Best Practices for Deploying Rational Team Concert on NetApp Storage

Agnes Jacob and Srinivas Kodiyalam, NetApp
October 2010 | TR-3882

ABSTRACT

This document provides an overview of the architecture and deployment scenarios of Rational Team Concert (RTC). It will provide performance numbers of Rational Team Concert in a cloud and standard environments using NetApp® storage. It will also describe data protection and disaster recovery options of RTC on NetApp storage.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION TO RATIONAL TEAM CONCERT (RTC)

Rational Team Concert (RTC) is a collaboration tool built on a scalable and extensible platform that seamlessly integrates tasks and processes across the software development lifecycle. It is IBM's next generation of products to address the changing and growing needs of product development. With the rise of global distributed teams and projects spanning multiple business units, tools that would enforce process and provide better collaboration became a requirement. Rational Team Concert addresses these needs by providing an infrastructure for agile development, project management, source control, build management, transparency, and iteration planning. The goal is to enable software delivery teams to be more effective and productive in order to deliver a high-quality product on schedule.

Rational Team Concert can be deployed in a stand-alone environment or in a public and private cloud. RTC in a cloud transforms software delivery by enabling self-service, rapid provisioning, ease of use and access, and reduction of IT operations and infrastructure costs. In an RTC private cloud, the application is managed and owned by the customer. Deploying in a private cloud allows for customization, efficiency, security, and privacy. In contrast, in an RTC public cloud a service provider owns, manages, and provides access by subscription to the application. Also, there are standardization, preservation of capital, and flexibility in a public cloud.

Other operational benefits of Rational Team Concert include low administrative overhead and a lighter weight framework that is extensible and scalable. Hence, RTC products are based on number of users. Their offerings range from small teams of 10 users to enterprise teams of unrestricted number of users. They also provided a facility in which existing artifacts that reside in other software control management such as Subversion, Clearcase, and ClearQuest can be integrated.

2 ADVANTAGES OF USING NETAPP STORAGE FOR RATIONAL TEAM CONCERT

NetApp's shared storage infrastructure is an ideal foundation for RTC primarily due to the collaboration requirements of RTC as well as for enabling a flexible and scalable infrastructure. NetApp provides shared storage infrastructure supporting:

- One or more applications
- Virtualized environments
- Advanced data centers that deliver IT as a service

All are based on a "unified" storage architecture that is much more than multiprotocol. Clearly, multiple protocols are an aspect of it, but the same architecture needs to be applicable for the entire range from low-end to high-end systems. Customers need different price-performance profiles in the same architecture, which you can do by using NetApp Flash assist technologies or caching techniques. These are all varying aspects of NetApp's unified architecture that provide a unified set of tools enabling a unified set of processes, disaster recovery, backup, provisioning, management, and maintenance; ultimately enabling the RTC environment and the customer to get a massive benefit in terms of complexity reduction.

Selecting the appropriate storage infrastructure to address the multitude of software development challenges and needs of development environments is critical. Some of the storage challenges include:

- Scaling (and performance)
- Storage efficiency
- Quick backup and restore
- Disaster recovery

2.1 SCALING

Scaling involves multiple dimensions: the storage infrastructure has to be able to scale in terms of performance. This necessitates arrays that have the right optimized file system, that are able to take advantage of new media types such as SSDs, the assist capabilities that Flash can provide, and techniques such as caching (which report the I/Os much closer to the end users and the applications without them having to traverse either the network or going really to the HDDs). NetApp's shared storage infrastructure provides for many different techniques for addressing performance. With application consolidation such as with RTC environments and other software configuration management (SCM) tools such as Clearcase and Clearquest, the performance requirements become even more acute because you're putting more applications onto the same infrastructure.

The second dimension of scaling is the sheer amount of data that this infrastructure needs to be able to handle. Clearly there will be more terabytes hanging off of each individual storage array, but that is no longer enough. For RTC, it is desirable to provide a storage infrastructure with a single logical pool that can be provisioned as a logical pool across multiple storage arrays vs. storage in an array by array by array fashion.

The third dimension of scaling, as we begin to build those clustered logical storage pools, is about making sure that it can operationally scale. No longer can the storage administrators and the application administrators continue to spend the time they were spending on the manual activities they were previously doing. Things such as provisioning, data protection, and any kind of tuning change into the storage infrastructure become harder and harder. It's one thing to make those changes to a few terabytes, but changes are far more complex when your scale is in multiple petabytes.

2.2 STORAGE EFFICIENCY

The NetApp approach to storage efficiency involves several components, both hardware and software. These components are designed as integral pieces of a complete solution:

- **Thin provisioning with FlexVol®:** When an RTC storage administrator allocates storage based on expected peak storage needs and anticipated growth, much of that storage is commonly wasted due to some inherent guesswork used in the process. The guessing is needed because the traditional methods do not allow for the volumes to grow if more storage is needed later. Neither do traditional provisioning methods allow storage administrators to easily reclaim the unneeded space. The resulting waste in resources comes at a cost for IT organizations in the form of buying extra storage and the extra costs of administering it. NetApp thin provisioning is based on FlexVol technology. FlexVol volumes can grow and shrink on demand and can share a common storage pool used to autogrow the volumes if needed. RTC users will see their requested amount of storage allocated as expected, but the volume can actually be smaller with the ability to autogrow as the storage is used and more storage is needed. The volume also has the ability to shrink as appropriate, returning unused storage to a shared storage pool.

- **Virtual volume copies:** FlexClone® volumes are writable Snapshot™ copies, allowing many instances of the core data to be used, while only using a small fraction of storage to save only the changes for each instance. Instead of provisioning a large storage capacity to perform application testing, a central copy of the application data is "shared" with the test data, resulting in extreme storage efficiency. FlexClone copies are invaluable in RTC test/dev environments where patches and updates are tested and prestaging is commonplace. In addition to space savings, there are rapid build and teardown because each clone is created by simply creating pointer references to the central data, and teardown is simply the deletion of the pointer references. Overall FlexClone copies provide means to deploy the preferred versions of RTC tool faster by allowing more simultaneous test instances of the code and by allowing rapid build and teardown of dev/test environments.

- **Deduplication:** Deduplication searches for and removes duplicate data by freeing up duplicated blocks. NetApp deduplication can be applied to primary storage, replicated storage, backup storage, and archival storage. Dedupe is at the block (subfile) file compared to other implementations, which work at the file level only. NetApp dedupe can be enabled on a volume-by-volume basis, which allows a focused dedupe effort and allows dedupe to skip volumes that do not have duplicate data blocks, such as encrypted data or compressed data. Dedupe can be scheduled to happen at off-peak times if the RTC workload is especially performance sensitive.

Storage efficiency from a hardware perspective could also be realized by getting more out of low-cost components such as SATA drives by putting Flash Cache modules to drive acceleration on top of SATA drives.

2.3   BACKUP AND RECOVERY

The RTC environment must be backed up to safeguard its business-critical components. NetApp Snapshot technology can provide quick and reliable backups and near-instantaneous online data recovery to help keep your business up and running. With NetApp Snapshot copies, you can save up to 255 instantaneous, point-in-time versions of each data volume. NetApp SnapCreator and SnapManager® products for DB2 and Oracle® can be used to create consistent backups of the database files and logs, as well as the volumes that house the RTC server code and configurations.

NetApp SnapRestore® software can provide rapid restores and immediate online data recovery. Using a simple single command, you can recover entire volumes or individual files in seconds.

2.4   DISASTER RECOVERY

RTC and other SCM environments must be able to maintain application availability during planned or unplanned downtime and have the ability to quickly reestablish full functionality in case of a disaster. NetApp unified architecture, with a single standard software platform that spans across the entire storage system, makes sure of common IT processes, helping to reduce likelihood of operator errors. NetApp provides a comprehensive solution set addressing all the causes of downtime and improves application and data availability.

- **Fast data recovery:** Snapshot and SnapRestore technologies make sure of fast data recovery on the primary storage with minimal data loss and negligible performance impact. SnapVault® backs up the Snapshot copies to a secondary storage, resulting in a cost-effective backup and recovery solution that is far superior in performance to a tape-based solution both from a backup and a recovery perspective.

- **Storage resiliency:** RAID-DP® provides the ultimate protection against double disk drive failures while maintaining a high level of performance. Better double disk failure protection than RAID 1+0 at a much lower cost. The measured performance overhead of RAID-DP is 3% or less, which is much smaller than other dual parity (RAID 6) implementations. SyncMirror® combines disk array mirroring with RAID-DP to provide data protection that is superior to conventional mirroring.

- **Site protection: metro and regional solution:** NetApp MetroCluster™ provides a unique, cost-effective integrated high-availability and disaster recovery solution. Providing site protection in addition to protecting against hardware outages, it makes sure of zero data loss and minimal downtime in the event of a disaster. MetroCluster combines array-based clustering with synchronous mirroring to provide continuous availability to an SCM environment. The combination of MetroCluster and SnapMirror® data replication solution provides the most comprehensive level of data protection in that customers can use to protect themselves from both unplanned events that occur inside and outside of the data center.

There are several considerations to be made when deciding which type of disaster recovery solution to implement. Not all applications are equal, so you need to tier applications based on their criticality and

have different sets of recovery point objectives (RPOs) for each application tier. RPOs define the amount of time for which work may be lost in the event of an unplanned outage at the primary site. Recovery time objective (RTO) is how quickly you need to failover to a disaster recovery site—seconds, minutes, hours, days. This will depend on the likely geographic impact of the disaster that you are protecting against. Also, many large organizations tend to mirror between their existing geographically distributed data centers, as it eliminates the cost of setting up a brand new DR site. SCM and RTC customers have to typically make tradeoffs between RPO, RTO, and cost.

3 RATIONAL TEAM CONCERT ARCHITECTURE

Architecture of Rational Team Concert is based on Open Services for Lifecycle Collaboration (OSLC) to enable interoperability of tools, flexibility, and scalability. The framework, which is developed using the Jazz technology platform (a set of REST APIs), is meant to be open and customizable to enable new business models. The main components of RTC include a client interface, application server, LDAP, and database.

3.1 CLIENT INTERFACE

The client interface for Rational Team Concert can be an Eclipse-based client and/or a Web interface. The Eclipse-based client interface offers developers a rich integrated development environment for developing, building, and creating artifacts. The Web-based interface is for the users who do not need a full feature development environment and need to access RTC in order to create and manage project work items. Thus Web-based users are mostly project managers, engineering managers, and other stakeholders who are contributing or interested in the projects. Another supported client interface is the Microsoft® Visual Studio .NET Client. There is a plug-in available which will allow users to access RTC functionality from within the Visual Studio IDE. For a supported list of Eclipse and Web browser versions, refer to the IBM Support site.

3.2 RTC SERVER

The Rational Team Concert server is a Web application based on Java® that runs within any Java EE 1.4 compliant application server. Currently, RTC supports two different application servers: Apache Tomcat 5.5.23 or later or IBM WebSphere (WAS 6.1.0.23 with Fixpack 85942). Other application servers will be supported in future releases. The application server includes the management of the repository and project team process components.

PROJECT PROCESS COMPONENT

The project process component includes the guidelines, rules, and practices that a team utilizes to manage their work. Thus the project component handles and processes the requests related to project management such as creating teams, definition of procedures, processes, schedules, agile plans, and deliverables the team should follow during the project.

REPOSITORY

The repository stores the project areas for teams to access all the data specific information related to the project. Thus, it contains all the data and artifacts related to the project such as work items, processes, schedules, project teams information, iteration plans, documentations, sources, and bug tracking items. Repositories can be configured to have a single project area or multiple project areas. Depending on whether project teams would like tighter controls and access to data would determine how the repositories are configured. Usually, for a large team working on a single code base a single project area per repository is appropriate. In a group where there are multiple teams working on related projects and users may need to access each other project areas, multiple project areas per repository would be ideal.

For customers who have repositories using other tools such as ClearQuest and ClearCase, RTC has provided connectors or bridges that allow RTC projects to access data from these repositories. This allows for development teams to preserve existing infrastructures and at the same time allow flexibility to use the new features of RTC.

3.3 AUTHENTICATION

For authentication, Rational Team Concert relies on the application server's standard LDAP-based authentication for login and identity. The RTC server will map the identity of the logged-in user to an RTC contributor item and will further do permission checks. Depending on the level of security required for projects, the choice of LDAP can be limited. The supported LDAP servers for identity management include Apache Tomcat 5.5.23, IBM Tivoli Directory Server 6.0, and Microsoft Windows® Server 2003 Active Directory.

3.4 DATABASE

The repository for Rational Team Concert resides in a relational database. There are three relational database systems that are currently supported: Apache Derby 10.3, IBM DB2 9.5, Oracle Database 10g release 2, and Microsoft SQL Server® 2008 or 2005 SP2.

3.5 CONNECTORS/BRIDGES

RTC provides interoperability and integration with existing IBM tools such as ClearCase and ClearQuest. These connectors provide bidirectional synchronization between RTC and ClearCase/ClearQuest repositories. There are other integrations available with other vendors. For a list of the vendors who have integrations with RTC, refer to the IBM Rational Ensemble.

4 RATIONAL TEAM CONCERT DEPLOYMENT CONFIGURATIONS

The deployment of RTC ranges from simple to large configuration as well as setup in a cloud infrastructure. Choice of configurations depends largely on the requirements and needs of development teams.

4.1 SIMPLE CONFIGURATION

In a simple configuration, there is a Web client or Eclipse client, a single RTC Server which contains a repository and project areas, and a database to store the artifacts and data.

Figure 1) Simple RTC configuration
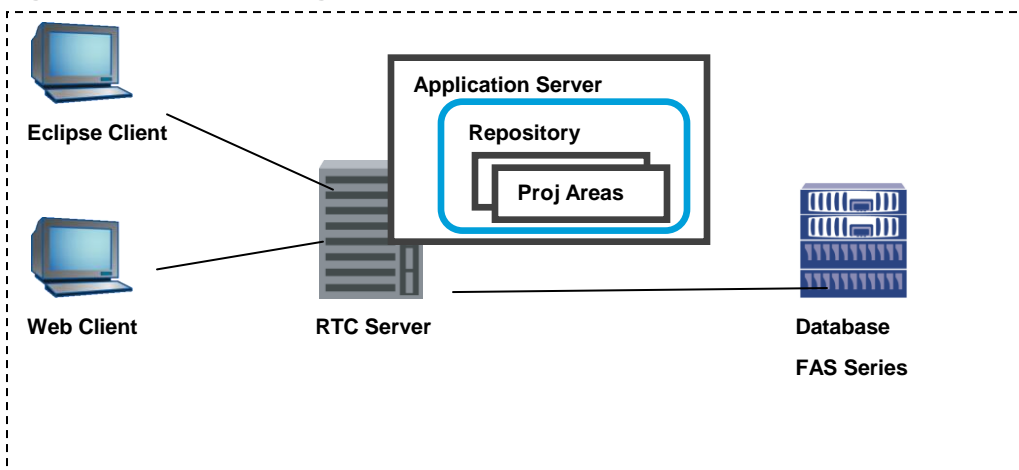
4.2   LARGE COMPLEX CONFIGURATION

For a large and complex configuration, there will be numerous Web and Eclipse clients which can be dispersed globally accessing multiple RTC servers residing on one or several servers. It is also possible that there are connectors in order to interoperate with existing repositories. The figure below shows an example of an RTC environment with connectors to a ClearCase environment.

Figure 2) Large complex RTC configuration
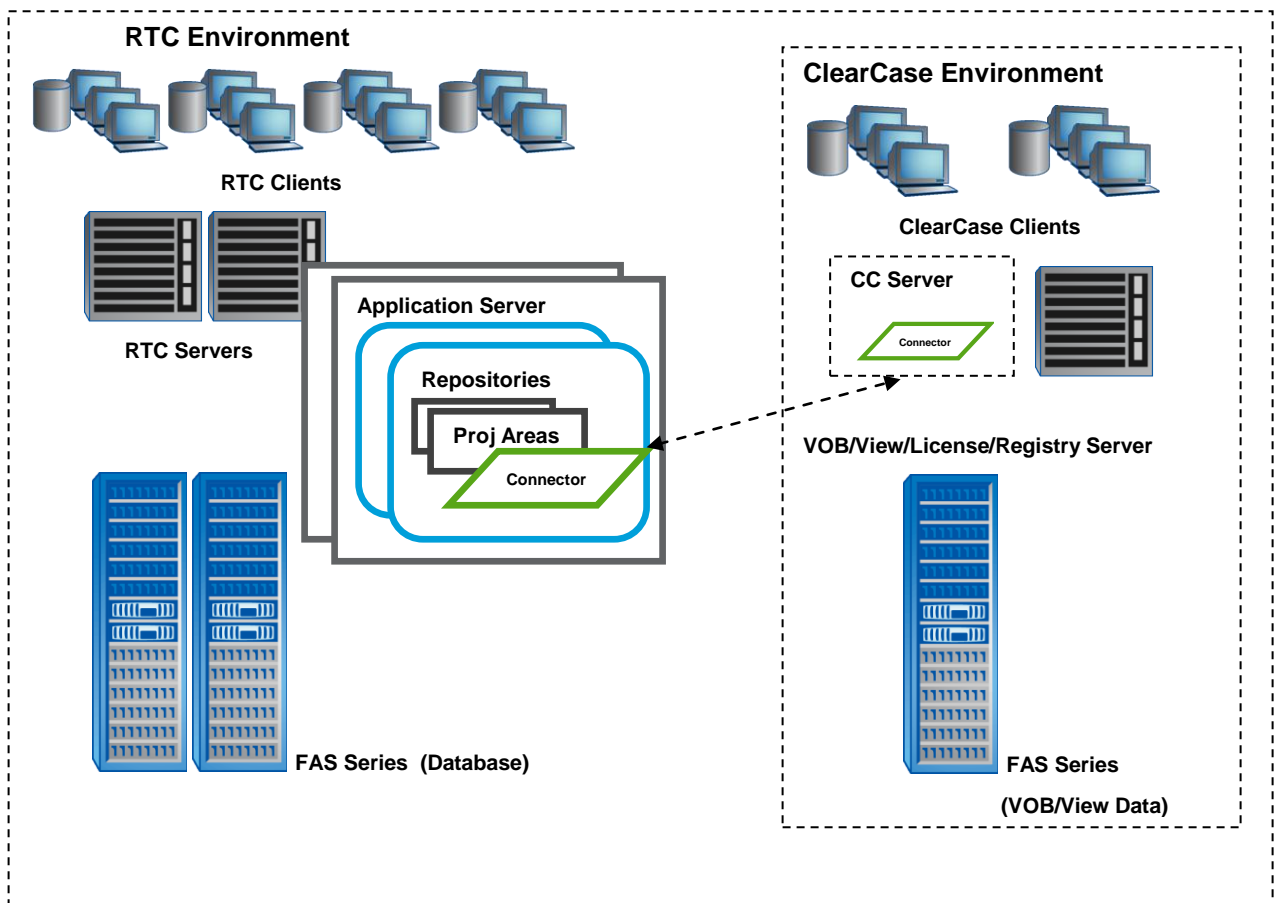
4.3   CLOUD CONFIGURATION

In a cloud environment, each team is accessing a customized, virtualized RTC environment which has been carved out of a pool of hardware resources. The cloud could be a private or public cloud. In a private cloud, the hardware resources reside in the corporate site and are customized according to requirements of development teams. In a public cloud, resources are managed and hosted by an application server provider and customers are charged based on a subscription basis.

Figure 3) RTC cloud environment

Pool of servers that hosts a myriad of virtualized RTC environment and FAS series for the database
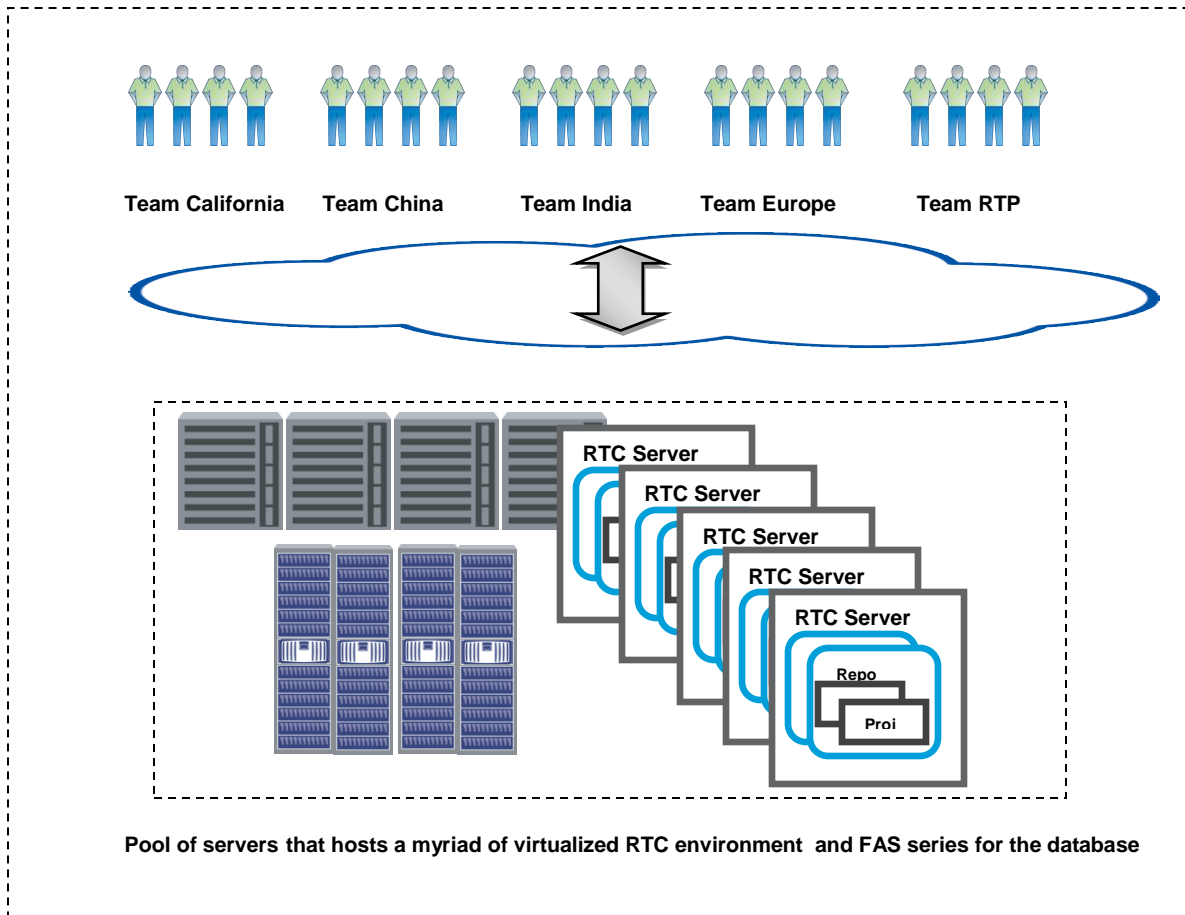
5 PERFORMANCE OF RATIONAL TEAM CONCERT ON NETAPP STORAGE

5.1 TEST DESCRIPTION

Rational Performance Tester (RPT) was utilized to conduct the tests for both standard and cloud configuration. It was used to simulate users running source control management (workspace checkouts, checkin, and builds) and project management (work item creations/deletions, subscription feeds, and content management) jobs. Several Rational Performance Tester agents are spawned to simulate active eclipse workbench and Web users to exercise the RTC servers and database components. The performance metrics generated from the tests are interpreted as the estimated number of developers, builds, feeds, and work items counts. These were totaled together to provide real number of real users that can be supported at an acceptable response time.

5.2 STANDARD ENVIRONMENT (NONCLOUD)

CONFIGURATION

- 62 users and 9 agents (5 vm Linux®, 1 vm Windows, 4 Windows)
- Intel® Nehalem Server
- FAS3170 (RAID-DP, RAID Group size of 16, Flexible volume of SAS Drives)

RESULTS

Performance testing of RTC on a standard single-tier configuration was done on FCP and NFS storage protocols. The performance differences between NFS and FCP were minimal, and thus conclusions from these tests indicate that NFS is comparable in performance to FCP in a standard RTC environment.

Table 1) Standard single-tier RTC configuration results.

Storage Protocols	Type	Counts	Totals
FCP	Developer	834.2	2989.5
	Builds	50	
	Work Items	1379.0	
	Feeds	726.3	
NFS	Developer	918.0	3055.6
	Builds	53	
	Work Items	1367.3	
	Feeds	717.3	

Figure 4) Graph of RTC results for a single-tier standard configuration.
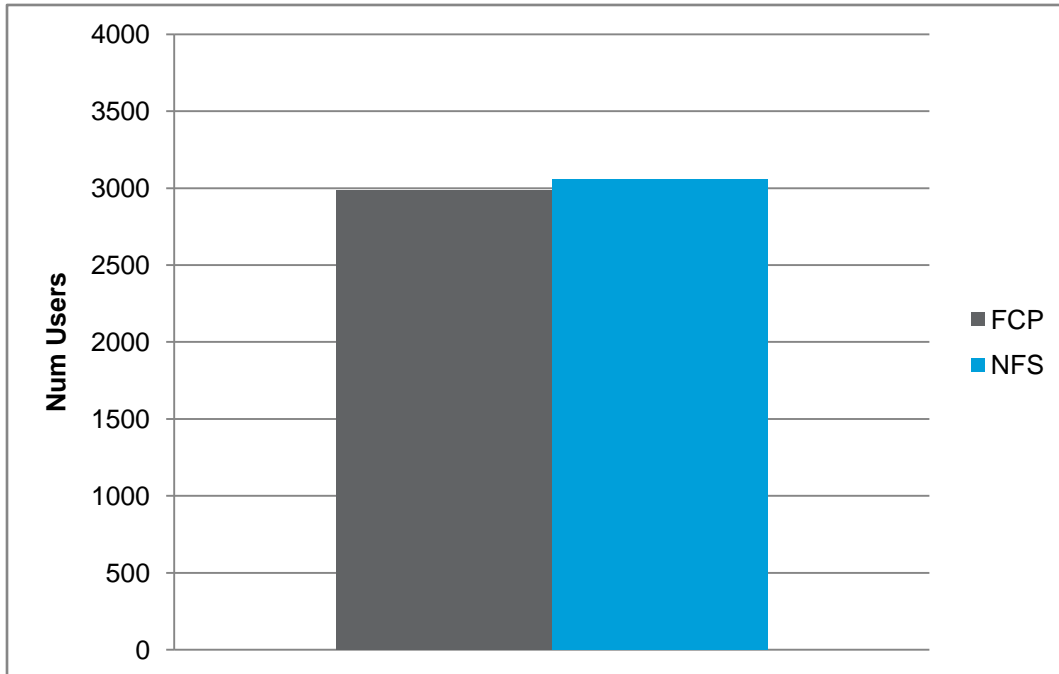
5.3 PRIVATE CLOUD

CONFIGURATION

- Server
  - Intel Westmere Nehalem-EX server 4 x 8 core CPUs and 64GB Memory
  - Hyperthreading enabled
  - Turbo mode enabled
- Virtualization with Kernel-based Virtual Machine
  - 4 – 64 bit KVM RHEL 5.5 images
    - RTC 2.0.0.2 Server (64 Bit) and DB2 9.7 – 16 virtual cores and 16GB memory
    - RTC 2.0.0.2 Client, RPT 8.1.0.2 workbench – 16 virtual cores and 16GB memory
    - RPT 8.1.0.2 agent – 16 virtual cores and 12GB memory
    - RPT 8.1.0.2 agent – 14 virtual cores and 12GB memory
- FAS3170
  - RAID-DP, RAID Group size of 16, Flexible volume of SAS/SATA drives
  - NFS, iSCSI storage protocols
 - Data ONTAP® 7.3.2

RESULTS

Performance testing was conducted on local SAS drives and NetApp's ISCSI and NFS storage protocols using SAS and SATA drives inside a private cloud environment. From these results, the best performance numbers were on the NAS devices when compared with local drives. The performance difference between NAS and local drives ranged from 34%-37%. In comparing ISCSI and NFS, NFS was 5% better than ISCSI. Also when using SAS and SATA drives, the performance is within the 3.5% difference. Thus in a private cloud, low-cost drives are sufficient to handle the workload of over 1800 users. The results for RTC private cloud are lower compared to a standard (noncloud) environment since the resources are shared as oppose to being dedicated.

Table 2) RTC cloud environment results.

Storage Protocol	Drives	Type	Counts	Totals
Local	SAS	Developers	834	1239.2
		Builds	19.5	
		Work Items	343.9	
		Feeds	41.8	
	SAS	Developer	1483.8	1880.4
		Builds	25.5	
		Work Items	342.9	

Storage Protocol	Drives	Type	Counts	Totals
ISCSI		Feeds	28.2	
	SATA	Developers	1521.2	1917.3
		Builds	24	
		Work Items	343.1	
		Feeds	28.8	
NFS	SAS	Developers	1572.2	1970.4
		Builds	26.2	
		Work Items	343.3	
		Feeds	29.0	
	SATA	Developers	1504.1	1900.8
		Builds	25	
		Work Items	342.6	
		Feeds	29.1	

Figure 5) Graph of RTC results in a cloud for local, ISCSI, and NFS.
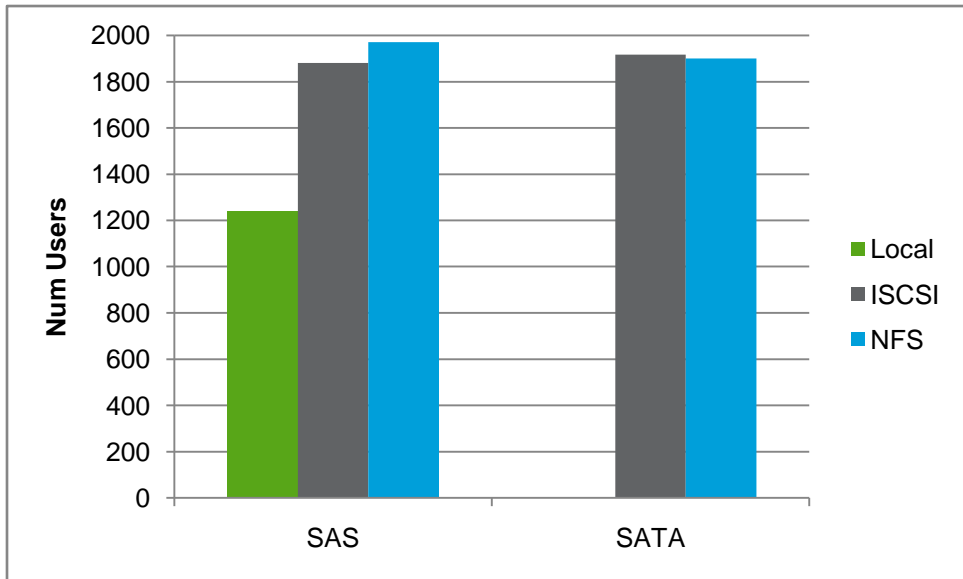
5.4 FLASH CACHE

Further testing still needs to be conducted to investigate the performance effects of Flash Cache in the RTC environments.

6 DEDUPLICATION

There were space savings observed when deduplication was run on RTC databases. It was observed to be 18% for data stored in IBM DB2 and 9% savings for data stored in an Oracle Database. These numbers can vary depending on size of repositories. Deduplication was also conducted on the virtual images used for the cloud environment testing. The images included the RTC server, IBM DB2 9.7, RTC clients and RTC agents, and the workbench. The amount of space savings observed was 61%. In a cloud environment, deduplication can benefit since there are multiple images which have duplicate or shared binaries and libraries. Also it was observed that RTC performance did not degrade when deduplication was enabled.

7 FLEXCLONE

In a software lifecycle, testing of unreleased or production code is key to make sure of the quality and robustness of the product. NetApp FlexClone technology can help facilitate testing without the need to duplicate all the storage. It can create fast, inexpensive, low-overhead copies of the RTC environment for testing purposes. IBM was able to utilize this feature while conducting performance and scalability testing of RTC. Engineers created FlexClone copies of the RTC baseline database in different stages of the testing such that if the changes that were made did not work, or if it is needed to restart from the baseline to conduct another test, it was easy to revert back to a previous working stage. Since FlexClone is easy to create and manage, performance and test engineers were able to create the FlexClone volumes without assistance from the system administrators, thus improving productivity and efficiency of engineers.

8 BEST PRACTICES

Since all the repository data of Rational Team Concert is stored in a relational database, common NetApp best practices for the specific database should be followed. A high-level view is described below; however, for more detailed descriptions refer to NetApp's best practices for IBM DB2 (TR-3272) and Oracle (TR-3633). As for the Rational Team Concert Application Server components, these can be stored inside a single flexible volume for ease of backup and recovery. There are also considerations when running in virtualized or cloud environments. Refer to NetApp's best practices for running in KVM environment (TR-3848).

8.1 DISK AND DATA LAYOUT BEST PRACTICES

- Create and lay out volumes with recovery in mind. Store system tables, user data, temporary data, and archive logs on separate volumes on a single aggregate on the storage system. This will make the recovery process easier since there is data separation. It is also advisable to place the configuration and install components of RTC Server and index in separate flexible volumes.

- Consider the number of spindles for your database. For a database with an I/O intensive workload, selection of the number of spindles can affect performance. The RAID group size defines the number of spindles available for the flexible volume within the aggregate. Setting this number too large or too small can affect overall performance and recovery of disks. Using NetApp default values would be a recommended best practice to start. To achieve results conducted in this paper, use a RAID group size of 16.

- When planning disk storage space of volumes or LUNs, reserving space for Snapshot copies must be considered.

8.2   DATABASE SETTING CONSIDERATIONS

DB2

These are some DB2 parameters that should be considered for optimum performance.

DB2_PARALLEL_IO - parallels the I/O if the number of containers in a tablespace is greater than 1 when reading and writing data in the tablespace containers.

When using a file system as opposed to raw devices, consider these variables:

- EXTENTSIZE – at creation time of database this variable specifies the number of PAGESIZE pages written to a container before going to the next container. Since NetApp's disk segment size is 256KB (32 x 4 KB blocks), use an extent size that is multiples of 256KB. Setting this variable appropriately will allow for a more efficient big-block I/O performed by prefetchers.

- PREFETCHSIZE – defines the prefetch size of a tablespace. Setting of the prefetch size reduces query response time since it will read ahead the pages of table data which will be needed for the subsequent query. Best practice would be to define a prefetch value that is large enough to parallelize use of available containers. As a rule of thumb, the prefetch size should be a multiple of the extent size used multiplied by number of data disks used. Thus with a RAID group size of 18, the prefetch size would be 16*256=4096KB.

- NO FILE SYSTEM CACHING - Disabling File System Caching parameter in the tablespace can improve performance since it will not allow for double caching (caching in the kernel file system level and database caching).

Use of the DB2 Configuration Advisor would also be helpful in further tuning the configuration parameters.

ORACLE 11G

Oracle parameters that would assist I/O performance include the following:

- Direct NFS (DNFS) - If using NFS protocol, use of Oracle DNFS client provides for a direct access to the database engine without relying on the host's operating system NFS client. This enhances performance, manageability, availability, and scalability.

- DB_FILE_MULTIBLOCK_READ_COUNT – determines the maximum number of database blocks read in one I/O operation during a full table scan. Setting this value to a multiple of the NFS READ/WRITE size (convert the NFS size in bytes to blocks) will limit the amount of fragmentation occurrences.

- DB_BLOCK_SIZE – the size of Oracle database blocks. This value should be set to be a multiple of the physical block size at the device level. For best performance, it is recommended that the DB_BLOCK_SIZE be a multiple of the OS block size. Also, if using the NFS protocol, the NFS rsize and wsize should also be a multiple of the DB_BLOCK_SIZE.

- DISK_ASYNCH_IO – controls whether I/O to data files, control files, and log files are asynchronous. If asynchronous I/O is supported on the platform, it is advisable to leave the default setting of TRUE in order parallelize I/O requests with CPU processing during table scans.

- DB_WRITER_PROCESSES – setting of this parameter parallelizes the gathering and writing of buffers. Usually used in conjunction with DISK_ASYNCH_IO set to TRUE.

8.3   RESILIENCY BEST PRACTICES

- Enable the nvfail feature for the volumes used for the database. Setting this value provides special error processing which allows for additional data protection.

- For a highly available environment, storage controllers should be clustered for failover.

8.4   CLOUD CONSIDERATIONS

- Disk Alignment – due to the abstractions between virtual and physical disks in a cloud environment, the starting block offset can be misaligned between blocks viewed at the virtual layer versus the physical layer. Misalignment can cause extra reads and writes. To prevent this latency, the starting of a file system should line up with the start of block in the next layer down. For more detailed information, refer to TR-3848, "Best Practices for KVM and Red Hat Linux on NetApp Storage."

9   BACKUP AND RECOVERY

Backups are imperative for development environments to recover from potential disasters and loss of data scenarios. For RTC, full backups as oppose to incremental backups are recommended to make sure of database integrity. There are two backup processes that can be done for RTC which are cold offline backups and/or hot online mode backups. In cold backups, the servers and databases are shut down for a period of time, whereas in hot online backup mode, the servers are placed in temporary suspend mode and the databases are quiesced to do backups. Thus, cold backups can take longer than hot backups. With the use of NetApp Snapshot technology together with NetApp SnapManager and SnapCreator solutions, the backup and recovery are fast and the administration is simple. The processes for each are described below. Refer to the Appendix for how to set up SnapCreator to do a backup of RTC database using DB2. Refer also to the References section for pointers to technical reports and administration guides for SnapManager and SnapCreator.

9.1   COLD OFFLINE BACKUPS

BACKUP

1) Stop RTC Server.
2) Stop the database.
3) Use SnapCreator for DB2 or SnapManager for Oracle to create Snapshot copies of database files and logs and the flexible volumes which contains the RTC Server Code and configurations which include the jazz.war file, the workitem index directory, the teamserver properties, the profile.ini file, the update-site directory and the conf directory.
4) Restart database. This can be done using SnapCreator for DB2 and SnapManager for Oracle.
5) Restart RTC server.

RESTORE

1) Stop RTC Server.
2) Stop the database.
3) Use SnapCreator for DB2 or SnapManager for Oracle to restore the database files and logs from a specific Snapshot copy and the flexible volumes that contain the RTC Server components and configurations.
4) Redeploy the RTC application to the Application Server.
5) Restart database.
6) Restart RTC server.

9.2   HOT ONLINE-MODE BACKUPS

BACKUP

1) Use repotools to place the RTC Server and RDF store (resource index directory and the baseline index directory) in suspend mode.

2) Use SnapCreator for DB2 and SnapManager for Oracle to quiesce database and create Snapshot copies of database files and logs as well as to create Snapshot copies of flexible volumes that has the RTC Server code and configurations (teamserver.properties, web.xml, server.xml, WorkItem Index, Profile Backups [for WAS], and so on).

3) SnapCreator will also resume the database once all Snapshot copies are created.

4) Use repotools to resume RTC Server.

RESTORE

1) Use repotools to place the RTC Server and RDF store (resource index directory and the baseline index directory) in suspend mode.

2) The restore process is an interactive process that would require user input. Use SnapCreator for DB2 and SnapManager for Oracle to suspend the database, restore the database and the flexible volumes that has the RTC Server Code and configuration files to a specific Snapshot copy. After the restore, restart the database with the write resume option.

3) Use repotools to resume RTC server.

10 DISASTER RECOVERY

SnapMirror is NetApp's disaster recovery solution. It enables data replication to improve data protection by automatically maintaining duplicate copies of data locally or remotely. To minimize storage and network bandwidth impact, SnapMirror replicates only the changed blocks from the primary storage controller after the initial synchronization. There are three configurable modes in SnapMirror:

* Asynchronous: updates from source to destination are defined in a schedule. The frequency is configurable by user.

* Semisynchronous: updates are sent to the destination as they occur; however, application does not need to wait for acknowledgement from secondary storage before continuing with the transaction.

* Fullly synchronous: updates are sent to the destination as they occur rather than on a schedule and application would need to wait from secondary storage that the write request has completed.

These modes define the frequency of the synchronization. Replicating to secondary storage allows for faster backup and recovery compared to tape backups. IBM has validated and certified NetApp SnapMirror on Rational Team Concert.

11 CONCLUSION

NetApp storage is an ideal choice to complement the flexibility and scalability design as well as ease of administration and backup and recovery of Rational Team Concert. Engineering development environments share common challenges that affect the productivity of teams and product delivery. NetApp's storage technology provides scalability, performance, storage efficiency, reduced backup and recovery windows, and disaster recovery solutions to meet and address the challenges and needs of development environments.

REFERENCES

1. "IBM DB2 on NetApp Storage: Deployment and Best Practices"
    http://www.netapp.com/library/tr/3272.pdf
2. "IBM DB2 for UNIX: Backup and Recovery Using NetApp Technologies"
    http://media.netapp.com/documents/tr-3114.pdf
3. "NetApp Best Practice Guidelines for Database 11g"
    http://media.netapp.com/documents/tr-3633.pdf
4. "Best Practices for KVM and Red Hat Enterprise Linux on NetApp Storage"
    http://media.netapp.com/documents/tr-3848.pdf
5. "SnapCreator 3.2 Installation and Administration Guide"
    http://media.netapp.com/documents/tr-3841.pdf
6. "SnapManager 3.0 for Oracle Best Practices"
    http://media.netapp.com/documents/tr-3761.pdf
7. "SnapMirror Sync and SnapMirror Semi-Sync Overview and Design Considerations"
    http://media.netapp.com/documents/tr-3326.pdf
8. "Using SnapMirror for Disaster Protection with Block Access Protocols"
    http://media.netapp.com/documents/TR-3703.pdf
9. "Backup and Restoration of Jazz Resources"
    https://jazz.net/library/content/articles/rtc/2.0.0.2/jazz-admin-
    guide/#Backup_and_Restoration_of_Jazz
10. "Jazz Foundation Server Administration"
    https://jazz.net/wiki/bin/view/Main/JFSServerAdmin#Server_Backup_Maintenance

APPENDIX

A.1   STEPS TO SET UP SNAPCREATOR FOR IBM DB2

Refer to the SnapCreator 3.2 Installation and Administration Guide for more detailed explanations.
Flexible volumes that contain the RTC Server and Configuration components can be added to the
VOLUMES parameter in step 6. Also set PRE_NTAP_CMD<#> and POST_NTAP_CMD<#> to define the
commands to place the RTC Server in quiesce mode (repotools) and to resume, respectively.

Table 3) Steps to setup SnapCreator for IBM DB2.

Steps	Commands
1. On RTC Server/DB Server unzip scServer_v3.2.zip in /usr/local/	cd /usr/local unzip scServer_v3.2.zip
2. Go to bin directory and select the correct snapcreator binary for your operating system	cd /usr/local/scServer_v3.2/bin/<osx> cp snapcreator /usr/local/scServer_v3.2

Steps	Commands
3. Run snapcreator setup from the /usr/local/scServer directory	cd /usr/local/scServer_v3.2 ./snapcreator --profile setup
4. On the NetApp storage system create user roles and set admin privileges for snapcreator	useradmin role add scrole –a login-*,api-snapshot-*,api-system-*,api-ems-*,api-snapvault-*,api-snapmirror-*,api-volume-*,api-lun-*,api-cg-* usreradmin group add scgroup –r scrole useradmin user add scuser –g scgroup Note: When you create a user for scuser, it will ask for a password; make note of password to put in the profile *.conf file in scServer
5. On RTC Server/DB Server, create a directory to store the profile	mkdir –p /usr/local/scServer_v3.2/configs/oct1lr Note: The directory name should be the same as the name first part of the profile conf file. By convention, the name of profile is name of database to back up. cp /usr/local/scServer_v3.2/configs/default/default.conf /usr/local/scServer_v3.2/configs/oct1lr/oct1lr.conf
| 6. Edit the oct1lr.conf file and set the following parameters | ```
SNAME=oct1lr
VOLUMES=qntap100:qesx202_nfs
NTAP_USERS=qesx202:scuser/<password from step 4>
NTAP_SNAPSHOT_RETENTIONS=daily:7
NTAP_SNAPSHOT_RETENTION_AGE=7
NTAP_CONSISTENCY_GROUP_SNAPSHOT=Y
APP_NAME=db2
DB2_DATABASES=oct1lr:db2inst4
DB2_CMD=/usr1/home/db2inst4/sqllib/bin/db2
``` |
| 7. Run snapcreator (this can be added to a cron job or done manually). From /usr/local/scServer_v3.2 directory run: | ./snapcreator --profile oct1lr --action snap --policy daily --verbose |

A.2 SAMPLE SNAPCREATOR OUTPUT OF IBM DB2 BACKUP

This is only a sample output of backup of database using SnapCreator.

[root@qesx202 scServer_v3.2]# ./snapcreator --profile oct1lr --action snap --policy daily --verbose

########## PRE APPLICATION QUIESCE COMMANDS ##########
[Mon Oct 4 10:10:31 2010] INFO: No commands defined

########## PRE APPLICATION QUIESCE COMMANDS FINISHED SUCCESSFULLY ##########

########## Application quiesce ##########
[Mon Oct 4 10:10:31 2010] INFO: Quiescing databases
[Mon Oct 4 10:10:31 2010] INFO: Quiescing database OCT1LR
[Mon Oct 4 10:10:32 2010] INFO: Quiescing database OCT1LR finished successfully
[Mon Oct 4 10:10:32 2010] INFO: Quiescing databases finished successfully

########## POST APPLICATION QUIESCE COMMANDS ##########
[Mon Oct 4 10:10:32 2010] INFO: No commands defined

########## POST APPLICATION QUIESCE COMMANDS FINISHED SUCCESSFULLY ##########

########## PRE NETAPP COMMANDS ##########
[Mon Oct 4 10:10:32 2010] INFO: No commands defined

########## PRE NETAPP COMMANDS FINISHED SUCCESSFULLY ##########

########## Generating Info ASUP on qntap100 ##########
[Mon Oct 4 10:10:33 2010] INFO: NetApp ASUP create on qntap100 finished successfully

########## Gathering Information for qntap100:qesx202_03102010 ##########
[Mon Oct 4 10:10:33 2010] INFO: Performing NetApp Snapshot Inventory for qesx202_03102010 on qntap100
[Mon Oct 4 10:10:33 2010] INFO: NetApp Snapshot Inventory of qesx202_03102010 on qntap100 completed Successfully

########## Running NetApp Snapshot Rename on Primary qntap100 ##########
[Mon Oct 4 10:10:33 2010] INFO: oct1lr-daily_20101004101031 is the first snapshot taken for qntap100:qesx202_03102010, Skipping!

########## Starting Consistency Group Snapshot for qntap100 ##########
[Mon Oct 4 10:10:33 2010] INFO: Starting NetApp Consistency Group Snapshot oct1lr-

daily_20101004101031 for volumes qesx202_03102010 on qntap100
[Mon Oct 4 10:10:33 2010] INFO: NetApp Consistency Group Start for qntap100 volumes qesx202_03102010 succeeded!

########## Committing Consistency Group Snapshot for all filers ##########
[Mon Oct 4 10:10:33 2010] INFO: Committing NetApp Consistency Group Snapshot for qntap100 succeeded!
[Mon Oct 4 10:10:33 2010] INFO: Creation of NetApp Consistency Group Snapshot oct1lr-daily_20101004101031 on qntap100 Completed Successfully

########## PRE APPLICATION UNQUIESCE COMMANDS ##########
[Mon Oct 4 10:10:33 2010] INFO: No commands defined

########## PRE APPLICATION UNQUIESCE COMMANDS FINISHED SUCCESSFULLY ##########

########## Application unquiesce ##########
[Mon Oct 4 10:10:33 2010] INFO: Unquiescing databases
[Mon Oct 4 10:10:33 2010] INFO: Unquiescing database OCT1LR
[Mon Oct 4 10:10:35 2010] INFO: Unquiescing database OCT1LR finished successfully
[Mon Oct 4 10:10:35 2010] INFO: Unquiescing databases finished successfully

########## POST APPLICATION UNQUIESCE COMMANDS ##########
[Mon Oct 4 10:10:35 2010] INFO: No commands defined

########## POST APPLICATION UNQUIESCE COMMANDS FINISHED SUCCESSFULLY ##########

########## ARCHIVE COMMANDS ##########
[Mon Oct 4 10:10:35 2010] INFO: No commands defined

########## ARCHIVE COMMANDS FINISHED SUCCESSFULLY ##########

########## Generating Info ASUP on qntap100 ##########
[Mon Oct 4 10:10:35 2010] INFO: NetApp ASUP create on qntap100 finished successfully

########## POST NETAPP DATA TRANSFER COMMANDS ##########
[Mon Oct 4 10:10:35 2010] INFO: No commands defined

########## POST NETAPP DATA TRANSFER COMMANDS FINISHED SUCCESSFULLY ##########

########## Running NetApp Snapshot Delete on Primary qntap100 ##########

########## POST NETAPP COMMANDS ##########
[Mon Oct 4 10:10:35 2010] INFO: No commands defined

########## POST NETAPP COMMANDS FINISHED SUCCESSFULLY ##########

########## SnapCreator finished successfully ##########
[Mon Oct 4 10:10:35 2010] INFO: INFO: SnapCreator Backup for oct1lr ACTION: snap
POLICY: daily finished successfully
[root@qesx202 scServer_v3.2]#