Оглавление

[Введение: 1](#_Toc421802969)

[**О чем данная статья:** 1](#_Toc421802970)

[**Кому подойдет данная статья:** 2](#_Toc421802971)

[**Список связанных статей:** 2](#_Toc421802972)

[**Теория (todo: Изменить название)** 2](#_Toc421802973)

[1. Журнал транзакций 3](#_Toc421802974)

[2. Разностная копия БД 4](#_Toc421802975)

[3. Системные базы данных 5](#_Toc421802976)

[4. План бекапирования 6](#_Toc421802977)

[5. Общие рекомендации по резервному копированию 7](#_Toc421802978)

[**Используем приложение** 7](#_Toc421802979)

[1. Настройка уведомления администратора 7](#_Toc421802980)

[2. Дополнительные уведомления для администратора 9](#_Toc421802981)

[3. Решение проблем при настройке Database Mail 10](#_Toc421802982)

[4. Настраиваем резервное копирование с помощью приложения для SQL Standart: 11](#_Toc421802983)

[5. Настраиваем резервное копирование с помощью приложения для SQL Express: 13](#_Toc421802984)

[6. Удаление задач из БД 14](#_Toc421802985)

[Как восстанавливать бэкап (Руками, с помощью приложений) 15](#_Toc421802986)

Автоматизация настройки резервного копирования MS SQL

# Введение:

Всем привет. Я долго созревал, чтобы написать данную статью и выложить свое приложение. Надеюсь вам будет интересно.

# **О чем данная статья:**

В ней описан тот способ, как с помощью разработанного мною .NET приложения можно распространять план резервного копирования и проводить все необходимые настройки над БД средствами СУБД с уведомлением администратора о выполнении задач.

Данное приложение так же помогает администратору с восстановлением БД из полученных резервных копий.

По максимум постараюсь описать те нюансы, с которыми мне пришлось столкнуться в ходе разработки приложения и настройки БД.

Для описанных ниже задач можно использовать мастер планов обслуживания, но мне больше понравился такой подход. (пример можно посмотреть здесь <http://tavalik.ru/sozdanie-plana-obsluzhivaniya-v-microsoft-sql-server-200-r2/> ). Основное преимущество описанного мною метода, что данный можно применять практически ко все базам (кроме Express, там немного другой подход). План обслуживания можно переносить <http://mssqltrek.com/2011/08/27/how-to-transfercopy-maintenance-plans-from-one-server-to-other/>, но у вас должна быть соответствующая в версия MS SQL и все равно будет создан Job для запуска плана обслуживания.

Сравнить редакции MS SQL и их функционал вы можете вот здесь <https://www.microsoft.com/sqlserver/ru/ru/product-info/compare.aspx>.

**Осторожно**: перфекционисты могут испытать жжение и боль при просмотре кода моего приложения, т.к. оно было написано на скорую руку и заточено под решение конкретной задачи.

# **Кому подойдет данная статья:**

1. Тем, у кого MS SQL Express и нет возможности запускать с помощью Job задачи
2. Тем, кто в ближайшем будущем планирует перейти с MS SQL 2008 на более новую версию и не хочет настраивать зеркалирование БД, а сразу на новой версии настроить AlwaysOn <https://msdn.microsoft.com/ru-ru/library/hh510230.aspx>
3. Тем, у кого нет средств для поднятия еще резервных серверов и приходится обходиться тем, что есть.
4. У кого нет сжатых сроков на время восстановления БД, главное, чтобы она восстановилась и потеря данных была как можно меньше.
5. Кому лень что-то делать
6. Просто любопытным людям.

# **Список связанных статей:**

1. Создание и хранение резервных копий баз данных в MS SQL. Практические советы <http://habrahabr.ru/post/177085/>
2. Построение цепочки восстановлений баз данных MS SQL <http://habrahabr.ru/post/154907/>
3. Настройка Database Mail в MS SQL Server 2005 и старше <http://habrahabr.ru/post/132902/>
4. SQL Server 2008: бэкапим с умом. Часть 1: Теория <http://habrahabr.ru/post/131300/>
5. Всё что вы стеснялись спросить о бэкапах Microsoft SQL Server <http://habrahabr.ru/post/162497/>

Исходники на github для MS SQL Standart и для MS SQL Express <https://github.com/kinglamer/AutobackupPlan>

Если появится желающие добавить свои мысли в код, принимаю pull request. Готов выслушать конструктивную критику и доработать приложение.

# **Теория (todo: Изменить название)**

MS SQL Server поддерживает 3 модели резервного копирования.

1. Простую <https://technet.microsoft.com/ru-ru/library/ms191164(v=sql.105).aspx>
2. Модель полного восстановление <https://technet.microsoft.com/ru-ru/library/ms190217(v=sql.105).aspx>
3. Модель полного восстановления с неполным протоколированием <https://technet.microsoft.com/ru-ru/library/ms190692(v=sql.105).aspx>

Я выбрал для приложения модель полного восстановления, т.к. мне необходимо было иметь возможность всегда восстановить последнюю версию БД после любой операции и у меня не было одномоментных массовых операций по вставке данных. Если вы только начинаете и не знаете, как правильно выбрать, вам может помочь вот эта статья Microsoft <https://technet.microsoft.com/ru-ru/library/ms175987(v=sql.105).aspx> .

Для включения данного режима, необходимо выполнить следующий скрипт

ALTER DATABASE [Имя базы данных] SET RECOVERY FULL;

При переключении модели восстановления на полную у нас появится следующие возможности:

1. СУБД начнет вести записи журнал транзакций <https://msdn.microsoft.com/ru-ru/library/ms190925.aspx>
2. Создание разностной резервной копии <https://msdn.microsoft.com/ru-ru/library/ms175526.aspx>
3. Создание полной резервной копии <https://msdn.microsoft.com/ru-ru/library/ms186289.aspx>

Ниже будут описаны некоторые нюансы, связанные с резервным копирование журналов транзакций и разностный копий. По поводу полного копирования у меня замечаний никаких нет, просто делайте ее периодически и все у вас будет хорошо ☺

## Журнал транзакций

Журнал транзакций является критическим компонентом базы данных и в случае системного сбоя может потребоваться для приведения базы данных в согласованное состояние

Преимущества при восстановлении БД с помощью журнала транзакций

1. восстановление отдельных транзакций;
2. восстановление всех незавершенных транзакций при запуске SQL Server;
3. накат восстановленной базы данных, файла, файловой группы или страницы до момента сбоя и т.д

Рекомендации

1. Вынести на быстрый жесткий диск, чтобы при большом потоке операций не было задержек при записи.
2. Необходимо делать резервные копии журнала транзакций не реже чем каждый час.
3. После создания полной (разностной) копии базы данных, все старые журналы можно удалять, т.к. они теряют свою актуальность.
4. Внимательно следите за размером диска на котором хранятся журналы транзакций, если оно закончится, то записать новые данные в БД будет невозможно, пока не произойдет уменьшение размеров журнала транзакций или не добавиться новый дополнительный файл транзакций.
5. Журнал транзакций необходимо регулярно усекать, чтобы избежать его переполнения.
6. Возможна ситуация, когда невозможно сразу сделать усечение журнала. Они описаны в данной статье <https://msdn.microsoft.com/ru-ru/library/ms190925.aspx>
7. Для получения информации о состоянии базы данных можно с помощью следующего запроса: select name,log\_reuse\_wait, log\_reuse\_wait\_desc from sys.databases
8. При необходимости можно получить информацию о последних открытых транзакциях DBCC OPENTRAN (Имя базы данных) WITH TABLERESULTS

Пример SQL скрипта для выполнения резервного копирования журнала транзакции с последующим усечением файла.

BACKUP LOG [Имя базы данных] TO DISK = N'C:\Program Files\Microsoft SQL Server\MSSQL10\_50.SQL\MSSQL\Backup\[Имя файла].bak' WITH NOFORMAT, NOINIT,

NAME = N'Журнал транзакций Резервное копирование', SKIP, NOREWIND, NOUNLOAD, STATS = 10

GO

USE [Имя базы данных]

GO

DBCC SHRINKFILE (N'Имя файла лога БД' , 25)

GO

Эти же операции можно проделать с помощью SSMS <https://msdn.microsoft.com/ru-ru/library/ms179478.aspx>

## Разностная копия БД

Разностное резервное копирование основано на самой последней предыдущей полной резервной копии данных. В разностной резервной копии сохраняются только те изменения, которые были произведены с момента создания последней полной резервной копии.

Рекомендации:

1. Используйте разностные копии БД, если создание полной копии БД занимает большой промежуток времени
2. Периодически делайте полную копию БД, чтобы уменьшить объемы создаваемых разностных копий.
3. После создания полной копии БД, все предыдущие разностные копии теряют свою актуальность.

Более подробно о рекомендациях по частоте созданию разностных резервных копий, можно прочитать здесь <https://technet.microsoft.com/ru-ru/library/ms181092(v=sql.105).aspx> .

Приведу небольшой пример из практики, почему мы стали использовать разностную копию. Со временем у нашего клиента разрослась база данных до таких размеров, что создание полной резервной копии занимало 8 часов, еще несколько месяцев и возможно к началу рабочего дня не успевало бы завершиться данная операция. После перевода на разностное резервное копирование, мы сократили время с 8 часов до 2-4 минут (в зависимости от дня недели). Раз в неделю мы делали полную копию БД.

Пример SQL для создания резервной разностной копии БД с проверкой копии по завершению (отличается от полного копирования флагом DIFFERENTIAL вместо него нужно использовать NOFORMAT)

declare @pathBackup as varchar(55)

set @pathBackup = N'C:\Backup\[Имя файла БД]\_' + REPLACE(convert(varchar,GETDATE(), 104),'.','\_') + '.bak'

BACKUP DATABASE [Имя базы данных] TO DISK = @pathBackup

WITH DIFFERENTIAL, NOFORMAT, INIT, NAME = N'Полная База данных Резервное копирование', SKIP, NOREWIND, NOUNLOAD, STATS = 10, CHECKSUM

GO

declare @backupSetId as int

declare @pathBackup as varchar(55)

set @pathBackup = N'C:\Backup\[Имя файла БД]\_' + REPLACE(convert(varchar,GETDATE(), 104),'.','\_') + '.bak'

select @backupSetId = position from msdb..backupset where database\_name=N'[Имя базы данных]'

and backup\_set\_id=(select max(backup\_set\_id) from msdb..backupset where database\_name=N'[Имя базы данных]')

if @backupSetId is null

begin

raiserror(N'Ошибка верификации. Сведения о резервном копировании для базы данных "[Имя базы данных]" не найдены.', 16, 1)

end

RESTORE VERIFYONLY FROM DISK = @pathBackup

WITH FILE = @backupSetId, NOUNLOAD, NOREWIND

GO

## Системные базы данных

Помимо основной базы и связанных с ней файлов, я настоятельно рекомендую делать копии и системных баз данных. Начнем с того, что рассмотрим какие базы существуют в MS SQL. Их всего 5:

|  |  |
| --- | --- |
| **Название** | **Описание** |
| База данных master | В этой базе данных хранятся все данные системного уровня для экземпляра SQL Server. |
| База данных msdb | Используется агентом SQL Server для планирования предупреждений и задач. |
| База данных model | Используется в качестве шаблона для всех баз данных, создаваемых в экземпляре SQL Server. Изменение размера, параметров сортировки, модели восстановления и других параметров базы данных model приводит к изменению соответствующих параметров всех баз данных, создаваемых после изменения. |
| База данных resource | База данных только для чтения. Содержит системные объекты, которые входят в состав SQL Server. Системные объекты физически хранятся в базе данных Resource, но логически отображаются в схеме sys любой базы данных. |
| База данных tempdb | Рабочее пространство для временных объектов или взаимодействия результирующих наборов. |

Более подробно можете прочитать о них тут <https://msdn.microsoft.com/ru-ru/library/ms178028.aspx> и еще вот тут <http://tavalik.ru/sistemnye-bd-ms-sql-server-2012-2008/> .

Я выбрал резервировать только 2 системные БД:

1. msdb – потому что, там хранятся настроенные задачи и другие
2. master – хранятся все произведенные настройки SQL Server.

Данная информация все равно не сильно критична и ее можно восстановить руками, но зачем тратить лишнее время, когда можно просто взять из резервной копии.

## План бекапирования

На основе выше описанного составим наш план резервного копирования данных. Он может отличаться от того, что потребуется вам, все зависит от требований к восстановлению БД. Когда я подготавливал план, мне пришлось учесть, что необходимо восстановить данные максимально и потеря данных составляла не больше одного часа.

Мы будем делать следующие резервные копии:

* Полная копия основной БД, чаще чем раз в неделю нет необходимости
* Разностная копия основной БД, каждый день
* Копии журнала транзакций основной БД, каждый час
* Копия системной БД master, раз в неделю
* Копия системной БД msdb, раз в неделю

В итоге у нас получился следующий план резервного копирования данных:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **День недели** | **Время** | **Действия** | **Частота** | **Описание** |
| Понедельник - Пятница | С 8-00 до 21-00 | Резервные копии  Журнала транзакций | Каждый час | После выполнения резервной копии БД идет сжатие и усечение журнала транзакций |
| Суббота - Воскресенье | С 8-00 до 18-00 |
| Понедельник – Воскресенье | 22-00 | Разностная копия основной БД | 1 раз в день | После успешного выполнения разностной копии удаляются все старые копии журнала транзакций |
| Суббота | 12-00 | Проверка БД | 1 раз в день | Проверка БД Дело на целостность. |
| Суббота | 18-00 | Создание полной копии БД | 1 раз в день | По завершению данной операции идет уведомление на почту.  Если создание резервной копии прошло удачно, удаляется   * старая полная резервная копия * все старые разностные копии * все старые журналы транзакций |
| Понедельник – Воскресенье | 23-30 | Создание копии системной базы master | 1 раз в день | Хранится всегда только последний экземпляр БД |
| Воскресенье | 12-30 | Создание копии системной базы msdb | 1 раз в месяц | Хранится всегда только последний экземпляр БД |

## Общие рекомендации по резервному копированию

1. Используйте опцию BACKUP WITH CHECKSUM, чтобы убедиться, что все прошло хорошо. Недостатком такого решения является то, что для больших баз данных проверка контрольной суммы может серьезно загрузить систему.
2. Не выполняйте резервное копирование файлов на тот же физический диск, на котором хранится база данных или протокол транзакций.
3. Если вы используете MS SQL 2008 или выше, рекомендую вам использовать сжатие резервных копий средствами SQL. Следующий код включит сжатие по умолчанию: USE master; GO EXEC sp\_configure ‘backup compression default’, '1'; RECONFIGURE WITH OVERRIDE;
4. держите резервные копии по нескольку дней на случай, если одна из них будет повреждена – старая резервная копия лучше, чем никакой.
5. Используйте DBCC CHECKDB для проверки каждой базы данных перед копированием, это своевременно предупредит вас о надвигающихся проблемах. DBCC CHECKDB ('Имя базы данных') WITH NO\_INFOMSGS, ALL\_ERRORMSGS;

Примечание: на практики мы использовали данную проверку, только перед выполнением полной резервной копии.

1. Выполняйте периодически обновление статистики и реорганизации индексов БД

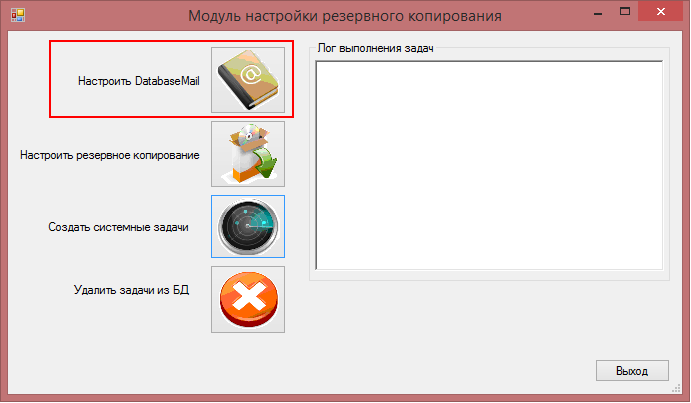
# **Используем приложение**

## Настройка уведомления администратора

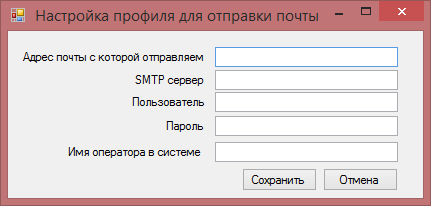
Мне было лень каждый раз заходить на сервер и проверять, сработал ли задача или произошла какая-то ошибка. Да и хотелось иметь возможность получать другие уведомления, не только о выполнения задач.

Для данной цели используется Database Mail MS SQL <https://technet.microsoft.com/ru-ru/library/ms175887(v=sql.105).aspx> (для версии Standart и выше)

В своем приложение я сделал специальный раздел для автоматизации данной задачи



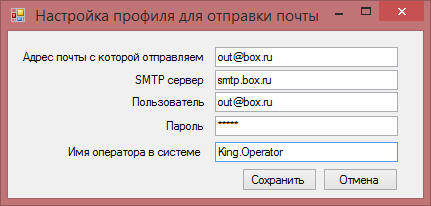
При нажатии появится форма для заполнения информации необходимой для создания профиля рассылки писем:



Приложение автоматически настроено на стандартный 25 SMTP порт для адреса с которого отправляются письма. При необходимости его можно изменить в процедуре sysmail\_add\_account\_sp <https://msdn.microsoft.com/en-us/library/ms182804.aspx> (она находится в файле AutoCreateBackupPlan.Standart.DatabaseMail.DatabaseMailQuerys.

Пользователь и пароль требуются на случай, если у почтового сервиса настроена аутентификация.

Имя оператора в системе указывается для того, чтобы у нас нормально создался профиль в DatabaseMail. Пишите любое название, которое будет для вас понятным. Ниже приведен пример заполненной формы.



Дальше с данного ящика от указанного оператора, к нам будут приходить уведомления об успешном выполнении операций.

Выполняемые действия на данном этапе:

1. Меняются системные параметры MS SQL.
2. Создается Database Mail Profile <https://technet.microsoft.com/en-us/library/ms189879(v=sql.105).aspx>
3. Активируется в SQL Agente профиль
4. Создается Database Mail Account <https://technet.microsoft.com/en-us/library/ms188668(v=sql.105).aspx>
5. Добавляется Database Mail Account к Database Mail Profile
6. Создается Database Mail Operator

Более подробно описано в следующей статье <http://habrahabr.ru/post/179819/> и частично я брал отсюда <http://sqlblog.com/blogs/jonathan_kehayias/archive/2010/01/25/setting-up-database-mail-to-use-gmail-account-for-presentations.aspx>. Естественно данные действия можно выполнить с помощью SSMS <http://blogs.msdn.com/b/suhde/archive/2009/07/12/how-to-configure-sql-server-database-mail-to-send-email-using-your-windows-live-mail-account-or-your-gmail-account.aspx>

## Дополнительные уведомления для администратора

В программе предусмотрены 2 задачи применяемые к БД:

1. проверка целостности БД. Для проверки базы данных использовалась стандартная процедура DBCC CHECKDB <https://msdn.microsoft.com/ru-ru/library/ms176064.aspx> .
2. информирование о свободном месте в файловых группах.

Вторая задача была реализована с помощью запроса к системной таблице dbo.sysfiles <https://msdn.microsoft.com/en-us/library/ms178009.aspx>

Вот пример данного запроса, который выполнялся к базе:

Select

NAME = left(a.NAME,15),

a.FILEID,

[FILE\_SIZE\_MB] = convert(decimal(12,2),round(a.size/128.000,2)),

[SPACE\_USED\_MB] = convert(decimal(12,2),round(fileproperty(a.name,'SpaceUsed')/128.000,2)),

[FREE\_SPACE\_MB] = convert(decimal(12,2),round((a.size-fileproperty(a.name,'SpaceUsed'))/128.000,2)) ,

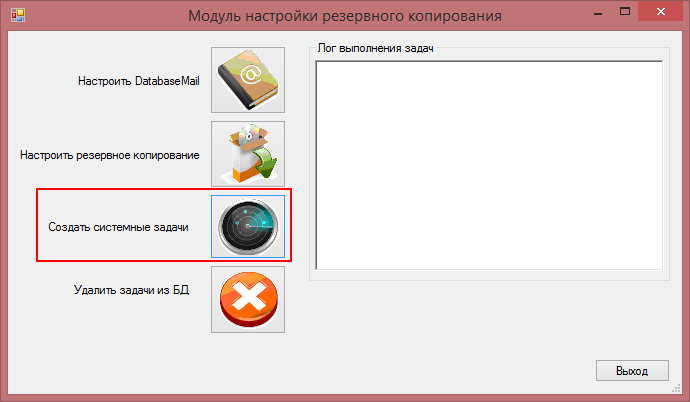
FILENAME = a.FILENAME

From dbo.sysfiles a

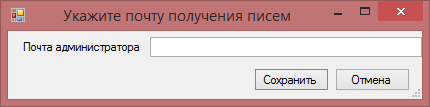
Ответ с сервера приходит на почту администратора в виде html разметки. Данный синтаксис возможен благодаря следующей стандартной функции MS SQL FOR XML <https://msdn.microsoft.com/ru-ru/library/ms190922.aspx>

Так же пока я искал, как преобразовать в возвращаемый результат выполнения запросов в html текст, наткнулся на следующую страницу, где человек создал целую процедуру для этих целей <https://www.virtualobjectives.com.au/sqlserver/saving_to_html.htm>

Настроить эти операции можно с помощью соответствующего пункта в меню программы:



Появиться окно для указания почтового ящика, на который необходимо высылать html текст отчета



Код можете найти в папке AutoCreateBackupPlan.Standart.DatabaseTasks.SystemTask файлы TaskCheckDB и TaskFileStatistic.

## Решение проблем при настройке Database Mail

В MS SQL 2008 я столкнулся с проблемой при настройке SQL Server Agent. Симптомы следующие, после настройки невозможно запустить SQL Agent. В основном это решается с помощью установки update на SQL сервер.

Убедитесь, что установлен sp1 <http://www.microsoft.com/ru-ru/download/details.aspx?id=26727>, а потом можно уже ставить обновление <http://support.microsoft.com/kb/2926028>

Если данные обновления не помогают, необходимо скачать fix

Его можно найти на данном сайте <http://support.microsoft.com/select/default.aspx?target=assistance> конечную ссылку не могу указать сейчас, для того что бы дойти до пакета фикса, нужно будет ответить на ряд вопросов.

Если есть проблемы с модулем DatabaseMail. После настройки данного модуля с помощью приложения, необходимо зайти в SQL Agent и просмотреть журнал событий. Если там будут ошибки «невозможно подключиться к почтовому ящику». Значит есть проблема, даже если через проверку отправляется письмо.

Исправляется это следующими манипуляциями:

1. Management Studio - SQL Server Agent - Properties.

2. Alert System

3. Уберите галочку с Enable mail profile

4. Нажмите OК

5. Зайдите снова и поставьте галочку

6. Перезагрузите SQL Server Agent.

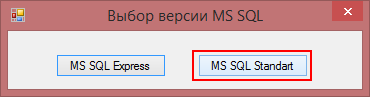
А потом

Проверьте учетную запись для SQL Agent service и поменяйте ее, если это доменная учетная запись на системную или наоборот.

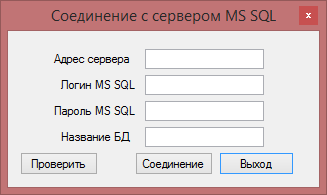
Все должно заработать.

## Настраиваем резервное копирование с помощью приложения для SQL Standart:

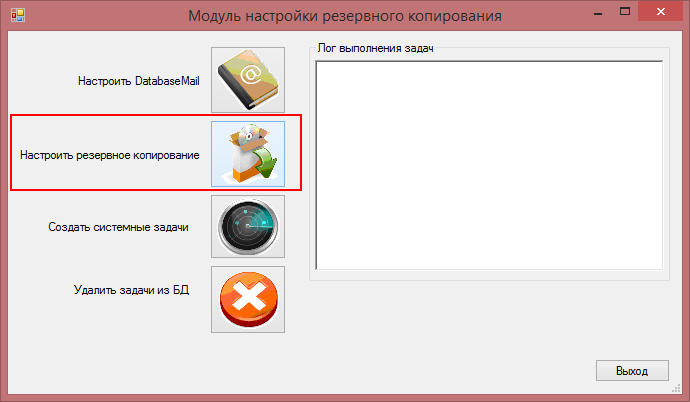
Выбираем версию Standart. Настраиваем уведомления. (см. раздел, настройки уведомления)



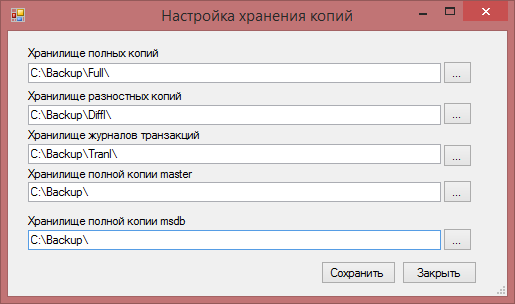
Соединяемся с БД, заполняя данные для соединения и указываем БД, для которой будет применяться Job



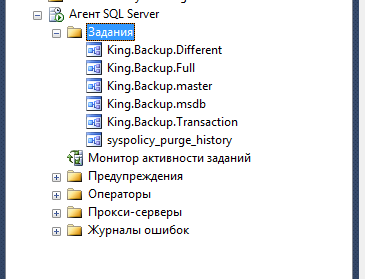
Выбираем настройку резервного копирования



Указываем пути для сохранения копий БД. Если указанные папки не существует, то программа попытается их создать (нужны соответствующие права).



Нажимаем сохранить и базе настраиваются соответствующие задачи. Желательно настроить для каждого бэкапа разные папки, т.к. при удаление будут удаляться все файлы с расширением bak. (см. раздел удаление копий БД)

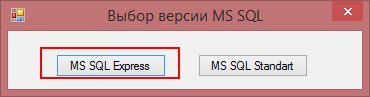


## Настраиваем резервное копирование с помощью приложения для SQL Express:

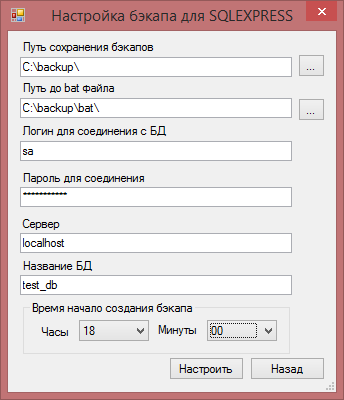
Принцип работы приложения.

Так как в SQL Express отсутствует SQL Agent, задачу по автоматизации резервного копирования пришлось решить другим путем. В указанной пользователем папке создается bat файле в котором описан SQL запрос, отвечающий за создание резервной копии. В случае необходимости можно редактировать его напрямую. По мимо этого должен работать стандартный планировщик Windows, в нем создается задача, которая будет запускать раз в сутки в указанное время.

Для этого запускаем приложение. Выбираем пункт MS SQL Express



Появляется форма для заполнения параметров

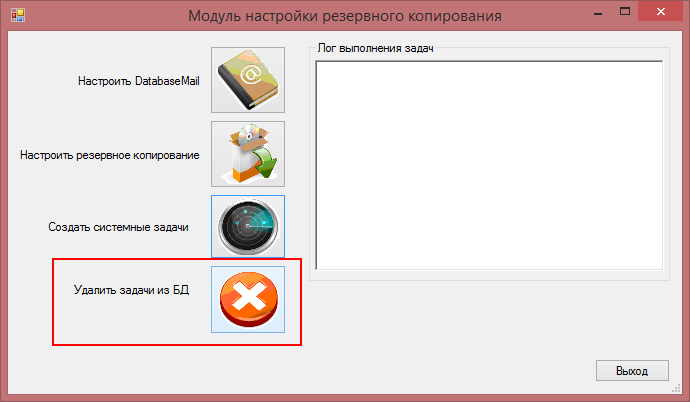


Указываем где будет сохраняться наша копия, а также где будет лежать bat файл для создания копии базы (имя файла указывать не надо, оно будет задано автоматически). Далее указываем настройки соединение и время когда необходимо запускать задачу.

Единственный минус данного подхода в том, что приходится храниться в открытом виде пароль для соединения с БД.

## Удаление задач из БД.

Если необходимо удалить все задачи из БД (например, захотели изменить пути сохранения БД). Для этого используем соответствующий пункт в меню программы. Из SQL Agent будут удалены все задачи с определенным начальным префиксом (в моем случае King)



## Удаление копий БД

В некоторых задачах, настроено удаление старых копий БД. Для этого я использую процедуру master.dbo.xp\_delete\_file

Пример использования: Удалит все файлы с расширением bak из указанной папки, дата создания которых превышает 14 дней.

EXECUTE master.dbo.xp\_delete\_file 0,"E:\backups",N'bak',dateadd(d,-14,getdate()),0;

И вот еще один более подробный пример <http://www.patrickkeisler.com/2012/11/how-to-use-xpdeletefile-to-purge-old.html> и информация о том, какие параметры принимает данная функция <http://stackoverflow.com/questions/24582996/sql-server-xp-delete-file-parameters>

# Как восстанавливать резервные копии

# 

С помощью SQL скрипта

Для восстановления базы данных используется команда RESTORE <https://msdn.microsoft.com/ru-ru/library/ms186858.aspx>

Если необходимо восстановить просто базу из полной копии, то достаточно выполнить следующий скрипт:

RESTORE DATABASE [Имя базы данных] FROM DISK = 'Z:\SQLServerBackups\back.bak' WITH REPLACE

В случае, если необходимо восстановить последовательно сначала полную копию, разностные копии и журналы транзакций, тогда необходимо написать следующий SQL скрипт

RESTORE DATABASE TEST\_DB –восстанавливаем полную копию

FROM test\_db\_full

WITH NORECOVERY;

GO

RESTORE DATABASE TEST\_DB –восстанавливаем разностную копию

FROM test\_db\_diff

WITH FILE = 1,

NORECOVERY;

GO

RESTORE LOG TEST\_DB –восстанавливаем журнал транзакций №1

FROM test\_db\_tran\_1

WITH FILE = 1,

WITH NORECOVERY;

GO

RESTORE LOG TEST\_DB –восстанавливаем журнал транзакций №2

FROM test\_db\_tran\_2

WITH FILE = 1,

WITH NORECOVERY;

GO

RESTORE DATABASE TEST\_DB

WITH RECOVERY;

GO

Для восстановления БД можно использовать так же и SSMS <https://support.managed.com/kb/a1788/how-to-manually-restore-an-mssql-database-management-studio.aspx>