



IP DIRECTOR APPLICATION NOTE

SQL Server maintenance

09-Apr-2013

Corporate

Headquarters
+32 4 361 7000

North & Latin America

Headquarters
+1 947 575 7811

Asia & Pacific

Headquarters
+852 2914 2501

Other regional offices

Available at
www.evs.com/contact

TABLE OF CONTENTS

TABLE OF CONTENTS	2
INTRODUCTION	3
SQL SERVER MAINTENANCE.....	3
DATABASE BACKUP PLAN	3
INDEX OPTIMISATION.....	3
FULLTEXT CATALOGS REORGANIZE	4
SYSTEM DATABASES BACKUP PLAN.....	4
PROCEDURE.....	4
EVS SQL DATABASE TOOL (VERSION 5.9.2).....	5
[CREATE MAINTENANCE JOBS]	6
[BACKUP A DATABASE].....	9
[RESTORE A DATABASE FROM HISTORY].....	10
[RESTORE A DATABASE FROM FILE]	12
[MANUALLY REBUILD DATABASE INDEXES]	13
[MANUALLY REBUILD FULLTEXT CATALOGS].....	14
[CREATE ROBOCOPY JOB]	15
EVS SQL LOGS (EVS-DB-LOGS.HTA)	16
[GET SQL CONFIGURATION].....	16
[GET BLOCKING CONNECTIONS LIST]	16
[GET LOCK LIST].....	17
[GET INDEX FRAGMENTATION REPORT].....	17
[GET MIRRORING STATISTICS]	17
[GET FULLTEXT STATISTICS]	18
[DUMP ALL THE INFORMATION TO A LOG FILE]	19
SQL SERVER CONFIGURATION.....	20
MEMORY.....	20
SERVICE PACKS.....	20
WINDOWS CONFIGURATION	20
EVENTLOG	20

INTRODUCTION

This application note describes the entire SQL server configuration used with the IP Director database. You can also find some troubleshooting tools described at the end of the document

IP Director 6.15 is compatible with SQL2005 and SQL2008 R2

SQL SERVER MAINTENANCE DATABASE BACKUP PLAN

It's important to regularly backup the media database to ensure data recovery.

Several backup plans could be implemented depending on the infrastructure used by IP Director.

A daily full backup is considered as the minimal backup strategy but the following procedure ensures an optimal recovery plan:

- > Full Backup every hour
- > Transaction Log Backup every 15 minutes

On the standard platform, a full backup takes less than 10 seconds (DB size:450 Mb => 2Sec/100Mb). Even if this backup operation uses a lot of disk resources, the backup window is so short that this kind of backup could be done every hour without any substantial performance loss.

To shorten the data loss window, a Transaction Log backup is launched every 15 minutes. This complementary backup ensures that we can recover all the data changes up to 15 minutes before the DB corruption.

All these backup files should be stored on a local disk with enough free space (typically on E:\DB_BACKUPS). The backup files retention period is configured to 4h. That means that 4 full backup files and 4h of transactions will be stored in the backup folder (it could be a lot of data. i.e. with a 200 Mb DB and 500 Mb Transaction Log : $(4*200)+500=1300$ Mb).

The backup files can also be automatically copied to a network share to ensure that they are available if the SQL Server totally crashes. This file copy can be done with Robocopy and is included in the SQL toolkit.

INDEX OPTIMISATION

To keep the database performances at the highest level day after day, it's important to periodically rebuild the indexes defined on each table. This procedure can be done automatically with a SQL job and launched by SQLSERVERAGENT every night.

Index rebuild is a performance consuming process and should be avoided during production hours.

FULLTEXT CATALOGS REORGANIZE

The IP Director database intensively uses the SQL fulltext catalogs to perform searches. In order to keep them optimized a night job will reorganize them every day at 04:10

SYSTEM DATABASES BACKUP PLAN

In order to complete the backup strategy, MASTER and MSDB must also be saved through a backup procedure.

MASTER and MSDB are usually small, their backup-up should be performed quickly.

A job will backup these two system databases every night at 03:00.

PROCEDURE

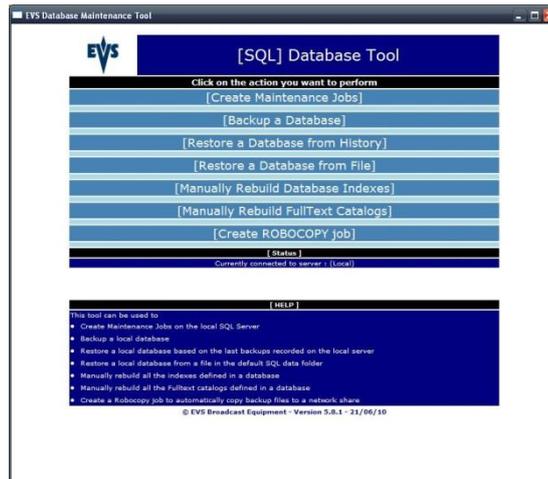
All these maintenance actions can be created with the EVS_DB_Tool.hta file contained in the Remote Installer plugins or in the SQL_Toolkit.

EVS SQL DATABASE TOOL (VERSION 5.9.2)

The EVS_DB_Tool is an HTA file (HTML Application).

It has been developed to easily perform administrative tasks on a SQL server .

These tasks are displayed in the following screenshot :



[CREATE MAINTENANCE JOBS]

This part of the tool creates the maintenance jobs explained in the first part of the document.

Jobs created are :

1. A DB Full backup every hour
2. A DB Transaction Log backup every 15 Minutes
3. A DB Index rebuild every day at 02:00
4. Fulltext catalogs reorganize every day at 04:10
5. The system database full backup every day at 03:00
6. A msdb database clean-up every day at 03:10
7. Errorlog cycle every week on sunday

[BACKUP A DATABASE]

Creates a manual full or transaction log backup from the specified database

[RESTORE A DATABASE FROM HISTORY]

Restores a database from the lasts backups. The backup file list is obtained from the local SQL Server. It can be used to easily restore a previous version of the local database but not to restore a database coming from another server.

[RESTORE A DATABASE FROM FILE]

Restores a database contained in a backup file.

The database will be restored with its original name in the local server default directory.

This part of the tool is specially designed to restore databases coming from another server especially with different database folder path.

[MANUALLY REBUILD DATABASE INDEXES]

Rebuild all the indexes defined in the specified database

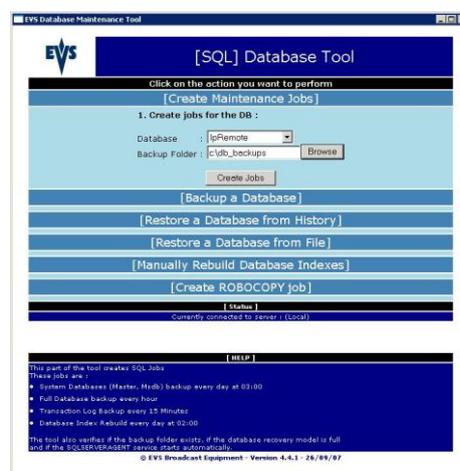
[MANUALLY REBUILD FULLTEXT CATALOGS]

Rebuild all FullText indexes defined in the specified database

[CREATE ROBOCOPY JOB]

Creates a jobs that replicates a folder to a network share

[CREATE MAINTENANCE JOBS]



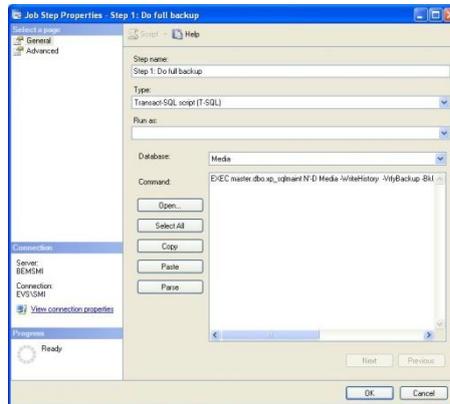
- > Click on [Create Maintenance Jobs] to display the tool options
- > Choose the database to “maintain” in the local database list
- > Choose the folder where the backups will be stored (default : C:\DB_Backups or E:\DB_Backups on DB Servers). The folder will be created if it doesn't exist.
- > Click on “Create Jobs” to create the maintenance jobs for the chosen database.

The tool creates

1. A Full backup job scheduled every hour
2. A DB Transaction Log backup job scheduled every 15 Minutes
3. A DB Index rebuild job scheduled every day at 02:00
4. A Fulltext Catalogs reorganize job scheduled every day at 04:00
5. The msdb and master full backup job scheduled every day at 03:00
6. A job that cleans the msdb database and keep a 15 days backup history
7. A ERROLOG cycle every week on sunday

The jobs are first deleted if they already exist.

If the tool detects that the service is configured to be launch with the LocalSystem Account, you will see the following information :



You can carefully modify the command line and change the 4HOURS parameter to the retention time you want (12HOURS,24HOURS,2DAYS,...)

EXEC master.dbo.xp_sqlmaint N'-D IPRemote -WriteHistory -VrfyBackup -BkUpMedia DISK -BkUpDB C:\DB_BACKUPS -DelBkUps 4HOURS -BkExt BAK'

Click OK to save your changes.

SQLServerAgent is not installed with a SQL2005 Express Edition. The HTA detects which SQL version is used by IP-Director and creates windows jobs instead of SQL Jobs in case of Express edition.

The jobs created by the HTA are :

a full database backup of all databases every hour

a DB Reindex on all databases every day at 2:00

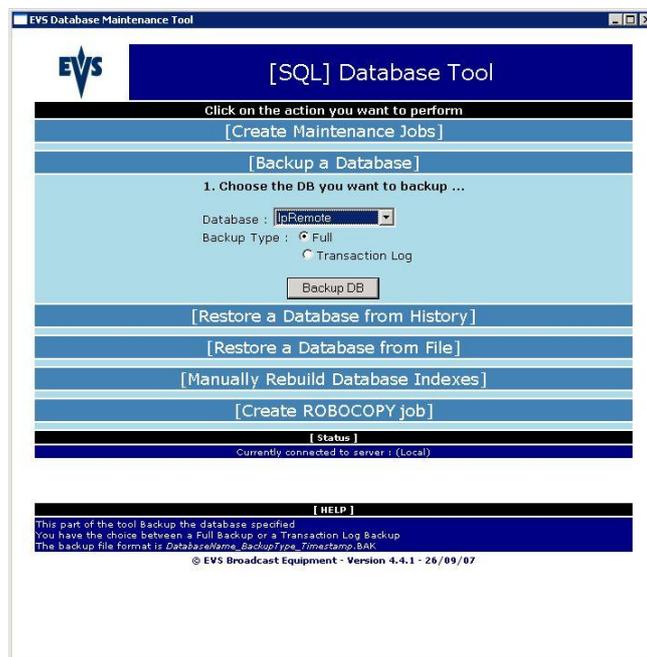
The jobs use the application ExpressMaint.exe that must exist in the c:\windows folder

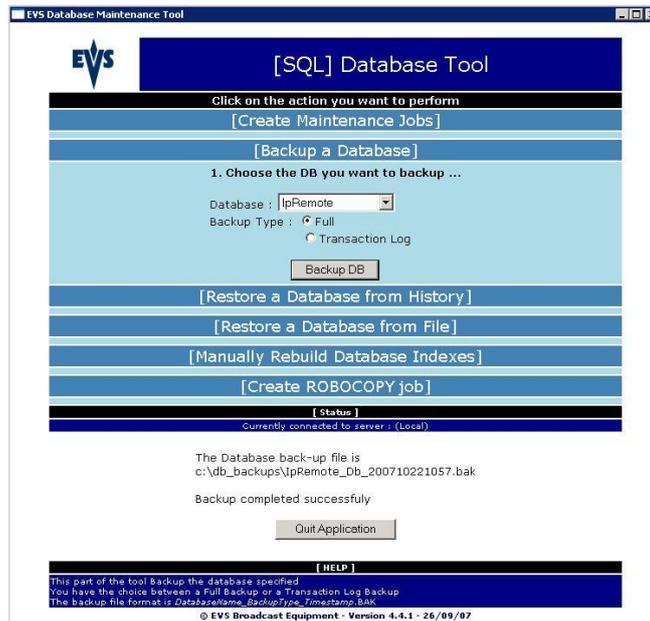
[BACKUP A DATABASE]

The EVS_DB_Tool can also be used to easily backup a database by choosing [Backup a Database]

- > Choose the Local Database name
- > Choose the database type (full or Transaction Log)
- > And Click on "Backup DB"

The backup file will be stored in the c:\DB_Backups folder and will be named *DatabaseName_BackupType_DateTime.BAK* (i.e. Media_db_201206291100.BAK)





[RESTORE A DATABASE FROM HISTORY]

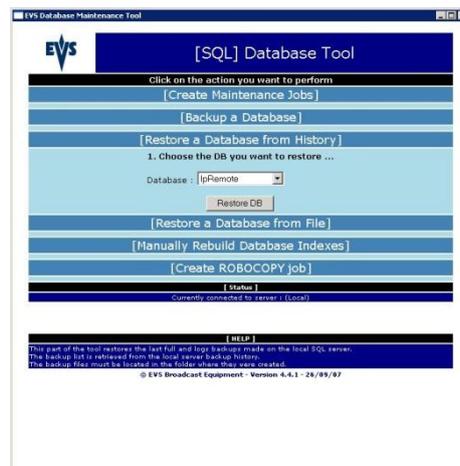
Two Restore types are available in this tool.

The first is based on the backup history stored on the local SQL server.

You have just to specify the local database to restore.

The tool will access the local SQL server and list the lasts backups performed on the database.

It then takes the last full database backup and the following transaction log backups and restores them on the specified database.



This first restore tool is designed to restore a corrupted database from the last backups performed on this database (locally)



Author Sebastien MAINDIAUX

10-avr.-13

File
Application_Note_External_SQL_Maintenance_v6.0.docx

name

Page 11 of 20

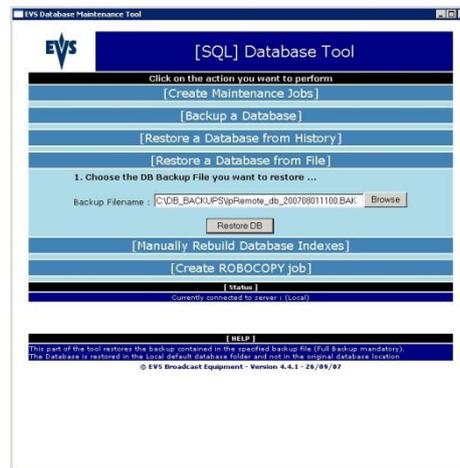
[RESTORE A DATABASE FROM FILE]

The second restore tool is based on a file restore and can be used to restore any database.

You have just to specified in which file the backup is stored (the entire file path)

The tool opens the backup file and reads the database name associated to the backup (only one backup must be stored in the file).

It restores this database in the local default database folder or on the local database location if it already exists (and not the originating file location stored in the backup file)



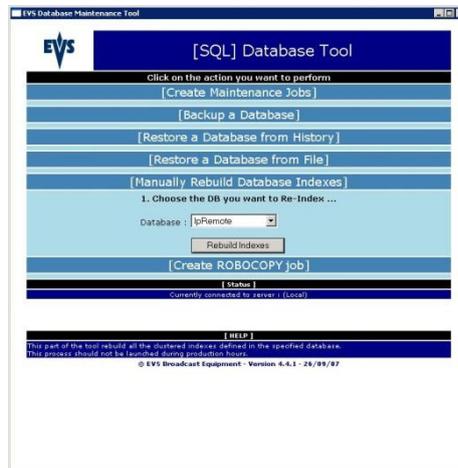
The tool is designed to restore a backup file whatever the database and the originate server configuration.

[MANUALLY REBUILD DATABASE INDEXES]

This part of the tool gives the opportunity to an operator to manually rebuild the database indexes. This operation is resource consuming and should be launched during low activity period otherwise it could cause locks and application slowdowns.

Index Rebuild can improve database performance by reorganising data in the database.

This functionality has been added to easily rebuild indexes when the corresponding maintenance job is disabled or not executed (regarding its schedule)



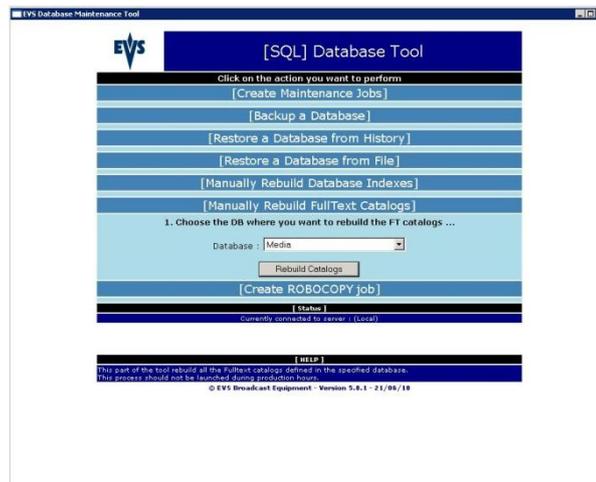
[MANUALLY REBUILD FULLTEXT CATALOGS]

IP Director 6 uses the SQL Fulltext Search Engine to perform 'Google' like searches on the database.

The Fulltext Search Engine stores indexed words in the Fulltext Catalogs. These catalogs can then be rebuilt by this HTA.

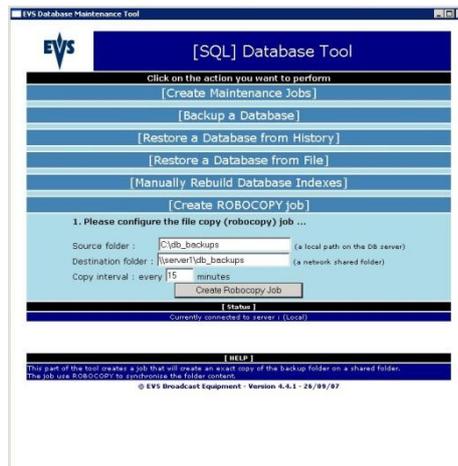
You typically need to rebuild the Fulltext Catalogs on an IP Director database if :

1. You don't find anything when you search for a word in the IP Director Database Explorer but you find something when you search for **word**
2. You don't find your new clips containing the word you are searching for but you find the older ones.



[CREATE ROBOCOPY JOB]

This part of the tool gives the opportunity to create a job that replicates files between a local drive and a network share



The Robocopy job can be used to replicate files between mirrored DB Servers.

In this case, the jobs should be created on both servers (main and backup) and have the following source and destination paths:

Source folder : \\Virtual_IP_Address\Share

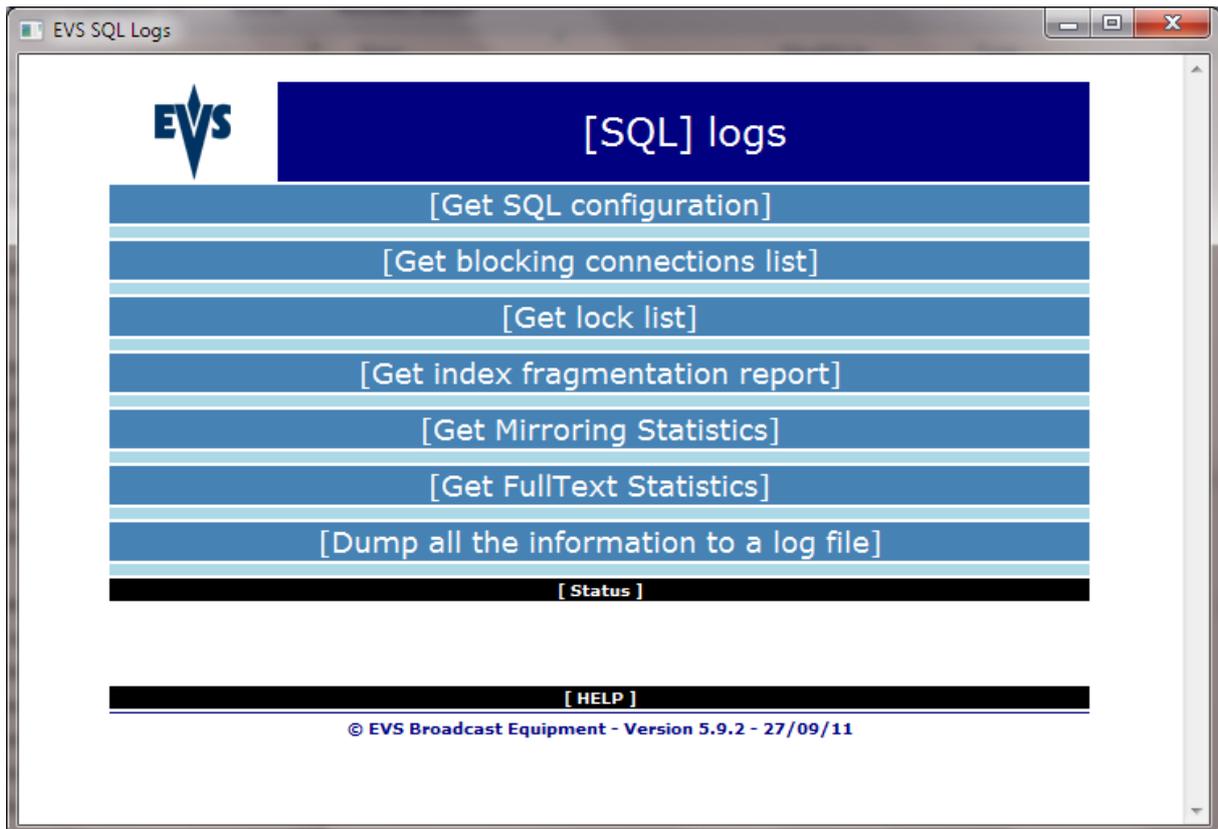
Destination folder : \\Local_IP_Address\Share

Remark : the "SQLServer" user must exist on the destination server or must be recognized by the destination folder

EVS SQL LOGS (EVS-DB-LOGS.HTA)

The “EVS SQL Logs” tool retrieves the major monitoring information related to an IP Director database.

You can either perform each operation one by one or either launch them all at once and store their result in a log file, with the last option.



[GET SQL CONFIGURATION]

This option retrieves the version of the SQL server used by IP Director

[GET BLOCKING CONNECTIONS LIST]

This option tries to list the connection that blocks other connections.

If you detect blocking connections, click several times on the [Get blocking connections list] menu to verify that the blocking connection disappears by itself.

A blocking connection is not necessarily an issue and is part of the normal behaviour on a loaded DB server.

However, if the same blocking connection stays for a long period of time (typically more than a minute), it could become a real issue for IP Director and it could explain misbehaviours in IP Director.

[GET LOCK LIST]

This option list the locks asked by the connections on the SQL objects.

This list can be really big on loaded servers.

[GET INDEX FRAGMENTATION REPORT]

The option performs a check on the database you have selected in the list to detect if indexes are fragmented.

[Get index fragmentation report]

Please choose the database you want to analyse ...

Database :

Get Fragmentation info

The result shows several statistics for all tables defined in the database.

[Status]												
DBCC SHOWCONTIG												
ObjectName	Rebuild	IndexName	IndexId	Rows	Pages	Extents	Extent Switches	Scan Density	Best Count	Actual Count	Logical Fragmentation	Extent Fragmentation
KeywordGridAccess		PK_KeyWordGridAccess	1	4	1	1	0	100	1	1	0	0
GroupVarID		PK_GroupVarID	1	0	0	0	0	100	0	0	0	0
LogsheetAssetMaterialID			0	18646	81	18	17	61.11	11	18	0	77.78
Log		PK_Log	1	14	1	1	0	100	1	1	0	0
TC2Field			0	180	5	5	4	20	1	5	0	60

Some of them will be highlighted in colour to reflect the size and the fragmentation of the table.

RemoteBSendToList		PK_RemoteBSendToList	1	0	0	0	0	100	0	0	0	0
TargetToDo			0	0	0	0	0	100	0	0	0	0
BinElem	rebuild	PK_ElemBin	1	44863	179	33	53	42.59	23	54	26.82	90.91
ConfigGP1			0	0	0	0	0	100	0	0	0	0

If a table is identified as too fragmented, a “rebuild” link will be added next to the table name.

Pressing this link will rebuild the indexes on the table.

[GET MIRRORING STATISTICS]

This option gives statistics for database in mirroring state.

[DUMP ALL THE INFORMATION TO A LOG FILE]

This option launches all tasks available in the tool and stores their result in a log file in c:\levslogs\.

This option can be useful to send monitoring result to EVS support team.

SQL SERVER CONFIGURATION MEMORY

The standard server used to host SQL Server is configured with 3 GB RAM.

SQL server must be configured to take a maximum of 2,5 GB Ram to prevent the service to disturb other processes.

This configuration can be changed manually through the SQL Server Properties (Memory tab) or automatically with the script "SQL_MaxMemory.sql" contained in the SQL toolkit

SERVICE PACKS

To ensure that SQL Server is up-to-date, it's recommended to apply the latest Service Pack available for SQL Server.

For SQL Server 2005, the last validated Service Pack is the SP3.

The file SQLServer2005SP3-KB955706-x86-ENU.exe EXE can be applied to SQL Server 2005 Standard Edition.

The file SQLEXP32_ADV.EXE integrating SP3 can be applied to SQL Server 2005 Express Edition.

SQL 2005 Express Edition Advanced is needed for IP Director 6

WINDOWS CONFIGURATION

EVENTLOG

SQL Server records a lot of messages in the application log. The log size should be configured to store all the application messages during at least one week.

To avoid any kind of message loss, it's important to configure the Application Log "Max Log size" property to more than 5 Mb without retention period (Overwrite Events as Needed)

A script can be applied to the servers to configure the maximum log size to 25 Mb. "Set_AppLog_Size.vbs" is available in the SQL toolkit