Siebel Maximum Availability Architecture

Oracle Maximum Availability Architecture White Paper
October 2008

Maximum Availability Architecture

Oracle Best Practices for High Availability

ORACLE

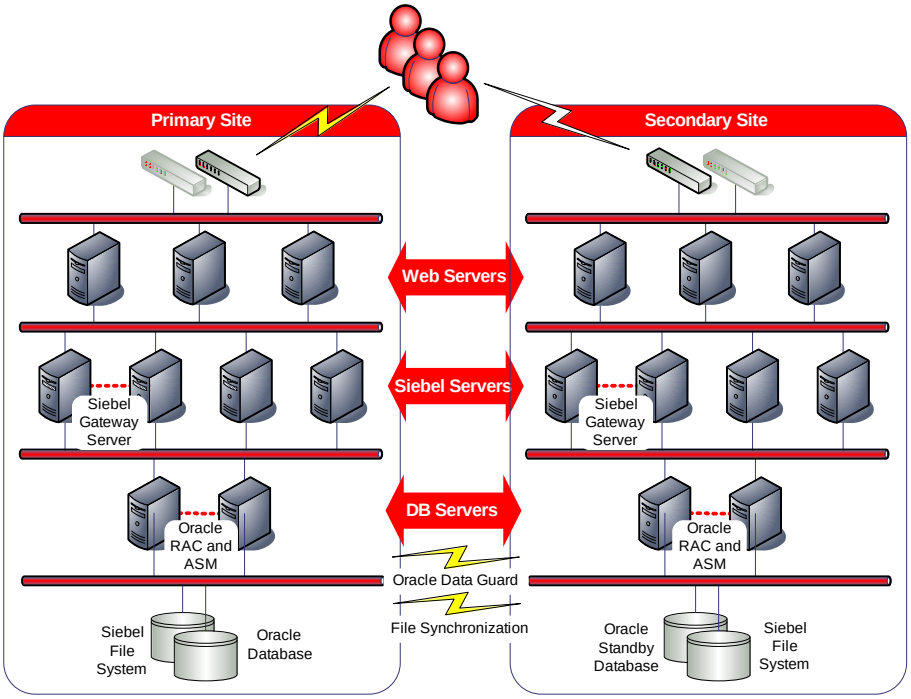
Siebel Maximum Availability Architecture

Siebel Maximum Availability Architecture

EXECUTIVE SUMMARY

Oracle Maximum Availability Architecture (MAA) is Oracle's best practices blueprint based on proven Oracle high availability technologies and recommendations. The goal of MAA is to achieve the optimal high availability architecture at the lowest cost and complexity. Papers are published on the Oracle Technology Network (OTN) - http://www.oracle.com/technology/deploy/availability/htdocs/maa.htm.

This paper describes the Siebel Maximum Availability Architecture (MAA), a best practice blueprint for achieving an optimal Siebel high availability deployment using Oracle high availability technologies and recommendations.



To achieve maximal Siebel availability, we make the following high-level recommendations:

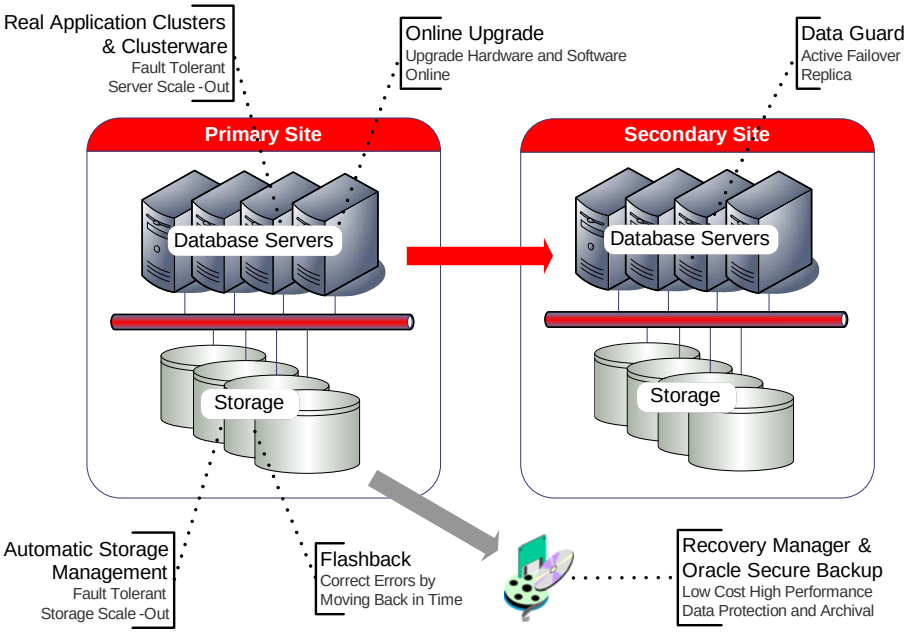
1. Deploy the Siebel database in an Oracle Database MAA configuration.
2. Deploy the Siebel application in a Siebel High Availability configuration.
3. Establish a secondary standby Siebel site for disaster recovery, testing and other planned maintenance activities.

The following chapters describe each recommendation in detail and the solutions that are available for planned and unplanned outages.

For demonstrations of Siebel behavior during RAC node failure and Site failure see http://www.oracle.com/technology/deploy/availability/demonstrations.html.

This document was written in the context of Siebel Business Applications, Version 8.0 running on Oracle Database 11g Release 1 and where necessary we make reference to the documentation of these product releases. In most cases, our recommendations and best practices can be applied to earlier releases too. If you are running earlier product versions we recommend you refer instead to the earlier product documentation.

ORACLE DATABASE MAA



To achieve maximum Siebel availability, Oracle recommends deploying Siebel on an Oracle Database MAA foundation that includes the following technologies:

- Oracle Real Application Clusters and Oracle Clusterware
- Oracle Data Guard
- Oracle Flashback
- Oracle Automatic Storage Management
- Oracle Recovery Manager and Oracle Secure Backup
- Oracle Online Upgrade

See also: Oracle® Database High Availability Overview for a thorough introduction to Oracle Database high availability products, features and best practices.

ORACLE REAL APPLICATION CLUSTERS AND CLUSTERWARE

Oracle Real Application Clusters (RAC) allows the Oracle database to run any packaged or custom application unchanged across a set of clustered nodes. This capability provides the highest levels of availability and the most flexible scalability. If a clustered node fails, the Oracle database will continue running on the surviving nodes. When more processing power is needed, another node can be added

without interrupting user access to data. See also: Oracle Real Application Clusters Administration and Deployment Guide.

Oracle Clusterware is a cluster manager that is designed specifically for the Oracle database. In a RAC environment, Oracle Clusterware monitors all Oracle resources (such as database instances and listeners). If a failure occurs, Oracle Clusterware will automatically attempt to restart the failed resource. During outages, Oracle Clusterware relocates the processing performed by the inoperative resource to a backup resource. For example, if a node fails, Oracle Clusterware will relocate database services being used by the application onto a surviving node in the cluster. See also: Oracle Clusterware Administration and Deployment Guide.

Oracle Clusterware may also be used as the cluster manager for Siebel mid-tier components.

ORACLE DATA GUARD AND ONLINE UPGRADE

Oracle Data Guard provides a comprehensive set of services that create, maintain, manage, and monitor one or more standby databases to enable production Oracle databases to survive failures, disasters, user errors, and data corruption. Data Guard maintains these standby databases as transactionally consistent copies of the production database. If the production database becomes unavailable due to a planned or an unplanned outage, Data Guard can switch any standby database to the production role, thus greatly reducing the application downtime caused by the outage. Data Guard can be used with traditional backup, restore, and clustering solutions to provide a high level of data protection and data availability. Siebel supports both physical and logical standby databases. See also: Oracle Data Guard Concepts and Administration.

A physical standby database provides a physically identical copy of the primary database, with on disk database structures that are identical to the primary database on a block-for-block basis. A physical standby database is kept synchronized with the primary database, through Redo Apply, which recovers the redo data received from the primary database and applies the redo to the physical standby database. As of Oracle Database 11g release 1 (11.1):

A physical standby database can receive and apply redo while it is open for read-only access and so may be used for other purposes as well as disaster recovery.

With a single command, a physical standby database can be converted into a Snapshot Standby and become an independent database open read-write, ideal for QA and other testing. The Snapshot Standby continues to receive and archive redo data from the primary database while it is open read-write, thus protecting primary data at all times. When testing is complete, a single command will convert the snapshot back into a standby database, and automatically resynchronize it with the primary.

A physical standby database can be used for rolling database upgrades using the SQL Apply process – and return to its function as a physical standby database once the upgrade is complete.

A logical standby database contains the same logical information as the production database, although the physical organization and structure of the data can be different. The logical standby database is kept synchronized with the primary database though SQL Apply, which transforms the data in the redo received from the primary database into SQL statements and then executes the SQL statements on the standby database. A logical standby database can be used for disaster recovery and reporting requirements, and can also be used to upgrade the database software and apply patch sets while the application is online and with almost no downtime.

It is possible to deploy a local standby database at the primary site as well as a remote standby at the secondary site. This offers the advantage that a failover to the local standby can be performed while the Siebel Servers continue running and can be done almost transparently to the end users. It also offers the ability to perform an online database upgrade without the need to switch to another site. We would recommend that a local and remote standby be deployed for maximum availability.

ORACLE FLASHBACK

Oracle Flashback quickly rewinds an Oracle database, table or transaction to a previous time, to correct any problems caused by logical data corruption or user error. It is like a 'rewind button' for your database. Oracle Flashback is also used to quickly return a previously primary database to standby operation after a Data Guard failover, thus eliminating the need to recopy or re-instantiate the entire database from a backup. See also: Oracle Flashback Technology.

ORACLE AUTOMATIC STORAGE MANAGEMENT

Oracle Automatic Storage Management (ASM) provides a vertically integrated file system and volume manager directly in the Oracle kernel, resulting in:

- Significantly less work to provision database storage
- Higher levels of availability
- Elimination of the expense, installation, and maintenance of specialized storage products
- Unique capabilities for database applications

For optimal performance, ASM spreads files across all available storage. To protect against data loss, ASM extends the concept of SAME (stripe and mirror everything) and adds more flexibility in that it can mirror at the database file level rather than the entire disk level.

Oracle Recovery Manager and Oracle Secure Backup

Recovery Manager (RMAN) is an Oracle database utility that can back up, restore, and recover database files. It is a feature of the Oracle database and does not require separate installation. RMAN integrates with sessions running on an Oracle database to perform a range of backup and recovery activities, including maintaining a repository of historical data about backups.

Oracle Secure Backup (OSB) is a centralized tape backup management solution providing performant, heterogeneous data protection in distributed UNIX, Linux, Windows, and Network Attached Storage (NAS) environments. By protecting file system and Oracle database data, OSB provides a complete tape backup solution for your IT environment. OSB is tightly integrated with Recovery Manager (RMAN) to provide the media management layer for RMAN.

Deploying or Transitioning to Oracle Database MAA

The mid-tier and database tier are cleanly separated in the Siebel application architecture which results in a looser coupling of application and database concerns. One benefit of this separation is that it allows us to apply a great deal of the generic Oracle database knowledge and best practices to the Siebel database without specific Siebel application concerns. In particular, it is possible to setup and configure the Siebel database in an MAA configuration, including RAC, ASM and Data Guard using the standard documentation and best practices. By using Oracle Data Guard it is possible to transition to RAC and ASM with almost no Siebel application downtime.

To tune and refine your configuration refer to the database MAA papers located at http://www.oracle.com/technology/deploy/availability/htdocs/maa.htm.

We recommend that Siebel is configured for Oracle transparent application failover (TAF) and uses Oracle Clusterware managed database services. We also recommend that TCP Keep Alive Timeout is reduced on the Siebel Servers.

Configuring Transparent Application Failover

Siebel supports Oracle Transparent Application Failover (TAF) which means that the Siebel Servers can move work to a surviving database instance in the event of a database instance failure. In most cases this failover will be transparent to the Siebel end user. Siebel can also be configured to transparently fail over to an Oracle Database Guard local standby when the primary database is lost.

To take advantage of TAF it is necessary to configure TAF for the database services used by Siebel or to configure TAF in the Oracle NET configuration of each Siebel Server. The "SELECT" failover type and "BASIC" failover method should be adequate for most configurations.

The "PRECONNECT" failover method may be used to improve failover times. With this method, a connection to a backup database instance is established for

every primary connection. If the primary connection is lost the backup connection is used instead, saving the need to reconnect at the time of failure. This will effectively double the database session overhead and so care must be taken to size the system appropriately before using this method.

See Metalink Note 460982.1 - How To Configure Server Side Transparent Application Failover for details on how to configure database services for TAF.

See Metalink Note 453293.1 - 10g :Configuration of TAF(Transparent Application Failover) and Load Balancing for details on how to configure SQL*NET for TAF.

The following table summarizes Siebel behavior during RAC or Data Guard failover when TAF is configured. Besides a short pause as the failover occurs, the failure is transparent to the end user:

Siebel Operation	Behavior
Web client user is updating data and steps-off (saves) the updates during or just after the DB failure.	Oracle reconnects and reconstructs the database session on a surviving node and Siebel resubmits the update.
Web client user is paging through queried data when the DB failure occurs.	Oracle reconnects and reconstructs the database session on a surviving node, re-executes the query, repositions the SQL cursor, and returns the next set of rows.
Web client user is issuing a new query or switching screens just after the DB failure.	Oracle reconnects and reconstructs the database session on a surviving node.

Configuring Oracle Clusterware Managed Database Services

It is recommended that an Oracle Clusterware managed database service be created for Siebel connections to the database. This is necessary to ensure that connections will only be made to open database instances. Oracle Clusterware managed database services are created through Enterprise Manager or with the srvctl command. Here is an example using srvctl:

```
srvctl add service -d SEBLRAC -s SIEBEL -r
"SEBLRAC1,SEBLRAC2" -P basic
```

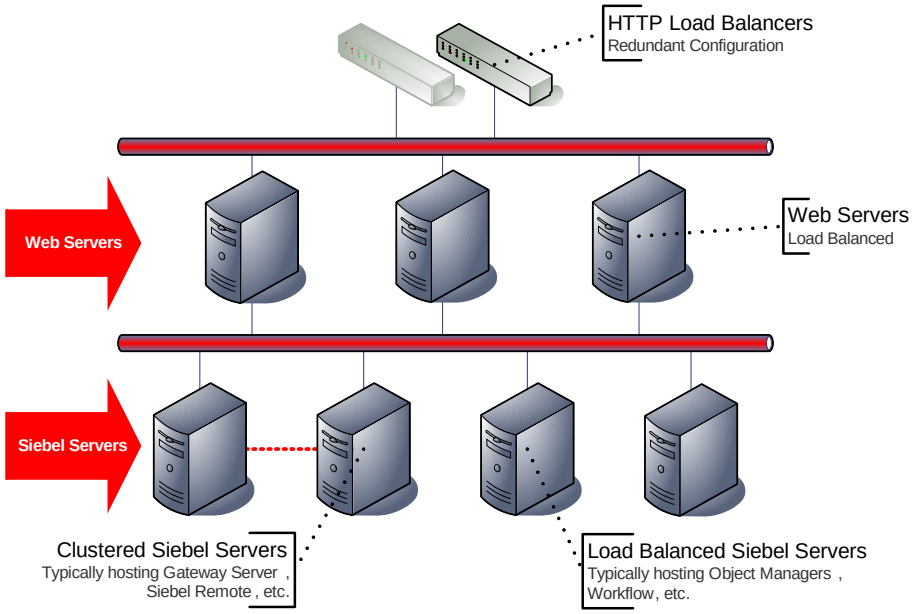
Configuring TCP Keepalive Timeout

Siebel currently does not support Oracle Fast Application Notification (FAN) and so it is necessary to reduce the TCP Keepalive Timeout for the Siebel Servers to release database connections in the event of a database node crash. This is only for

the rare case where the database node crashes before the TCP connections can be cleaned up, and only for connections where a database request was in-flight at the time of failure or a new request was started before the Virtual Internet Protocol (VIP) Address could be switched to a surviving node.. In all other cases the database connection failure is detected and a new connection is established on a surviving node.

Please refer to Metalink Note 249213.1 - Performance problems with Failover when TCP Network goes down (no IP address) for details on how to configure the TCP Keepalive timeout. Changing this parameter may have adverse effects on other network users.

SIEBEL HIGH AVAILABILITY DEPLOYMENT



In this section we discuss the high availability deployment of the Siebel application software that is layered on top of the Database MAA foundation.

SIEBEL HA DEPLOYMENT OPTIONS

Siebel offers the following high availability component deployment options:

Load Balancing

Siebel components (where noted below) are installed and deployed on multiple servers, and run in an "active/active" configuration for high availability and scalability purposes. Client initiated workload is distributed across multiple component instances running on multiple servers through load balancing. Web Server load is distributed by an HTTP load balancer. Siebel Server load may be delivered by an HTTP load balancer or by native Siebel load balancing.

Distributed Services

Many Siebel Components are implemented as Business Services and in some cases they can be deployed in a redundant configuration across multiple Siebel Servers known as distributed services. Business services are invoked by other components to complete their business function. The Siebel Server Request Broker (SRB) balances Service requests across the component instances. In the event that a component instance is lost, the request is re-routed to the surviving instances. An SRB instance will typically be running on all Siebel Servers.

Clustering

Siebel Server clusters consist of two or more physical servers linked together so that if one server fails, resources such as disks, network addresses, and Siebel components can be switched over to another server. Siebel components run in an active/passive configuration where a specific Siebel component instance is running on only one physical host at a time. We use Oracle Clusterware (or other 3rd party clusterware) to monitor and manage the configuration to ensure the components are enabled on only one node of a hardware cluster at a time.

Not all deployment options are supported by all components. The following table gives an example of the supported and preferred options for some of the most commonly deployed components. The Siebel Deployment Planning Guide has a comprehensive list.

Component	Clustering	Load Balancing	Distributed Services
Object Manager	Supported	Preferred	
EAI Object Manager	Supported	Preferred	
Siebel Remote	Preferred		
Workflow Process Manager	Supported		Preferred
Siebel Web Server	Supported	Preferred	
Siebel Gateway Server	Preferred		

LOAD BALANCING DEPLOYMENT

Web Servers and many Siebel Server Components can be load balanced. A third party load balancer is required to balance Web Server load. Siebel native (software) or third party load balancing may be used to balance Siebel Server load. The load balancer monitors the servers and improves availability by routing traffic appropriately when outages occur. Third party load balancers should be deployed in a redundant configuration. Please refer to your load balancer documentation for specific detail on how to configure Siebel load balancing.

See also: Siebel Deployment Planning Guide.

SIEBEL CLUSTER DEPLOYMENT

To create a Siebel cluster you need a Cluster Manager and a Shared Siebel software home.

Cluster Manager:

- Supports service virtual IP management with failover. The virtual IP address is used as a single network address for the Siebel Server or Gateway Server independent of the physical service location
- Performs service monitoring so it will know when services fail.
- Will restart and relocate Siebel services in the event of failure.

Shared Siebel software home:

- Shared by all cluster nodes for failover but accessed by only one node at any given time.
- Contains Siebel software, name server backing file, remote docking folders, etc.
- Must be deployed in a HA configuration to avoid a single point of failure. Typically a cluster file system or clustered NFS solution would be used.

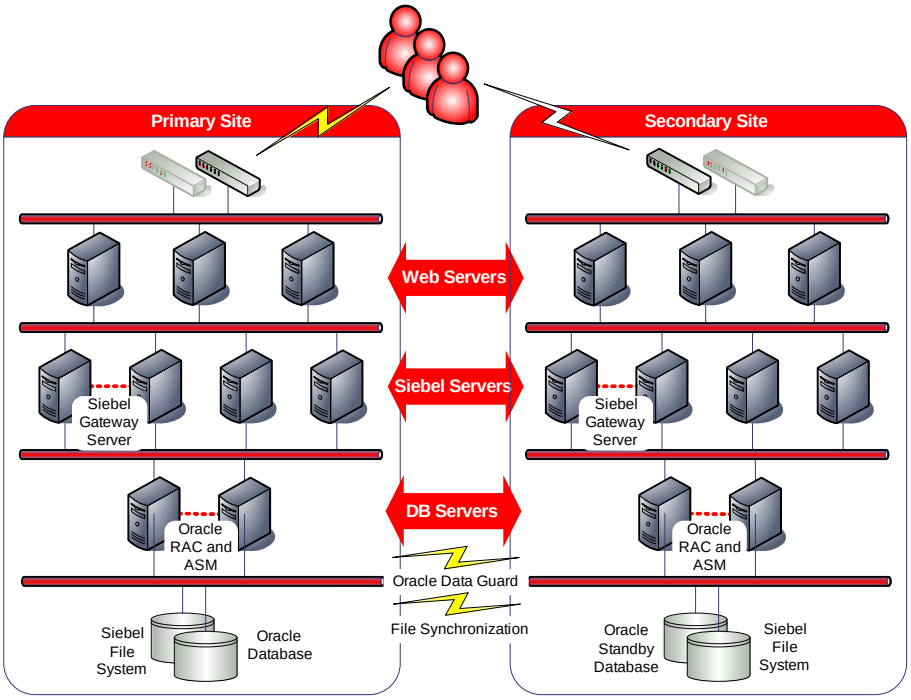
Oracle Clusterware may be used as the Cluster Manager for protecting Siebel components, see http://www.oracle.com/technology/products/database/clusterware/pdf/siebel_crm_protected_by_clusterware.pdf.

Please see the "Siebel System Requirements and Supported Platforms Guide" for details of Oracle's Support requirements for a clustered solution.

SIEBEL FILE SYSTEM DEPLOYMENT

The Siebel file system is used to store file attachments and other documents in the Siebel application and is accessed in parallel by all Siebel Servers. It is a critical part of the Siebel application and so must be deployed in a HA configuration to avoid a single point of failure. Typically, this would be achieved through a cluster file system or clustered NFS solution.

SIEBEL SECONDARY SITE DEPLOYMENT



To guard against an entire site failure and to reduce downtime during certain planned maintenance operations it is recommended that a secondary site be established. We will refer to this as the secondary site for the remainder of this document.

Care must be taken to size the secondary site appropriately so that business continuity can be assured. To achieve the same service levels on failover it will be necessary to replicate the primary site. Many customers also use their secondary site for development and testing purposes to increase system utilization. It is technically possible to deploy a reduced configuration on the standby site if necessary, for example the standby database may be non-RAC, or there may be fewer Siebel Servers on the secondary site.

In the following sections we describe how to establish the Oracle Data Guard Standby Database and Standby Siebel Enterprise environments. We also describe the secondary site operational procedures and how they can be automated.

Once in place, a secondary site must be regularly tested to make sure that it is operational in a real emergency.

ORACLE DATA GUARD STANDBY DATABASE DEPLOYMENT

Establishing a Data Guard standby database requires the following steps:

- Install and Configure Oracle Clusterware, ASM and Database on the standby site.
- Backup, transport and restore the database to the standby site.
- Begin standby operation.

Please refer to Oracle Data Guard Concepts and Administration for the detailed steps.

STANDBY SIEBEL ENTERPRISE DEPLOYMENT

To complete the standby site it is necessary to install Siebel and establish a standby Siebel enterprise. The standby Siebel enterprise will be configured to connect to the standby database. In the event of a switchover or failover to the standby site, the standby database will be opened and the standby Siebel enterprise will be started.

The Siebel File System contains critical data files that must be made available on the standby site in the event of a disaster. To ensure that this is available straight away we recommend that a standby copy of the file system be maintained. Any number of file copying tools could be used to create and maintain the copy. "rsync", for example, is a readily available utility that is well suited to maintaining copies over a wide area network. It will be necessary to switch the direction of synchronization in the event of a switchover or failover.

In the event of a site loss it is advantageous to have access to recent copies of trace files to aid in recovery on the secondary site. One way to do this is to create and maintain an offsite copy of the log folders, say using "rsync".

After a site failover it will be necessary for the Siebel end users to connect to the standby site to access the Siebel application. The switchover can be made transparent to the end users by implementing a Domain Name Server (DNS) push. In a DNS push, the IP address associated with the Siebel service is changed in DNS and then propagated to the end user's browser. When the user tries to connect they pick up the new address and are routed to the alternate location.

OPERATIONAL PROCEDURES

Switchover Procedure

The switchover procedure involves the following steps:

1. Verify standby is up-to-date and operating correctly
2. Shut down Siebel
3. Switch over to standby database
4. Enable flashback on standby (optional)
5. Open standby

6. Start original standby as primary
7. Ensure the standby Siebel File System is up-to-date and reverse the synchronization direction
8. Start Siebel. Siebel Remote user will need to be re-extracted.

See the Oracle Data Guard documentation for detailed database switchover steps. The same procedure is followed to switch back to the primary.

Failover Procedure

The failover procedure involves the following steps:

1. Fail over to standby database
2. Enable flashback on standby (optional)
3. Open standby
4. Start Siebel. Siebel Remote users will need to be re-extracted
5. If/when the primary site becomes available, flash the primary database back and start standby operation, and reverse the synchronization direction of the Siebel File System

See the Oracle Data Guard documentation for the detailed database failover steps.

Standby Testing Procedure Using Flashback Database

You can verify the viability of the standby site while the primary site is in live operation, using Oracle Flashback Database to quickly restore the standby site to standby operation afterwards.

This procedure assumes the primary site is in live operation and the standby site is in standby mode and applying redo.

- Activate and open the standby database
- Perform testing
- Flash the database back and resume standby operation

See the Oracle Data Guard documentation for detailed failover steps.

Patching and Maintenance Procedure

Once in place, a secondary site must be kept up-to-date with the primary. When Siebel software and configuration changes are made to the primary they must also be applied to the standby. Database data changes will be propagated automatically by Oracle Data Guard. Siebel File System changes will be propagated to Standby Siebel File System by using the appropriate copying mechanism as described above.

Care must be taken to ensure that the standby site is available at all times just in case it is needed. Try to avoid or minimise the time when the secondary site is out of sync or the different parts of the secondary site are out of sync with each other. In a Siebel application upgrade scenario, for example, it may be necessary to suspend Oracle Data Guard synchronization and Standby Siebel File System synchornization until an upgraded Standby Siebel Enterprise has been established.

AUTOMATING SWITCHOVER AND FAILOVER PROCEDURES

This section details the steps to achieve automation of Oracle Data Guard administration including switchover and failover procedures by utilizing Oracle Data Guard Broker. We also describe how we may optionally configure automatic database failure detection and Siebel database and application failover using Oracle Fast-Start Failover.

The Oracle Data Guard broker is a distributed management framework that automates and centralizes the creation, maintenance, and monitoring of Data Guard configurations. The broker automates and simplifies the following operations:

- Managing an entire Data Guard configuration, including all databases, redo transport services, and log apply services, through a client connection to any database in the configuration.
- Managing the protection mode for the broker configuration.
- Invoking switchover or failover with a single command to initiate and control complex role changes across all databases in the configuration.
- Configuring failover to occur automatically upon loss of the primary database, increasing availability without manual intervention. This is known as Fast-Start Failover (FSFO). Use of this feature is optional.
- Monitoring the status of the entire configuration, capturing diagnostic information, reporting statistics such as the log apply rate and the redo generation rate, and detecting problems quickly with centralized monitoring, testing, and performance tools.

All management operations can be performed locally or remotely through the broker's easy-to-use interfaces: the Data Guard management pages in Oracle Enterprise Manager, which is the broker's graphical user interface (GUI), and the Data Guard command-line interface called DGMGRL.

In our example we have automated the entire switchover/failover process including automatic triggering of failover. It is possible to omit certain pieces, such as automatic triggering, if they do not make sense for your implementation.

The process works like this:

- Fast-Start Failover (FSFO) determines that a failover is necessary and initiates a failover to the standby database automatically, or the administrator instructs the broker to switch or fail over.
- When the database switchover/failover has completed the DB_ROLE_CHANGE database event is fired.
- The event causes a trigger to be fired which calls a script that configures and starts the Siebel application.

As a starting point for our setup procedures we assume that the primary site is in live operation and the standby site is in standby mode and applying redo. Here are the steps we follow to implement FSFO.

Develop Siebel Startup Script

Develop a script that will automate the Siebel startup process. An example script is provided for your reference in Appendix A. Modify the script to suit your environment and requirements.

Make sure ssh (or equivalent) is configured so that remote shell scripts can be executed without password prompts.

Make sure that the operating system user has permissions to execute the script.

Automate Script Execution by Trigger

Create a database event "DB_ROLE_CHANGE" trigger, which fires after database role changes from standby to primary. For example:

```
CREATE OR REPLACE TRIGGER postover
AFTER DB_ROLE_CHANGE ON DATABASE
DECLARE
  v_db_unique_name varchar2(30);
BEGIN
  select upper(VALUE) into v_db_unique_name
  from v$parameter where NAME='db_unique_name';

  dbms_scheduler.create_job(
    job_name=>'postover',
    job_type=>'executable',
    job_action=>
      '/NAS/oracle/FSFO/' ||
v_db_unique_name||'.fsfo.sh',
    enabled=>TRUE
  );
END;
```

The trigger calls a wrapper script named <DB_UNIQUE_NAME>.fsfo.sh which in turn calls the fsfo.sh script. This is done because it is not possible to directly pass arguments to a script from dbms_scheduler. A wrapper script must be created for the production and standby databases.

In our case, we create a script named SEBLRAC.fsfo.sh as follows:

```
#!/bin/sh
/NAS/oracle/FSFO/fsfo.sh SEBLRAC
```

And, we create a script named SEBLRAC_DR.fsfo.sh as follows:

```
#!/bin/sh
/NAS/oracle/FSFO/fsfo.sh SEBLRAC_DR
```

Configure Fast-Start Failover

Follow the steps in Oracle Database 10g Release 2 Best Practices: Data Guard Fast-Start Failover to configure Fast-Start Failover.

PLANNED AND UNPLANNED OUTAGE SOLUTIONS

In the following sections we summarize the outages that may occur in a Siebel environment and the Oracle solution that would be used to keep application downtime to a minimum. In all cases, we are focused on Siebel Application downtime as perceived by the end user, not the downtime of the individual component.

UNPLANNED OUTAGE SOLUTIONS

In the following table we describe the unplanned outages that may be caused by system or human failures in a Siebel environment and the technology solutions that would be used to recover and keep downtime to a minimum.

Outage Type	Oracle Solution	Benefits	Recovery Time
Siebel Node or Component Failure	Load Balancing	Surviving nodes pick up the slack	Affected users reconnect
	Distributed Services	Surviving nodes continue processing	No downtime
	Clustering	Automatic failover to surviving node	Seconds to < 2 minutes
Database Node or Instance Failure	RAC	Automatic recovery of failed nodes and instances, transparent application and service failover	Users transparently fail over Updates may need to be re-submitted
Site Failure	Data Guard	Fast Start Failover	Seconds to 5 minutes
Storage Failure	ASM	Mirroring and automatic rebalance	No downtime
	RMAN with flash recovery area	Fully managed database recovery and disk based backups	Minutes to hours

Outage Type	Oracle Solution	Benefits	Recovery Time
	Data Guard	Fast Start Failover	Seconds to 5 minutes
Human Error	Oracle Flashback	Database and fine grained rewind capability	Minutes
	Log Miner	Log analysis	Minutes to hours
Data Corruption	RMAN with flash recovery area	Online block media recovery and managed disk-based backups	Minutes to hours
	Data Guard	Automatic validation of redo blocks before they are applied, fast failover to an uncorrupted standby database	Seconds to 5 minutes

Site failure will require Siebel Remote re-extract

PLANNED MAINTENANCE SOLUTIONS

In the following table we summarize the planned maintenance activities that may typically occur in a Siebel environment and the technology solutions that we would recommend to keep downtime to a minimum.

Maintenance Activity	Solution	Siebel Outage
Mid-Tier Operating System or Hardware Upgrade	Siebel Load balancing, distributed services and clustering	No downtime
Siebel Application Patching	Siebel rolling patch application	No downtime
Siebel Application Configuration Change	Siebel Application Restart	Minutes
Siebel Upgrades	Siebel Upgrade and Upgrade Tuner	Hours to days (depending on DB size) [1]

[1] In reality there are a number of ways to mitigate the impact of extended upgrade downtime, for example, by providing a read-only replica. Oracle Consulting Services have significant experience in this area and can help to plan and execute the upgrade.

Database Tier Operating Systen or Hardware Upgrade	Oracle RAC	No downtime
Oracle Database interim patching	Oracle RAC rolling apply	No downtime
Oracle Database 11g online patching	Online Patching	No downtime
Oracle Clusterware upgrade and patches	Rolling apply/upgrade	No downtime
Database storage migration	Oracle ASM	No downtime
ASM upgrade	10g: Oracle Data Guard	Seconds to minutes
	11g: Rolling Upgrade	No downtime
Migrating to ASM or migrating a single-instance database to Oracle RAC	Oracle Data Guard	Seconds to minutes
Patch set and database upgrades	Oracle Data Guard logical standby	Seconds to minutes

REFERENCES

MAA home page:
http://www.oracle.com/technology/deploy/availability/htdocs/maa.htm

MAA demonstrations page:
http://www.oracle.com/technology/deploy/availability/demonstrations.html

Overview of Oracle database high availability products and features:
Oracle® Database High Availability Overview

Guide to Oracle Clusterware Administration and Deployment:
Oracle Clusterware Administration and Deployment Guide

Definitive guide to Oracle Data Guard concepts and administration:
Oracle Data Guard Concepts and Administration

Summary of Flashback technologies in the Oracle Database Concepts Guide:
Oracle Flashback Technology

Guide to Siebel Deployment including HA deployment:
Siebel Deployment Planning Guide

Guide to using Oracle Clusterware to create a Siebel mid-tier cluster:
http://www.oracle.com/technology/products/database/clusterware/pdf/siebel_crm_protected_by_clusterware.pdf

MAA best practices for Data Guard Fast-Start Failover deployment:
Oracle Database 10g Release 2 Best Practices: Data Guard Fast-Start Failover

Metalink Note 460982.1 - How To Configure Server Side Transparent Application Failover

Metalink Note 453293.1 - 10g :Configuration of TAF(Transparent Application Failover) and Load Balancing

Metalink Note 249213.1 - Performance problems with Failover when TCP Network goes down (no IP address)

APPENDIX A - SAMPLE CONFIGURATION AND STARTUP SCRIPT

This script is named fsfo.sh in our example and is executed by the postover trigger on change to primary database operation. It is designed to be executed by any database node as any node can be first to open:

```
#!/bin/sh

# Enable/Disable the script,
# set value to 1 to perform the steps in the script
#################################################################
ENABLED=1

# Arg1 DB_UNIQUE_NAME determines the site
# that needs to be activated.
#################################################################
DB_UNIQUE_NAME=$1

# Constants, modify according to your environments
#################################################################
DB_NAME=SEBLRAC

SITE1=SEBLRAC
SITE2=SEBLRAC_DR

DB_NODES_SITE1="ha1dbh01 ha1dbh02"
DB_NODES_SITE2="ha2dbh01 ha2dbh02"

SS_NODES_SITE1="ha1imh01 ha1imh02"
WS_NODES_SITE1="ha1mth03 ha1mth04"
SS_NODES_SITE2="ha2imh01"
WS_NODES_SITE2="ha2imh01"

OH=/u01/dbhome/siebelrac

DBOSUSER=oracle
APPSOSUSER=siebel

# Logfile
#################################################################
LOGF=/NAS/oracle/fsfo/SEBLfsfo.log
DETAILLOGF=/NAS/oracle/fsfo/SEBLdetailfsfo.log
exec >>$LOGF 2>>$DETAILLOGF

# Start executing
#################################################################

echo ""
echo "-------------------------------------------------"
echo "script started at `date`"
echo "-------------------------------------------------"
echo ""

# Initialize the variables for the correct Site
#################################################################
if [ ${DB_UNIQUE_NAME}x = ${SITE1}x ]; then
  DB_NODES=$DB_NODES_SITE1
  SS_NODES=$SS_NODES_SITE1
  WS_NODES=$WS_NODES_SITE1
elif [ ${DB_UNIQUE_NAME}x = ${SITE2}x ]; then
  DB_NODES=$DB_NODES_SITE2
  SS_NODES=$SS_NODES_SITE2
  WS_NODES=$WS_NODES_SITE2
else
  echo "`date` -- Error !"
  echo "(Err) Missing/Invalid argument DB_UNIQUE_NAME: \"$DB_UNIQUE_NAME\""
  exit 1
fi

echo "Site: $DB_UNIQUE_NAME on `hostname` as `id`"
echo "-------------------------------------------------"

echo "`date` -- Start Gateway Server"
```

```
echo "------------------------------------------------"

if [ ${DB_UNIQUE_NAME}x = ${SITE1}x ]; then
  ssh siebel@ha1imh01 /NAS/oracle/fsfo/startGW >>$DETAILLOGF &
elif [ ${DB_UNIQUE_NAME}x = ${SITE2}x ]; then
  ssh siebel@ha2imh01 /NAS/oracle/fsfo/startGW >>$DETAILLOGF &
fi

echo "`date` -- Start Siebel Server on All Nodes"
echo "------------------------------------------------"

for node in $SS_NODES; do
  ssh ${APPSOSUSER}@$node /NAS/oracle/fsfo/startSS >>$DETAILLOGF &
done
wait

echo ""
echo "`date` -- Start Web Server on All Nodes"
echo "------------------------------------------------"

for node in $WS_NODES; do
  ssh ${APPSOSUSER}@$node /NAS/oracle/fsfo/startWS >>$DETAILLOGF &
done
wait

echo ""
echo "------------------------------------------------"
echo "script completed at `date`"
echo "------------------------------------------------"
echo ""
```

Siebel Maximum Availability Architecture
October 2008
Authors: Richard Exley

Oracle USA, Inc.
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