Scaling for the Enterprise

by Pete Dohner and Jeremy Vorndam
June 2, 2017
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who Should Read This Paper?</td>
<td>1</td>
</tr>
<tr>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>N-Tier Architecture</td>
<td>2</td>
</tr>
<tr>
<td>SBM Clients</td>
<td>3</td>
</tr>
<tr>
<td>SBM Server Components</td>
<td>3</td>
</tr>
<tr>
<td>The Process App Development Lifecycle</td>
<td>5</td>
</tr>
<tr>
<td>Security</td>
<td>6</td>
</tr>
<tr>
<td>Scalability</td>
<td>7</td>
</tr>
<tr>
<td>Increasing Scalability</td>
<td>8</td>
</tr>
<tr>
<td>A General Approach for Scalability</td>
<td>11</td>
</tr>
<tr>
<td>Summary</td>
<td>13</td>
</tr>
</tbody>
</table>
Who Should Read This Paper?

This paper provides information on scaling Micro Focus® Solutions Business Manager (SBM) 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 or alliances professionals.

Introduction

IT organizations are challenged by a growing backlog of requests from lines of business. They face 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 requirement leaves many requests from the business unattended.

SBM is a powerful solution that empowers business users to collaborate with IT to build applications that solve business problems. It includes key capabilities for creating workflows, orchestrations, work item forms, and streamlined management of application staging and deployment.

SBM enables the creation of business applications that are:

- Visually composed and assembled
- A combination of content and services from multiple applications and services
- Process-centric and data-enabled
- Delivered through multiple channels such as email, 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
- The Process App Development Lifecycle
- Security
- Scalability

SBM is a powerful solution that empowers business users to collaborate with IT to build applications that solve business problems. It includes key capabilities for creating workflows, orchestrations, work item forms, and streamlined management of application staging and deployment.
N-Tier Architecture

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

Fig. 1

SBM Architecture
SBM Clients

- **MSBM Composer**
  SBM Composer is the SBM client component that designers use to create SBM Composer process apps. SBM Composer is a modern smart-client application that is 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 and redo. SBM Composer is the design environment from which designers can find, customize, and create process apps. 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.

- **SBM Work Center and SBM User Workspace**
  End users interact with SBM through the SBM Work Center and SBM User Workspace interfaces, which are the client components from which users can access deployed process apps. These interfaces are 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 interfaces use AJAX (Asynchronous JavaScript and XML) to deliver a rich user experience.

- **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.

- **SBM System Administrator**
  A Windows application that administrators use to configure various system settings and run database utilities, such as the Data Import Wizard.

- **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. SBM Application Repository provides repository, deployment, promotion, and logging services.

SBM Server Components

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 SBM Work Center and 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 end users. SBM Application Engine primarily coordinates human workflow activities in a process app. SBM Application
Engine powers the following:

- **SBM Work Center**
- **SBM User Workspace**
- **SBM Application Administrator**
- **SBM System Administrator**

**SBM Tomcat**
The SBM Tomcat server is a J2EE container for a number of services provided in SBM Server.

These services include:

- **SBM Application Repository**
  SBM Application Repository stores design and deployment artifacts that are created in SBM Composer. Once process apps are built and versioned in SBM Composer, designers and administrators can deploy them to the SBM Server, which makes them available to end users. SBM Application Repository also enables business users and administrators to promote process apps from a test/UAT environment into the production environment.

- **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 Smart Search Lucene engine in Work Center, agile services for Work Center backlogs, Serena Localization Service (SLS), and the SBM proxy server. The SBM proxy server is used by the REST Grid Widget and the PDF Widget, which generates PDF documents from data in the end user interfaces.

- **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 email 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 email to the Mail Client email box. The Mail Client communicates with mail servers using one of the following protocols: POP3, IMAP, or Exchange (via web services).

- **SBM Logging Services**
  Powers solution usage reports in SBM Work Center and performs active diagnostics, which capture SBM Application Engine Web Server events that occur without requiring you to stop the IIS services.

**SBM Database Server**
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 GSOAP web services, the C++ API, and the JSON/REST API.
SBM separates application design from configuration and use. This offers many benefits, including the ability to aggregate many changes and deploy them at once.

For details on setting up the SBM database, refer to the SBM Installation and Configuration Guide.

The Process App Development Lifecycle

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 in the process app development lifecycle:

1. **Design**
   All design tasks take place in SBM Composer. 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, designers 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**
   Designers and administrators can deploy a process app and its applications to the runtime SBM Server. Deployment activates the applications in Work Center and 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.
   **Tip:** The Quick Deploy feature in SBM Composer enables you to validate, publish, and deploy process apps in one step.

4. **Configure**
   Administrators use SBM Application Administrator to configure deployed applications. Configuration covers four general areas: user and group management, project configuration, field overrides, and notification management. Administrators can access SBM Application Administrator via links in Work Center and SBM User Workspace.

5. **Use**
   Once an application is fully configured, it is ready for users in Work Center and SBM User Workspace.

Best practices in software management advocate multiple environments for testing, staging, and production where changes are propagated from one environment to another in an automated manner. SBM provides full fidelity promotion of a process app and all related configuration data (called a snapshot) from one SBM environment to another via the path to production model.
These snapshots can also be exported and imported into the SBM Server in case there is a firewall in between environments. For more information, refer to the *Path to Production for Enterprises* white paper.

### 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.

- **Identity and Access Management**
  SBM provides single sign-on (SSO) authentication out of the box while interacting with components at runtime and design time. It also provides a complete audit trail of all interactions and changes that are performed by either humans or applications during a session.

- **User Authentication**
  Requests to the SBM Server are authenticated using one of the following schemes:
  - **Internal Passwords**
  - **Windows Domain Authentication**
    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 can validate users against an LDAP system. The system can be configured to use LDAP authentication alone or LDAP and then internal passwords if the user is not found in LDAP.

- **Data and Function Security**
  SBM provides granular control over privileges for all logical components within the SBM Server. Users may be granted varying roles that 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 configuration options to control authorization down to the field level.
Scalability

Applications need to scale well in order to be successfully deployed in enterprise environments. This is a complex task because 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 SBM Application Engine, whereas a process app with complex orchestrations will require more resources to be allocated to SBM Tomcat.

IT professionals must carefully consider the following factors when planning a scalable deployment:

- **Network Configuration and Topology**
  SBM is distributed across clients, servers, and a database server. Consequently, network hardware and software configuration and capabilities have a significant impact on the performance characteristics.
  The closer in proximity that these pieces are on the network, 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**
  SBM 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. 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**
  SBM Composer provides designers the ability to create a variety of process apps. Each customer uses SBM 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.

Micro Focus utilizes standard Microsoft and J2EE technologies and approaches for building and deploying a scalable enterprise solution. The Windows 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 servers are consistently among the best as measured by the Transaction Processing Performance Council (TPC; reference http://www.tpc.org).
Increasing Scalability

Once you have an understanding of your usage model and you have taken an inventory of what machines are available, 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 or servers. 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) enables the system to operate more efficiently. If possible, place the operating system on a drive of its own. Reducing contention for data access from the physical disk drive improves the scalability of the system.

- **Memory**
  You must ensure that each process that you scale vertically has adequate memory. As of SBM 11.3, Windows 2012 and Windows 2016 platforms are supported. For a comprehensive list of memory limits for each platform, see: [http://msdn.microsoft.com/en-us/library/aa366778%28VS.85%29.aspx](http://msdn.microsoft.com/en-us/library/aa366778%28VS.85%29.aspx)
 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 first.

You can scale SBM Application Engine even further by implementing a web farm that contains multiple IIS servers, 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 Tomcat servers can be configured to create a cluster. You can control the amount of activity that is directed to any one Tomcat server by specifying particular endpoints during the process app deployment process. Contact Support or search the Knowledgebase for more information on clustering the Tomcat application server components or launching a backup Java Virtual Machine (JVM) that provides failover for Tomcat.
You can also install the SBM Mail Services on separate machines to move the processing overhead of these services to another set of servers. You can also cluster the SBM Mail Services to increase scalability and reliability.
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.

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:

<table>
<thead>
<tr>
<th>Zone</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Single system with all SBM components installed</td>
</tr>
<tr>
<td>SBM server: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>Database server: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Single system with all SBM components installed</td>
</tr>
<tr>
<td>SBM server: Windows 2016 Server, Dual CPU 3.2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>Database server: Windows 2016 Server, Dual CPU 3.2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>SBM IIS / Tomcat components distributed on separate servers</td>
</tr>
<tr>
<td>SBM IIS server: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>SBM Tomcat servers: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>Database Server: Windows 2016 Server, Quad CPU 3.2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>SBM IIS Web Farm / Tomcat Cluster</td>
</tr>
<tr>
<td>SBM IIS Web Farm:</td>
<td></td>
</tr>
<tr>
<td>– SBM IIS server 1: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>– SBM IIS server 2: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>– And more as needed</td>
<td></td>
</tr>
<tr>
<td>SBM Tomcat Cluster:</td>
<td></td>
</tr>
<tr>
<td>– Node 1: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>– Node 2: Windows 2016 Server, Dual CPU 2GHz, 16GB RAM</td>
<td></td>
</tr>
<tr>
<td>– And more as needed</td>
<td></td>
</tr>
<tr>
<td>Database Server: Windows 2016 Server, Quad CPU 3.2GHz, 32GB RAM</td>
<td></td>
</tr>
</tbody>
</table>
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.

**Summary**

SBM has been built, 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, SBM provides a platform for innovation that will grow to meet the ever-expanding needs of the enterprise.