Backing up IBM Rational DOORS by using snapshots

Note

Before using this information, be sure to read the general information under “Notices” on page 17.

Contents

Chapter 1. About this paper

IBM® Rational® has investigated snapshots as a method of backing up a Rational DOORS® database. The techniques that we used maintained data integrity while the database was taken offline for a minimal amount of time. On Windows® 2003, we were able to back up a 60 GB database while the database was offline for approximately 10 seconds. On Solaris, a 60 GB database was offline for approximately 70 seconds.

This paper contains the background information and sample scripts for you to customize and run in your own environment.

Chapter 2. Introduction

Every business needs to have a backup strategy that strikes a balance between ensuring the integrity of its data and minimizing the disruption to its ongoing operations. For the many businesses that have Rational DOORS as the cornerstone of their success, one of the major challenges facing system administrators is how to integrate backing up Rational DOORS into their work schedules. The following process shows you an efficient way to back up your Rational DOORS database using snapshots.

1. Stop the Rational DOORS database server. Before you stop the Rational DOORS database server, make sure all your users are disconnected. When you stop the server, users will lose any unsaved changes and will need to restart their Rational DOORS client.
2. Take a snapshot of the data.
3. Restart the database server.
4. Back up the snapshot of the data while the Rational DOORS database itself continues to be available to your users.

IBM Rational carried out tests on Windows 2003 and Solaris. Instructions and information about each test is contained in the following sections:

- Backing up a Windows 2003 database
- Backing up a Solaris database

Though we did not run any tests on Linux® and HP-UX, this paper also contains information about them:

- Backing up a Linux/HP-UX database

If your operating system is not included in this paper, a similar procedure might work for your system, but it has not been tested.

Chapter 3. Backing up a Windows 2003 database

There is a Microsoft® command-line tool called VShadow that allows you to take a snapshot of a Windows 2003 Rational DOORS database. It is included in the Microsoft Windows SDK for Windows Vista, which you can download from http://msdn.microsoft.com/windowsvista. For information about VShadow, follow this link: http://msdn2.microsoft.com/en-us/library/bb530725(VS.85).aspx.

In trials, IBM Rational used VShadow to back up a 60 GB Rational DOORS database where the database was offline for *less than 10 seconds*.

Trial hardware and software configuration

In the trials, we used the following hardware and software configuration:

- Dell 2950
- Inter Xeon 3.20 GHz 2 Duals
- 4 GB Ram
- 1 X 80 GB hard disk drive
- 1 X 250 GB hard disk drive
- Win2003 Server SP2
- Partition size used: 220 GB

Sample script

This sample script (**vshadow.cmd**):

1. Stops the database server.
2. Deletes any old versions of the snapshot of the database.
3. Runs VShadow.exe to take a snapshot of the database.
4. Restarts the database server.

vshadow.cmd

This script takes two arguments: the drive or volume that you want to take the snapshot of and where to mount the snapshot. You can mount the snapshot either on another drive or an empty folder on an NTFS partition. The folder can also be on the same drive as the original Rational DOORS data.

```
setlocal
if NOT "%CALLBACK_SCRIPT%"=="" goto :IS_CALLBACK
@REM Stop DOORS DB service.
net stop "DOORS DB Server 8.3"
@REM Get the source and destination paths.
set SOURCE_VOLUME=%1
set DESTINATION_VOLUME=%2
set CALLBACK_SCRIPT=%~dpnx0
set TEMP_GENERATED_SCRIPT=GeneratedVarsTempScript.cmd
@REM Remove any existing snapshots of your target; you only want a snapshot of the db as it is now.
%~dp0\vshadow.exe -do=%SOURCE_VOLUME%
@REM Create the snapshot.
%~dp0\vshadow.exe -nw -p -script=%TEMP_GENERATED_SCRIPT% -exec=%CALLBACK_SCRIPT% %SOURCE_VOLUME%
del /f %TEMP_GENERATED_SCRIPT%
@goto :EOF
:IS_CALLBACK
setlocal
call %TEMP_GENERATED_SCRIPT%
%~dp0\vshadow.exe -el=%SHADOW_ID_1%,%DESTINATION_VOLUME%
@REM Restart DOORS DB server.
net start "DOORS DB Server 8.3"
```

Note: Scripts are included in a smaller font to minimize the number of unnecessary line, which can result in the script not working.

Hints and tips

- The original Rational DOORS database and the snapshot must both be on the same physical machine.
- If you need to restore a backup, you need to restore to the location of the original data, not the location of the snapshot.
- Although the snapshot only takes a small amount of time, it is always best to run the script out of hours. If your business does not recognize the concept of "out of hours", run the script when the database is being used the least. Although the database is only offline for a short time, backing up the snapshot places a load on the disk that can affect the performance of your Rational DOORS database.
- Depending on the amount of activity on the Rational DOORS database, the disk space needed for the snapshot should be between 10% and 20% of the partition's total size. If there is little usage, the disk space can be even smaller.
- Rational recommends that you have only Rational DOORS data on the partition.
- If there is a power failure during the backup, the backup will fail. Rational recommends that you use an APC RS high performance backup and protection system, or the equivalent.

Performing your own backup

Backing up your Windows 2003 Rational DOORS database:

1. Make a copy of the sample VShadow command script and customize it to the demands of your environment.
2. Choose a time to run your script.
3. Run the script and create the snapshot.
4. While your users continue to use the Rational DOORS database, back up the snapshot.
5. When the backup has completed, it is good practice to free up disk space by deleting the snapshot.

You can automate this procedure by splitting it into a pair of related scripts, running one as a pre-snapshot script, and the other as a post-snapshot script.

Chapter 4. Backing up a Solaris database

A command-line tool called UFS Snapshot is available as part of the Solaris product. With it, you can take a snapshot of a Solaris Rational DOORS database.

In trials, IBM Rational used UFS Snapshot to back up a range of databases from 1 GB to 60 GB, with the following results:

Table 1. UFS Snapshot times

Database size	Elapsed time for snapshot
1 GB	10 seconds
5 GB	10 seconds
10 GB	10.2 seconds
20 GB	57 seconds
40 GB	64 seconds
60 GB	70 seconds

Trial hardware and software configuration

In the trials, we used the following hardware and software configuration:

Table 2. Hardware and software configuration

Smaller back ups	Larger backups
Sun SFV125 1 X 1 GHz UltraSPARCIIIi processor 2 GB Ram 2 X 73 GB hard disk drive (software raid 1) OS Solaris 10 44 GB Partition	Sun V240 2 X 1.5 GHz UltraSPARCIIIi processor 4-way Superscalar 4 GB Ram 4 X 73 GB hard disk drive (software raid 1) OS Solaris 10 200 GB Software Raid Partition

Note: As a rule, Rational has found that the configuration is not crucial to the time it takes to take the snapshot. Rather, it is the size of the Rational DOORS database that is the single most important factor.

Sample scripts

The following two scripts are sample scripts:

- **doors.txt**
- **ufs_snapshot.txt**

The doors.txt script stops and starts the Rational DOORS server, and ufs_snapshot.txt calls doors.txt and performs the snapshot.

doors.txt

```
#!/bin/sh
# Start/stop script for UNIX DOORS DB server

# Set config parameters
# Set the DOORS username, normally doors
DOORSUSER=doors
# Set tcp port server will run on
PORTNUMBER=36677
# Set location of DOORS installation
DOORSHOME=/path/to/doors
# Set location of DOORS data
SERVERDATA=/path/to/doorsdata
# Add DOORS bin folder to path
PATH=$PATH:$DOORSHOME/bin
export PORTNUMBER DOORSHOME LOCALDATA SERVERDATA PATH

# On Solaris we need to use the right echo, shell built in doesn't support -n
if [ `uname` = SunOS ]
then
  ECHO=/usr/ucb/echo
else
  ECHO=/usr/bin/echo
fi

ERSPID=`pgrep -x doorsd`

case $1 in
start) if [ -x $DOORSHOME/bin/doorsd ]
        then
          if [ -n "$ERSPID" ]
          then
            $ECHO "DOORS DB server already running: PID $ERSPID"
            exit 1
          else
            $ECHO -n "Starting DOORS Database Server: "
            su doorsd -c "$DOORSHOME/bin/doorsd -inactiveClientPoll 3600 > /dev/null &" && echo started || echo failed
          fi
        fi
        ;;

stop) if [ -x $DOORSHOME/bin/dbadmin ]
        then
        if [ -n "$ERSPID" ]
        then
          $ECHO -n "Stopping Doors Database Server PID $ERSPID: "
          su $DOORSUSER -c "$DOORSHOME/bin/dbadmin -d $PORTNUMBER@localhost -K" && echo stopped || echo failed
        else
          echo "DOORS Database not running."
          exit 1
        fi
      fi
        ;;

status) if [ -n "$ERSPID" ]
        then
          $ECHO "DOORS Database running: PID $ERSPID"
        else
          $ECHO "DOORS Database not running"
        fi
        ;;

*) echo "Useage: $0 [start|stop|status]"
        ;;
esac
exit 0
```

ufs_snapshot.txt

```
#!/bin/sh
# Script for creating Solaris UFS snapshot of DOORS for backup

# Location of DOORS start/stop script
DOORS_SCRIPT=/path/to/script
# Mount point of UFS file system to be snapped
BACKUP_VOLUME=/mount/point
# Location of backing store, this must be on a partition with
# enough space to hold the changes 10-20% of the size of the
# partition being snapped is common but depends on useage
BACKING_STORE=/path/to/buffer_file
# Where to mount the snapshot, it is this we will then back up,
# be aware that any backup will have this path so a restore will
# need to be relocated to the appropriate location
BACKUP_MOUNT=/backup/mount/point
```

```
SNAP_SCRIPT=`basename $0`

case $SNAP_SCRIPT
in
  backup_snapshot) if [ "$#" -ne 1 ]
                   then
                        echo "Useage: backup_snapshot [create|remove]"
                        exit 1
                   fi
                   MODE=$1
                   ;;
  backup_snapshot_create) MODE=create
                   ;;
  backup_snapshot_remove) MODE=remove
                   ;;
esac

case $MODE
in
  create) if [ `fssnap -i $BACKUP_VOLUME | wc -l` -ne 0 ]
   then
     echo "Snapshot already exists for volume: $BACKUP_VOLUME"
     $SNAP_SCRIPT remove
   fi
   $DOORS_SCRIPT stop
   echo "Setting up snapshot of $BACKUP_VOLUME"
   VDEV=`fssnap -F ufs -o bs=$BACKING_STORE,unlink $BACKUP_VOLUME`
   mount -F ufs -o ro $VDEV $BACKUP_MOUNT
   $DOORS_SCRIPT start
   ;;
  remove) echo "Removing snapshot of $BACKUP_VOLUME"
   umount $BACKUP_MOUNT && fssnap -d $BACKUP_VOLUME || echo "Couldn't unmount $BACKUP_VOLUME"
   ;;
  *) echo "Useage: backup_snapshot [create|remove]"
     exit 1
     ;;
esac

exit 0
```

Hints and tips

- The original Rational DOORS database and the snapshot must both be on the same physical machine.
- If you need to restore a backup, you need to restore to the location of the original data, not the location of the snapshot.
- Although the snapshot only takes a small amount of time, it is always best to run the script out of hours. If your business does not recognize the concept of "out of hours", run the script when the database is being used the least. Although the database is only offline for a short time, backing up the snapshot places a load on the disk that can affect the performance of your Rational DOORS database.
- Depending on the amount of activity on the Rational DOORS database, the disk space needed for the snapshot should be between 10% and 20% of the partition's total size. If there is little usage, the disk space can be even smaller.
- Rational recommends that you have only Rational DOORS data on the partition.
- If there is a power failure during the backup, the backup will fail. Consider using an APC RS high performance backup and protection system, or the equivalent.

Performing your own backup

Backing up your Solaris Rational DOORS database:

1. Make a copy of the sample command scripts and customize them to the demands of your environment.
2. Choose a time to run your script.
3. Log in as a superuser.

4. Run the scripts and create the snapshot.
5. While your users continue to use the Rational DOORS database, back up the snapshot.
6. When the backup has completed, it is good practice to free up disk space by deleting the snapshot.

You can automate this procedure by splitting it into a pair of related scripts, running one as a pre-snapshot script, and the other as a post-snapshot script.

Chapter 5. Backing up a Linux/HP-UX database

Rational has not run any tests on using snapshots to back up a Linux Rational DOORS database or an HP-UX database. However, we have found that there is a Linux tool that allows you to take snapshots, and we have included sample scripts for you to customize to your own environment.

A tool called Logical Volume Manager (LVM) is available as part of the Linux product. You can use it to take a snapshot of a Linux Rational DOORS database.

Sample scripts

The following two scripts are sample scripts:

- **doors.txt**
- **lvm.txt**

The doors.txt script stops and starts the Rational DOORS server, and lvm.txt calls doors.txt and performs the snapshot.

Although these scripts are primarily for a Linux database, you can amend them for use in an HP-UX environment.

doors.txt

```
#!/bin/sh
# Start/stop script for UNIX DOORS DB server

# Set config parameters
# Set the DOORS username, normally doors
DOORSUSER=doors
# Set tcp port server will run on
PORTNUMBER=36677
# Set location of DOORS installation
DOORSHOME=/path/to/doors
# Set location of DOORS data
SERVERDATA=/path/to/doorsdata
# Add DOORS bin folder to path
PATH=$PATH:$DOORSHOME/bin
export PORTNUMBER DOORSHOME LOCALDATA SERVERDATA PATH

# On Solaris we need to use the right echo, shell built in
# doesn't support -n
if [ `uname` = SunOS ]
then
  ECHO=/usr/ucb/echo
else
  ECHO=/usr/bin/echo
fi

ERSPID=`pgrep -x doorsd`

case $1 in
start) if [ -x $DOORSHOME/bin/doorsd ]
        then
          if [ -n "$ERSPID" ]
          then
            $ECHO "DOORS DB server already running: PID $ERSPID"
            exit 1
          else
            $ECHO -n "Starting DOORS Database Server: "
            su doorsd -c "$DOORSHOME/bin/doorsd -inactiveClientPoll 3600 > /dev/null &" && echo started || echo failed
          fi
        fi
        ;;

stop) if [ -x $DOORSHOME/bin/dbadmin ]
        then
        if [ -n "$ERSPID" ]
        then
          $ECHO -n "Stopping Doors Database Server PID $ERSPID: "
```

```
          su $DOORSUSER -c "$DOORSHOME/bin/dbadmin -d $PORTNUMBER@localhost -K" && echo stopped || echo failed
        else
          echo "DOORS Database not running."
          exit 1
        fi
      fi
        ;;

status) if [ -n "$ERSPID" ]
        then
          $ECHO "DOORS Database running: PID $ERSPID"
        else
          $ECHO "DOORS Database not running"
        fi
        ;;

*) echo "Useage: $0 [start|stop|status]"
        ;;
esac
exit 0
```

lvm.txt

```
#!/bin/sh
# Script for creating Linux LVM snapshots of DOORS for
# backup - this script can be adapted for HP-UX.

# Location of DOORS start/stop script
DOORS_SCRIPT=/path/to/script
# Volume group that contains the logical volume
VOLUME_GROUP=VolGrp01
# Logical volume to be snapped
BACKUP_VOLUME=LogVol01
# Amount of space to allocate to the snapshot to buffer the
# changes on the live volume. This is commonly 10-20% of the
# size of the original volume depending on useage. The following
# would assign 10 gigabytes.
SIZE=10G
# Name of backup volume
NAME=backup
# Where to mount the snapshot, it is this we will then back up,
# be aware that any backup will have this path so a restore will
# need to be relocated to the appropriate location
BACKUP_MOUNT=/backup/mount/point

SNAP_SCRIPT=`basename $0`

case $SNAP_SCRIPT
in
  backup_snapshot) if [ "$#" -ne 1 ]
                   then
                        echo "Useage: backup_snapshot [create|remove]"
                        exit 1
                   fi
                   MODE=$1
                   ;;
  backup_snapshot_create) MODE=create
                   ;;
  backup_snapshot_remove) MODE=remove
                   ;;
esac

case $MODE
in
  create) if lvdisplay /dev/$VOLUME_GROUP/$NAME > /dev/null 2>&1
   then
     echo "Snapshot already exists for volume: $BACKUP_VOLUME"
     $SNAP_SCRIPT remove
   fi
   $DOORS_SCRIPT stop
   echo "Setting up snapshot of $BACKUP_VOLUME"
   lvcreate --size $SIZE --snapshot --name $NAME /dev/$VOLUME_GROUP/$BACKUP_VOLUME
   mount -o ro  /dev/$VOLUME_GROUP/$NAME $BACKUP_MOUNT
   $DOORS_SCRIPT start
   ;;
  remove) if lvdisplay /dev/$VOLUME_GROUP/$NAME > /dev/null 2>&1
   then
     echo "Removing snapshot of $BACKUP_VOLUME"
     umount $BACKUP_MOUNT && lvremove /dev/$VOLUME_GROUP/$NAME || echo "Couldn't unmount $NAME"
   else
     echo "No snapshot of $BACKUP_VOLUME found"
   fi
   ;;
  *) echo "Useage: backup_snapshot [create|remove]"
```

```
    exit 1
    ;;
esac

exit 0
```

Hints and tips

- The original Rational DOORS database and the snapshot must both be on the same physical machine.
- If you need to restore a backup, you need to restore to the location of the original data, not the location of the snapshot.
- Although the snapshot only takes a small amount of time, it is always best to run the script out of hours. If your business does not recognize the concept of "out of hours", run the script when the database is being used the least. Although the database is only offline for a short time, backing up the snapshot places a load on the disk that can affect the performance of your Rational DOORS database.
- Depending on the amount of activity on the Rational DOORS database, the disk space needed for the snapshot should be between 10% and 20% of the total partition size. If there is little usage, the disk space can be even smaller.
- The partition containing the Rational DOORS data should not be the same size as the volume. Space on the same volume is needed to store the snapshot.
- Rational recommends that you have only Rational DOORS data on the partition.
- If there is a power failure during the backup, the backup will fail. Consider using an APC RS high performance backup and protection system, or the equivalent.

Performing your own backup

Backing up your Linux/HP-UX Rational DOORS database:

1. Make a copy of the sample command scripts and customize them to the demands of your environment.
2. Choose a time to run your script.
3. Log in as a superuser.
4. Run the scripts and create the snapshot.
5. While your users continue to use the Rational DOORS database, back up the snapshot.
6. When the backup has completed, it is good practice to free up disk space by deleting the snapshot.

You can automate this procedure by splitting it into a pair of related scripts, running one as a pre-snapshot script, and the other as a post-snapshot script.

Chapter 6. Summary

Intelligent use of available snapshot technology allows you to maintain the integrity of your Rational DOORS data while minimizing the disruption to your work schedules.

Rational has tested Windows 2003 and Solaris, but if you are using a different operating system, there may be similar tools available for you to use. Taking the approach outlined in this paper, you may be able to devise your own solutions.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this documentation in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Intellectual Property Dept. for Rational Software
IBM Corporation
1 Rogers Street
Cambridge, MA 02142
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks and service marks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at www.ibm.com/legal/copytrade.html.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.