



SERENA SOFTWARE

Scaling for the Enterprise

**Authors: Bill Weingarz, Pete Dohner, Kartik Raghavan, Amitav
Chakravartty**

2012-05-07

Table of Contents

Who Should Read This Paper?	3
Introduction	3
Architecture.....	3
Steps for Implementing Applications	5
Path to Production	6
Security.....	6
Introduction to Scalability.....	7
Hardware Requirements	8
Increasing Scalability	9
A General Approach For Scalability	11
Case Study: Serena Business Manager SAAS Deployment	14
Summary	15

Who Should Read This Paper?

This paper provides information on scaling Serena Business Manager to meet the performance and scalability needs of the enterprise. This includes a brief overview of the components and architecture followed by a description of the scalability options.

This information is relevant to technical contributors including sales architects, services, support, customers, and partner/alliances professionals.

Introduction

IT organizations are challenged by a growing backlog of requests from lines of business. They are faced with tighter budgets and limited resources. Because the startup cost of IT projects is high, projects that the IT organization undertakes must demonstrate a high return on investment. This leaves many requests from the business unattended.

SBM is a powerful solution from Serena Software, Inc. that empowers business users to collaborate with IT in building out applications to solve business problems. It includes key capabilities for creating workflows, orchestrations, and forms as well as SBM Application Repository for streamlined management of application staging and deployment.

SBM enables the creation of a new breed of business applications which are:

- Visually composed and assembled.
- A combination of content and services from multiple applications and services.
- Delivered on-premise or on-demand.
- Process-centric and data-enabled.
- Delivered through multiple channels such as e-mail, browser, IM, and mobile device.

SBM promotes a process-focused approach to productivity. It coordinates processes across teams and systems and provides repeatability, traceability, and auditability. To meet varying configuration, performance, and scalability needs of the enterprise, SBM employs several techniques including:

- N-Tier architecture
- Process app development lifecycle
- Path to production
- Security
- Scalability

Architecture

SBM employs a three-tier configuration which consists of the SBM Server, the database server, and one or more client machines.

The SBM Server is implemented using J2EE and ISAPI extensions. The primary components of SBM are:

- **SBM Application Engine** – Executes as an in-process ISAPI plug-in to Internet Information Services (IIS), which powers the SBM User Workspace. It executes processes and responds to HTTP and Web service requests from the IIS Web server. This is the entry point into the SBM Server for SBM User Workspace and SBM System Administrator users. The SBM Application

Engine primarily coordinates human workflow activities in a process app. SBM Application Engine components you will commonly use include:

- **SBM User Workspace** – End-users interact with SBM through the SBM User Workspace. The SBM User Workspace is the client component from which users can access deployed process apps. The interface is completely implemented using HTML and JavaScript, and requires no client installation. (This enables any client with a standard Web browser to render the SBM User Workspace). The SBM User Workspace uses AJAX (Asynchronous JavaScript and XML) to deliver a rich user experience. Adobe Flash is required for users to render Rich Graphical Reports in the SBM User Workspace.
- **SBM Application Administrator** – Enables Administrators to perform application configuration tasks—such as adding projects and assigning them to workflows, creating user accounts and assigning them to roles and groups, and creating notifications.
- **Serena Common JBoss** – The JBOSS server is a J2EE container for a number of services provided in SBM Server.

These services include:

- **SBM Application Repository** – The browser client from which administrators control the deployment of process apps into multiple environments. The interface uses advanced AJAX techniques and a single-page interface to deliver a rich internet application. The SBM Application Repository provides repository, deployment, promotion, and logging services.

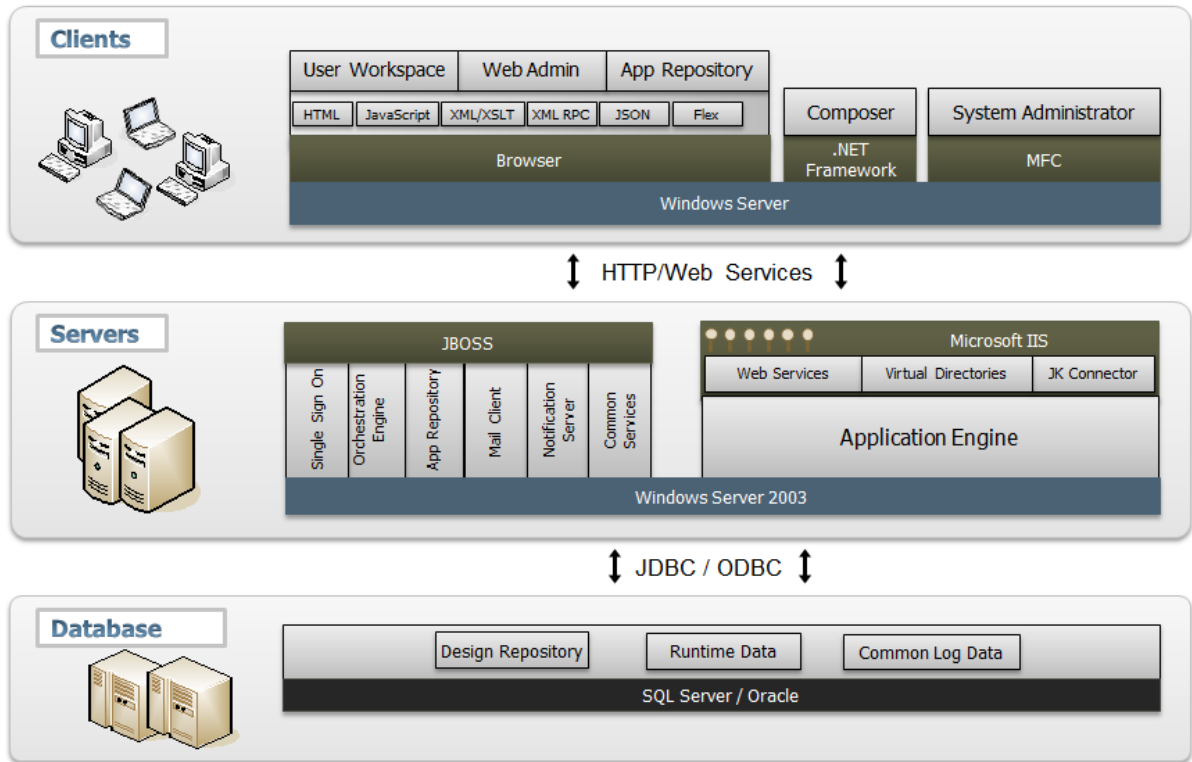
The repository is used by SBM Composer and SBM Application Repository to store design and deployment artifacts. SBM Composer accesses the repository via Web services. The SBM Application Repository client uses deployment and promotion services to provide business users and administrators with a path to production. Once process apps are built and versioned from SBM Composer, you deploy them to the SBM Server, which makes them available to end users in the SBM User Workspace. See [Steps for Implementing Applications \[page 5\]](#) for more information.

Users can access the SBM Application Repository using the following URL in a standard Web browser: `http://serverName:port/mashupmgr`.

- **SBM Orchestration Engine** – Receives events from applications and executes associated orchestrations as BPEL processes. These BPEL processes use Web services to execute a business process across multiple tools.
- **Single Sign-On (SSO)** – Provides common authentication across all components of SBM. SSO can be extended to include other applications by writing or configuring providers.
- **SBM Common Services** – Powers the proxy server (which is used by the REST Grid Widget) and the PDF Widget (which generates PDF documents from data in the SBM User Workspace).
- **SBM Mail Services** – Consists of the Notification Server and Mail Client. The Notification Server evaluates notification and escalation rules and performs specified actions such as sending reminders via e-mail or performing specific actions like running scripts, or calling Web service functions. The Notification Server communicates with mail servers using either SMTP or Exchange (via Web services). The Mail Client provides inbound services in which items can be submitted by sending an e-mail to the Mail Client e-mail box. The Mail Client communicates with mail servers using one of the following protocols: POP3, IMAP, or Exchange (via Web services).

The SBM client component is SBM Composer. SBM Composer is a modern smart-client application built using the Microsoft .NET platform. It provides an extremely rich interface with features such as

ribbon bars, drag and drop, zoom and unlimited undo/redo. SBM Composer is the design environment from which you can find, customize, and create process apps. SBM Composer uses Web services to interact with the SBM Server. It can be deployed in a variety of network topologies since it interacts with the SBM Server using industry standard HTTP and Web services protocols.



The SBM database resides in a relational database. Data design includes storage of system metadata information along with user data. Users can access data using Web services, APIs, or direct SQL access.

Detailed information about supported platforms and software configuration is available in the Supported Platform Matrix, which is available from the **My Downloads** tab at <http://www.serena.com/support>.

Steps for Implementing Applications

SBM separates application design from configuration and use. This offers many benefits, including the ability to aggregate many changes and deploy them at once. You can also deploy your changes to a test environment before deploying changes into your production environment.

Once SBM is installed, there are five main steps to implementing an application:

1. **Design** – All design tasks take place in SBM Composer. Application designers use SBM Composer to create and edit existing applications by adding workflows, tables, fields, roles, custom forms, and more. Orchestrations that integrate with external systems or other applications can also be added to your process app during the design phase.
2. **Publish** – After designing an application, you publish it as part of the process app in SBM Composer. Publishing takes completed process apps, moves them into the repository, and makes them visible in Application Repository. The published process app is versioned in the

repository, meaning that the particular set of changes is saved as one version. This enables another designer to open the process app from the repository and make changes, if necessary.

3. **Deploy** – You can deploy a process app and its applications to SBM Server, also known as the runtime environment. Deployment activates the applications in the SBM User Workspace, pushing the changes made in SBM Composer to end users. Depending on how you configure your environments, you can deploy directly from SBM Composer or from Application Repository.



Note: The **Quick Deploy** feature in SBM Composer enables you to validate, publish, and deploy process apps in one step.

4. **Configure** – Use SBM Application Administrator to configure deployed applications. Configurations cover four general areas: user and group management, project configuration, field overrides, and notification management. SBM Application Administrator is accessed via the SBM User Workspace.
5. **Use** – Once you've configured an application, it is ready for users in the SBM User Workspace.

Path to Production

Best practices in software management advocate multiple environments for development, testing, staging, and production where changes are propagated from one environment to another in an automated manner. SBM Server provides full fidelity promotion of a process app and all related configuration data (called a snapshot) from one environment to another. These snapshots can also be exported and imported into the SBM Server in case there is a firewall in between environments. For detailed information on creating an installation that supports path to production, see "Implementing SBM in a Multiple Environment Setting" in the *SBM Installation and Configuration Guide*.

Security

SBM provides authenticated users with appropriate access to data. This is accomplished via authentication and a highly flexible privilege system.

- **Network Security** – Clients can use HTTP over Secure Sockets Layer (SSL) encryption to communicate with the SBM Server. With SSL, all data transmitted between the SBM Server and the clients is encrypted.
- **User Authentication** – Every request to the SBM Server is authenticated using one of the following schemes:
 - **Internal Passwords** – User credentials can be validated against those stored within SBM. Passwords within the SBM Server are stored using a hashing algorithm (SHA-256). For more information, see <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf>.
 - **NT Challenge/Response** – The Windows security system integrated into IIS can be used to authenticate user access. In this case, IIS verifies login names and passwords against Windows user accounts.
 - **Central Directory via LDAP** – SBM Server can be configured to use LDAP systems for authentication. The system can be configured to use LDAP authentication alone or LDAP followed by Internal Passwords authentication.

-
- **Data and Function Security** – SBM provides granular control over privileges for all logical components within the SBM Server. Users may be granted varying roles which allow them access to particular areas of the system. The scope of a particular privilege may be limited to one or more projects within a process app. The system provides exceptional configurability to control authorization down to the field level.

The SBM Server also provides a complete audit trail and logging for all business and administrative events. Users may view the change history for all items and field values. Business data is retained indefinitely and may be archived as needed by the system administrator.

Introduction to Scalability

Applications need to scale well in order to be successfully deployed in enterprise environments. This is a complex task since many factors are involved: network hardware and software, WAN latencies, server hardware and software, network load, server load and data volume. In addition, each type of application activity utilizes resources in a unique way. For example, a process app with many human workflow interactions will consume more resources in the SBM Application Engine. Whereas a process app with complex orchestrations will require more resources to be allocated to Serena Common JBoss.

The following factors must be carefully considered by IT professionals along with Serena Professional Services when planning a scalable deployment:

- **Network Configuration/Topology** – Serena Business Manager is distributed across clients, servers, and one or more database servers. Consequently, network hardware and software configuration and capabilities have a significant impact on the performance characteristics. The closer in proximity on the network these pieces are the better the performance. As a general rule, the SBM Server and SBM Database should be located on the same network segment.
- **CPU speed** – Server and database transactions are CPU intensive. Faster CPUs translate directly to improved server performance. Do not under invest in the quality and speed of the server CPUs. The more processes that are running on a single machine, the higher the CPU demand will be once additional load is applied.
- **Software Configuration** – Serena Business Manager is a highly configurable product. While being one of the product's greatest advantages, it can lead to performance issues if not implemented in accordance with best practices. Serena Customer Support and Professional Services are well versed in these best practices and should be engaged to review implementations and help diagnose performance issues.
- **Usage Model** – The addition of SBM Composer and the SBM Orchestration Engine has provided users with the ability to create a variety of process apps. Serena Business Manager is used by each customer to solve different types of business problems. There are many possible usage models and each one will utilize systems resources in a unique way. Understanding the usage model of the user base will help in determining how best to scale the servers.

Serena utilizes standard Microsoft and J2EE technologies and approaches for building and deploying a scalable enterprise solution. The Windows 2003 Server and Windows 2008 Server platforms provide industry-leading capabilities to manage and grow enterprise application performance and availability in an economical way. The price and performance of Microsoft SQL Server and Windows 2003/2008-based servers are consistently among the best as measured by the Transaction Processing Performance Council (TPC; reference <http://www.tpc.org>).

Hardware Requirements

The hardware requirements for SBM running on a Windows 2003 or Windows 2008 32-bit operating system are as follows:

- **Recommended Requirements** – 2 GHz or higher multi-processors; 4 GB memory; 10 GB operational disk space.
- **Minimum Requirements** – 800 MHz or higher single processor; 2 GB memory; 2.5 GB operational disk space.

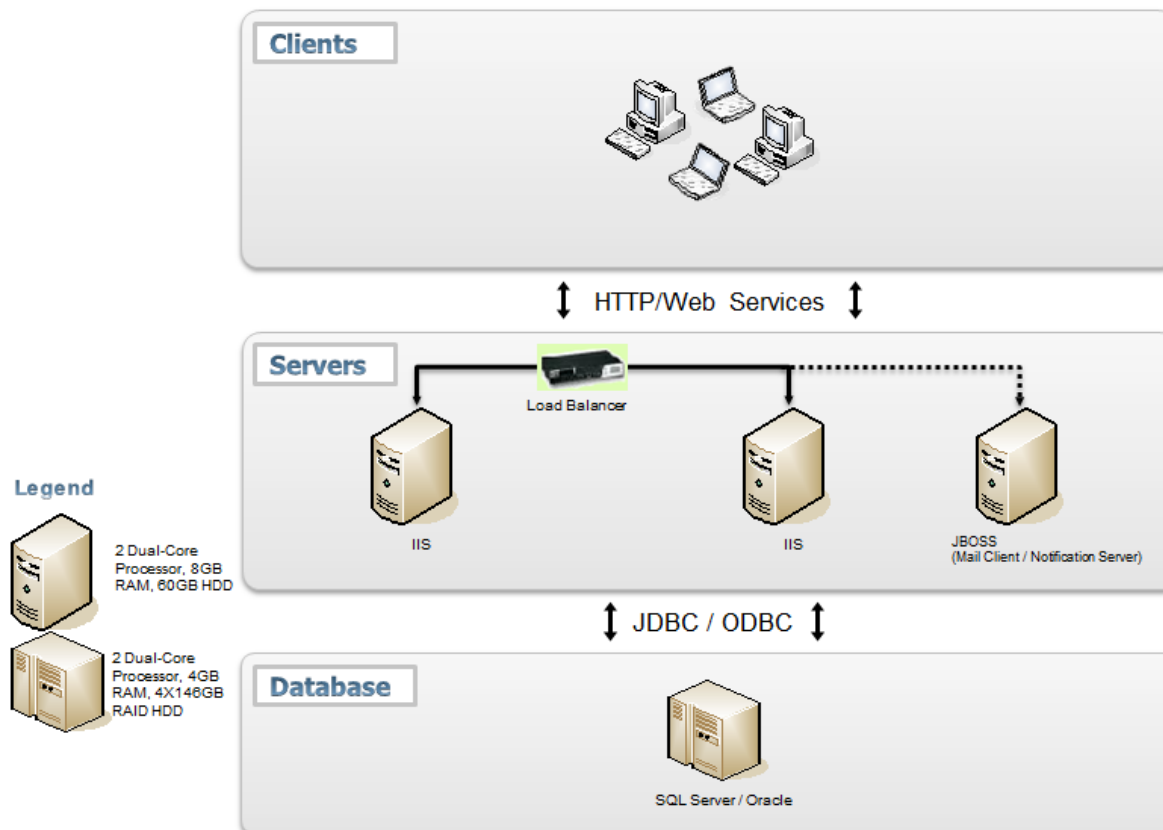
The hardware requirements for SBM running on a Windows 2008 R2 64-bit operating system are as follows:

- **Recommended Requirements** – 2 GHz or higher multi-processors; 16 GB memory; 10 GB operational disk space.
- **Minimum Requirements** – 800 MHz or higher single processor; 8 GB memory; 2.5 GB operational disk space.

When you install all SBM Server components on one machine, the recommended requirements will allow SBM Server to have an average response time under a normal load. With the minimum requirements, the response time will be much slower; however, SBM Server will continue to run. You can improve performance and accommodate greater usage by scaling your system vertically and horizontally.

For high load situations, it is recommended that you place your SBM Server and database on different machines.

The following diagram depicts a typical SBM production environment configuration. This configuration is suitable for process design with a level of high user interaction, but low Orchestration Engine usage:



Increasing Scalability

Once you have an understanding of your usage model and you have taken an inventory of what machines are available in your SBM system, you can begin to scale your environment accordingly.

Review the following sections to help you increase the scalability of your enterprise environment.

VERTICAL SCALING

Vertically scaling your environment involves applying higher-powered hardware on the SBM server. This involves implementing faster CPUs, multiple CPUs, more memory, faster network cards, or more likely, some combination of all of these. Consider the following areas:

- **Processors** – The SBM Server can take advantage of additional processors to improve overall performance. Using multiple processors enables you to scale your system vertically, provided your system has the capability to upgrade or expand the number of processors. For smaller installations in which all the SBM components are installed on a single server, you will find that implementing more processors enables your server to handle a higher work load.
- **Networking** – System networking is an important aspect of scaling. If the network card is saturated with network traffic in your configuration, you can add an additional network card to help scale the application. It is highly recommended that you place the SBM Server and Database server on a single subnet that has low latency.
- **Disk Subsystem** – The disk subsystem in a single system configuration can also be used to get additional scalability. The faster the disk drive speed, the faster disk I/O activity. Separating database data and logs onto separate physical drives (or spindles) will allow the system to operate more efficiently. If possible, the operating system should be placed on a drive of its

own. Reducing contention for data access from the physical disk drive will improve the scalability of the system.

- **Memory** – As of SBM 2009 R4, the following Windows server platforms are supported: Windows Server 2003, Windows Server 2008, and Windows Server 2008 R2 (for 64-bit implementations of SBM). For a comprehensive list of memory limits for each platform, see <http://msdn.microsoft.com/en-us/library/aa366778%28VS.85%29.aspx>.



Note: Ensure that each process that you scale vertically has adequate memory. Note that the per process memory limitations are reached much more quickly on 32-bit based systems compared to 64-bit systems.

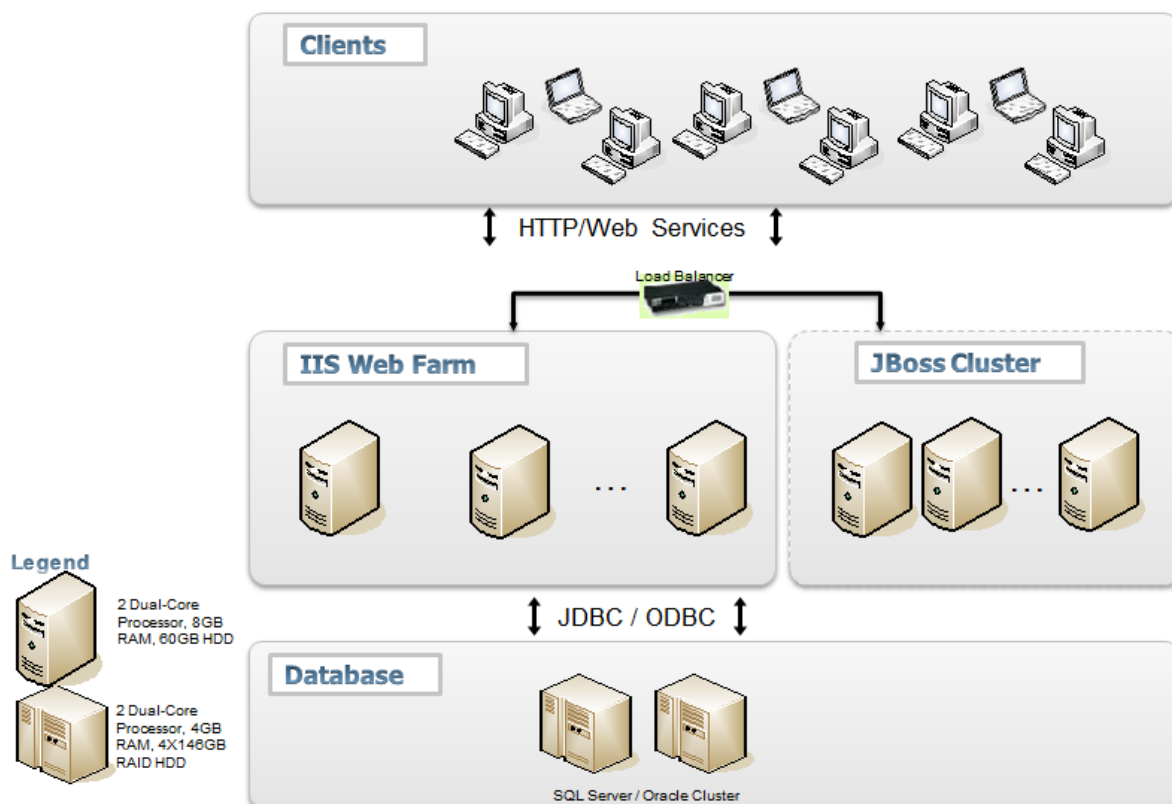
HORIZONTAL SCALING

To achieve the highest levels of concurrency, scale your servers horizontally by moving some components to separate machines. Since the SBM Server and Database processes contend for the most CPU and disk I/O, you should separate these components first.

You can scale the SBM Application Engine even further by implementing a Web farm that contains multiple IIS servers or JBoss clusters using either hardware or software. Hardware Web farms tend to be more costly to implement because they require you to have unique skills and specialized load-balancing hardware. The network load balancing capabilities of these Windows servers automatically distribute the load across the servers in the farm. A Web farm implementation is also transparent to end users—once the Web farm is implemented, users will continue to access the system via a single URL. Using a load-balancer also provides failover capabilities. In the event that one of the IIS servers becomes unavailable, the load-balancer redirects incoming requests to other available servers.

For deployments with complex or high volume orchestration activity, additional JBoss servers can be configured. You can control the amount of activity that is directed to any one JBoss server by specifying particular end-points during the process app deployment process. Contact Support or search the Knowledgebase at serena.com for more information on clustering the JBoss application server components or launching a backup Java Virtual Machine (JVM) that provides failover for JBoss. As of SBM 10.1.1.1, support for creating and configuring JBoss clusters using SBM Configurator is available.

You can install the SBM Mail Services on separate machines to move the processing overhead of these services to another system. As of SBM 10.1.1.1, you can also cluster the SBM Mail Services to increase scalability and reliability.



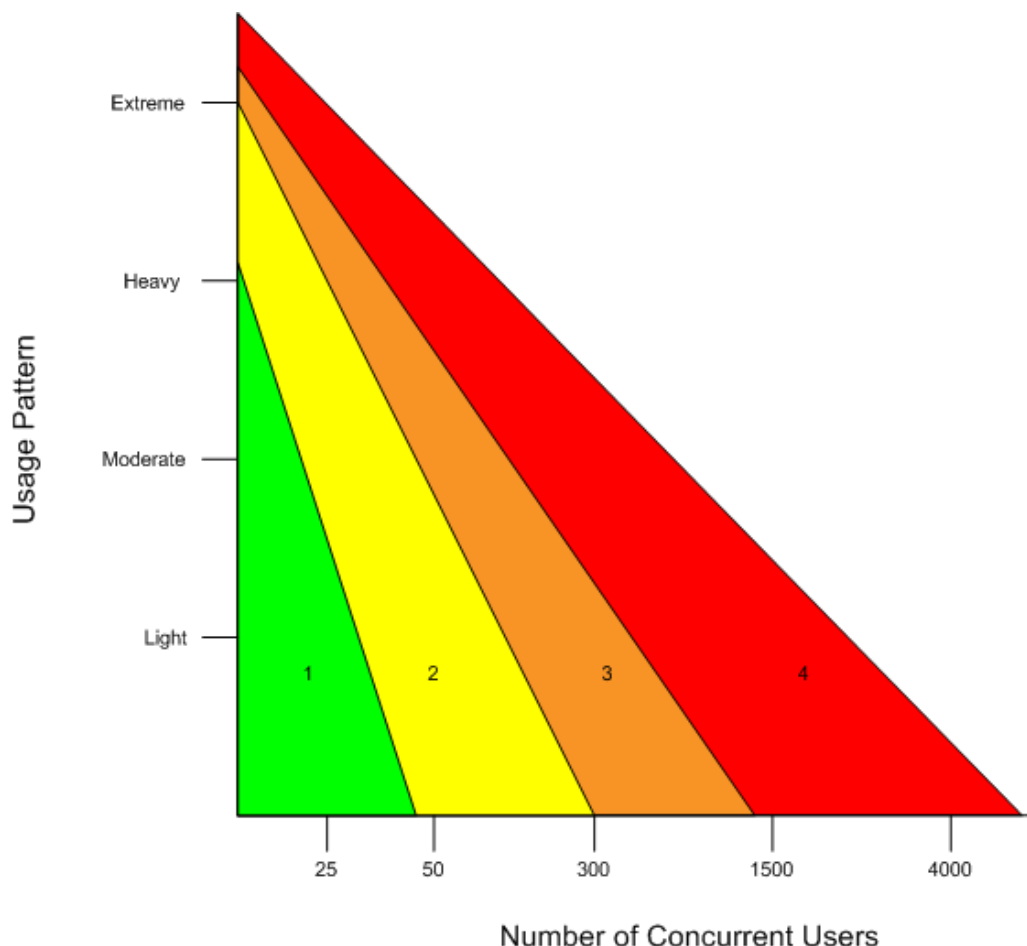
A General Approach For Scalability

When optimizing performance, focus on obtaining servers with the fastest possible CPUs first. Second, examine memory utilization closely and ensure that your servers have enough available memory. Next, consider implementing multiple processor configurations. You should apply these principles when scaling your servers both vertically and horizontally. Any single SBM operation is optimized for performance when CPU speed, network bandwidth, and available memory are high. Using the fastest possible CPU at the client, server, and database tiers results in the best overall end user experience. The following chart illustrates sample scalability zones for 32-bit and 64-bit SBM implementations.

Zone	Configuration
1	<ul style="list-style-type: none"> • Single system with all SBM components installed • Windows 2003 Server (32-bit), Single CPU 2GHz, 4GB RAM • Windows 2008 R2 Server (64-bit), Single CPU 2GHz, 16GB RAM

Zone	Configuration
2	<ul style="list-style-type: none">• Single system with all SBM components installed• Windows 2003 Server (32-bit), Quad CPU 3.2GHz, 4GB RAM• Windows 2008 R2 Server (64-bit), Single CPU 2GHz, 16GB RAM
3	<ul style="list-style-type: none">• Web Server: Windows 2003 Server (32-bit), Quad CPU 3.2GHz, 4GB RAM• Web Server: Windows 2008 R2 Server (64-bit), Single CPU 2GHz, 16GB RAM• Database Server: Windows 2003 Server (32-bit), Quad CPU 3.2GHz, 4GB RAM
4	<ul style="list-style-type: none">• 4+ system IIS Web Farm and Database Server• Web Server: Windows 2003 Server (32-bit), Dual CPU 3.2GHz, 4GB RAM• Web Server: Windows 2008 R2 Server (64-bit), Dual CPU 3.2GHz, 16GB RAM• Database Server: Windows 2003 Server (32-bit), Quad CPU 3.2GHz, 4GB RAM

Figure 1. Serena Business Manager Scalability Zones



In this chart there are four zones. Zone 1 represents low concurrency and simple configuration and usage patterns. Zone 4 is at the highest end of the spectrum. It represents high concurrent user load and complex data and usage patterns.

Use this table to determine appropriate server configurations based on usage patterns and the expected number of concurrent users. The configuration recommendations are the result of data analysis collected as part of the rigorous SBM acceptance and certification process. The data set that was used was enterprise-sized and reflected configurations common to many customers.

Concurrent users are users that are actively performing work in SBM (for example, two users running a report at the same time). Additional users can be logged into the system but are not considered "concurrent" until they perform some activity in SBM. You must consider the usage pattern and number of concurrent users when determining your optimal hardware configuration for SBM. For example, in order to support a population of 250 users in which 25 users are performing moderate operations at any given time—submitting new requests, updating records, reporting, querying for a record—start at Zone 1. To scale up to a 20,000 user population, Zone 3 (at 300 concurrent users) would be more appropriate.

Case Study: Serena Business Manager SAAS Deployment

Serena offers SBM in a Software as a Service (SaaS) model. This section describes the deployment architecture that Serena uses to host SBM in a way that can scale to support hundreds of distinct customers, each having hundreds of users on a single deployment of the SBM Server.

The SaaS model requires that the system is highly fault tolerant and secure. It must also ensure good performance under high concurrent load while supporting a large number of customers and users. Also, because SBM is a platform, customers can choose to create and deploy large and complex process apps that may require significant memory and CPU resources.

SBM is designed to operate as a multi-tenant server. This means that a single deployment of the SBM Server can be used to serve many different customers that are all isolated from each other. Using many of the scaling techniques described in this paper (including clustering, vertical scaling, and load balancing), Serena hosts a highly-scaled and secure deployment of our software that simultaneously supports all of our on-demand customers including trials and paid accounts. The initial configuration described below is designed to support 100,000 users which belong to multiple customers.

Serena also offers a hybrid cloud model in which customers interact with existing applications. Contact Serena for more information.

The following sections describe the main characteristics of our SaaS deployment.

SECURITY

The security features of the SaaS deployment includes the following:

- **Network Security** – The hosted deployment is firewalled with access permitted only through specified SSL-enabled ports on specified external virtual IP addresses. All communication between components inside the firewall happens over internal virtual IPs.
- **Single Sign-On (SSO)** – SSO is enabled on the hosted infrastructure. This allows users to centrally log in once to all of the SBM components. As a part of the log in process, the user's namespace (the customer account that he or she belongs to) and access levels are determined and all access is governed by those access rules.
- **Physical Security** – The hosted infrastructure is fully SaaS 70 level 2 certified. All physical access to machines is restricted by physical isolation. Network access to the machines is restricted through firewalls and authenticated access. Users can access SBM Web pages and the Web services API only through secured HTTPS ports.

For more details, refer to Serena's whitepaper on *Security in the Cloud* at <http://www.serena.com>.

SCALABILITY AND RELIABILITY

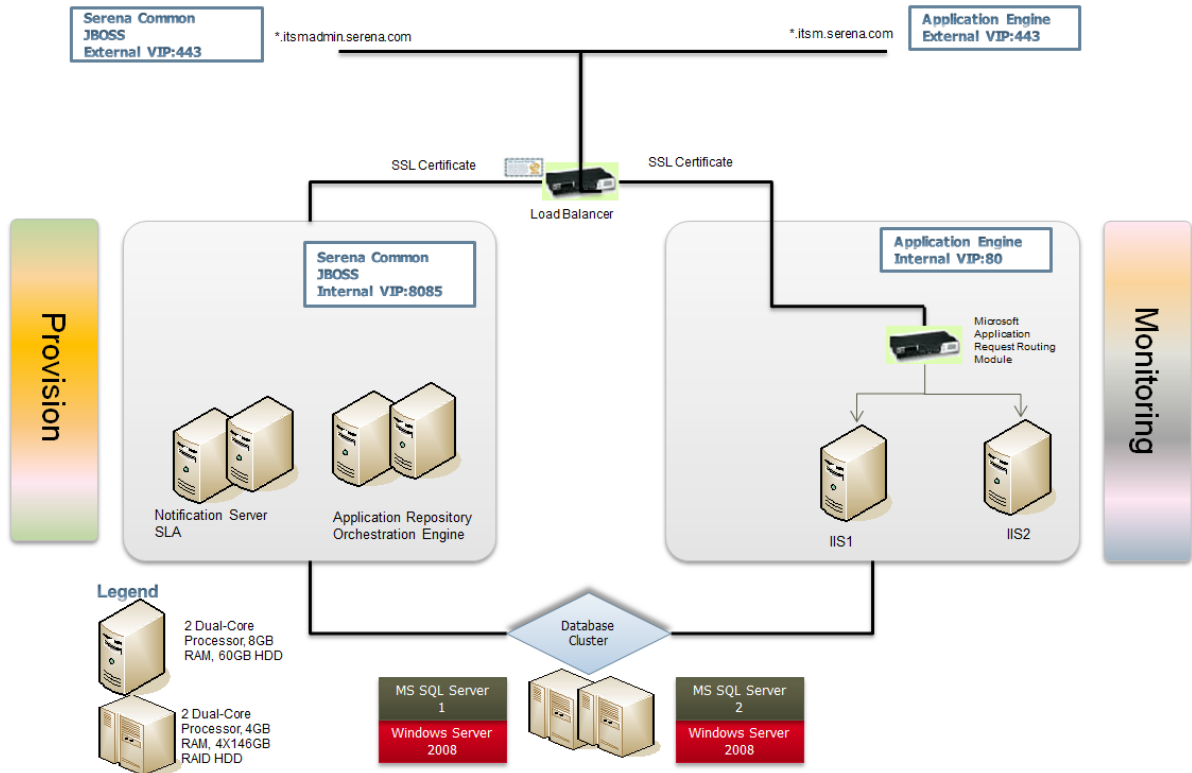
Multiple instances of SBM Application Engine and SBM Orchestration Engine have been deployed in clustered configurations. The configuration allows for each of the components to be independently scaled by adding additional server instances as needed. Each server instance is serviced by a separate network card. The distribution includes:

- JBoss Server instances (which host all Java-based components in SBM such as SBM Orchestration Engine, SBM Application Repository, and the Event Manager) are deployed as nodes in a load-balanced Java cluster that allows the load to be distributed and ensures reliability if any of the nodes fail.
- Multiple IIS Server instances (that serve the SBM Application Engine and the SBM User Workspace) are hosted in a load-balanced configuration to distribute load and ensure high

transaction speed. Clustered configurations for each of the components above are hosted on multiple physical machines to allow the system to remain functional if any of the physical machines fail.

- The database server is hosted on a separate SQL cluster configuration consisting of two physical machines, which ensures performance and reliability.
- The hosted infrastructure also includes additional components that enable automated creation of customer accounts and assignment of DNS addresses (C-Names). (These components are not shown in the deployment diagram below).

The diagram below describes the configuration of SBM components in Serena's SaaS deployment.



Summary

Serena Business Manager has been architected, implemented, and tested to meet the complexity and scalability challenges of the enterprise. Its modern architecture provides secure and highly scalable solutions. By leveraging state-of-the-art technology, Serena Business Manager provides a platform for innovation that will grow to meet the ever-expanding needs of the enterprise.

ABOUT

Serena Software, the Change Governance leader, helps more than 15,000 organizations around the world—including 96 of the Fortune 100 and 90 of the Global 100—turn change into a business advantage. Serena is headquartered in Redwood City, California, and has offices throughout the U.S., Europe, and Asia Pacific.

CONTACT

Website: <http://www.serena.com>

Copyright © 2012 Serena Software, Inc. All rights reserved. Serena, TeamTrack, ChangeMan, PVCS, StarTool, Collage, and Comparex are registered trademarks of Serena Software, Inc. Change Governance, Command Center, Dimensions, Mover and Composer are trademarks of Serena Software, Inc. All other product or company names are used for identification purposes only, and may be trademarks of their respective owners. Revised 8 May 2012.