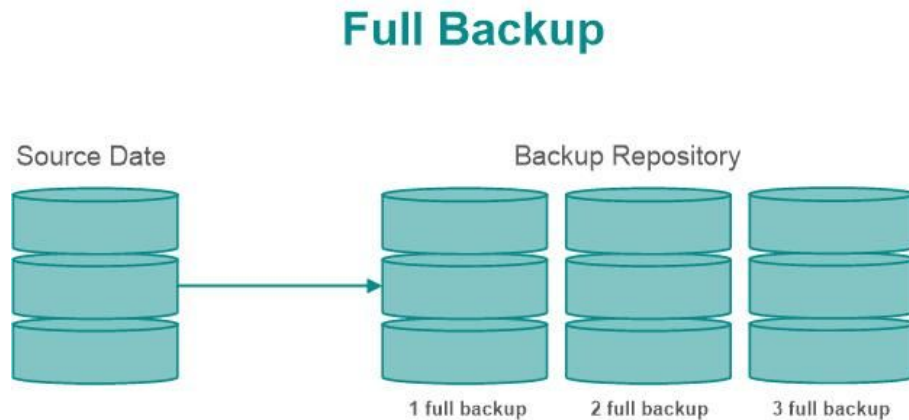
Полное резервное копирование

Полная резервная копия базы данных создает резервную копию всей базы данных.

Он включает в себя следующие данные:

- Метаданные базы данных, такие как имя, дата создания, параметры базы данных, пути к файлам и т. д.
- Используемые страницы данных каждого файла данных.

Кроме того, полная резервная копия включает в себя часть журнала транзакций. Он представляет базу данных на момент завершения резервного копирования.



При выполнении полного резервного копирования SQL Server может использовать значительный объем дискового ввода-вывода. Поэтому следует выполнять полное резервное копирование в то время, когда рабочая нагрузка невелика, например, ночью.

Обратите внимание, что вы должны выполнить хотя бы одно полное резервное копирование, чтобы выполнять резервное копирование других типов, таких как дифференциальное резервное копирование и резервное копирование журнала транзакций .

Создание полной резервной копии базы данных с помощью T-SQL.

```
BACKUP DATABASE database_name  
TO DISK = path_to_backup_file  
WITH options;
```

database_name — это имя базы данных, резервную копию которой вы хотите создать. База данных должна существовать и нормально работать на сервере.

path_to_backup_file — это путь к файлу резервной копии. По соглашению расширение резервной копии — bak. Если вы храните файл резервной копии в папке, эта папка должна существовать, потому что оператор не будет неявно создавать папку для вас.

option следующее **за WITH** ключевым словом указывает один или несколько параметров резервного копирования

INIT/NOINIT NAME FORMAT FILE

Server Properties - dbserverv

Select a page

- General
- Memory
- Processors
- Security
- Connections
- Database Settings
- Advanced
- Permissions

Script Help

Default index fill factor:

0

Backup and restore

Specify how long SQL Server will wait for a new tape.

☐ Wait indefinitely

☒ Try once

☐ Try for 0 minute(s)

Default backup media retention (in days):

0

☐ Compress backup ☐ Backup checksum

Recovery

Recovery interval (minutes):

0

Database default locations

Data: D:\MSSQLData\

Log: D:\MSSQLData\

Backup: D:\MSSQLBackup

☒ Configured values ☐ Running values

Задание места хранения резервной копии

```
BACKUP DATABASE HR
TO DISK = 'C:\work\CLR\hr.bak'
WITH NOINIT,
NAME = 'HR-Full Database Backup';

RESTORE HEADERONLY
FROM DISK = 'C:\work\CLR\hr.bak';
```

	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserName	ServerName	DatabaseName	Da
1	HR-Full Database Backup	NULL	1	NULL	0	1	2	sa	DBSERVERV	HR	86
2	HR-Full Database Backup	NULL	1	NULL	0	2	2	sa	DBSERVERV	HR	86

Восстановление базы данных из полной резервной копии

	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserName	ServerName	DatabaseName	DatabaseVer
1	HR-Full Database Backup	NULL	1	NULL	0	1	2	sa	DBSERVERV	HR	869
2	HR-Full Database Backup	NULL	1	NULL	0	2	2	sa	DBSERVERV	HR	869

RESTORE DATABASE database_name
FROM DISK = path_to_backup_file
WITH options;

```
--ВОССТАНОВЛЕНИЕ
USE master;
DROP DATABASE HR;
RESTORE DATABASE HR
FROM DISK = N'C:\work\CLR\hr.bak'
WITH FILE = 1;
Use HR;
SELECT * FROM People;
```

	Id	FirstName	LastName
1	1	John	Doe
2	2	Jane	Doe
3	3	Upton	Luis
4	4	Dach	Keon

```
--ВОССТАНОВЛЕНИЕ
USE master;
DROP DATABASE HR;
RESTORE DATABASE HR
FROM DISK = N'C:\work\CLR\hr.bak'
WITH FILE = 2;
Use HR;
SELECT * FROM People;
```

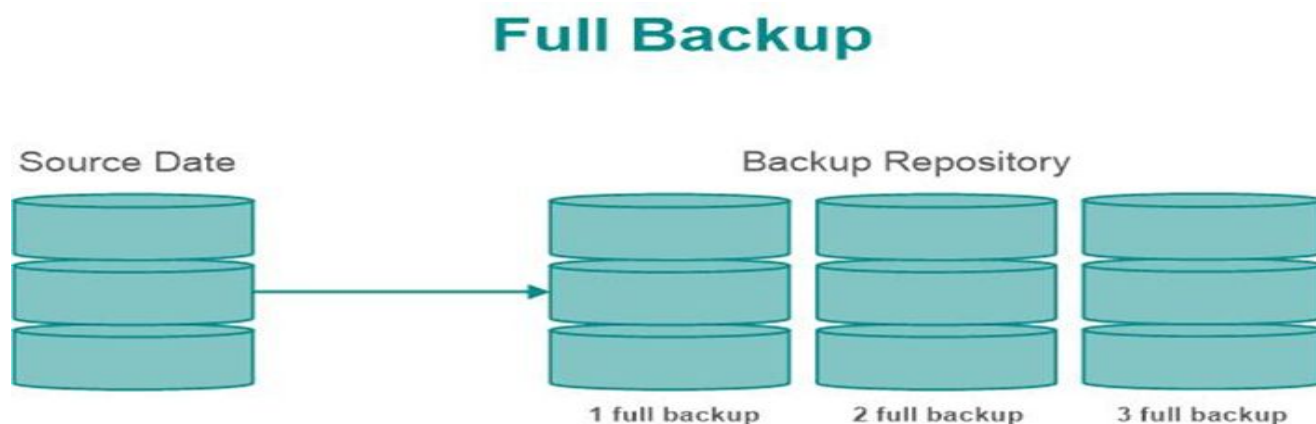
	Id	FirstName	LastName
1	1	John	Doe
2	2	Jane	Doe
3	3	Upton	Luis
4	4	Dach	Keon
5	5	Bob	Climo

Преимущества Full Backup:

- быстрое восстановление данных
- простое управление
- все данные содержатся в одной резервной копии

Недостатки Full Backup:

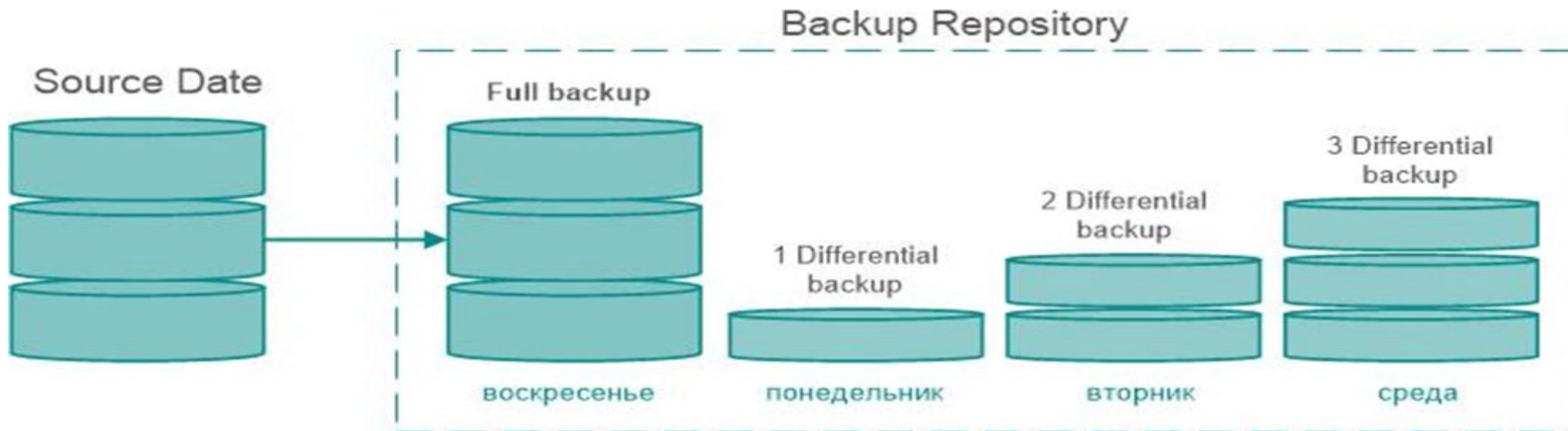
- требует много места для хранения резервных копий
- высокая загрузка сети
- длительное выполнение резервного копирования



Дифференциальное резервное копирование SQL Server

Дифференциальная резервная копия фиксирует все изменения с момента последней полной резервной копии. И эта полная резервная копия называется базой для дифференциальной резервной копии.

Differential Backup



Дифференциальное резервное копирование SQL Server

BACKUP DATABASE

database_name

TO DISK = path_to_backup_file

WITH DIFFERENTIAL;

Results		Messages						
	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserName
1	HR-Full Database Backup	NULL	1	NULL	0	1	2	sa
2	HR-Differential Database Backup	NULL	5	NULL	0	2	2	sa
3	HR-Differential Database Backup	NULL	5	NULL	0	3	2	sa
4	HR-Full Database Backup	NULL	1	NULL	0	4	2	sa
5	HR-Differential Database Backup	NULL	5	NULL	0	5	2	sa

Дифференциальные резервные копии и полные резервные копии

Дифференциальное резервное копирование имеет следующие **преимущества** по сравнению с полным резервным копированием:

Скорость — создание дифференциальной резервной копии может быть очень быстрым по сравнению с созданием полной резервной копии, поскольку дифференциальная резервная копия захватывает только данные, которые изменились с момента последней полной резервной копии.

Хранилище — дифференциальная резервная копия требует меньше места для хранения, чем полная резервная копия.

Меньший риск потери данных — поскольку для дифференциального резервного копирования требуется меньше места для хранения, вы можете выполнять дифференциальное резервное копирование чаще, что снижает риск потери данных.

Дифференциальные резервные копии и полные резервные копии

Недостатки Differential Backup:

каждый последующий бэкап выполняется дольше по времени и занимает больше дискового пространства в хранилище

Восстановление из разностной резервной копии требует больше времени, чем восстановление из полной резервной копии, поскольку вам необходимо восстановить как минимум два файла резервной копии:

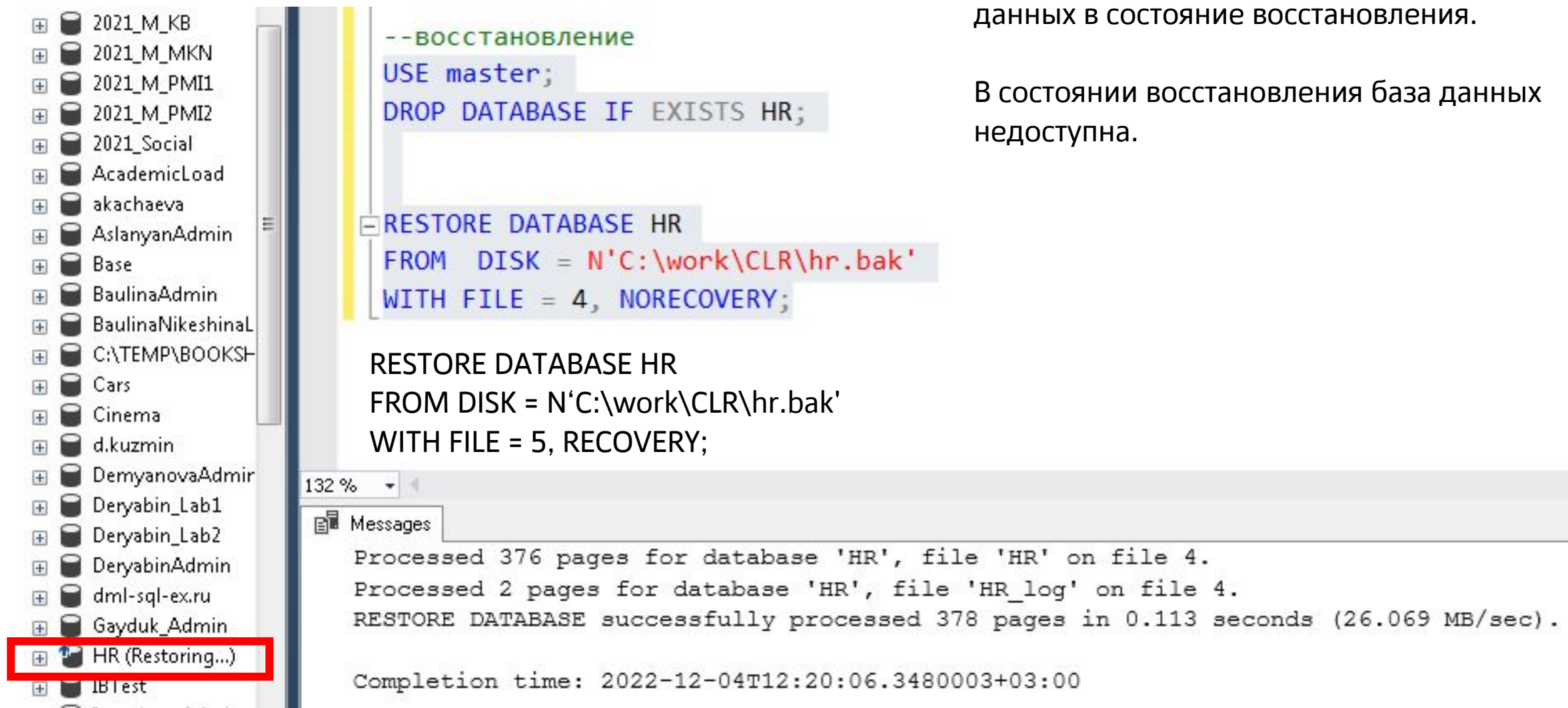
- Сначала восстановите из самой последней полной резервной копии.
- Затем выполните восстановление из разностной резервной копии.

Results		Messages						
	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserName
1	HR-Full Database Backup	NULL	1	NULL	0	1	2	sa
2	HR-Differential Database Backup	NULL	5	NULL	0	2	2	sa
3	HR-Differential Database Backup	NULL	5	NULL	0	3	2	sa
4	HR-Full Database Backup	NULL	1	NULL	0	4	2	sa
5	HR-Differential Database Backup	NULL	5	NULL	0	5	2	sa

Восстановление дифференциальной резервной копии

NORECOVERY параметр переводит базу данных в состояние восстановления.

В состоянии восстановления база данных недоступна.



```
-- ВОССТАНОВЛЕНИЕ
USE master;
DROP DATABASE IF EXISTS HR;

RESTORE DATABASE HR
FROM DISK = N'C:\work\CLR\hr.bak'
WITH FILE = 4, NORECOVERY;
```

RESTORE DATABASE HR
FROM DISK = N'C:\work\CLR\hr.bak'
WITH FILE = 5, RECOVERY;

132 %

Messages

Processed 376 pages for database 'HR', file 'HR' on file 4.
Processed 2 pages for database 'HR', file 'HR_log' on file 4.
RESTORE DATABASE successfully processed 378 pages in 0.113 seconds (26.069 MB/sec).

Completion time: 2022-12-04T12:20:06.3480003+03:00

Резервная копия журнала транзакций SQL Server

Для резервного копирования журнала транзакций модель восстановления базы данных должна быть **FULL** либо **BULK_LOGGED**.

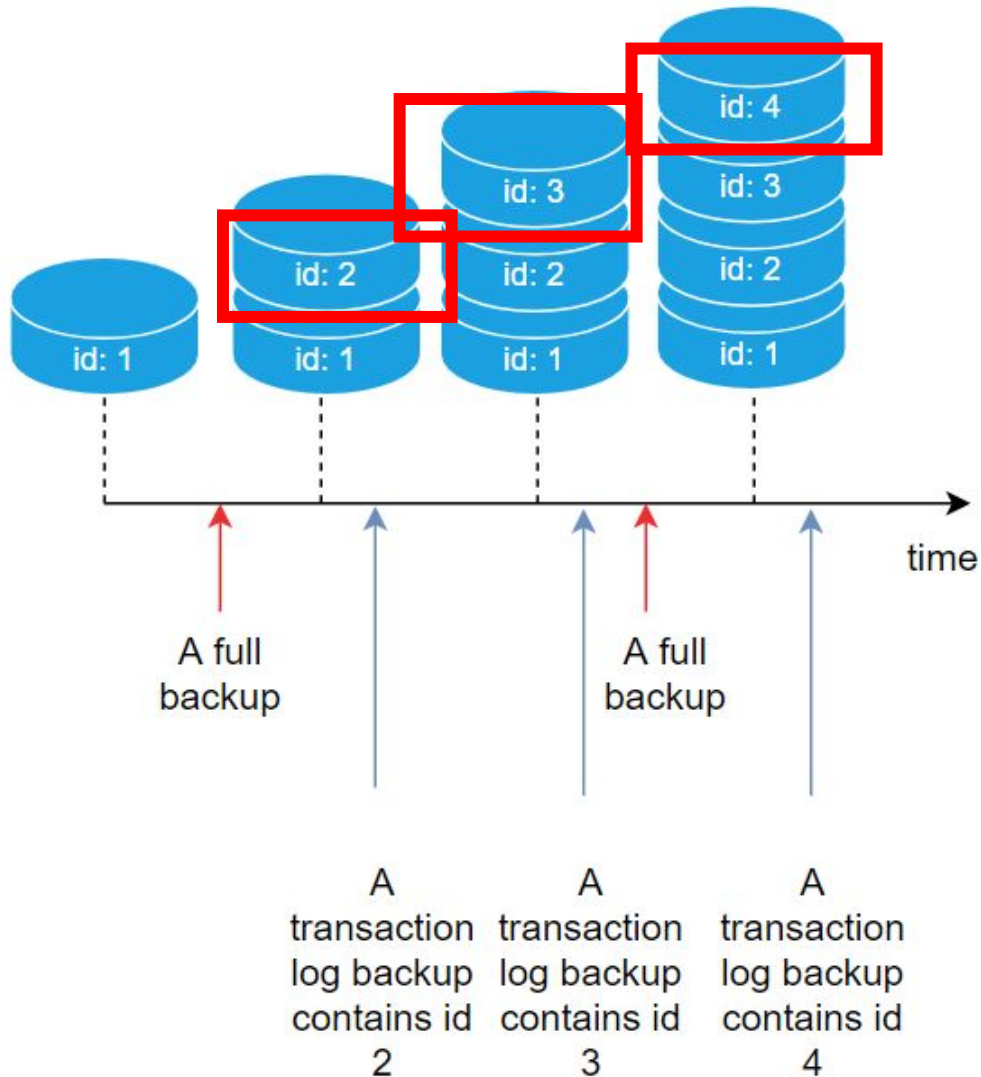
Резервная копия журнала транзакций содержит журнал транзакций базы данных. Перед созданием резервной копии журнала транзакций необходимо создать ***хотя бы одну полную резервную копию***. После этого вы можете создать любое количество резервных копий журнала транзакций.

Рекомендуется чаще делать резервные копии журнала транзакций, чтобы:

- Свести к минимуму потерю данных
- Сократить файлы журнала

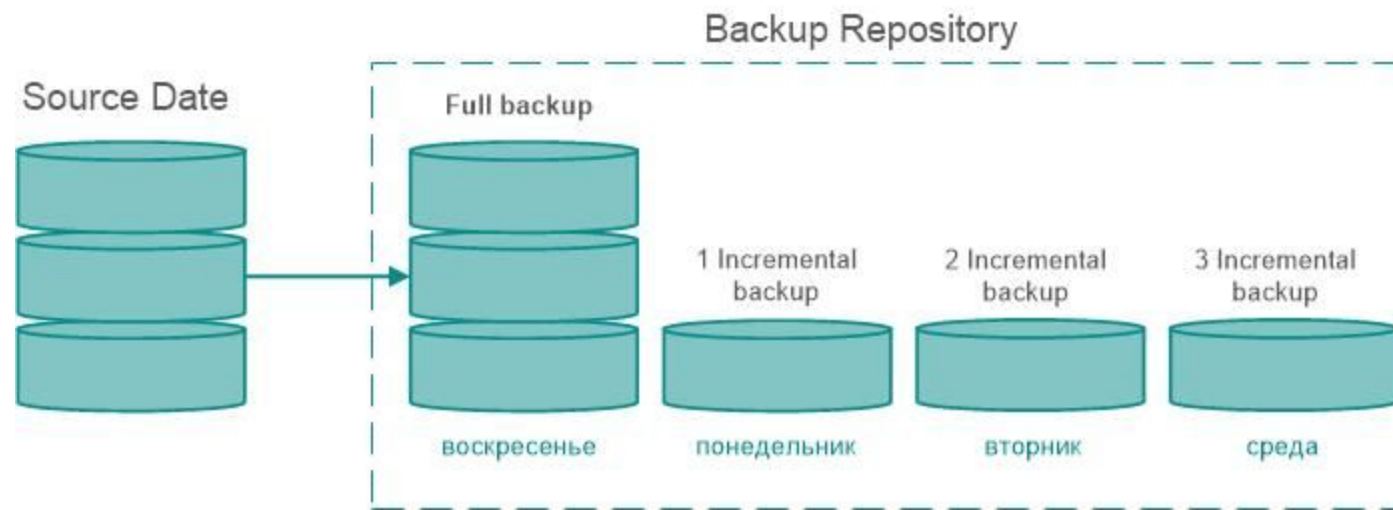
Как правило, вы создаете ***полную резервную*** копию время от времени, например, **еженедельно**, и создаете серию ***дифференциальных резервных копий*** с более короткими интервалами, например **ежедневно**. Независимо от резервных копий базы данных, вы создаете ***копии журнала транзакций*** с более частыми интервалами, например **ежечасно**.

Резервная копия журнала транзакций SQL Server

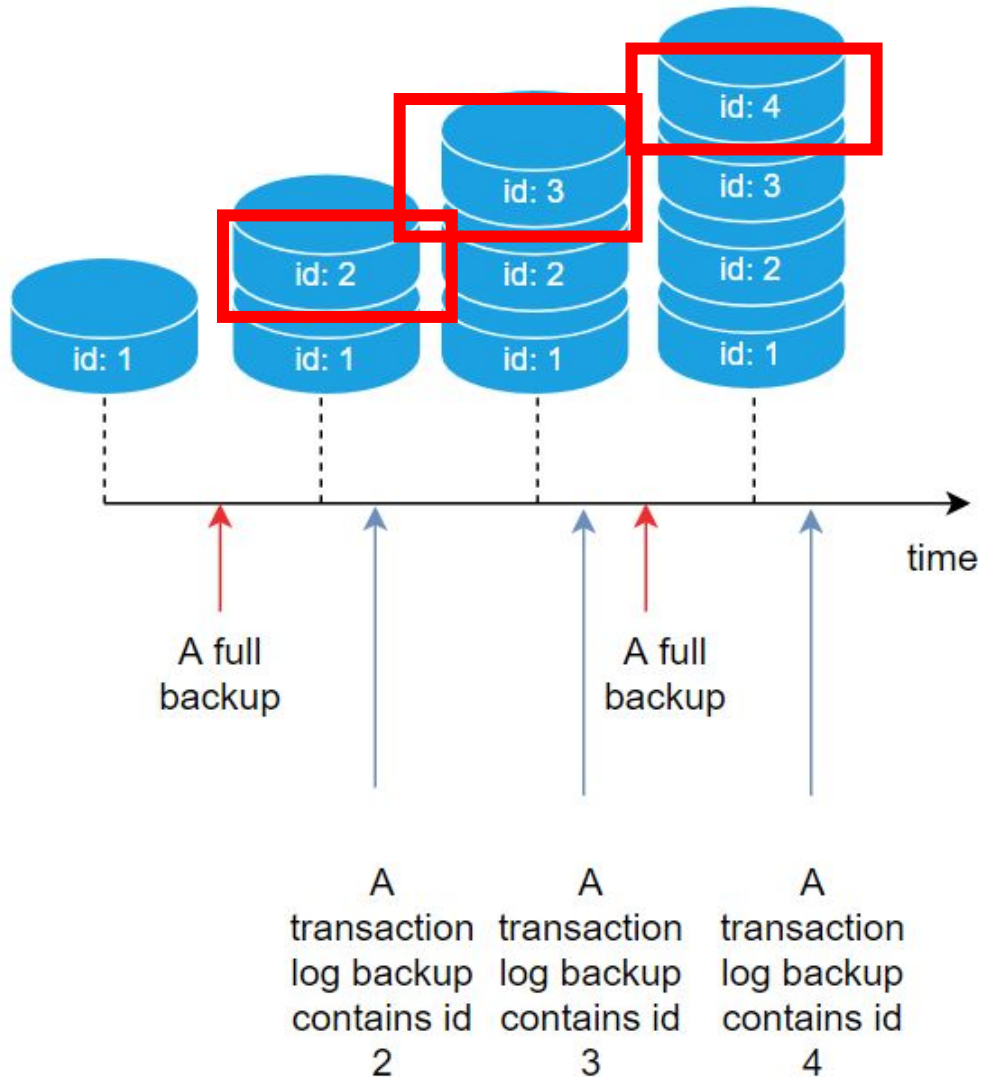


Обратите внимание, что резервные копии журнала транзакций **не содержат повторяющихся данных**, таких как полная резервная копия или дифференциальная резервная копия.

Incremental Backup



Резервная копия журнала транзакций SQL Server



Обратите внимание, что резервные копии журнала транзакций **не содержат повторяющихся данных**, таких как полная резервная копия или дифференциальная резервная копия.

BACKUP LOG database_name
TO DISK = path_to_backup_file
WITH options;

Пример резервной копии журнала транзакций SQL Server

	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserN
1	HR-Full Database Backup	NULL	1	NULL	0	1	2	sa
2	HR-Transaction Log Backup	NULL	2	NULL	0	2	2	sa
3	HR-Transaction Log Backup	NULL	2	NULL	0	3	2	sa
4	HR-Full Database Backup	NULL	1	NULL	0	4	2	sa
5	HR-Transaction Log Backup	NULL	2	NULL	0	5	2	sa

```
ALTER DATABASE HR  
SET RECOVERY FULL;
```

```
-- полная  
BACKUP DATABASE HR  
TO DISK = N'C:\work\CLR\hr.bak'  
WITH INIT,  
NAME = 'HR-Full Database Backup';  
  
INSERT INTO People(FirstName, LastName)  
VALUES ('Jane', 'Doe');  
  
-- первая  
BACKUP LOG HR  
TO DISK = N'C:\work\CLR\hr.bak'  
WITH NAME = N'HR-Transaction Log Backup';
```


Пример резервной копии журнала транзакций SQL Server

Results Messages								
	BackupName	BackupDescription	BackupType	ExpirationDate	Compressed	Position	DeviceType	UserN
1	HR-Full Database Backup	NULL	1	NULL	0	1	2	sa
2	HR-Transaction Log Backup	NULL	2	NULL	0	2	2	sa
3	HR-Transaction Log Backup	NULL	2	NULL	0	3	2	sa
4	HR-Full Database Backup	NULL	1	NULL	0	4	2	sa
5	HR-Transaction Log Backup	NULL	2	NULL	0	5	2	sa

--Восстановить базу данных из резервной копии журнала транзакций.

USE master;

DROP DATABASE HR;

```
RESTORE DATABASE HR
FROM DISK = N'C:\work\CLR\hr.bak'
WITH FILE = 4, NORECOVERY;
```

```
RESTORE LOG HR
FROM DISK = N'C:\work\CLR\hr.bak'
WITH FILE = 5, RECOVERY;
```

Сравнение методов резервного копирования

	Полный	Дифференциальный	Инкрементный
Пространство для хранения	Высокая	От среднего до высокого	Низкий
Скорость резервного копирования	Самый медленный	Быстро	Самый быстрый
Скорость восстановления	Самый быстрый	Быстро	Самый медленный
Носитель, необходимый для восстановления	Только самая последняя резервная копия	Самая последняя полная резервная копия и самая последняя дифференциальная резервная копия	Самая последняя полная резервная копия и все инкрементные резервные копии с момента полной резервной копии
Дублирование	Хранит много дубликатов	Сохраняет дубликаты	Нет дубликатов

Золотое правило бэкапа 3-2-1



3

Создайте три копии данных



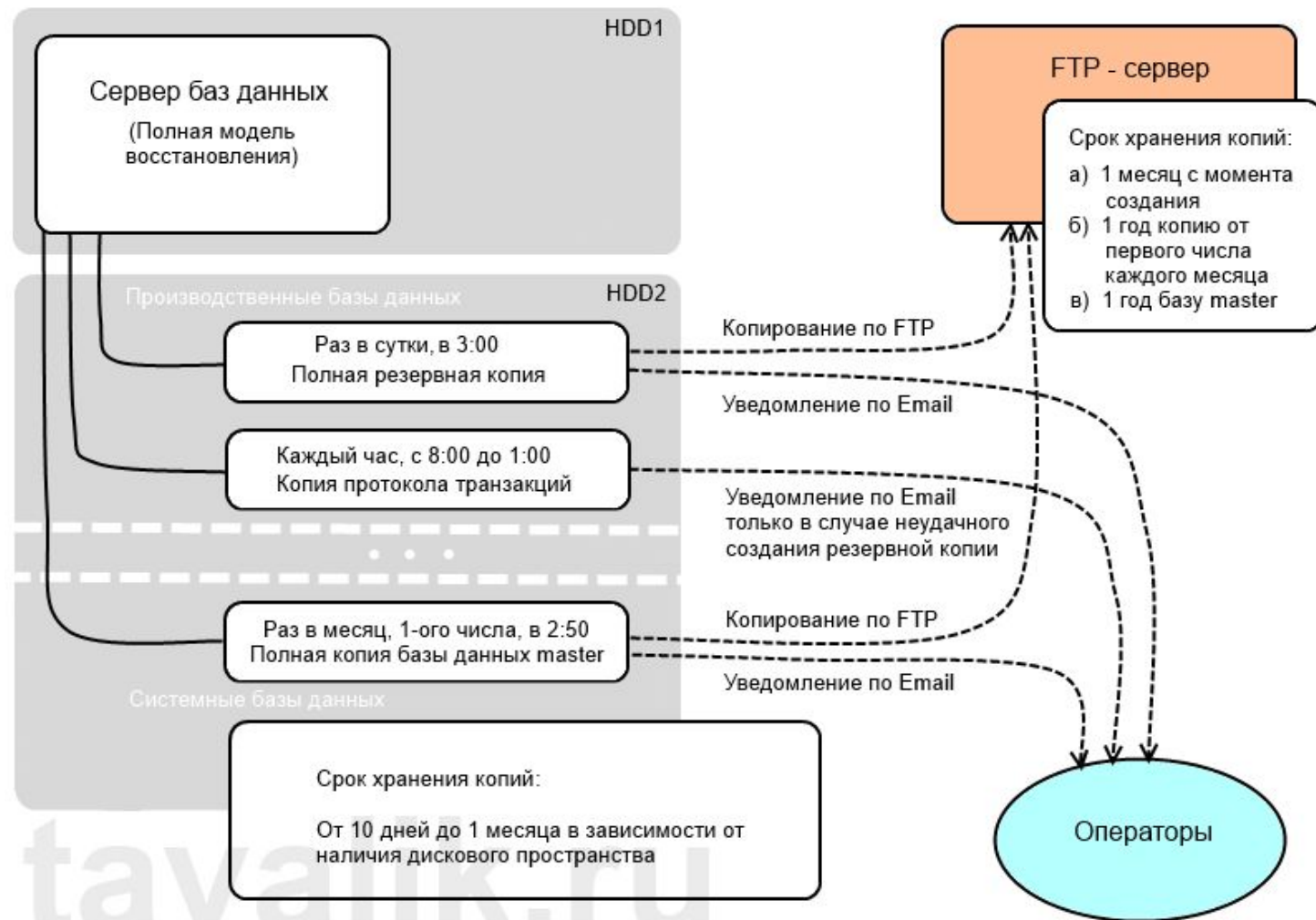
2

Храните копии на двух разных устройствах хранения



1

Одну копию храните на удаленном хранилище



Разрешения

Разрешения ***BACKUP DATABASE*** и ***BACKUP LOG*** по умолчанию назначаются участникам предопределенной роли сервера sysadmin и предопределенным ролям базы данных db_owner и db_backupoperator.

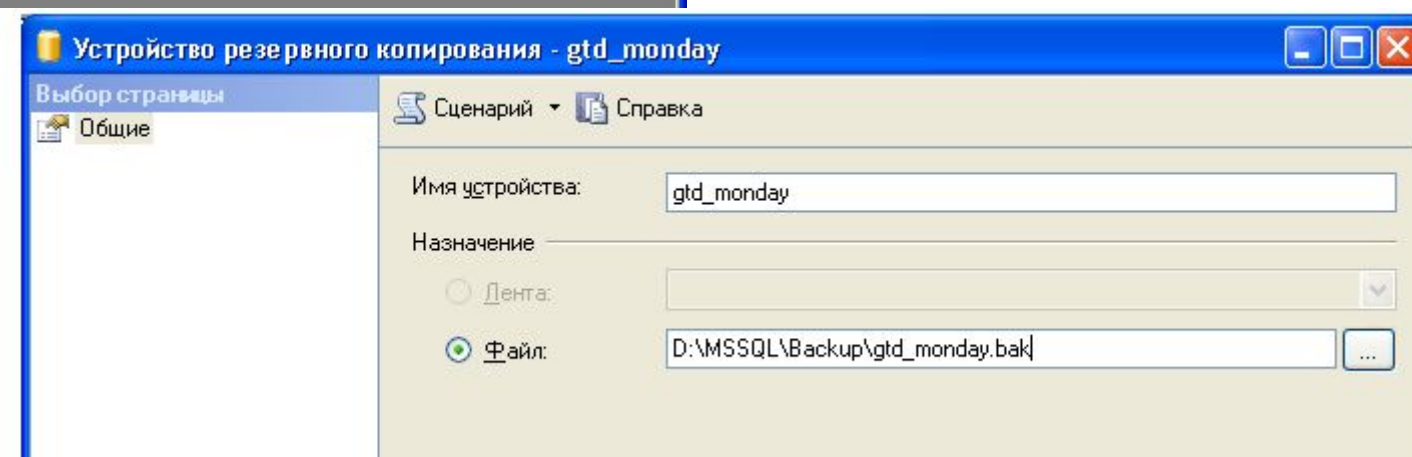
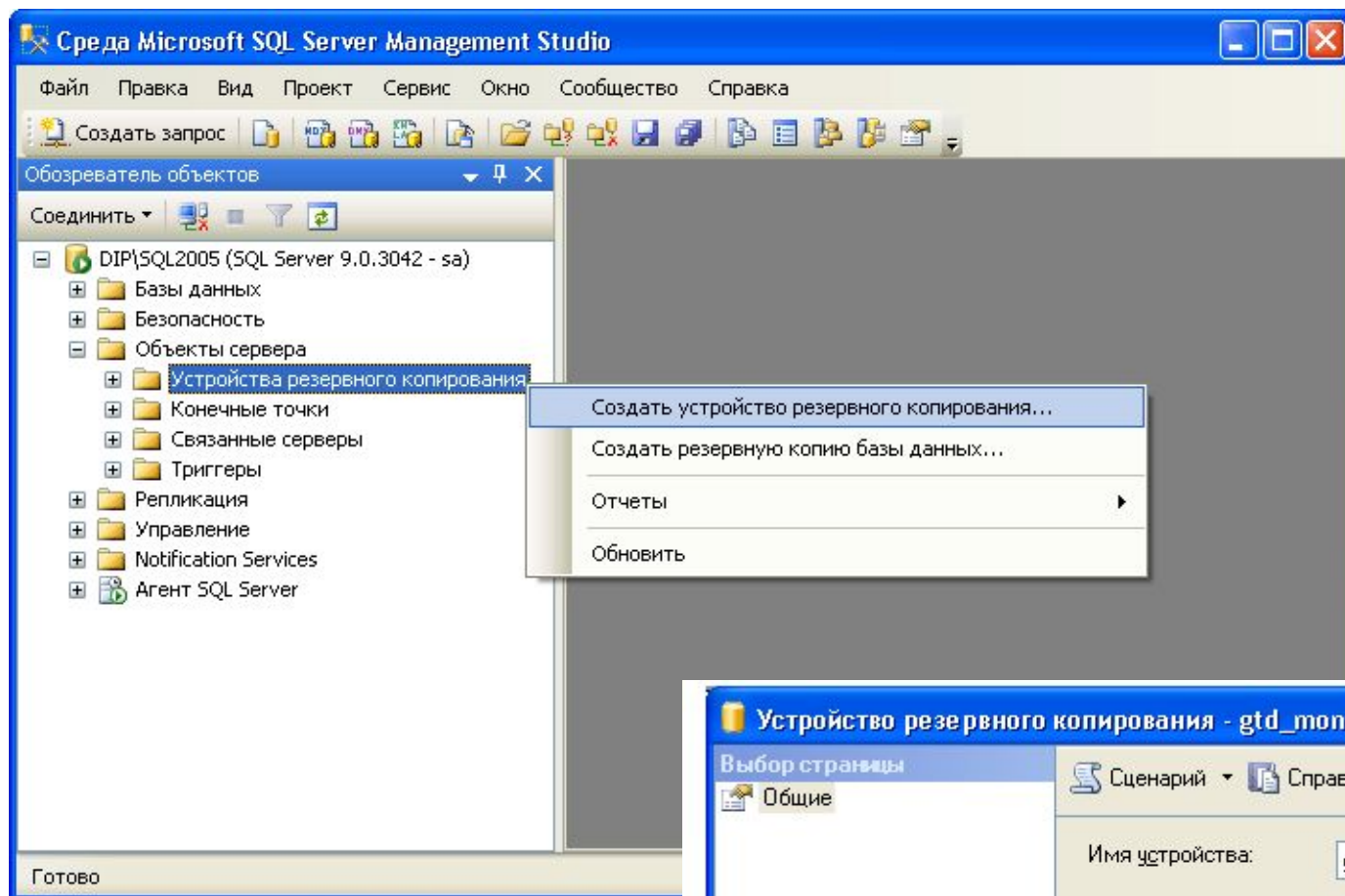
Проблемы, связанные с владельцем и разрешениями у физических файлов на устройстве резервного копирования, могут помешать операции резервного копирования. Служба SQL Server выполняет операции чтения и записи на устройстве. Учетная запись, под которой работает служба SQL Server, должна иметь разрешения на запись на устройстве резервного копирования. Однако процедура sp_addumpdevice, добавляющая запись для устройства резервного копирования в системные таблицы, не проверяет разрешения на доступ к файлу. Проблемы с физическим файлом устройства резервного копирования могут не проявиться до тех пор, пока эта резервная копия не будет применена или не будет выполнена попытка восстановления.

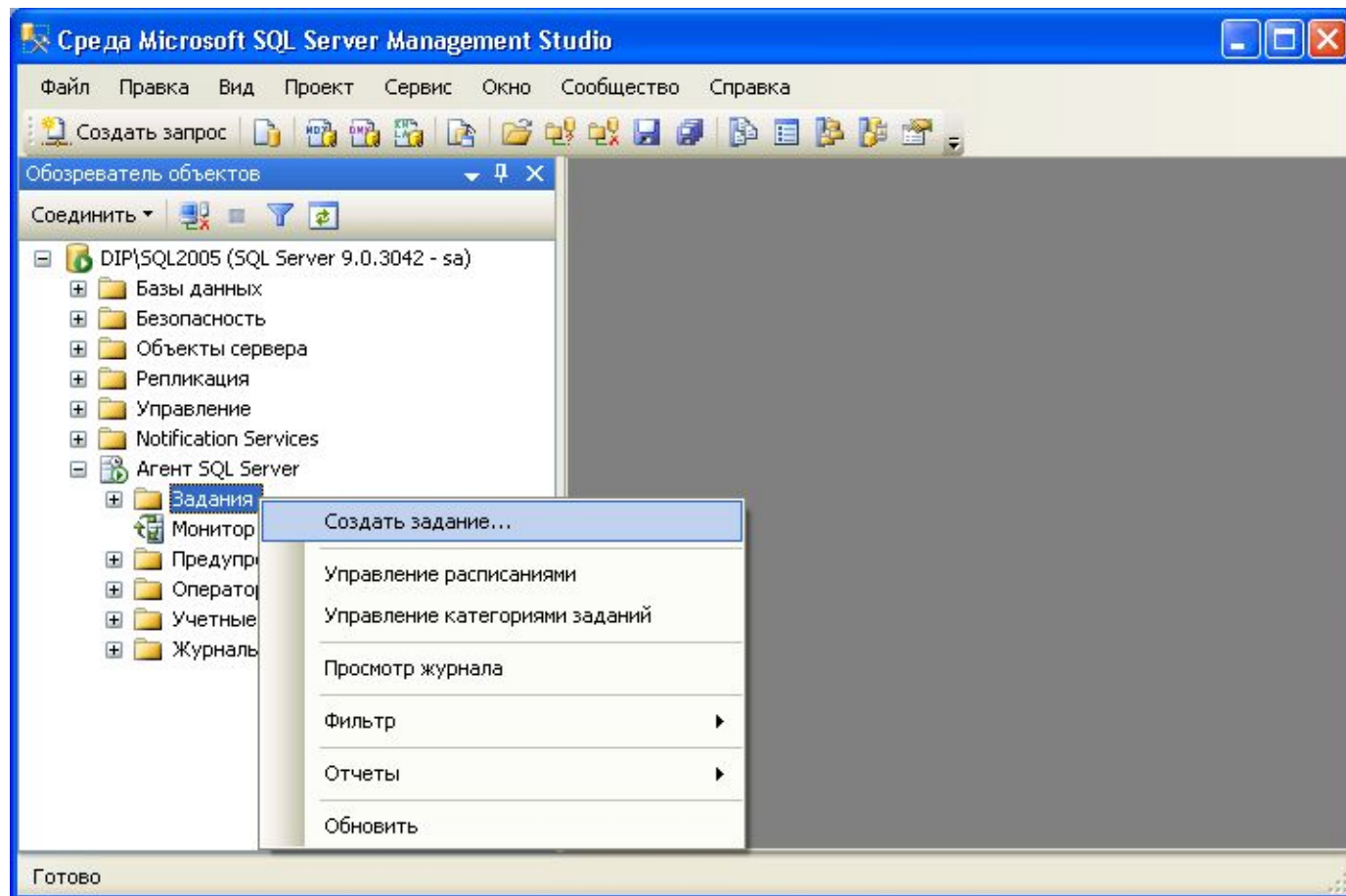
Агент SQL Server представляет собой систему планирования, обработчик заданий, средство общения и круглосуточного администрирования для Microsoft SQL Server.

Агент SQL Server выглядит как единая служба, но состоит из нескольких компонентов:

- планирование заданий;
- уведомления и предупреждения;
- операторы;
- журнал ошибок;
- серверы-посредники.

Задание: Настройка регулярного резервного копирования БД MS SQL Server





Создание шага задания

Выбор страницы

Сценарий Справка

Общие
Дополнительно

Имя шага: Backup

Тип: Сценарий Transact-SQL (T-SQL)

Выполнять как:

База данных: master

Команда: BACKUP DATABASE gtd TO gtd_monday WITH INIT, NOFORMAT, SKIP, NOUNLOAD

Открыть...
Выделить все
Копировать
Вставить
Анализ

Соединение

Сервер: DIP\SQL2005
Соединение: sa

Просмотреть свойства соединения

Выполнение

Готово

OK Отмена

Создание шага задания

Выбор страницы

Сценарий Справка

Общие
Дополнительно

Имя шага: Copy

Тип: Операционная система (CmdExec)

Выполнять как: Учетная запись службы "SQL Server, агент"

Код завершения процесса успешной команды: 0

Команда: XCOPY D:\MSSQL\BACKUP\gtd_monday.bak \\BACKUP_SERVER\Fc

Открыть...
Выделить все
Копировать
Вставить

Соединение

Сервер: DIP\SQL2005
Соединение: sa

Просмотреть свойства соединения

Выполнение

Готово

Подсказка: имена команд, содержащие пробелы, следует заключать в кавычки. Например: "имя команды" <аргументы>.

OK Отмена

Создание расписания задания

Имя:Monday

Задания в расписании

Тип расписания:Повторяющееся задание

☒ Включено

Однократное выполнение

Дата:29.10.2008

Время:22:45:17

Частота

Выполняется:Еженедельно

Повторяется каждые:1 нед.

☒ 1. Понедельник

☐ 3. Среда

☐ 5. Пятница

☐ 6. Суббота

☐ 2. Вторник

☐ 4. Четверг

☐ 7. Воскресенье

Сколько раз в день

☒ Однократное задание3:00:00

☐ Выполняется каждые1 ч

Начинается в:0:00:00

Завершается в:23:59:59

Продолжительность

Дата начала:29.10.2008

☐ Дата окончания:29.10.2008

☒ Без даты окончания

Сводка

Описание:Выполняется раз в неделю в понедельник в 3:00:00. Расписание будет использоваться с 29.10.2008.

OK

Отмена

Справка