

An Oracle White Paper  
March 2010

# Disaster Recovery Solution for Oracle Databases Using Remote Replication Feature of Sun Storage 7000 Unified Storage System (2010.Q1)



1.0 Introduction.....	1
2.0 Remote Replication Feature of the Sun Storage 7000 Unified Storage System .....	2
2.1 Terminology and Operations.....	2
2.2 Modes of Replication .....	3
2.3 Advantages of Deploying the Remote Replication Feature .....	4
2.4 Considerations for Deploying the Remote Replication Feature	4
3.0 Replicating an Oracle Database (DB) .....	5
3.1 Replicate Full DB.....	5
3.2 Replicate Online Redo Logs .....	7
4.0 DR Scenarios for Full DB Replication .....	8
4.1 Setting Up for Full DB Replication.....	8
4.2 Full DB Replication: Failover for Testing and Backup Purposes	10
4.3 Full DB Replication: Failover Without Role-Reversal .....	11
4.4 Full DB Replication: Failover With Role-Reversal .....	12
4.5 Failback to Source Site After a Failover .....	14
5.0 DR Scenarios for Online Redo-Log Replication .....	15
5.1 Setting Up the Standby DB at the Target Using Remote Replication	15
5.2 Online Logs Replication: Failover to Standby Site .....	17
6.0 Glossary .....	17
7.0 Conclusion.....	18
8.0 For More Information .....	18

## 1.0 Introduction

Oracle's Sun Storage 7000 Unified Storage System has comprehensive data services that provide instant snapshots, cloning, rollback, and remote replication. These features can simplify database protection and maintenance tasks for system and database administrators.

This paper shows how to use Sun Storage 7000 Unified Storage System remote replication with Oracle databases and how to perform disaster recovery (DR) at the target site. The article is intended for Oracle database administrators, system administrators, and anyone who wants to implement replication techniques with an Oracle database. This paper is based on version 2010.Q1 software.

This paper covers three topics:

- Overview and capabilities of the remote replication feature
- DR scenarios for full database replication
- DR scenarios for online redo-log replication

## 2.0 Remote Replication Feature of the Sun Storage 7000 Unified Storage System

The remote replication feature asynchronously streams the data to the remote storage appliance. The data being streamed is modified at the source site at the granularity of the storage transaction in a project or in a share. In earlier versions of the software, only project-level replication was allowed.

The initial replication requires replicating all the data of either the project or the share, depending on the replication setup. For subsequent replications, only the differential data between the initial replication and the current point in time is replicated.

The data can be encrypted using Secure Sockets Layer (SSL) and transmitted. The replication can be configured to replicate either over a dedicated line or over a public network. The following are some highlights and capabilities of the remote replication feature.

- If the data is replicated at the project level, all the shares, snapshots, and clones that reside in the project are replicated.
- For flexibility, replication can also be done by selecting a specific share.
- The data is consistent at the source every time the replication is initiated.
- Write ordering is preserved across the shares within the project at the target site, which is critical for databases.
- Multiple projects from one source can be replicated to the same or multiple targets simultaneously.
- The replication supports one-to-one, one-to-many, and many-to-one replication topologies.
- The replication doesn't require a dedicated network for replication, although it is recommended to replicate on a private link between the storage systems.
- Replication can be done across different Sun Storage 7000 Unified Storage System platforms.

### 2.1 Terminology and Operations

TABLE 1. TERMINOLOGY AND OPERATIONS	
TERMINOLOGY	DESCRIPTION
Source	The primary (production) site of the replication project. The activities around replication are done only at the project level. In the browser user interface (BUI), click Shares → Projects → [click on the project] → Replication.
Target	The receiving site of the replication. A target can receive one or more projects from one or more storage systems. The

	replication information at the target is maintained at the system level instead of the project level.
Replica or Package	The replicated copy of the project at the target site. Read-only option of the package is available with 2010.Q1. However, in order to access the replica for read/write purposes, the package needs to be cloned.
Clone or Export Package/Replica	The process to access the replica at the target. A new project is created. All the shares, snapshots, clones, and so on, are accessible under the cloned project.
Role Reversal	The direction of the replication is reversed from source -> target to target -> source for a replica. This transfers the package into a local project at the target and sets up a replication back to the original source. This is the recommended mode for the failover process.
Destroy Local Replica	This procedure creates a new local project out of the package and disables the replication. All clones associated with the target instantly become independent projects by themselves.
Add Target	The replication process is established between the source and that target. More than one target can be defined at the source. The root credential is required to establish a target.
Delete Target	Target is deleted at the source. This operation destroys any established replication between the source and the target.

## 2.2 Modes of Replication

The Sun Storage 7000 Unified Storage System supports two modes of replication: Scheduled and Continuous. In both the modes, the underlying architecture is similar and the replication occurs in asynchronous mode. It is necessary to place all the application data that requires write ordering in a single project. The mode of replication from one mode to another can be changed any time.

### 2.2.1 Scheduled Replication

In this mode, you have the choice to either define a schedule or leave the schedule undefined. If a schedule is established, the replication occurs at the defined schedule. If no schedule is defined, the replication occurs only when you manually initiate replication. In the manual mode, you initiate replication either by clicking the Update button on the browser user interface (BUI) or through a cron job using the command-line interface (CLI).

In either mode, the target site can be refreshed any time by performing a manual update using either the BUI or the CLI.

### 2.2.2 Continuous Replication

In this mode, the replication happens continuously without any user interference. As soon as the replica from the source is fully received at the target, the next replication starts. This is not a standard synchronous mode of replication where a two-phase commit occurs. This mode is deployed when the target site is expected to be as synchronous as possible with the source. As soon as the continuous mode is established, the replication starts immediately.

## 2.3 Advantages of Deploying the Remote Replication Feature

The remote replication feature provides the following advantages:

- The replica at the target site is cloned to be used for test, development, backup, and various other purposes.
- Any number of clones can be created out of the package. When the replica is exported, the exported clone will have the point-in-time image of the package that was last successfully received. In this way, multiple clones can be created from different points-in-time of the same replica.
- These clones and the ongoing replication are independent of each other, providing greater flexibility.
- A read-only version of the replicated package allows you to view and back up the content without a need for cloning.
- Snapshots can be taken on the exported clones. This makes the target site even more flexible for various purposes—including backup, test/development, and so on—without impacting the production site.
- The exported clone (project) can assume independent project status by deleting the underlying replica. That project can then be used as a source and can be replicated to a third site.
- Since only the last successful replica is made available for cloning, the target site can provide a consistent copy at all times.

## 2.4 Considerations for Deploying the Remote Replication Feature

Keep the following consideration in mind when deploying the remote replication feature:

- The first replication replicates the entire project, but subsequent updates replicate only the incremental changes. Proper planning is required in terms of how long it will take to establish the initial replication.
- The storage system supports only asynchronous mode of replication. The continuous mode of replication shouldn't be confused with synchronous replication.

- Business processes, such as Recovery Time Objective (RTO), Recovery Point Objective (RPO), and Service Level Agreement (SLA), should be considered in deciding the mode of replication. The rate of change, latency, bandwidth, number of projects to replicate, and so on, all influence the decision-making process. It is possible to change from one mode to another.
- The target site is not verified for the space requirement when the replication is established. Before initiating the replication, verification is necessary to ensure the target site has enough storage space to receive the replica.
- If the target site is used to take a backup of an Oracle database, the replica is cloned and then the database is opened. Then the database is either shut down or placed in hot backup mode before taking a snapshot for backup purposes.
- If the target site is used for production failover, it is strongly recommended to use Role Reversal instead of regular export of the package. This allows a faster synchronization back to the original source.

## 3.0 Replicating an Oracle Database (DB)

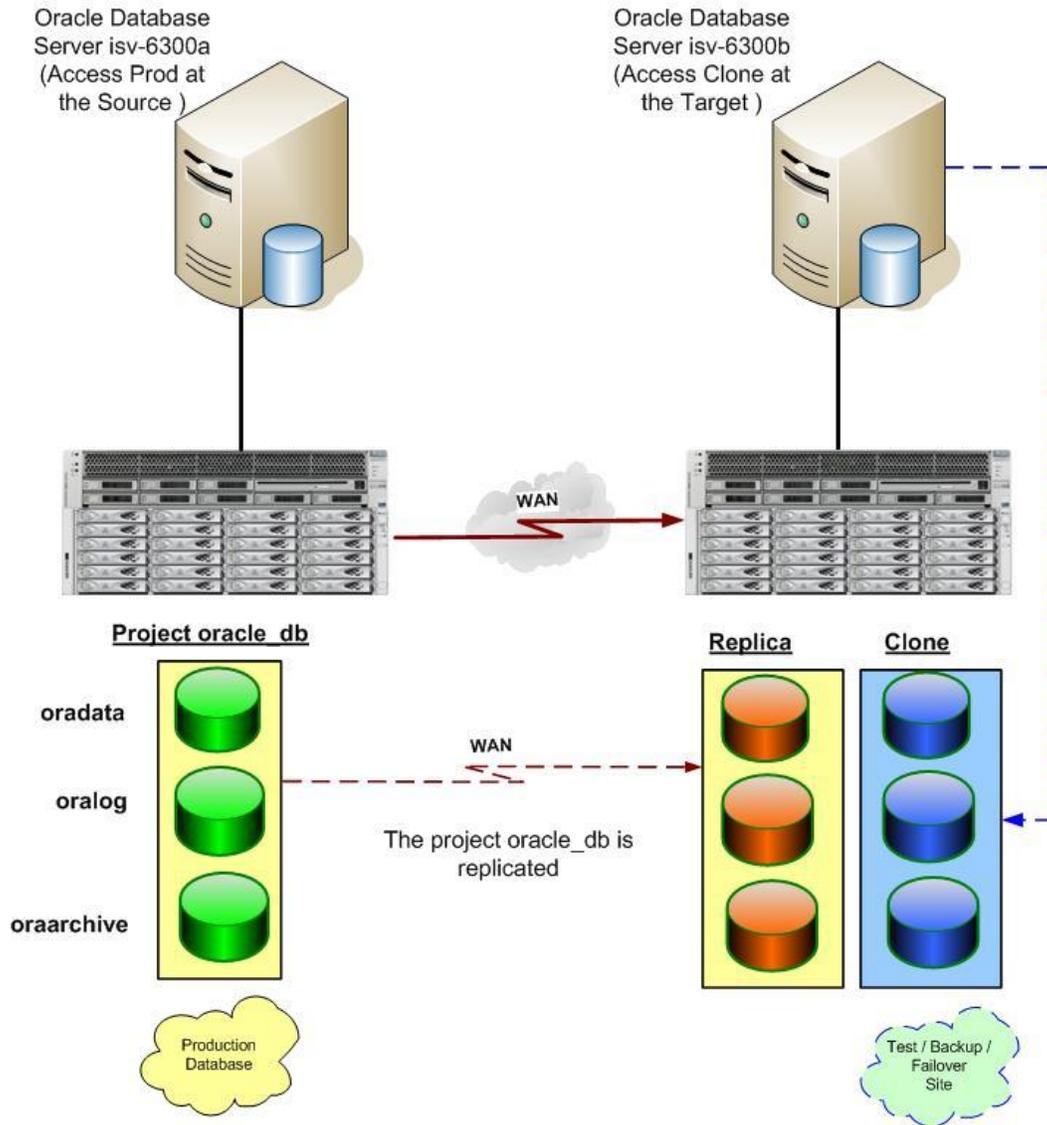
For deploying a DR solution with Oracle databases, either the entire database is replicated or only online redo logs are replicated.

### 3.1 Replicate Full DB

In this mode, all the physical files of the database that are stored in the project are replicated. Either continuous or scheduled replication is used, depending on your requirements.

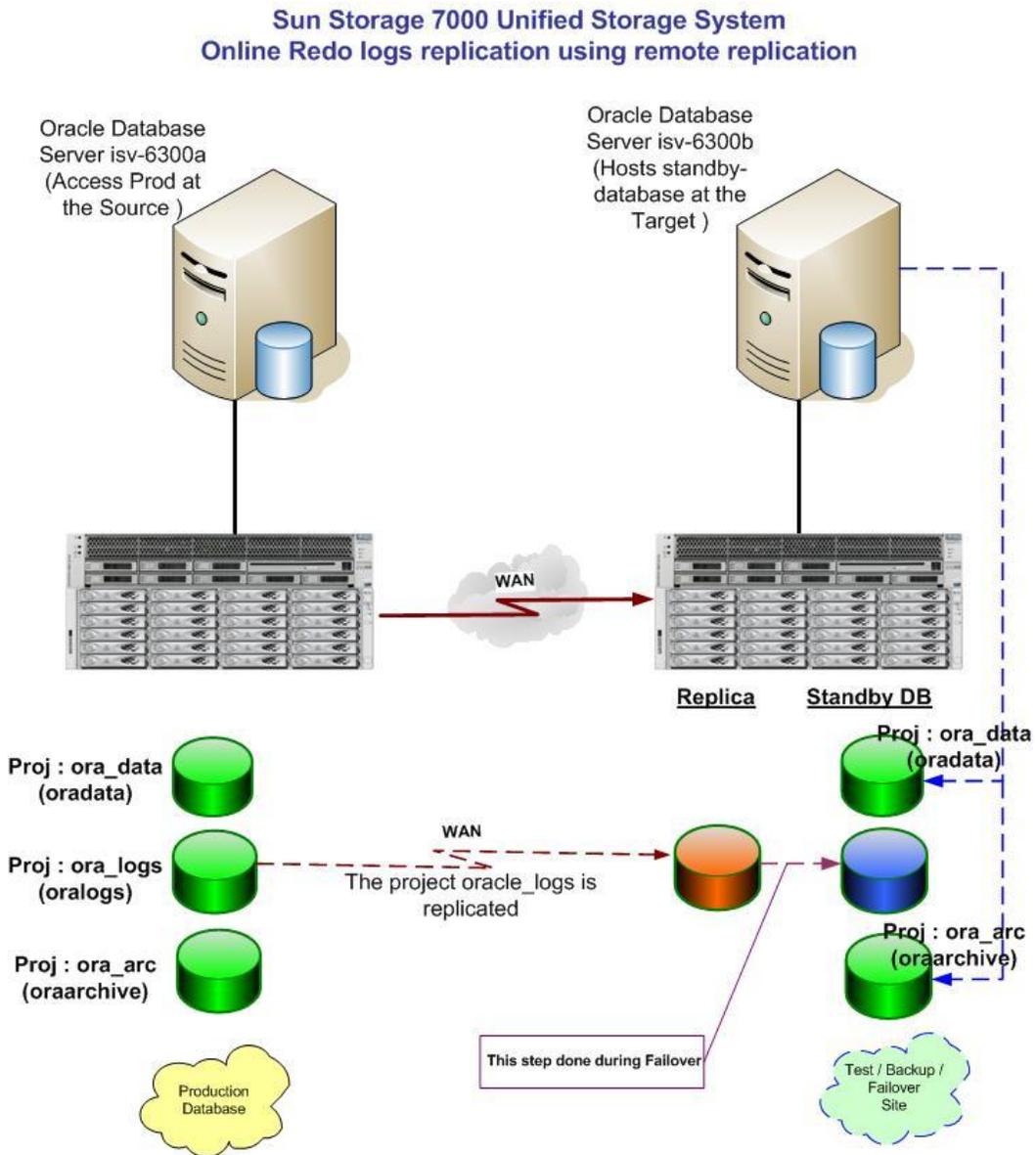
The replica at the target site is cloned and then used for any read/write or backup purposes. The target site has the database copy as of the time of the last successful replication, and the database is started which, in turn, causes an instance recovery to be performed. The database can then be used for various purposes.

### Sun Storage 7000 Unified Storage System Full database replication using remote replication



### 3.2 Replicate Online Redo Logs

This mode requires a standby database to be maintained at the target site. At the source site, the Oracle database is stored in two different projects. One project is used to store the data files, control files, and, optionally, a multiplexed copy of online redo logs. The other project contains the online redo logs only, and only this project is replicated. The archived logs are shipped to the standby site using the Oracle log shipping method. The standby database is created at the target by initially replicating both the projects.



## 4.0 DR Scenarios for Full DB Replication

In this mode of replication, the entire database is replicated to the target site.

Configuration details:

- Source site: *isv-USS-1* (also called *isv-7210b* in figures)
- Target site: *isv-USS-2* (also called *isv-7310b* in figures)
- Project: *oracle\_sparc\_smalldb*
- Filesystems (shares): *oradata, oralog, oraarchive*
- Database server (client at Source): *isv-6300b*
- Database server (client at Target): *isv-6300b*
- Mount points at the database server: */oradata/data, /oradata/logs, /oradata/archive*

### 4.1 Setting Up for Full DB Replication

Perform the following steps at the source site:

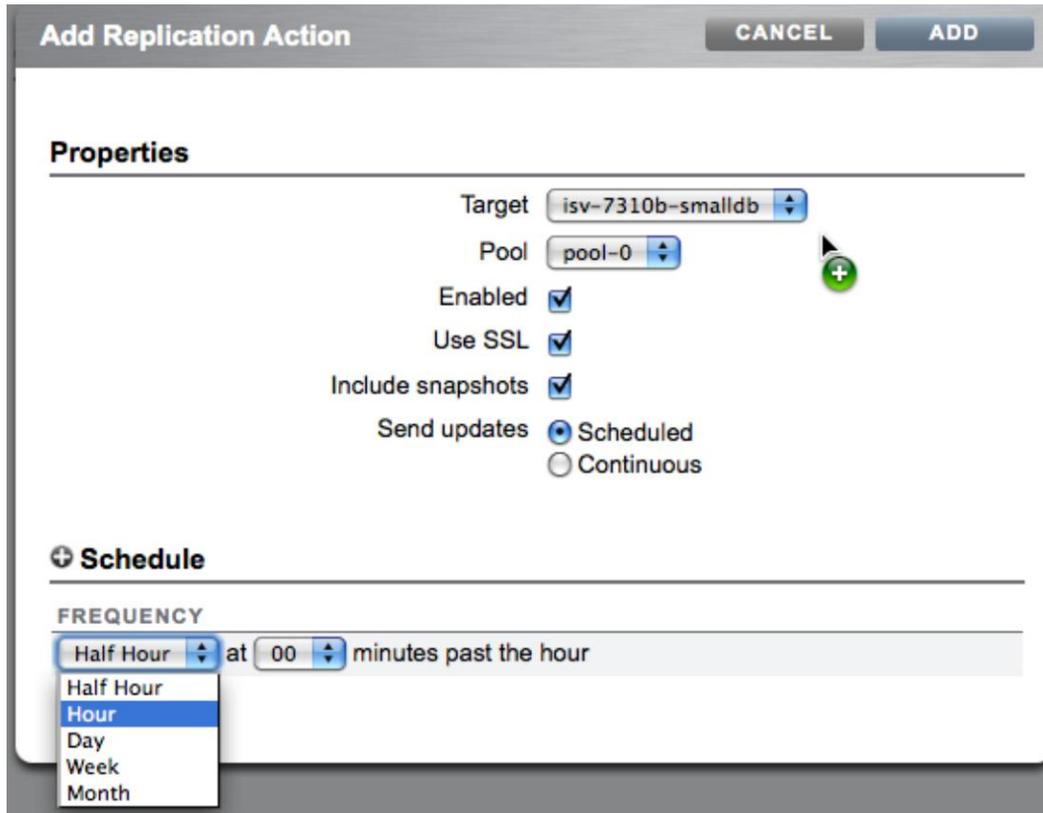
1. From the database server, shut down the database cleanly:

```
SQL> SHUTDOWN IMMEDIATE;
```

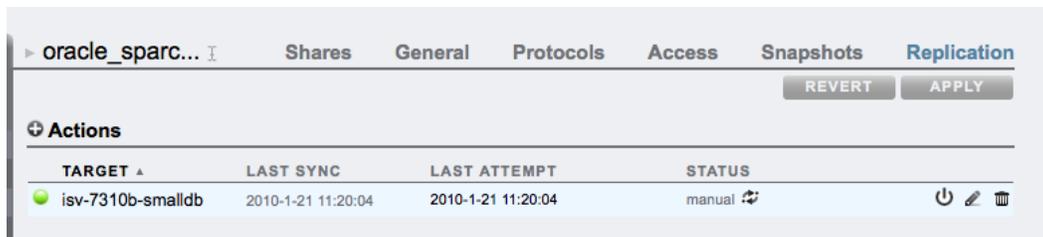
2. Create the replication from the BUI by clicking the following:

Shares → Projects → *oracle\_sparc\_smalldb* → Replication

3. Click (+) Target and provide the target site's name, for example, *isv-USS-2*, and provide the root password. Choose the Pool on the target where the replica will be created. Choose either Continuous or Scheduled. If you select Scheduled, click (+) next to Schedule if there is any specific schedule you want to use to perform the replication. Click Apply.



If you choose Continuous replication, the replication starts immediately. Otherwise, depending on the schedule, the replication starts at a later time or you can start a manual update by clicking the Restart sign in the STATUS column.



4. Start up the database at the source:

```
SQL> STARTUP PFILE=<init param>
```

Once the replication for the project is complete, the target site's replica can be used for test, backup, and other purposes. Refer to the appropriate failover section for more details.

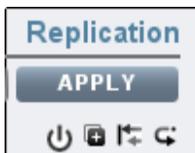
**Note:** The target site's replication information shows the time stamp of the last successfully received package so you can understand the time lag between the sites.

## 4.2 Full DB Replication: Failover for Testing and Backup Purposes

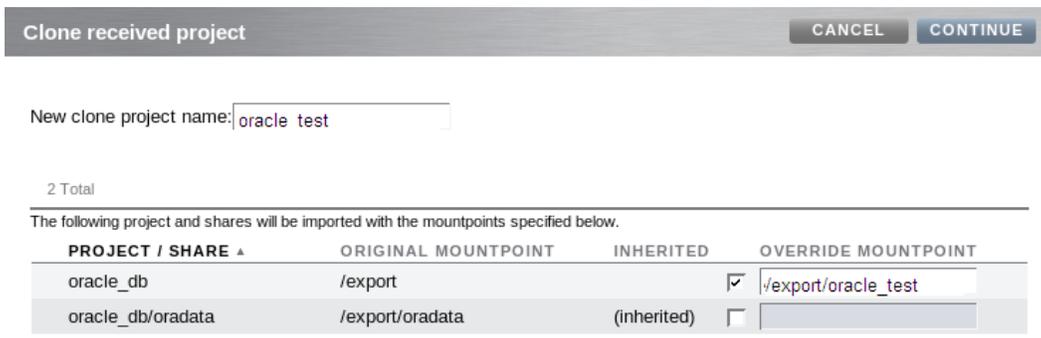
This scenario is for deploying an architecture where the target site is to be used for backup, test, and other development purposes without interrupting the ongoing replication. The prerequisite is that the steps specified in section "4.1 Setting Up for Full DB Replication" are complete and the replica is available at the target site.

At the target site, perform the following steps:

1. In the BUI, click Shares and then Replication. Look for and then click the name of the source (for example, `isv-USS-1`) and the project name.
2. Click the (+) Export the Clone icon (below the Apply button next to the Power icon), which opens a pop-up.



3. Provide a new project name, `oracle_test`. The BUI shows a few lines of information for the shares, with the first line being the project itself and the rest for the shares. In the first project line, override the mount point and input `/export/oracle_test/`. Click Apply.



The shares are now accessible by the client system. The mount points are as follows:

```
isv-USS-2:/export/oracle_test/oradata
```

```
isv-USS-2:/export/oracle_test/oralogs
```

```
isv-USS-2:/export/oracle_test/oraarchive
```

4. On the database server, `isv-6300b`, mount those Network File System (NFS) shares under the directories `/oradata/data`, `/oradata/logs`, and `/oradata/archive`, respectively. Make sure the correct NFS mount options are specified:

```
rw,bg,hard,nointr,rsiz=32768,wsiz=32768,proto=tcp,
```

```
vers=3,noac,forcedirectio
```

5. Start up the database:

```
SQL> STARTUP PFILE=<init*.ora>
```

The database can now be used for test purposes.

6. After the testing is complete, shut down the database and unmount the shares from the `isv-6300b` system.

7. Destroy the clone project by clicking Shares → Projects → `oracle_test` → Destroy.

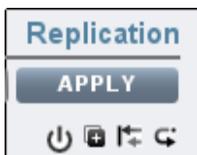
Note that while this clone is accessing the replica, the subsequent replica updates do not impact it. In other words, the data is not refreshed.

### 4.3 Full DB Replication: Failover Without Role-Reversal

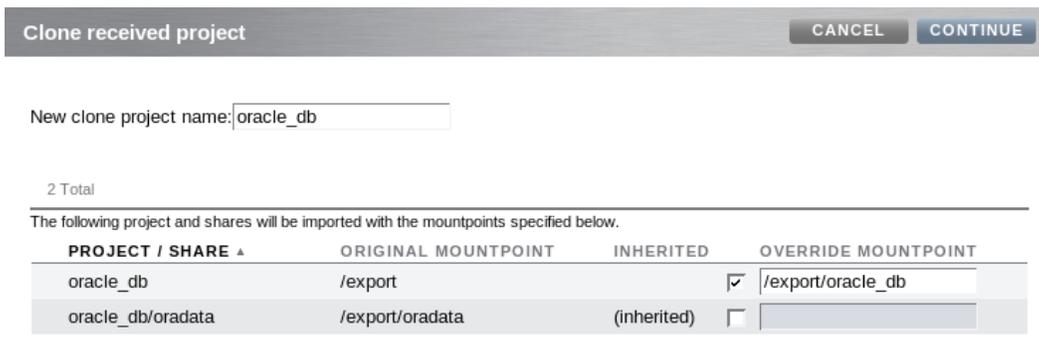
In this scenario, the source site is inaccessible and the target site becomes a production site. The prerequisite for establishing the full database replication is explained in section "4.1 Setting Up for Full DB Replication" and a copy of the replica is available at the target for the failover.

At the target site, perform the following steps:

1. In the BUI, click Shares and then Replication. Look for and then click the name of the source (for example, `isv-USS-1`) and the project name.
2. Click the (+) Export the Clone icon (next to the Power icon), which opens a pop-up.



3. Provide a new project name, `oracle_db`. Click Apply. The project inherits all the properties based on the original project from the source.



4. Select the “Destroy the local replica and the source project” or the “Destroy the local replica” option. This step makes `oracle_db` an independent project at the target site.

The shares are now accessible by the client system. The mount points are as follows:

```
isv-USS-2:/export/oracle_db/oradata
```

```
isv-USS-2:/export/oracle_db/oralogs
```

```
isv-USS-2:/export/oracle_db/oraarchive
```

5. On the database server, `isv-6300b`, mount those NFS shares under the directories `/oradata/data`, `/oradata/logs`, and `/oradata/archive`.

6. On `isv-6300b`, start the database and open the database. Instance recovery is automatically done and the database is then active.

With that step, the failover is complete, but note that there is no replication established for this project.

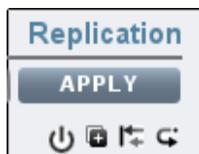
7. (Optional) Once the source site (`isv-USS-1`) is accessible again, to make that the target for this project, follow the steps provided in the section "4.1 Setting Up for Full DB Replication."

#### 4.4 Full DB Replication: Failover With Role-Reversal

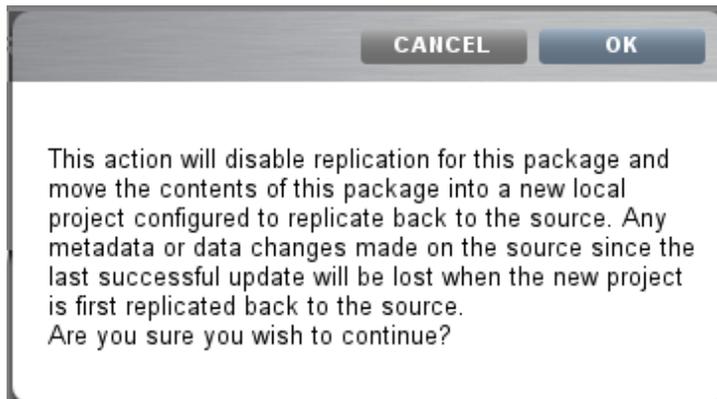
In this scenario, both the source and target are accessible but the roles of the source and target for the project are reversed. The target site becomes the active production site and the source site becomes the new target site. The steps described in this section could be performed in a DR situation where the roles need to be reversed for a quick failover and for faster failback. The prerequisites are that the steps described in section "4.1 Setting Up for Full DB Replication" have been completed, and the target site has the replica on which the failover occurs.

At the target site, perform the following steps:

1. In the BUI, click Shares and then Replication. Look for and then click the name of the source (for example, `isv-USS-1`) and the project name (`oracle_db`).
2. Click the Reverse Replication icon (the fourth icon below the Apply button), which opens a pop-up.



3. Provide a new project name, `oracle_db`. Click Apply. A warning is displayed:



4. Click OK. The following actions occur automatically:

- A clone is created with the project name. It inherits all the properties associated with the original project.
- The local replica is destroyed.
- The source and target destinations are reversed for this project.
- The replication from the new source to the new target is established without the need for full reverse updates. (However, the system will not perform a replication if replication is set to be done manually, and the replication will show up on the source only when a manual replication is started.)

The shares are now accessible by the client system. The mount points are as follows:

```
isv-USS-2:/export/oracle_db/oradata  
isv-USS-2:/export/oracle_db/oralogs  
isv-USS-2:/export/oracle_db/oraarchive
```

5. In the database server, `isv-6300b`, mount those NFS shares under the directories `/oradata/data`, `/oradata/logs`, and `/oradata/archive`.

The database can now be opened for read/write access.

The full replication is initiated implicitly.

With that, the role-reversal for the database is complete.

## 4.5 Failback to Source Site After a Failover

In this scenario, the source site becomes accessible again. The data from the target site should be replicated back to the source and then the replication between this source and target needs to be established. The prerequisites are that the steps in section "4.3 Full DB Replication: Failover Without Role-Reversal" and section "4.4 Full DB Replication: Failover With Role-Reversal" have already been done. This means the replication is already established between the new source (old target) and the new target (old source). The objective is to make the old source the source again and to make the old target the target.

Once the old source (new target) site `isv-USS-1` is updated with the latest replica, perform the following steps.

At the target site:

1. Shut down the database.

At the source site:

1. Destroy or rename the project `oracle_db` if it already exists.
2. In the BUI, click Shares and then Replication. Look for and then click the name of the source (for example, `isv-USS-2`) and the project name (`oracle_db`).
3. Click the Reverse Replication button, which opens up a pop-up.
4. Provide a new project name, `oracle_db`. Click Apply. The project inherits all the properties based on the original project.

With that step, the shares are now accessible by the client system. The mount points are as follows:

```
isv-USS-1:/export/oracle_db/oradata
```

```
isv-USS-1:/export/oracle_db/oralogs
```

```
isv-USS-1:/export/oracle_db/oraarchive
```

5. On the database server, `isv-6300a`, mount those NFS shares under the directories `/oradata/data`, `/oradata/logs`, and `/oradata/archive`.

The database can now be opened for read/write access. Since the full replication is done, it doesn't require any recovery.

With that step, the replication is established between `isv-USS-1` and `isv-USS-2`.

The failback to the original source for the database is complete.

## 5.0 DR Scenarios for Online Redo-Log Replication

In cases where a standby database is maintained at the target site, the archived redo logs are shipped by the primary Oracle instance itself. However, the unarchived latest transactions are still stored in online redo logs. In order for the standby database to apply all the latest transactions when it is promoted to be the production database, the standby database requires access to the online redo logs.

For replicating the online redo-logs only, the Oracle database is stored in two different projects in the Sun Storage 7000 Unified Storage System. In one project, the control files, database files, and archived logs are stored. The other project is used to store online redo logs. The continuous mode of replication is preferred to capture the latest online redo logs.

Configuration details:

- Source site: `isv-USS-1`
- Target (standby) site: `isv-USS-2`
- Project (non-replicated): `oracle_db` (control files, data files, archived logs)
- Project (replicated): `oracle_logs` (online redo logs)
- Filesystems (database shares): `oradata`, `oralog`, `oraarchive`
- Database server (client at Source): `isv-6300a`
- Database server (client at Target): `isv-6300b`
- Mount points at the DB server: `/oradata/data`, `/oradata/logs`,  
`/oradata/archive`

### 5.1 Setting Up the Standby DB at the Target Using Remote Replication

The standby database can be created by either taking a cold backup of the database and restoring it at the target site, or using the remote-replication feature for full database replication and converting the target to a standby database. This procedure explains the second scenario.

The following steps assume that the production database is up and running at the source site.

At the source site, perform the following steps:

1. Create a standby control file:

```
SQL> ALTER DATABASE CREATE STANDBY CONTROLFILE AS  
'/oradata/data/standby.ctl';
```

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE;
```

2. Perform the steps specified in section “4.1 Setting Up for Full DB Replication,” which will also replicate the standby control file, `standby.ctl`.

At the target site:

1. In the BUI, click Shares and then Replication. Look for and then click the name of the source (for example, `isv-USS-1`) and the project name (`oracle_db`).
2. Click the (+) Export the Clone icon, which opens up a pop-up.
3. Provide a new project name, `oracle_stby`. Click Apply. The project inherits all the properties based on the original project.
4. Perform the “Destroy the local replica” operation. This step makes `oracle_stby` an independent project at the target site, which has the data files and the archived logs.

With that step, the shares are now accessible by the client system. The mount points are as follows:

```
isv-USS-2:/export/oracle_stby/oradata
```

```
isv-USS-2:/export/oracle_stby/oraarchive
```

5. On the database server, `isv-6300b`, mount those NFS shares under the directories `/oradata/data` and `/oradata/archive`.

6. Rename `standby.ctl` to `control1.ctl` and also make a copy called `control2.ctl`.

7. Set `ORACLE_SID` to `stby`, and set `listener.ora` to reflect the instance name. Also make sure the control file information is correct.

8. Do the following to activate the standby database:

```
SQL> STARTUP MOUNT;
```

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
```

At the source site, perform the following steps:

1. Modify `init<sid>.ora` to reflect the second archive destination for the standby.

```
log_archive_dest_2 = 'SERVICE=<standby_instance> LGWR ASYNC  
valid_for=(ONLINE_LOGFILES,PRIMARY_ROLE) db_unique_name=<db_name>'
```

2. Shut down and restart the database. Verify in `alert.log` that the archived logs are being shipped to the target site.

From here on, the standby database keeps applying the archived logs received from the source site. However, the latest updates that are captured in the online redo logs are not updated yet to the standby database. Refer to the next section (“5.2 Online Logs Replication: Failover to Standby Site”) failover process to make the replicated online redo logs accessible.

## 5.2 Online Logs Replication: Failover to Standby Site

Following are the steps involved in the failover to the standby site.

At the target site, perform the following steps:

1. Under Shares → Replication → <source\_name> (for example, isv-USS-1), make sure the ora\_logs project was replicated.
2. Click (+) to clone and provide the project name ora\_logs. Update the project default mount path to /export/ora\_logs.
3. From the system isv-6300b, mount isv-USS-2:/export/ora\_logs under /oradata/logs.
4. Take a snapshot of ora\_db, just as a precautionary measure.
5. Activate the standby database:

```
SQL> ALTER DATABASE ACTIVATE STANDBY DATABASE;
```

```
SQL> RECOVER AUTOMATIC DATABASE;
```

```
SQL> ALTER DATABASE OPEN;
```

At this point, the target site hosts the regular database and can be used as a production site for backup purposes, for test and data verification purposes, and so on.

## 6.0 Glossary

The following table contains common abbreviations and acronyms that are used in this document.

TABLE 2. GLOSSARY	
ABBREVIATION	EXPLANATION
BUI	Browser User Interface
CLI	Command Line Interface
DR	Disaster Recovery
NFS	Network File System
RPO	Recovery Point Objective
RTO	Recovery Time Objective
SLA	Service Level Agreement
SSL	Secure Sockets Layer

## 7.0 Conclusion

This paper described how to use Oracle's Sun Storage 7000 Unified Storage System (version 2010.Q1) for remote replication with Oracle databases and how to perform disaster recovery for full database replication and for online redo-log replication.

## 8.0 For More Information

Here are additional Sun resources.

Storage resources:

- Sun Storage 7000 Unified Storage Systems web site:  
[http://www.sun.com/storage/disk\\_systems/unified\\_storage/](http://www.sun.com/storage/disk_systems/unified_storage/)
- Sun Disk Storage web page: <http://www.sun.com/storagetek/open.jsp>
- Sun Storage Solutions web page: <http://www.sun.com/storagetek/solutions.jsp>
- Discussions, such as the Storage forums  
(<http://forum.java.sun.com/category.jspa?categoryID=66>) and the Sun Hardware - Servers forums  
(<http://forums.sun.com/forum.jspa?forumID=830>)
- Wikis, such as the Storage Administration wiki  
(<http://wikis.sun.com/display/StorageAdmin/Home>) and the BigAdmin Storage Tech Tips wiki  
(<http://wikis.sun.com/display/BigAdmin/Storage+Tech+Tips>)
- Storage Stop Blog: <http://blogs.sun.com/storage>
- Resources on BigAdmin, such as the Storage resource collection (includes community submissions): <http://www.sun.com/bigadmin/collections/storage.jsp>

General links:

- Sun download site: <http://www.sun.com/download/>
- Oracle University web site: <http://www.sun.com/training/>
- Discussions, such as Sun forums (<http://forums.sun.com/index.jspa>) and the BigAdmin Discussions collection (<http://www.sun.com/bigadmin/discussions/>)
- Sun product documentation at <http://docs.sun.com> and the Sun Documentation Center  
(<http://www.sun.com/documentation/>)
- Sun wikis, such as the Sun BluePrints wiki  
(<http://wikis.sun.com/display/BluePrints/Main>) and the BigAdmin wiki  
(<http://wikis.sun.com/display/BigAdmin/Home>)

- Support:
- Sun resources:
  - Register your gear: <https://inventory.sun.com/inventory/>
  - Sun Services: <http://www.sun.com/service/index.jsp>
  - SunSolve Online: <http://sunsolve.sun.com>
- Community system administration experts:  
<http://www.sun.com/bigadmin/content/communityexperts/>



Disaster Recovery Solution for Oracle  
Databases Using Remote Replication Feature of  
Sun Storage 7000 Unified Storage System  
(2010.Q1)  
March 2010  
Authors: Jignesh Shah and Sridhar  
Ranganathan

Oracle Corporation  
World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065  
U.S.A.

Worldwide Inquiries:  
Phone: +1.650.506.7000  
Fax: +1.650.506.7200  
oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2010, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.