

Best Practices for Running Oracle Siebel CRM on AWS

March 2018

This paper has been archived.

For the latest technical content about the AWS Cloud, see the AWS Whitepapers & Guides page:

<https://aws.amazon.com/whitepapers>



Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Archived

Contents

| | |
|--|----|
| Introduction | 1 |
| Benefits of Running Siebel Applications on AWS | 1 |
| Key Benefits of AWS versus On-Premises | 1 |
| Key Benefits over SaaS | 4 |
| AWS Concepts and Services | 5 |
| Regions and Availability Zones | 5 |
| Amazon EC2 | 7 |
| Amazon RDS | 7 |
| AWS DMS | 7 |
| Elastic Load Balancing | 7 |
| Amazon EBS | 8 |
| Amazon Machine Images | 8 |
| Amazon S3 | 8 |
| Amazon Route 53 | 8 |
| Amazon VPC | 9 |
| AWS Direct Connect | 9 |
| Siebel CRM Architecture on AWS and Deployment Best Practices | 9 |
| Traffic Distribution and Load Balancing | 10 |
| Scalability | 11 |
| Architecting for High Availability and Disaster Recovery | 12 |
| VPC and Connectivity Options | 16 |
| Securing Your Siebel Application on AWS | 17 |
| Siebel and Oracle Licensing on AWS | 19 |
| Siebel and Oracle Database License Portability | 19 |
| Amazon RDS for Oracle Licensing Models | 20 |
| Siebel on AWS Use Cases | 20 |

| | |
|---------------------------------------|-----------|
| Monitoring Your Infrastructure | 21 |
| AWS and Oracle Support | 22 |
| AWS Support | 22 |
| Oracle Support | 22 |
| Conclusion | 23 |
| Contributors | 23 |
| Further Reading | 23 |

Archived

Abstract

Oracle's Siebel Customer Relationship Management (CRM) is a widely used and popular CRM application. This whitepaper is intended for AWS customers and partners who want to learn about the benefits and options for running Oracle Siebel CRM on AWS.

This whitepaper provides architectural guidance and outlines best practices for high availability, security, scalability, performance, and disaster recovery for running Oracle Siebel CRM systems on AWS.

Archived

Introduction

Companies are increasingly adopting a “cloud first, mobile first” strategy. Migrating Oracle’s Siebel Customer Relationship Management (CRM) applications to a cloud platform is becoming a necessity. This paper is intended to help you understand Amazon Web Services (AWS) and how to leverage AWS to run Oracle Siebel CRM applications. The paper also discusses key benefits and best practices for running Oracle Siebel CRM workloads on AWS.

Benefits of Running Siebel Applications on AWS

Migrating your Siebel applications to AWS is relatively simple and straightforward. However, it’s important that you don’t view this as merely a physical to virtual conversion, or as just a “lift and shift” migration. Understanding and using the AWS services and capabilities will help you make the most of running your Siebel systems on AWS.

Key Benefits of AWS versus On-Premises

Migrating your on-premises Siebel environment to AWS offers you the following benefits:

- **Eliminate long procurement cycles** – Traditional deployment, as shown in the following figure, involves a long procurement process. Each stage is time intensive and requires large capital outlay and multiple approvals.

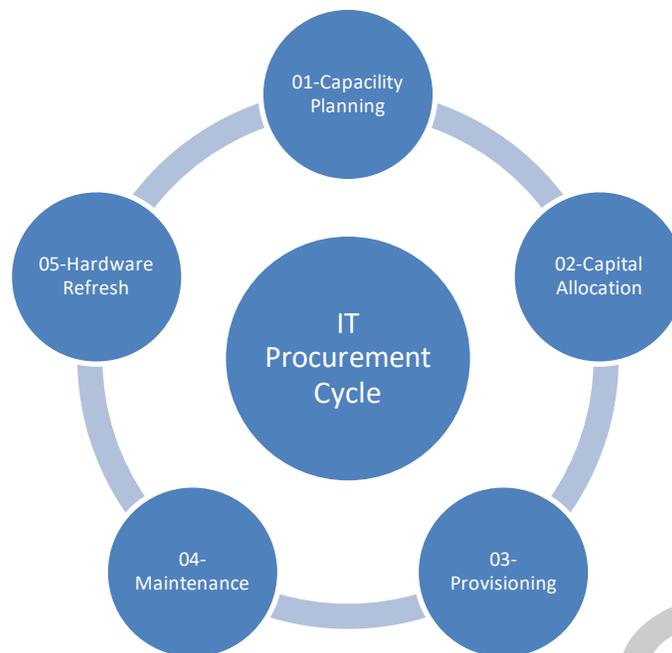


Figure 1: A typical IT procurement cycle

This process has to be repeated for the various environments, for example, development, testing, training, break-fix, and production, which compounds the costs and causes significant delays.

With AWS, you can provision new infrastructure and Siebel environments in minutes, compared to waiting weeks or months to procure and deploy traditional infrastructure.

- **Have Moore’s law work for you instead of against you** – In an on-premises environment, you end up owning hardware that depreciates in value every year. You’re locked in to the price and capacity of the hardware once you acquire it, and you have ongoing hardware support costs. With AWS, you can switch your underlying Siebel instances to newer AWS instance types as they become available.
- **Right size, anytime** – Often customers oversize environments for initial phases, and then can’t cope with growth in later phases. With AWS, you can scale the usage up or down at any time. You pay only for the computing capacity you use, for the duration you use it. You can change instance sizes in minutes through the AWS Management Console, the AWS API, or the AWS Command Line Interface (CLI).

- **Resilience and ability to keep recovering from multiple failures** – On-premises failures have to be handled on a case-by-case basis. Failed parts have to be procured and replaced. Key components such as the Siebel gateway name server have to be clustered using expensive clustering software. Deployment is still limited by the ability to handle only one failure in the primary gateway. With AWS, clustering of the Siebel gateway isn't required. The gateway can recover from multiple failures using the instance recovery feature of [Amazon EC2](#).
- **Disaster recovery** – You can build extremely low-cost, standby disaster recovery (DR) environments for existing deployments, and incur costs only for the duration of the outage.
- **Lower total cost of ownership (TCO)** – Siebel customers with on-premises data centers typically pay hardware support costs, virtualization licensing and support costs, data center costs, etc. You can eliminate or reduce all of these by moving to AWS. You benefit from the economies of scale and efficiencies that AWS provides, and pay for only the compute, storage, and other resources you use.
- **Ability to test application performance** – Performance testing is recommended before any major change to a Siebel environment. However, most customers performance test their Siebel CRM applications only during the initial launch on the yet-to-be-deployed production hardware. Later releases are usually never performance tested due to the expense and lack of the environment required for performance testing. With AWS, you can minimize the risk of discovering performance issues later in production. An AWS Cloud environment can be created easily and quickly just for the duration of the performance test and used only when needed. Again, you're charged only for the hours the environment is used.
- **No end-of-life (EOL) for hardware/platform** – All hardware platforms have EOL dates when the existing hardware is no longer supported and you are forced to buy new hardware. In the AWS Cloud, you can simply upgrade your Amazon EC2 instances to new AWS instance types in a single click, at no cost for the upgrade.
- **No need for clustering** – The Siebel gateway name server is a single point of failure. On-premises implementations require you to cluster the gateway. Clustering is complicated and expensive to implement. With AWS, no clustering is needed. In addition, you can automatically recover

a failed gateway name server using the instance recovery feature of Amazon EC2.

- **Unlimited environments** – Customers with on-premises data centers face the issue of limited environments. For example, a test environment will have a newer release compared to a production environment. This means that if a performance issue is found in production, you have no way to suddenly provision a performance debugging environment. On AWS, you can do this easily.

Key Benefits over SaaS

The following are some of the benefits of deploying Siebel CRM on AWS compared to moving to a CRM offering based on a Software-as-a-Service (SaaS) model:

- **Lower total cost of ownership (TCO)** – Existing Siebel customers don't have to purchase new CRM licenses or risk a reimplementing of their CRM—they can just move their existing Siebel CRM implementation to AWS. For new customers, the TCO is still low because they don't have to pay monthly SaaS license fees. Siebel is a proven CRM with rich verticals.
- **Unlimited usage** – SaaS applications have governor/platform limits to accommodate underlying multi-tenant architecture. Governor limits restrict usage, including the number of API calls, transaction times, datasets, and file sizes. With AWS, you can self-provision and use as much or as little capacity as you need. You pay only for what you use.
- **Multi-tenant versus elastic architecture** – SaaS products typically use a multi-tenant architecture, which ties you to a specific instance and the limits of that instance. With AWS, you have complete control over the computing capacity you provision—you can provision as much or as little as you need.
- **Single application management** – With Siebel CRM, you can manage everything—including marketing, sales, service, CPQ, and orders—in one application. On SaaS this requires multiple applications that you have to buy and integrate. The cost of integration with SaaS applications is easy to overlook in the buy decision, but these costs can add up significantly later.

AWS Concepts and Services

In this section, we introduce you to some AWS concepts and services that help you understand how Siebel CRM is deployed on AWS.

Regions and Availability Zones

The AWS Cloud infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple, physically separated and isolated Availability Zones that are connected with low latency, high throughput, and highly redundant networking. Availability Zones consist of one or more discrete data centers, each with redundant power, networking, and connectivity, and housed in separate facilities. These Availability Zones enable you to operate production applications and databases that are more highly available, fault tolerant, and scalable than is possible from a single data center.

Each region is a separate geographic area isolated from the other regions. Regions enable you to place resources, such as Amazon EC2 instances, and data in multiple locations. Resources aren't replicated across regions unless you do so specifically. An AWS account provides multiple regions so that you can launch your application in locations that meet your requirements. For example, you might want to launch in Europe to be closer to your European customers or to meet legal requirements. The following diagram illustrates the relationship between regions and Availability Zones.

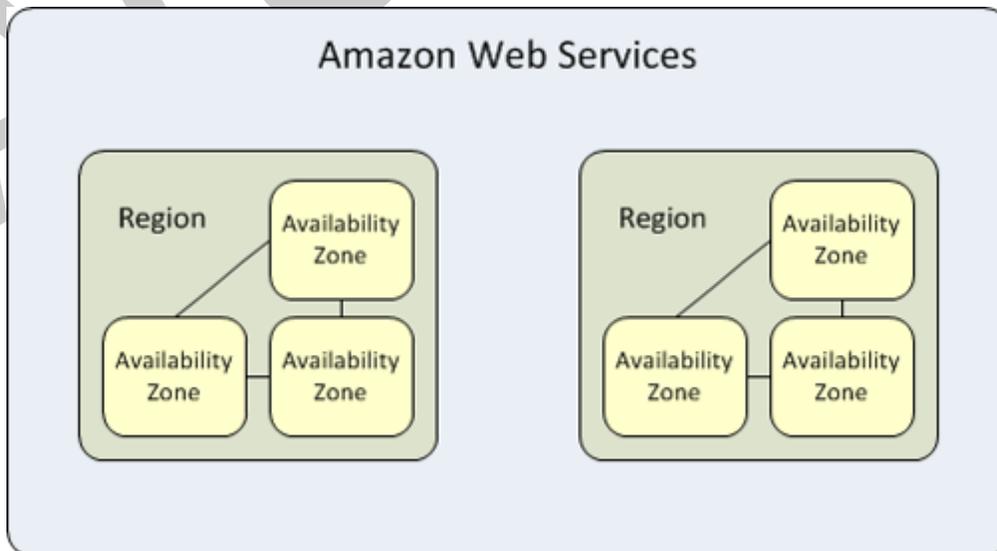


Figure 2: Relationship between AWS Regions and Availability Zones

The following figure shows the regions and the number of Availability Zones in each region provided by an AWS account, at the time of this publication. For the most current list of regions and Availability Zones, see <https://aws.amazon.com/about-aws/global-infrastructure/>.



Figure 3: Map of AWS Regions and Availability Zones

Amazon EC2



Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud, billed by the hour. You can run virtual machines with various compute and memory capacities. You have a choice of operating systems, including different versions of Windows Server and Linux.

Amazon RDS



Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity, while managing time-consuming database administration tasks. This allows you to focus on your applications and business. For Siebel, both Microsoft SQL Server and Oracle databases are available.

AWS DMS



AWS Database Migration Service (AWS DMS) helps you migrate databases to AWS quickly and securely. AWS DMS can also be used for continuous data replication with high availability, and supports most widely used commercial and open-source databases like Oracle, SQL Server, PostgreSQL, and SAP ASE.

Elastic Load Balancing



Elastic Load Balancing (ELB) automatically distributes incoming application traffic across multiple Amazon EC2 instances in the cloud. It enables you to achieve greater levels of fault tolerance in your applications, seamlessly providing the required amount of load balancing capacity needed to distribute application traffic. You can use ELB to load balance web server traffic.

Amazon EBS



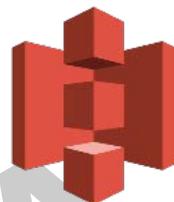
Amazon Elastic Block Store (Amazon EBS) provides persistent block-level storage volumes for use with EC2 instances in the AWS Cloud. Each EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability. EBS volumes offer the consistent and low-latency performance needed to run your workloads.

Amazon Machine Images



An Amazon Machine Image (AMI) is simply a packaged-up environment that includes all the necessary bits to set up and boot your EC2 instance. AMIs are your unit of deployment. Amazon EC2 uses Amazon EBS and Amazon S3 to provide reliable, scalable storage of your AMIs so that AWS can boot them when you ask AWS to do so.

Amazon S3



Amazon Simple Storage Service (Amazon S3) provides developers and IT teams with secure, durable, and highly scalable object storage. Amazon S3 is easy to use, with a simple web service interface to store and retrieve any amount of data from anywhere on the web. With Amazon S3, you pay only for the storage you actually use. There is no minimum fee and no setup cost.

Amazon Route 53



Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. It's designed to give developers and businesses an extremely reliable and cost-effective way to route end users to internet applications.

Amazon VPC



Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the AWS Cloud where you can launch resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own private IP address range, creation of subnets, and configuration of route tables and network gateways.

You can leverage multiple layers of security, including security groups and network access control lists, to help control access to EC2 instances in each subnet. Additionally, you can create a hardware virtual private network (VPN) connection between your corporate data center and your VPC, and use the AWS Cloud as an extension of your corporate data center.

AWS Direct Connect



AWS Direct Connect is a network service that provides an alternative to using the internet to utilize AWS Cloud services.

Using Direct Connect, you can establish private dedicated network connectivity between AWS and your data center, office, or colocation environment. In many cases this can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections.

Siebel CRM Architecture on AWS and Deployment Best Practices

The following architecture diagram illustrates how you can deploy Oracle Siebel CRM on AWS. Three required components of your Siebel CRM application (the Siebel gateway name server, Siebel application server, and Siebel web server) can be deployed to multiple EC2 instances behind an Elastic Load Balancing load balancer. The fourth required Siebel component (the Siebel database) can be set up on Amazon RDS for Oracle. You can deploy your Siebel web, application, and gateway name servers and the Siebel database across multiple Availability Zones for high availability of your application.

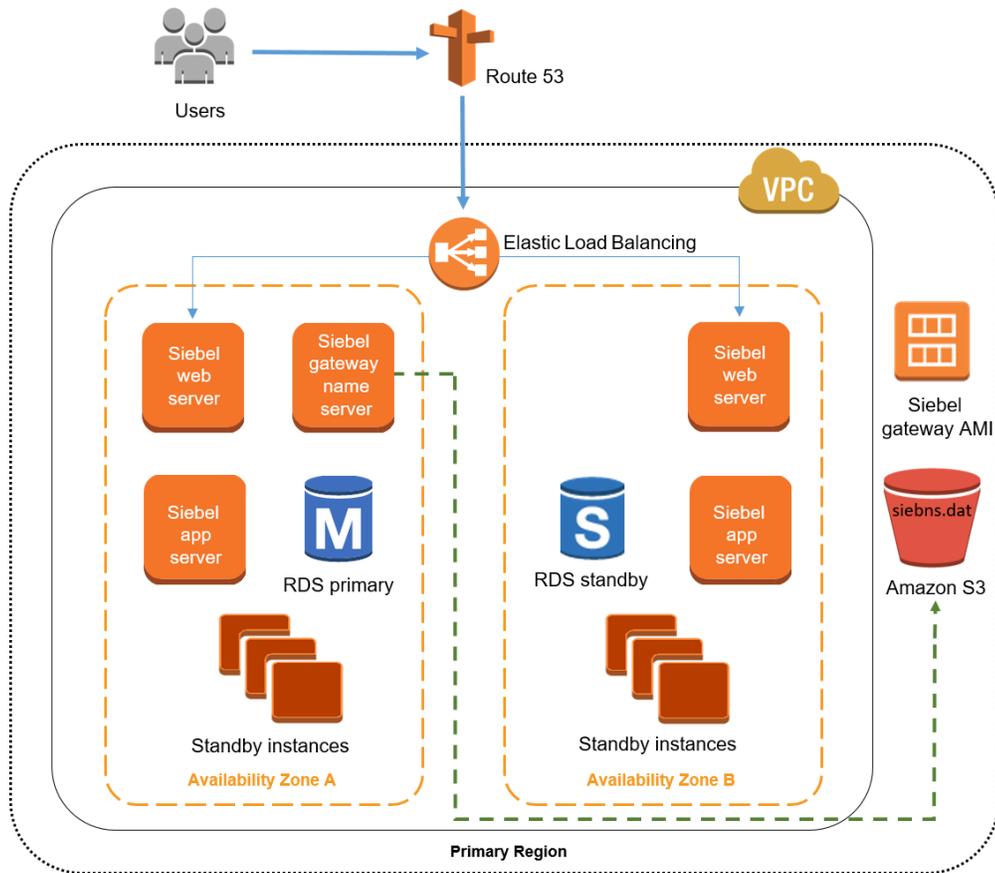


Figure 4: Architecture for deploying Siebel CRM on AWS

The following sections describe the elements of this architecture in detail.

Traffic Distribution and Load Balancing

Amazon Route 53 DNS is used to direct users to Siebel CRM hosted on AWS. Elastic Load Balancing (ELB) is used to distribute incoming application traffic across the Siebel web servers deployed in multiple Availability Zones. The load balancer serves as a single point of contact for client requests, which enables you to increase the availability of your application. You can add and remove Siebel web server instances from your load balancer as your needs change, without disrupting the overall flow of information. ELB ensures that only healthy Siebel web server instances receive traffic by detecting unhealthy instances and rerouting traffic across the remaining healthy instances. If a Siebel web server instance fails, ELB automatically reroutes the traffic to the

remaining running Siebel web server instances. If a failed Siebel web server instance is restored, ELB restores the traffic to that instance.

Scalability

When using AWS you can scale your application easily because of the elastic nature of the cloud. You can scale up the Siebel web and application servers simply by changing the instance type to a larger instance type.

For example, you can start with an r4.large instance with 2 vCPUs and 15 GiB RAM, and scale up all the way to an x1e.32xlarge instance with 128 vCPUs and 3,904 GiB RAM. After selecting a new instance type, you only need a restart for the changes to take effect. Typically, the resizing operation is completed in a few minutes, the Amazon EBS volumes remain attached to the instances, and no data migration is required.

For your Siebel database deployed on Amazon RDS, you can scale the compute and storage resources independently. You can scale up the compute resources simply by changing the DB instance class to a larger one. This modification typically takes only a few minutes and the database will be temporarily unavailable during this period. You can increase the storage capacity and IOPS provisioned for your database without any impact on database availability.

You can scale out the web and application tier by adding and configuring more instances when you need them. The Siebel gateway name server keeps track of available application and web servers. These are registered with the Siebel gateway name server when the Siebel application server or Siebel web server is installed.

To meet extra capacity requirements, additional instances of Siebel web servers and application servers should be preinstalled and configured on EC2 instances. These “standby” instances can be shut down until extra capacity is required. You don’t incur charges when instances are shut down—you incur only Amazon Elastic Block Store (Amazon EBS) storage charges. At the time of this publication, EBS General Purpose volumes are priced at \$0.10 per GB per month in the US East (Ohio) Region. Therefore, for an instance with 120 GB of hard disk drive (HDD) space, the storage charge is only \$12 per month. These preinstalled standby instances provide you the flexibility to use them to meet additional capacity needs as and when you need them.

Architecting for High Availability and Disaster Recovery

In this section, we discuss best practices and options for deploying Siebel CRM on AWS for high availability of your Siebel application and for disaster recovery.

Multi-Availability Zone Deployment for High Availability of a Siebel Database on Amazon RDS

As described [earlier](#), each Availability Zone is isolated from other zones and runs on its own physically distinct, independent infrastructure. The likelihood of two Availability Zones experiencing a failure at the same time is very small. Like the Siebel web and application servers, you can deploy the Siebel database on Amazon RDS in a Multi-AZ configuration. Multi-AZ deployments provide enhanced availability and durability for Amazon RDS DB instances, making them a natural fit for production database workloads. When you provision a [Multi-AZ DB instance](#), Amazon RDS automatically creates a primary DB instance and synchronously replicates the data to a standby instance in a different Availability Zone. In case of an infrastructure failure (for example, instance hardware failure, storage failure, or network disruption), Amazon RDS performs an automatic failover to the standby instance. Because the endpoint for your DB instance remains the same after a failover, your application can resume database operations as soon as the failover is complete, without manual administrative intervention.

To learn how to set up Amazon RDS for Oracle as the database backend of your Siebel CRM application, see this [documentation](#).¹

Configuring the Siebel Gateway Name Server for High Availability

With bare metal implementations, you can deploy Siebel gateway name servers in an active/passive cluster to ensure availability in case of underlying host failure. When deploying on AWS, you have several options for configuring Siebel gateway name servers to ensure high availability.

You can use the EC2 automatic instance recovery feature to recover the Siebel gateway if the underlying host fails. Instance recovery performs several system status checks of the Siebel gateway name server instance and the other components that need to be running for the instance to function as expected. Among other things, instance recovery checks for loss of network connectivity,

loss of system power, and software and hardware issues on the physical host. If a system status check of the underlying hardware fails, the instance will be rebooted (on new hardware, if necessary) but will retain its instance ID, IP address, Elastic IP addresses, EBS volume attachments, and other configuration details.

Another option is to put the Siebel gateway name servers in an Auto Scaling group that spans multiple Availability Zones, and set the minimum and maximum size of the group to one. Auto Scaling ensures that an instance of the Siebel gateway name server is running in the selected Availability Zones. This solution ensures high availability of the Siebel gateway name server in the unlikely event of an Availability Zone failure.

Note: You should back up the `siebns.dat` configuration file to Amazon S3 before and after making any configuration changes, especially when creating new component definitions and adding or deleting Siebel servers. When the Siebel gateway name server is restored after a failure, it should update itself with the latest copy of `siebns.dat` from Amazon S3. You don't have to buy additional software or run additional passive instances while using instance recovery or a fixed-size Auto Scaling group for high availability.

Finally, you can configure high-availability clusters of the Siebel gateway name servers. There are several third-party products such as [SIOS](#)² and [SoftNAS](#)³ that offer a shared storage solution on AWS for clustering the Siebel gateway name servers.

Multi-Region Deployment for Disaster Recovery

Although a single AWS Region architecture with Multi-AZ deployment might suffice for most use cases, you might want to consider a multi-region deployment for disaster recovery (DR), depending on business requirements. For example, you might have regulatory requirements or a business policy that mandates that the DR site be located a certain distance away from the primary site.

Cross-region deployments for DR should be designed and validated for specific use cases based on your uptime needs and budget. The following diagram shows a typical Siebel deployment across regions that addresses both high availability and DR requirements.

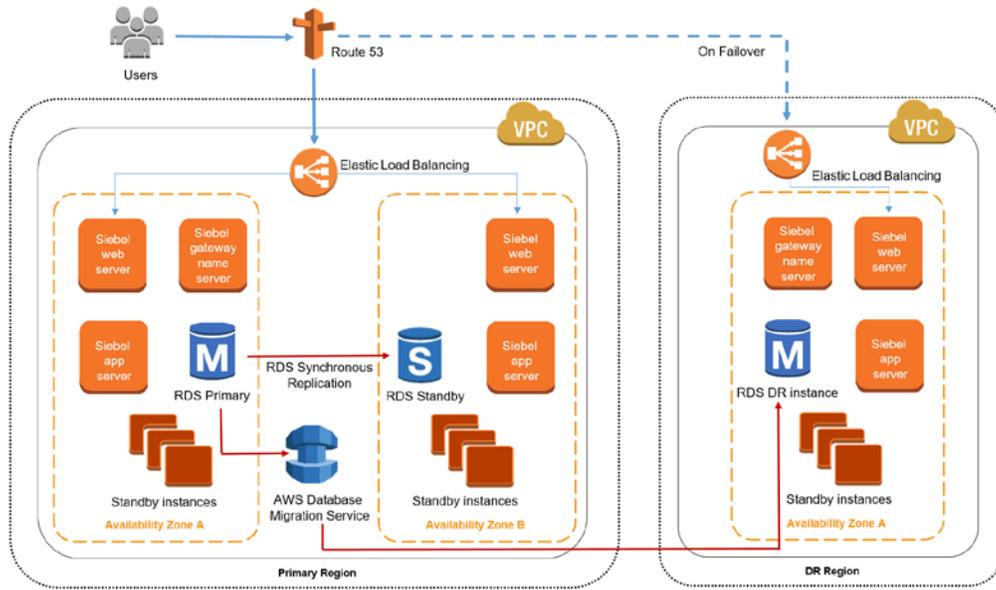


Figure 5: Multi-region deployment of Siebel on Amazon RDS for Oracle

In this scenario, users are directed to the Siebel application server in the primary region using Amazon Route 53. If the primary region is unavailable due to a disaster, failover is initiated and users are redirected to the Siebel application server deployed in the DR region. The primary database is deployed on Amazon RDS for Oracle in a Multi-AZ configuration. AWS DMS is used to replicate the data from the RDS DB instance in the primary region to another RDS DB instance in the DR region.

Note: AWS DMS can replicate only the data, not the database schema changes. The database schema changes in the RDS DB instance in the primary region should be applied separately to the RDS DB instance in the DR region. You can do this while updating the applications in the DR region.

Multi-Region Deployment of Siebel on Oracle Running on Amazon EC2 Instances

Although Amazon RDS for Oracle is the recommended option for deploying the Siebel database, there could be scenarios where Amazon RDS might not be suitable. For example, Amazon RDS might not be suitable in the unlikely scenario that the database size is close to or greater than the Amazon RDS for Oracle storage limit. In such scenarios, you can install the Siebel database on

Oracle on EC2 instances and configure Oracle Data Guard replication for high availability and DR, as shown in the following figure.

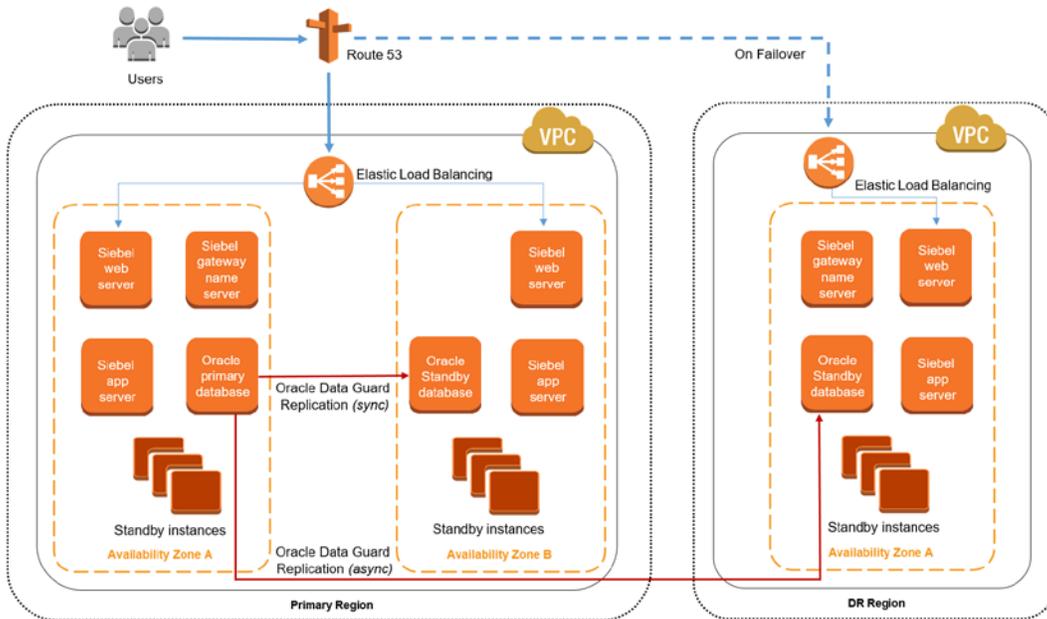


Figure 6: Multi-region deployment of Siebel on Oracle on Amazon EC2

In this DR scenario, the database is deployed on Oracle running on EC2 instances. Oracle Data Guard replication is configured between the primary database and two standby databases. One of the two standby databases is “local” (for synchronous replication) in another Availability Zone in the primary region. The other is a “remote” standby database (for asynchronous replication) in the DR region.

If the primary database fails, the local standby database is promoted as the primary database and the Siebel application server will connect to it. In the extremely unlikely event of a region failure or unavailability, the remote standby database is promoted as the primary database and users are redirected to the Siebel application server in the DR region using Route 53.

For more details on deploying Oracle Database with Data Guard replication on AWS, see the [Oracle Database on the AWS Cloud Quick Start](#).⁴

Refer to this [AWS whitepaper](#) to learn more about using AWS for disaster recovery.⁵

Using AWS as a DR Site and an On-Premises Production Environment

You can also deploy a DR environment on AWS for your Siebel applications running in an on-premises production environment. If the production environment fails, a failover is initiated and users are redirected to the Siebel application server deployed on AWS. The process is fairly simple and involves the following major steps:

- Setting up connectivity between your on-premises data center and AWS using a VPN connection or AWS Direct Connect
- Installing Siebel web, application, and gateway name servers on AWS
- Backing up siebns.dat to an Amazon S3 bucket
- Installing the standby database on AWS and configuring Oracle Data Guard replication or replication using AWS DMS between the on-premises production database and the standby database on AWS.

In this scenario, if the on-premises production environment fails, you can initiate a failover and redirect users to the Siebel application server on AWS.

VPC and Connectivity Options

Amazon VPC lets you provision a secure, private, isolated section of the AWS Cloud where you can launch AWS resources in a virtual network using IP address ranges that you define. Amazon VPC provides you with several options for securely connecting your AWS virtual networks with other remote networks. (Network security is discussed in greater detail in the section [Amazon VPC and Network Security](#).)

If users are accessing the Siebel application primarily from an office or on premises (e.g., a call center), you can use a hardware IPsec VPN connection or AWS Direct Connect to connect your on-premises network and Amazon VPC. If users are accessing the Siebel application from outside the office (e.g., a sales rep or customer accessing Siebel from the field or from home), you can use a software appliance-based VPN connection over the internet.

For detailed information about various connectivity options, see the whitepaper [Amazon Virtual Private Cloud Connectivity Options](#).⁶

Securing Your Siebel Application on AWS

The AWS infrastructure is architected to provide an extremely scalable, highly reliable platform that enables you to deploy applications and data quickly and securely.

Security in the cloud is slightly different from security in your on-premises data centers. When you move computer systems and data to the cloud, security responsibilities become shared between you and your cloud service provider. In this case, AWS is responsible for securing the underlying infrastructure that supports the cloud, and you are responsible for securing workloads you deploy in AWS. This shared security responsibility model can reduce your operational burden in many ways. It also gives you the flexibility to implement the most applicable security controls for your business functions in the AWS environment.

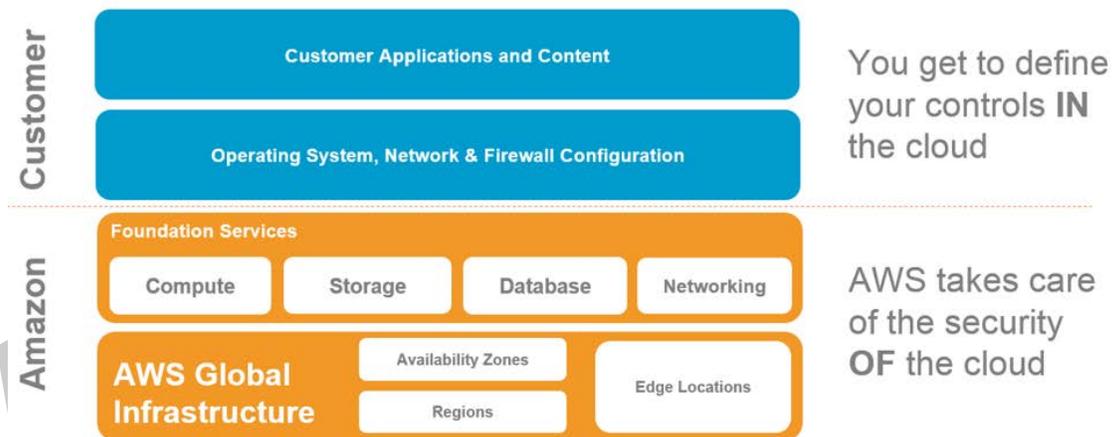


Figure 7: AWS shared responsibility model

We recommend that you take advantage of the various security features that AWS offers when deploying Siebel CRM on AWS. You can use the following AWS security features to control and monitor access to the infrastructure components of your Siebel deployment (e.g., OS-level access to your Siebel application servers, network-level security, limiting access to AWS services such as Amazon EC2, Amazon RDS, Amazon S3, etc.). The [Siebel application security architecture](#) (user authentication, authorization, field-level encryption, etc.) does not change when you deploy your Siebel application on AWS—you configure and manage it the same way as you would on premises.⁷

IAM

When you deploy your Siebel application on AWS, you can use AWS Identity and Access Management (IAM) to control access to the AWS environment in which your Siebel servers are deployed. With IAM, you can centrally manage users and security credentials (such as passwords, access keys, and permissions policies) that control which AWS services and resources users can access. IAM supports multi-factor authentication for privileged accounts. This includes options for hardware-based authenticators and support for integration and federation, with corporate directories to reduce administrative overhead and improve end-user experience.

Monitoring and Logging

You can use AWS CloudTrail for resource change tracking and compliance auditing of the AWS infrastructure components of your Siebel environment (such as Amazon EC2, Amazon RDS, Amazon S3, etc.). For Siebel application-level auditing, you can continue to use the [Siebel Audit Trail](#) feature.⁸

AWS CloudTrail is a web service that records AWS API calls for your AWS account and delivers log files to you. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service. This provides deep visibility into API calls, including who, what, when, and from where calls were made. The AWS API call history produced by CloudTrail enables security analysis, resource change tracking, and compliance auditing.

Amazon VPC and Network Security

Amazon VPC enables you to provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. It offers you an IPsec VPN device to provide an encrypted tunnel between the Amazon VPC and your data center.

You create one or more subnets within each Amazon VPC. Each instance launched in the Amazon VPC is connected to one subnet. Traditional Layer 2 security attacks, including MAC spoofing and ARP spoofing, are blocked. You can configure network access control lists (network ACLs), which are stateless traffic filters that apply to all traffic inbound or outbound from a subnet within Amazon VPC. These network ACLs can contain ordered rules to allow or deny

traffic based on IP protocol, by service port, as well as source/destination IP address.

Security groups are a complete firewall solution enabling filtering on both ingress and egress traffic from an instance. Traffic can be restricted by any IP protocol, by service port, and by source/destination IP address (individual IP or CIDR block).

Data Encryption

AWS offers you the ability to add a layer of security to your data at rest in the cloud by providing scalable and efficient encryption features. Data encryption capabilities are available in AWS storage services such as Amazon EBS, Amazon S3, and Amazon Glacier, and database services such as Amazon RDS for Oracle and Amazon RDS for SQL Server for use with the Siebel database. You can choose whether to have AWS manage your encryption keys using AWS Key Management Service (AWS KMS), or you can maintain complete control over your keys. Dedicated, hardware-based cryptographic key storage options (AWS CloudHSM) are available to help you satisfy compliance requirements.

For more information on AWS security, see the [Introduction to AWS Security](#)⁹ and [AWS Security Best Practices](#)¹⁰ whitepapers.

Siebel and Oracle Licensing on AWS

In this section, we will briefly discuss Siebel CRM and Oracle Database license portability and Amazon RDS for Oracle licensing models.

Siebel and Oracle Database License Portability

Most Oracle software licenses are fully portable to AWS, including the Enterprise License Agreement (ELA), Unlimited License Agreement (ULA), Business Process Outsourcing (BPO), and Oracle Partner Network (OPN). You can use your existing Siebel license and Oracle database licenses on AWS. However, you should consult your own Oracle license agreement for specific information.

Amazon RDS for Oracle Licensing Models

You can deploy your Siebel CRM applications on Amazon RDS for Oracle under two different licensing models: “License Included” and “Bring-Your-Own-License (BYOL)”. In the License Included service model (available only for Oracle Standard Edition One and Oracle Standard Edition Two), you don’t need to separately purchase Oracle licenses. The Oracle Database software has been licensed by AWS.

If you already own Oracle Database licenses, you can use the BYOL model to run Oracle databases on Amazon RDS. The BYOL model is designed for customers who prefer to use existing Oracle database licenses or purchase new licenses directly from Oracle.

Siebel on AWS Use Cases

The following are some of the common use cases for Siebel on AWS:

- **Migrate existing Siebel environments to AWS** – This is most suitable if you are on a recent release of Siebel. You should design your AWS deployments based on the best practices in this whitepaper.

For migrating large databases to Amazon RDS within a small downtime window, we recommend that you take a point-in-time export of your database, transfer it to AWS, import it into Amazon RDS, and then apply the delta changes from on premises.

You can use [AWS Direct Connect](#) or [AWS Snowball](#) to transfer the export dump to AWS.

You can use AWS DMS to apply the delta changes and sync the on-premises database with the Amazon RDS instance.

- **Siebel upgrade** – You can leverage AWS as the upgrade environment to keep the costs of upgrade to a minimum. You can either use this new environment only for test and development, or you can migrate your entire Siebel environment to AWS. Either way, you can reduce your overall TCO.
- **Performance testing** – Most customers only do performance testing for Siebel changes either on initial implementation or when they have Siebel upgrades to put in place. Performance testing for customer

enhancements is almost never continually done. AWS enables you to do performance testing at minimal cost because you are only charged for the resources you use, when you use them. This minimal cost enables more realistic testing, both for Siebel upgrades and for your own enhancements. You can budget for this on an annual basis, depending on your needs, for example, when Siebel repository file (SRF) changes are put in place. With additional real-world testing of your own planned changes or enhancements, you can reduce performance issues and avoid business-critical downtimes.

- **Siebel test and development environments on AWS** – You might want to set up test and development environments on AWS just to try AWS, or if the move of the production environment isn't urgent.
- **Disaster recovery on AWS** – You might want to set up a DR environment for your existing Siebel CRM on AWS. This can be done at a much lower cost than setting up traditional DR.

Monitoring Your Infrastructure

You can continue to use the existing tools that you are familiar with for monitoring your Siebel application, such as the Siebel Web Server Extension (SWSE) statistics page, the Server Manager GUI, or the Server Manager (srvrmgr) command-line interface. Optionally, you can use Oracle Enterprise Manager to monitor your Siebel environment by installing the Oracle Enterprise Manager Plug-in for Oracle Siebel. You can also use Amazon CloudWatch to monitor AWS Cloud resources and the applications you're running on AWS.

Amazon CloudWatch enables you to monitor your AWS resources in near real time, including EC2 instances, EBS volumes, load balancers, and RDS DB instances. Metrics such as CPU utilization, latency, and request counts are provided automatically for these AWS resources. You can also supply your own logs or custom application and system metrics, such as memory usage, transaction volumes, or error rates. Amazon CloudWatch will monitor these also.

You can use the Enhanced Monitoring feature of Amazon RDS to monitor your Siebel database. Enhanced Monitoring gives you access to over 50 metrics, including CPU, memory, file system, and disk I/O. You can also view the processes running on the DB instance and their related metrics, including percentage of CPU usage and memory usage.

AWS and Oracle Support

In this section, we discuss the support model when you deploy your Siebel CRM applications on AWS.

AWS Support

[AWS Support](#) is a one-on-one, fast-response support channel that is staffed around the clock with technical support engineers and experienced customer service professionals, who can help you get the most from the products and features provided by AWS.¹¹

All AWS Support tiers offer an unlimited number of support cases with pay-by-the-month pricing and no long-term contracts. The four tiers provide developers and businesses the flexibility to choose the support tiers that meet their specific needs.

AWS Support Business and Enterprise levels include support for common operating systems and common application stack components. AWS Support engineers can assist with the setup, configuration, and troubleshooting of certain third-party platforms and applications, including Red Hat Enterprise Linux, SUSE Linux, Windows Server 2008, Windows Server 2012, Windows Server 2016, Open VPN, RRAS, etc.

Oracle Support

Siebel CRM versions 15.0 and 16.0 are certified to run on AWS. Oracle's certification details for Siebel on AWS are available in the certification section of the [Oracle Support site](#).¹²

You can use the existing licenses for Siebel applications that you had with your on-premises implementations. You will have the same level of Oracle Support that you had with your on-premises implementation. Oracle's only requirement for Infrastructure as a Service (IaaS) clouds is that you use platforms and databases that are certified with Siebel. Certified versions of both Siebel and platforms and databases are documented on the Oracle support site.

You can submit issues in the same manner and provide information about your environments as before. When you contact Oracle Support, the fact that you are

running your Siebel CRM application in the cloud might not even enter the discussion, because there is nothing unique about using IaaS that would require any change to the application. This is the same approach for virtualization technology that Oracle Support has followed with Siebel for many years. Escalations would continue to go through the customer support site.

Conclusion

By deploying Siebel in the AWS Cloud, you can simultaneously reduce cost and enable capabilities that might not be possible or cost effective if deployed in an on-premises data center. Some benefits of deploying Siebel on AWS include:

- Low cost—resources are billed by the hour and only for the duration they are used.
- Changing from CapEx to OpEx eliminates the need for a large capital layout.
- Higher availability of 99.99% by deploying Siebel in a Multi-AZ configuration.
- Flexibility to add capacity elastically to cope with demand. This enables you to perform application upgrades faster.
- Flexibility to add environments and use them for short durations, such as for performance testing and training.

Contributors

The following individuals and organizations contributed to this document:

- Ashok Sundaram, Solutions Architect, Amazon Web Services
- Yoav Eilat, Sr. Product Marketing Manager, Amazon Web Services
- Mark Farrier, Director Product Management – Siebel CRM, Oracle
- Milind Waikul, CEO, Enterprise Beacon, Inc.

Further Reading

For additional information, see the following sources:

- Test drive Siebel running on Amazon EC2 and Amazon RDS
<http://www.enterprisebeacon.com/testdrives.html>
- Amazon EC2
<https://aws.amazon.com/ec2/>
- Amazon RDS
<https://aws.amazon.com/rds/>
- Amazon CloudWatch
<https://aws.amazon.com/cloudwatch/>
- AWS DMS
<https://aws.amazon.com/dms/>
- Elastic Load balancing
<https://aws.amazon.com/elasticloadbalancing/>
- Amazon EBS
<https://aws.amazon.com/ebs/>
- Amazon S3
<https://aws.amazon.com/s3/>
- Amazon Route 53
<https://aws.amazon.com/route53/>
- Amazon VPC
<https://aws.amazon.com/vpc/>
- AWS Direct Connect
<https://aws.amazon.com/directconnect/>
- AWS CloudTrail
<https://aws.amazon.com/cloudtrail/>
- AWS CloudHSM
<https://aws.amazon.com/cloudhsm/>
- Amazon Glacier
<https://aws.amazon.com/glacier/>
- AWS KMS
<https://aws.amazon.com/kms/>
- AWS Cost Estimator
<http://calculator.s3.amazonaws.com/index.html>

- AWS Trusted Advisor
<https://aws.amazon.com/premiumsupport/trustedadvisor/>
- Oracle cloud licensing
<http://www.oracle.com/us/corporate/pricing/cloud-licensing-070579.pdf>
- Oracle Processor Core Factor Table
<http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf>
- Amazon EC2 virtual cores by instance type
<https://aws.amazon.com/ec2/virtualcores/>
- Oracle Database on the AWS Cloud Quick Start (with Data Guard replication)
<https://s3.amazonaws.com/quickstart-reference/oracle/database/latest/doc/oracle-database-on-the-aws-cloud.pdf>

Notes

1

<http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Oracle.Resources.Siebel.html>

2 <http://us.sios.com/clustersyourway/products/windows/datakeeper-cluster>

3 <https://aws.amazon.com/whitepapers/softnas-architecture-on-aws/>

4 <https://s3.amazonaws.com/quickstart-reference/oracle/database/latest/doc/oracle-database-on-the-aws-cloud.pdf>

5 <https://d0.awsstatic.com/whitepapers/aws-disaster-recovery.pdf>

6 <https://d0.awsstatic.com/whitepapers/aws-amazon-vpc-connectivity-options.pdf>

7

https://docs.oracle.com/cd/E74890_01/books/Secur/secr_aboutsec005.htm

8

https://docs.oracle.com/cd/E74890_01/books/AppsAdmin/AppsAdminAuditTrail2.html

9 http://d0.awsstatic.com/whitepapers/Security/Intro_to_AWS_Security.pdf

10

https://d0.awsstatic.com/whitepapers/Security/AWS_Security_Best_Practices.pdf

11 <https://aws.amazon.com/premiumsupport/>

12 <http://support.oracle.com/>