

Virtual Backup and Restore of an IQ Database using Amazon Web Services

Introduction

This tutorial provides users step by step instructions to create a pair of batch files capable of, first, creating a virtual backup of an IQ database and uploading it to Amazon Web Services (AWS) and, second, downloading the backed up files from AWS and restoring the original IQ database. The virtual backup method creates a checkpoint in the IQ database, effectively allowing the live database files to be uploaded to AWS without creating a physical backup file for the IQ database while still allowing database access to users.

Prerequisites

- For the AWS part of each of the following tutorials, we have used third party software called S3Sync which can be downloaded from: <https://sprightlysoft.com/S3Sync/>.
- All referenced .bat and .sql files appear in the appendix at the end of this document. It may be useful to save these files before beginning the tutorial
- Bolded variable assignment values (eg. SET bucket-name=<bucket-name>) will need to be changed to suit your setup. Variables are used throughout the batch files surrounded in ‘%’ signs. (eg. -BucketName %bucket-name%).
- This tutorial assumes the batch files will be run using the Windows cmd command prompt
- It should be noted that batch files may be opened for editing in any text editor, for example Notepad
- For the remainder of this tutorial, Amazon Simple Storage Service will be abbreviated to “S3”
For lack of confusion, the machine running the server for the database we are working with will be known as the “server machine”

Backing up a database and uploading the backup files to S3

We will be working with the virtualBackupAndUpload.bat batch file for the remainder of this section.

The first command of the batch file is as follows:

```
dbisql -c "UID=dba;PWD=sql;" -host %<ip-address>% -port %<port>% virtualBackup.sql
```

First, a call to Interactive SQL is made using the “dbisql” command. Next, options such as username and password for the desired database are entered using the “-c” option.

Note: if you are running this batch file on the server machine, the command above can be simplified to:

```
dbisql -c "UID=dba;PWD=sql;LINKS=SharedMemory;" virtualBackup.sql
```

Following are the IP address (using the “-host” option) of the server machine and the port number (using the -port option) for which the database server is running on (on the server machine).

Last, and most importantly, the file name “virtualBackup.sql” is being passed to Interactive SQL to be run. The virtualBackup.sql file can be opened for editing using Notepad or a similar text editor and its code is also provided below:

```
BACKUP DATABASE
FULL VIRTUAL ENCAPSULATED
'<path-to-batch-file>\uploadToS3.bat'
TO '<backup-path>\backup'
```

The first line is fairly obvious, being the SQL key words to back up a database. The next line indicates that the type of backup will be “FULL VIRTUAL ENCAPSULATED”. This means that a checkpoint will be made in the database allowing the backup to occur while the database is still live. Virtual backups, however, require the user to back up the checkpointed files using a shell command.

The third line executes a batch file “uploadToS3.bat” residing on the server machine for which the single line of code will be provided after explaining the last line in “virtualBackup.sql”.

It may be useful to add that, in addition to creating a checkpoint in the database, an actual backup file of the “.db” file will be created. This file will be created in the specified path in the final line: “TO ‘<backup-path>\backup’”. Note that the backup file will be created as “backup.1” in the folder having path “<backup-path>” on the server machine, i.e. not necessarily the machine that is running the batch file. You will need to open “virtualBackup.sql” in a text editor and select a backup location. It is also good to know that this path will be accessed later to upload the backup file to S3.

Note: A folder having the same name as the file to back up (for example, a folder named “backup” in the “C:\myfolder” directory) will cause an error during execution. As well, the path specified (for example, “C:\myfolder”) must exist in order for the command to execute successfully.

For more information regarding virtual backup statements, please consult [BACKUP DATABASE Statement](#).

As promised, the code for the shell command (‘<path-to-batch-file>\uploadToS3.bat’) is provided below:

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSSecretAccessKey
%aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-
folder-name% -SyncDirection Upload -LocalFolderPath %local-folder-path% -
IncludeOnlyLocalFilesRegularExpression ".*\.$|\.iqtmp" -CompareFilesBy Timestamp
-LicenseKey %license-key% -LogFilePath %log-file-path%
```

Before explaining this command, it is good to know that a batch file similar to this command will be generated for you when installing S3Sync and it may be useful to reference.

First, %path-to-S3sync-exe% is the path to S3Sync.exe (eg. "C:\Program Files (x86)\SprightlySoft\S3 Sync\S3Sync.exe"). In this case, we are using the path to the executable instead of just calling it (in contrast to calling “dbisql” in the previous command) because the installer for S3Sync

does not add its directory location to the Path environment variable. The example path to “S3Sync.exe” is the default install path so it is likely that your executable file will be in the same location, however it is best to check and make changes if necessary.

Next, the “-AWSAccessKeyId” and “-AWSSecretAccessKey” options are for the Access and Secret Access keys unique to your AWS account. If you are unsure of where to find these keys, please consult this tutorial:

<http://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSGettingStartedGuide/AWSCredentials.html>.

Next is the “-BucketName” you wish to upload your backup to. The bucket name specified must exist for this command to be executed successfully. The following option is “-S3FolderKeyName” for which a folder may be (this is not a mandatory option) specified to upload to. If the folder does not exist within the specified bucket, one will be created.

The “-SyncDirection” option can be set to “Upload” or “Download” for expected results. In this case, we are going to be uploading the checkpointed database files to S3.

The “-LocalFolderPath” must lead to the location of your database files (more specifically, the folder containing the corresponding “.iq” and “.iqtmp” files). Note that the “-LocalFolderPath” is local to the server machine, not necessarily the machine executing the command. The following “-IncludeOnlyLocalFilesRegularExpression” option takes in a regular expression used to filter files from the “-LocalFolderPath” specified. In this case, the “.*\iq\$|.*\iqtmp” regular expression will result in uploading only files with “.iq” and “.iqtmp” extensions which happen to be the checkpointed database files we are interested in.

The “-CompareFilesBy” option specifies how S3Sync should determine whether a file in S3 should be overwritten or not. We have set this option to “Timestamp” meaning if the “Modified” timestamp and size of the local file and the timestamp and size of the file on S3 are the same, the files are considered the same and will not be uploaded.

You will need to provide your S3Sync license key after the “-LicenseKey” option. Note: A free 60 day trial is available by providing a valid email address on the S3Sync website.

Last is the “-LogFilePath” option (eg. “C:\Users\\AppData\Local\Temp\S3SyncLog.txt”). The example folder is dumped upon powering off the machine so if you would like to store log files for later reference, it is advised to choose a new path.

To recap, the first line of the batch file called the “virtualBackup.sql” file to create a database checkpoint and uploaded the corresponding files to S3 by calling the “uploadToS3.bat” file on the server machine. Stepping back, the remainder of the “virtualBackup.sql” file created a relatively small backup file (with a “.1” file extension) on the server machine which contains the catalogue database. This additional backup file will be required to complete the restoration process, so we need to copy it to your S3 repository as well.

The second line of “virtualBackupAndUpload.bat” is the line that copies the backup file containing the catalogue database up to your S3 repository:

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSSecretAccessKey  
%aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-  
folder-name% -SyncDirection Upload -LocalFolderPath %local-folder-path% -  
IncludeOnlyLocalFilesRegularExpression ".*\.1$" -CompareFilesBy Timestamp -  
LicenseKey %license-key% -LogFilePath %log-file-path%
```

This command will not be explained in detail as it is almost identical to the call in the “uploadToS3.bat” file. The only difference is that now we are only uploading files with the extension “.1” as stated in the regular expression specified for the “-IncludeOnlyLocalFilesRegularExpression” option. The “-LocalFolderPath” you specify should be the same location that you used in the “TO” clause in “virtualBackup.sql”.

Note: This S3Sync call will not access a folder on a remote machine. There are a few options for solving this problem. The first is to use “psexec.exe” which can be downloaded from the Microsoft site <https://technet.microsoft.com/en-ca/sysinternals/bb897553.aspx>. Instructions for using “psexec.exe”, however, are outside the scope of this tutorial. The second option is to simply copy the backup file from the remote machine to the machine currently running this batch file. Alternatively, if you can run the batch file on the server machine then the problem is avoided completely.

Downloading a database backup file from S3 and restoring it

When restoring the IQ database from the backup in your S3 repository, we need to first download the database files and then run the RESTORE DATABASE command. The process of restoring the database does require that the backup files are locally accessible in a directory or mount from the IQ server.

We will be working with the downloadAndVirtualRestore.bat batch file for the remainder of this section.

The first command of the batch file is as follows:

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSSecretAccessKey  
%aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-  
folder-name% -SyncDirection Download -LocalFolderPath %local-folder-path% -  
CompareFilesBy Timestamp -LicenseKey %license-key% -LogFilePath %log-file-path% -  
IncludeOnlyLocalFilesRegularExpression ".*\.1$"
```

This command should need very little explanation as it is nearly identical to the second and last command in “virtualBackupAndUpload.bat”. The only difference is that we are now downloading files from S3 having the extension “.1” instead of uploading them. Note that the “-LocalFolderPath” (save location for “.1” file(s)) will be local to the machine executing the command.

The second command of the batch file is as follows:

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSSecretAccessKey  
%aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-  
folder-name% -SyncDirection Download -LocalFolderPath %local-folder-path% -
```

```
CompareFilesBy Timestamp -LicenseKey %license-key% -LogFilePath %local-folder-  
path% -IncludeOnlyLocalFilesRegularExpression ".*\.iq$|.*\.iqtmp"
```

Again, this call is almost identical to that of “uploadToS3.bat” with the exception of “-SyncDirection” being “Download” instead of “Upload” and the slight modification of the regular expression option: “-IncludeOnlyS3FilesRegularExpression”. This call will download all files in the specified bucket/folder having the extension “.iq” or “.iqtmp” to the “-LocalFolderPath” specified (on the machine executing the command).

Now that all of the backup files have been downloaded from S3, the database can be restored. This is done in the third and final command in the batch file:

```
dbisql -c "DBN=utility_db;UID=dba;PWD=sql;" -host %ip-address% virtualRestore.sql
```

Note: before running this command, a server must be running on the machine on which the backup resides in order to connect to “utility_db”. This can be done via the Windows cmd command line on the respective machine using:

```
"C:\Users\Administrator\Desktop\IQ_16.0\SCC-  
3_2\bin\sa\bin_windows64\dbsrv##.exe" -n MyServer -su sql
```

The initial path leads to dbsrv##.exe where ## is the version of IQ installed on the machine running this command. The “-n” option specified the name of the server and the “-su” option sets a password for “utility_db”.

As well, the database to be restored must not be running for the dbisql command to execute successfully.

This dbisql command is similar to the first command of the “virtualBackupAndUpload.bat” tutorial but includes “DBN=utility_db” in the “-c” option specification. As you might expect, this connects to the “utility_db” allowing us to restore a database.

Next, the IP address of the server machine hosting “utility_db” is specified. If you are running this command from a machine other than the one hosting the server, you can perform the restore as long as the backup file resides on the server machine.

Finally, the “virtualRestore.sql” file, shown below, is executed to perform the actual restore:

```
RESTORE DATABASE '<db-name>'
FROM '<backup-path>\backup'
```

The **<db-name>** is the relative or absolute path of the database to be restored. This can be the original location, or a new location for the catalog store file.

Note: restores are not always backwards compatible. For example, if a database is backed up using IQ 16, an older version of IQ may not be able to restore it.

Note: for more information regarding the RESTORE statement, please consult [RESTORE DATABASE Statement](#).

Appendix

virtualBackupAndUpload.bat

REM variable declarations. variables used in commands are surrounded by '%'

SET ip-address=<ip-address>

SET port=<port>

SET path-to-S3sync-exe=<"path-to-S3sync-exe">

SET aws-access-key-id=<aws-access-key-id>

SET aws-secret-access-key=<aws-secret-access-key>

SET bucket-name=<bucket-name>

SET s3-folder-name=<s3-folder-name>

SET local-folder-path=<"local-folder-path">

SET license-key=<license-key>

SET log-file-path=<"log-file-path">

REM calls Interactive SQL to run the file "virtualBackup.sql" which performs

REM a virtual backup of a database on the database server and exports

REM the iq database to Amazon S3

```
dbisql -c "UID=dba;PWD=sql;" -host %<ip-address>% -port %<port>% virtualBackup.sql
```

REM upload catalog (SQL Anywhere files) to Amazon S3 (will have extension .1)

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSSecretAccessKey %aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-folder-name% -SyncDirection Upload -LocalFolderPath %local-folder-path% -IncludeOnlyLocalFilesRegularExpression ".*\..1$" -CompareFilesBy Timestamp -LicenseKey %license-key% -LogFilePath %log-file-path%
```

virtualBackup.sql

BACKUP DATABASE

FULL VIRTUAL ENCAPSULATED

--The following line executes uploadToS3.bat on the server machine

```
'<path-to-batch-file>\uploadToS3.bat'
```

--a file, backup.1, will be created in the <backup-path> directory

TO '**<backup-path>**\backup'

[uploadToS3.bat](#)

REM variable declarations. variables used in commands are surrounded by '%'

SET path-to-S3sync-exe=<"**path-to-S3sync-exe**">

SET aws-access-key-id=<**aws-access-key-id**>

SET aws-secret-access-key=<**aws-secret-access-key**>

SET bucket-name=<**bucket-name**>

SET s3-folder-name=<**s3-folder-name**>

SET local-folder-path=<"**local-folder-path**">

SET license-key=<**license-key**>

SET log-file-path=<"**log-file-path**">

REM upload iq files to Amazon S3

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSSecretAccessKey %aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-folder-name% -SyncDirection Upload -LocalFolderPath %local-folder-path% -IncludeOnlyLocalFilesRegularExpression *.*\iq$.*\.iqtmp" -CompareFilesBy Timestamp -LicenseKey %license-key% -LogFilePath %log-file-path%
```

[downloadAndVirtualRestore.bat](#)

REM variable declarations. variables used in commands are surrounded by '%'

SET ip-address=<**ip-address**>

SET path-to-S3sync-exe=<"**path-to-S3sync-exe**">

SET aws-access-key-id=<**aws-access-key-id**>

SET aws-secret-access-key=<**aws-secret-access-key**>

SET bucket-name=<**bucket-name**>

SET s3-folder-name=<**s3-folder-name**>

SET local-folder-path=<"**local-folder-path**">

SET license-key=<**license-key**>

```
SET log-file-path=<"log-file-path">
```

REM download full backup file from Amazon S3

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSecretAccessKey %aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-folder-name% -SyncDirection Download -LocalFolderPath %local-folder-path% -CompareFilesBy Timestamp -LicenseKey %license-key% -LogFilePath %log-file-path% -IncludeOnlyLocalFilesRegularExpression ".*\.$"
```

REM download the .iq and .iqtmp files to the database folder on the server machine

```
%path-to-S3sync-exe% -AWSAccessKeyId %aws-access-key-id% -AWSecretAccessKey %aws-secret-access-key% -BucketName %bucket-name% -S3FolderKeyName %s3-folder-name% -SyncDirection Download -LocalFolderPath %local-folder-path% -CompareFilesBy Timestamp -LicenseKey %license-key% -LogFilePath %local-folder-path% -IncludeOnlyLocalFilesRegularExpression ".*\.$|.iq|.iqtmp"
```

REM calls Interactive SQL to run the file "virtualRestore.sql" which performs a restore of a database on the database server

```
dbisql -c "DBN=utility_db;UID=dba;PWD=sql;" -host %ip-address% virtualRestore.sql
```

[virtualRestore.sql](#)

--restore from the IQ full backup file

```
RESTORE DATABASE '<db-name>'
```

--must include a filename at end of backup-location path (no extension)

```
FROM '<backup-path>\backup'
```




Copyright 2015 SAP SE. All rights reserved ©

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only.

National product specifications may vary.

These materials are subject to change without notice.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP NetWeaver "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.