# SQL_R2_Logo.jpg

SQL Server Technical Article

**Technical Writer:** Paul Jenkins

**Technical Reviewers:** Mark Mortimore, Nathan Johnson

**Date/Time of Draft: 6/30/2011 @ 5:00 PM**

**Published:** July 2011

**Applies to:** SQL Server 2008 R2 and SQL Server 2008

**Summary:** The purpose of this guide is to provide a description of the technologies and best practices utilized to design a database consolidation solution; guidance will be appropriately defined throughout to prescribe configurations and considerations to implement for best results. Documentation of specific tasks will be very limited.

Copyright

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2011 Microsoft. All rights reserved.

Contents

[ 1](#_Toc297215654)

[Introduction 5](#_Toc297215655)

[Purpose of Consolidation 5](#_Toc297215656)

[Goals of Consolidation 5](#_Toc297215657)

[Private Cloud and Consolidation 6](#_Toc297215658)

[The Benefits of Consolidation 6](#_Toc297215659)

[Methodology 7](#_Toc297215660)

[Workload Analysis 7](#_Toc297215661)

[Consolidation Options 7](#_Toc297215662)

[How to Choose a Strategy 7](#_Toc297215663)

[Hardware Design 7](#_Toc297215664)

[Manageability 8](#_Toc297215665)

[Licensing 8](#_Toc297215666)

[Workload Analysis 9](#_Toc297215667)

[MAP Toolkit 9](#_Toc297215668)

[Categorization 9](#_Toc297215669)

[Consolidation Options 10](#_Toc297215670)

[Database Stacking 10](#_Toc297215671)

[SQL Server Instance Stacking 10](#_Toc297215672)

[Virtualization 10](#_Toc297215673)

[Other Options 11](#_Toc297215674)

[How to Choose a Strategy 12](#_Toc297215675)

[Security 12](#_Toc297215676)

[High Availability and Disaster Recovery 14](#_Toc297215677)

[Resource Management 15](#_Toc297215678)

[Density 17](#_Toc297215679)

[Manageability 18](#_Toc297215680)

[A Note on Performance 19](#_Toc297215681)

[Hardware Design 20](#_Toc297215682)

[Hardware Sizing 20](#_Toc297215683)

[Start from the Server 20](#_Toc297215684)

[Start from the Workload 20](#_Toc297215685)

[Potential Bottlenecks 21](#_Toc297215686)

[New Virtualization Technologies 21](#_Toc297215687)

[Manageability 23](#_Toc297215688)

[Monitoring and Alerting 23](#_Toc297215689)

[Management and Administration 23](#_Toc297215690)

[Backup and Recovery 23](#_Toc297215691)

[Servicing and Patching 23](#_Toc297215692)

[Licensing 24](#_Toc297215693)

[Conclusion 26](#_Toc297215694)

[List of Sources 27](#_Toc297215695)

[Appendix 28](#_Toc297215696)

[Appendix A – SQL Server 2008 Consolidation Benefits 28](#_Toc297215697)

[Maximize Savings from Consolidation 28](#_Toc297215698)

[Flexible Options for Consolidation 28](#_Toc297215699)

[Features That Drive Consolidation Success 28](#_Toc297215700)

[Ease of Management 30](#_Toc297215701)

[Included Comprehensive Business Intelligence Platform 30](#_Toc297215702)

# Introduction

The goal of this document is to provide technical guidance to IT personnel who are considering or planning a database consolidation project. Hence, the intended audience for this paper will be IT Managers, Project Managers, and engineers who are looking to build their own consolidation solution, or have it built for them through a 3rd party. In this document, we will discuss, in high-level language, best practices and guidelines that should be follow to improve the success of their projects.

In general terms, consolidation is the combining of various smaller units into a more efficient stable larger unit. When applied to information technology (IT), consolidation translates into improved cost efficiency from higher utilization of resources, standardization and improved manageability of the IT environment, and a focus on **green** IT environments through reduced energy consumption. One of the more important applications in the IT environment is the database. Databases tend to be very widespread across the enterprise because they are very efficient for handling relational storage of data and are designed to be platforms for a wide variety of applications. Because databases form the foundation of so many business systems, IT departments can easily lose control of the number of databases that need to be maintained, because each group may simply create their own database to solve a specific problem. This leads to uncontrolled database growth. Thus, databases are one of the prime candidates for consolidation. When merging database applications, consider the following three potential strategies:

* Using a single physical machine to host multiple virtual machines running the Microsoft SQL Server database software
* Using a single machine to host multiple SQL Server instances
* Using a single instance of SQL Server to host multiple databases

Each of these strategies has different advantages and disadvantages related to security and compliance requirements, high availability and disaster recovery requirements, resource management benefits, level of consolidation density, and manageability tradeoffs. In this document, we will try to answer the following questions:

* What are the concerns when creating a consolidation plan for my environment?
* What are the key differentiators among the three consolidation options?
* How can I use these differentiators to choose the best option for my environment?

# Purpose of Consolidation

This paper describes a method for database consolidation that merges all SQL Servers within the infrastructure in order to centralize, simplify, and improve their management and significantly reduce facilities and operations costs.

## Goals of Consolidation

In the corporate infrastructure, especially in large organizations, numerous instances of SQL Server are installed to physical hardware such as Dell, HP, IBM, or other servers. The problem with hardware is that because it eventually gets old and outdated, it must be replaced. In addition, physical hardware takes up lots of space, consumes electricity, requires HVAC services, needs to be supported by IT personnel, and needs to be tracked and stored in the company’s inventory database. To address these problems, as well as others, a solution was developed. The database consolidation solution is a good answer to all of these problems, as they exist today.

## Private Cloud and Consolidation

Microsoft offers a comprehensive solution for optimizing your databases for the private cloud. Optimizing SQL Server for a private cloud solution can help ensure that your computer, network and storage resources are used efficiently, reducing physical footprint, capital, and operational expenses. It provides you with the ability to scale your resources efficiently and deploy the resources on demand without compromising control. It can also help you drive standardization and compliance in your new and efficient private cloud. Key capabilities include:

* Resource pooling - consolidating your databases
* Elasticity - scaling your computer, network, and storage resources efficiently
* Self-service - deploying resources on demand quickly
* Control and customize - driving standardization and compliance

Within the private cloud, you can consolidate databases to reduce capital and operational expenses and drive green IT; scale your computer, network, and storage resources efficiently to meet demand; deploy your computer, network, and storage resources on demand, allowing for faster time to market; and drive standardization based on policies and customized for your business needs.

## The Benefits of Consolidation

There are numerous benefits that result from implementing the database consolidation solution, including, but not limited to, the following:

* Improved performance
* High availability
* Retirement of older hardware
* Reuse of hardware for other purposes
* Facilitation of disaster recovery and business continuity
* Reduction of facilities costs (floor space, HVAC, electricity, and so on)
* Reduction of data center operations costs (IT support, service contracts, and so on)

For an example of how SQL Server augments and supports consolidation, see Appendix A.

# Methodology

In this section, we discuss the methodology that we used to design our database consolidation solution. This approach describes the steps necessary to gather essential information, categorize it in a way that best fits your business needs, and then use this data to design the right solution for your database workloads.

## Workload Analysis

Using our methodology, we first gather information about database workloads and then we categorize them. In this section, we introduce the Microsoft Assessment Planning (MAP) Toolkit, which was designed to discover SQL Server workloads running in the enterprise. The MAP Toolkit also performs its version of categorizing the database workloads. You can categorize workloads in any way that makes sense for your organization. This might be according to database size, database performance requirements, the service level agreement (SLA) for each workload, the business owner of the workload, the availability requirements of the workload, common security requirements, or any other reason that makes good sense for your organization.

## Consolidation Options

Each consolidation option provides a certain degree of isolation, which may impact the number of applications/databases that can be consolidated to one machine (referred to as density). Typically having higher isolation allows for greater flexibility in leveraging features, but it may increase management cost and reduce the density limit. Achieving higher density results in lower isolation in order to optimize resources and reduce management cost. This can be cost-effective, but it may reduce the ability to leverage certain features and increase the potential for resource contention. In this section, we’ll review several methods for consolidation.

## How to Choose a Strategy

Choose your strategy based on your organization’s priorities and how different consolidation options support those priorities. The key considerations for choosing a consolidation strategy can be broadly divided into the following categories:

* Security
* High availability and disaster recovery
* Resource management
* Density
* Manageability

How important each of these factors is depends on the priorities for your organization’s consolidation effort.

## Hardware Design

It may be worthwhile to establish a standardized server type or configuration for the consolidation machine. Standardization can help streamline the ordering process for additional machines, and it presents a common set of maintenance and configuration requirements. In this section, we’ll take a look at some options for building the right hardware configuration.

## Manageability

Here we introduce the activities we think of as management actions, and then we look at some of the different products that can be used to accomplish those functions. We discuss applications that perform monitoring and alerting, management and administration, backup and recovery, and servicing and patching.

## Licensing

Licensing software used in a virtualized environment can be extremely complicated. In this section, we review some of the key considerations for legally licensing software components and look at ways to correctly license some of the core components in the virtualized environment, including the host operating system environment (physical) and the guest operating system environments (virtual). This section also looks at licensing the guest SQL Server installations and the System Center product suite.

# Workload Analysis

Using our methodology, we first gather information about database workloads and then we categorize them. In this section, we introduce the MAP Toolkit, which was designed to discover SQL Server workloads running in the enterprise. The MAP Toolkit also performs its version of categorizing the database workloads. You can categorize workloads in any way that makes sense for your organization. This might be according to database size, database performance requirements, the SLA for each workload, the business owner of the workload, the availability requirements of the workload, common security requirements, or any other business related reason that makes good sense for your organization.

## MAP Toolkit

MAP 6.0 is the capacity planning toolkit for the database consolidation solution. It starts by discovering all of the instances of SQL Server in your environment and identifying the low utilized instances of SQL Server as candidates for consolidation. Then, it uses MAP to gather performance data for the list of candidate-servers. The final steps are to consolidate placement of all the instances of SQL Server by sizing them to a predefined virtual machine, place the virtual machines into the appliance, and generate recommendations. The results are fed into the on-boarding process for actual migration of servers. Please note, however, that although MAP is tuned to find SQL Server, it can also find competitor databases in the environment too.

## Categorization

Categorization is what you do after discovering your company has, for example, 5,000 instances of SQL Server that need to be grouped in some logical or meaningful way. You might do this by size, business unit, security criteria, SLA, high availability and disaster recovery requirements, or any number of other ways.

# Consolidation Options

Each consolidation option provides a certain degree of isolation, which may impact the number of applications that can be consolidated to one machine (referred to as density). Typically having higher isolation allows for greater flexibility in leveraging features, but it may increase management cost and reduce the density limit. Achieving higher density results in lower isolation in order to optimize resources and reduce management cost. This can be cost-effective, but it can also reduce the ability to leverage certain features and increase the potential for resource contention.

## Database Stacking

In database-level consolidation, multiple applications share and store data in one SQL Server instance; each application uses its own database or set of databases. Because all of the databases are in the same SQL Server instance, this also means that all applications share the same SQL Server patch level (that is, the major and minor version of SQL Server) and all server-level objects such as **tempdb**. This option is attractive in terms of reducing management and licensing costs because fewer SQL Server instances need to be maintained. In SQL Server 2008 R2, databases can be registered as data-tier applications for managing within a SQL Server control point after the host SQL Server instance is enrolled as a managed instance for centralized management.

## SQL Server Instance Stacking

In instance-level consolidation, multiple applications are moved onto a single physical server with multiple SQL Server instances. This option provides isolation of the SQL Server instance binaries, allowing for each application to be at different patch levels. However, potential exists for application conflicts because system resources (mainly CPU, memory, and I/O) are shared, although configuration settings such as the **CPU affinity mask** and **max server memory** can help provide resource isolation. Database system administration is isolated, but Windows system administration is shared for the host server. Each SQL Server instance on the machine can be enrolled within a SQL Server control point for management.

## Virtualization

In this approach, applications are migrated from their physical server into a virtual machine. A single machine hosts multiple virtual machines, and each virtual machine hosts a single SQL Server instance. Because the virtual machine can act as a dedicated physical machine, this approach provides an easier migration of the source environment to the consolidation environment. Virtual machines are fully isolated from other virtual machines and communicate with other servers on the network as if they were physical machines. Optimal resource governance between multiple virtual machines is automatically managed by the hypervisor. Additional high availability options are also available with features like Microsoft Hyper-V Live Migration. While this approach results in fewer physical servers to manage, it does maintain as many operating system and SQL Server images as the source environment. The SQL Server instances within the virtual machine can be enrolled in a SQL Server control point for management.

## Other Options

Other possibilities include further optimizations on an existing approach such as schema-level consolidation or hybrid approaches, such as mixing the instance consolidation and database consolidation approaches or having multiple SQL Server instances in a virtual machine. These approaches may require management at several levels, but they may provide more flexibility with a blend of the advantages and disadvantages of each approach. The key decision factors are similar to the higher-level consolidation options mentioned previously, so this document will focus only on those.

The following table describes how each of the consolidation options compares to the others. This perspective makes it easier to determine which consolidation methodology is best for your particular needs.

|  |  |  |  |
| --- | --- | --- | --- |
| **Consolidation Comparison Matrix** | | | |
| 1=poor, 2=low, 3=average, 4=good, 5=best | | | |
| **Feature** | **DB Stacking** | **Instance Stacking** | **Virtualization** |
| Scalability | 1 | 2 | 5 |
| Security | 1 | 3 | 5 |
| Density | 4 | 3 | 2 |
| High availability | 3 | 3 | 5 |
| Disaster recovery | 3 | 3 | 5 |
| Resource management | 2 | 3 | 5 |
| Manageability | 3 | 4 | 5 |

Table 1: Consolidation Feature Comparison

# How to Choose a Strategy

Choose your strategy based on your organization’s priorities and how different consolidation options support those priorities. The key considerations for choosing a consolidation strategy can be broadly divided into the following categories:

* Security
* High availability and disaster recovery
* Resource management
* Density
* Manageability

How important each of these factors is depends on the priorities established for your organization’s consolidation effort.

## Security

Security is a critical factor in creating a consolidation plan. Regulatory requirements around disclosure of information, compliance, separation of responsibilities, and privacy are important to recognize and establish up front, because these policies are usually set externally and monitored by auditors. Improper migration of applications can cause problems that are expensive and difficult to rectify. As a result, it is very important to identify what information is stored and who needs to be notified if that information is accessed improperly or lost so that you can identify the appropriate consolidation strategy. If an application has very strict security requirements, it is an ideal candidate for a virtualized approach to consolidation because the virtual machine has almost the same security isolation options as if the application had a dedicated physical host. Table 2 shows how security requirements are handled in different consolidation options.

| **Requirement** | **Virtualization** | **Instance** | **Database** |
| --- | --- | --- | --- |
| Equivalent to having a dedicated physical machine | Yes | No | No |
| Isolation of local Windows accounts | Yes | No | No |
| Isolation of SQL Server logins | Yes | Yes | No |
| Isolation of SQL Server binaries | Yes | Yes | No |
| Data protection through Windows BitLocker drive encryption | Yes | Partial – no isolation between applications | Partial – no isolation between applications |
| Data protection through Windows Encrypting File System (EFS) | Yes | Yes – if instances have separate service accounts | Partial – no isolation between applications |
| Data protection through Microsoft SQL Server Transparent Data Encryption (TDE) | Yes | Yes | Partial – all root certificates are stored in the ***master*** database |
| Data protection through Windows permissions | Yes | Yes | Partial – SQL Server service account and files shared for host instance |
| Data protection through SQL Server granular encryption | Yes | Yes | Yes |
| Data protection through SQL Server granular permissions | Yes | Yes | Yes |
| Auditing of actions with SQL Server Audit | Yes | Yes | Yes |

Table 2: Comparison of security considerations across consolidation options

In general, it is better to keep applications with different security requirements separate. For example, an application with customer data that requires restricted access should not be consolidated onto a machine with an application that is regularly accessed by users who do not normally have access to customer data. Such consolidation increases the risk that an improperly configured login or permission would grant access to the private data. This is even more important with the database-level approach; because the Windows accounts and SQL Server logins are shared and the binaries themselves are the same, a security exploit in one database can potentially affect another.

Instance-level consolidation can provide an additional layer of protection, because the binaries and the SQL Server logins are separate, but the instances still share the same Windows accounts and operating system configuration. At the instance level, we recommend that you use different service accounts for each instance to reduce security risks from one process affecting another.

Note: SQL Server 2008 and SQL Server 2008 R2 take advantage of the service security identifier (SID) support provided by Windows Server 2008 and Windows Server 2008 R2 to help mitigate these risks.

Before you decide on an approach, it is important to first identify where the specific vulnerabilities exist for the application. Certain types of vulnerabilities can help you rule out some consolidation approaches. For example, hard-coded dependencies on the SQL Server system administrator account, other server roles, credentials, or any other server objects (for example, **tempdb** or **msdb**) need to be explicitly identified, because these applications have an increased risk of inadvertent information disclosure. These applications need to be either modified or consolidated with an approach other than the database approach. Even if there isn’t sensitive information, these dependencies increase the risk of cross-application corruption or overwriting of data.

If sensitive information is being stored, it is important to identify what is being used to protect this data. It might be operating-system-based mechanisms such as Windows BitLocker and Windows EFS, or it might be database mechanisms such as SQL Server TDE. If the application relies on operating-system-based mechanisms, database-level or even instance-level consolidation options may not be viable, because these share the same operating system environment.

## High Availability and Disaster Recovery

As part of creating the consolidation plan, consider each application’s high availability and disaster recovery requirements. The virtualization approach leveraging Hyper-V has an advantage in minimizing planned downtime through Live Migration because the application can remain active during planned failover. Other high-availability solutions may require applications to restart or clients to reconnect after failover. For more information about Live Migration, see the [Hyper-V Live Migration Overview and Architecture](http://www.microsoft.com/downloads/details.aspx?FamilyID=fdd083c6-3fc7-470b-8569-7e6a19fb0fdf&displaylang=en) white paper (http://www.microsoft.com/downloads/details.aspx?FamilyID=fdd083c6-3fc7-470b-8569-7e6a19fb0fdf&displaylang=en). Live Migration also requires that the hosts share processors from the same manufacturer. See the Live Migration white paper for details.

All three approaches can leverage the various high-availability features built into SQL Server such as failover clustering, database mirroring, and replication. The unit of failover is different depending on the feature used. Table 3 compares these features across consolidation options.

| **Feature** | **Virtualization** | **Instance** | **Database** |
| --- | --- | --- | --- |
| Application remains available during planned host machine downtime without application restart | Yes – via Live Migration (database mirroring can also be used) | Yes – via database mirroring | Yes – via database mirroring |
| Application remains available during planned host machine downtime without client reconnect | Yes – via Live Migration | No | No |
| Application can be migrated between machines without downtime (restart or reconnect) | Yes – via Live Migration | No | No |
| SQL Server failover clustering | Yes | Yes | Partial – failover is at the instance level |
| SQL Server log shipping | Yes | Yes | Yes |
| SQL Server database mirroring | Yes | Yes | Yes |
| SQL Server replication | Yes | Yes | Yes |

Table 3: Comparison of high-availability features across consolidation options

You can place applications that require similar levels of availability on the same machine. Such grouping can take advantage of your best hardware, and it can help focus management resources on maintaining those applications. However, remember that the high availability technology determines the level at which failover takes place, and consider your choice of consolidation strategy accordingly. Virtualization or dedicated hardware may be the best choices in this scenario. For example, if SQL Server failover clustering is the high-availability solution, database-level consolidation may not be the best choice, because failover occurs at the instance level. If you have applications that are consolidated at the database level, these applications should rely on health monitoring based on the entire instance failing over.

On the other hand, instance-level or even virtualization consolidation strategies enable the application to take advantage of high availability features that are delivered at the database level, such as database mirroring. Virtualization, for example, enables the application to take full advantage of Live Migration, SQL Server failover clustering, database mirroring, and other high-availability features simultaneously to control specific degrees and stages of failover and availability. Finally, because the applications all share one machine, a failure in one application could cause machine issues, resulting in downtime for all of the other applications as well. Thus it is potentially better to rely on virtualization or dedicated hardware to achieve a high degree of isolation and avoid issues where failures from other applications affect availability.

## Resource Management

The consolidated server should be able to handle one or more applications at peak usage and applications that suddenly require more resources than normal, and it should be able to prevent situations where the resource contention caused by resource usage spikes could impact the reliability and consistency of other applications residing on the server. The consolidated server should also be able to handle any sustained usage growth from the application as well. Several features are available to handle resource management.

Virtualization provides perhaps the most specific boundaries, because CPU and memory allocation must be specified for the entire virtual machine container. Both of these settings can be modified later, but you may need to take the virtual machine offline to perform the changes. The benefit of this approach is that the resources are contained and isolated within the virtual machine (with the exception of the CPU, which can be over-committed), which thus reduces the impact of one application that is experiencing nonaverage workloads affecting other applications on the server. On the downside, these resources are allocated to the virtual machine regardless of whether or not they are fully utilized. In addition, the guest operating system of the virtual machine itself consumes some overhead of the allocated resources, and the host operating system also requires an additional allocation of resources, although these are generally relatively small. We recommend that you map one virtual processor to one physical processor. It can be fairly safe to over-commit processors (that is, to have multiple virtual processors map to one physical processor) but performance should be monitored. You should not over-commit memory, because doing so can create bottlenecks that impact performance. In fact, you *cannot* over-commit memory in Hyper-V.

Instance-level and database-level consolidation options provide direct access to the consolidated server’s physical hardware, which may help scalability by providing support for hot-add CPU and memory (that is, adding a logical processor or adding system memory on a live, running server), for example. However, direct access also creates an opportunity for resource contention. To address this, SQL Server provides the **max server memory** and **CPU affinity mask** settings to set limits on how much memory and how many logical processors the SQL Server instance can use. For more information about setting **max server memory**, see [Server Memory Options](http://msdn.microsoft.com/en-us/library/ms178067.aspx) (http://msdn.microsoft.com/en-us/library/ms178067.aspx) in SQL Server Books Online. For more information about setting **CPU affinity mask**, see [affinity mask Option](http://msdn.microsoft.com/en-us/library/ms187104.aspx) (http://msdn.microsoft.com/en-us/library/ms187104.aspx) in SQL Server Books Online.

**Note**: **CPU affinity mask** is ignored in a virtual machine.

The other consideration for resource contention is the interaction between the application and **tempdb**. If multiple databases are consolidated and they have dependencies on **tempdb**, I/O bottlenecks on **tempdb** can cause performance issues. If this is the case, consider using either instance or virtualization consolidation or modifying the applications. For more information about **tempdb**, see [tempdb Database](http://msdn.microsoft.com/en-us/library/ms190768.aspx) (http://msdn.microsoft.com/en-us/library/ms190768.aspx) in SQL Server Books Online.

If no contention exists between applications for server-level objects, database-level consolidation can be easier to manage from a resource standpoint because you only need to configure a single SQL Server instance per machine. This instance can thus use the full resources of the machine without concern for sharing CPU or memory for other SQL Server instances. The applications connecting to the instance may still contend for resources, but the Resource Governor feature introduced in SQL Server 2008 can be used to manage workloads (that is, groups of queries) through limits on CPU and memory resource consumption and prioritization between workloads. For more information about Resource Governor, see [Managing SQL Server Workloads with Resource Governor](http://msdn.microsoft.com/en-us/library/bb933866.aspx) (http://msdn.microsoft.com/en-us/library/bb933866.aspx) in SQL Server Books Online. Resource Governor can also be used in the other two consolidation options to further tune performance within the SQL Server instance, but it cannot be used across SQL Server instances or virtual machines.

| **Consideration** | **Virtualization** | **Instance** | **Database** |
| --- | --- | --- | --- |
| Isolation of **tempdb** | Yes | Yes | No |
| Isolation of server-level objects (credentials, linked servers, **msdb**, SQL Server Agent jobs, and so on) | Yes | Yes | No |
| Hard limits on CPU and memory usage set per application | Yes | Yes | No |
| Use of Resource Governor to provide query prioritization within a SQL Server instance | Yes | Yes | Yes |
| Hot-add CPU | No | Yes | Yes |
| Hot-add memory | No | Yes | Yes |
| Hot-add storage | Yes | Yes | Yes |

Table 4: Comparison of resource isolation considerations across consolidation options

## Density

In the context of consolidation, *density* is the number of applications that can be consolidated to a single machine. Virtual machines, SQL Server instances, and SQL Server databases all have different degrees of overhead, which affects consolidation density. Virtual machines have higher overhead, because a full operating system is maintained for each application. At the instance level, operating system resources are shared, but each application has an independent instance running, which is its own independent service. Database-level consolidation typically provides lower overhead, because all other resources are shared with the other databases on the single instance.

Note: SQL Server is currently limited to a maximum of 50 instances per operating system environment (physical or virtual). Hyper-V has a limit of 64 virtual machines per node and SQL Server has a limit of 32,767 databases per instance. For more information about limits related to virtual machines, see [Requirements and Limits for Virtual Machines and Hyper-V in Windows Server 2008 R2](http://technet.microsoft.com/en-us/library/ee405267(WS.10).aspx) (http://technet.microsoft.com/en-us/library/ee405267(WS.10).aspx).

Two data points to capture when measuring density are *throughput* and *response time*. These points determine the density limit, which this document defines as the moment at which adding an additional application causes the average response time or the average throughput for one or more other applications to become significantly lower than it was with the original hardware.

Ensure that the target consolidation server includes extra capacity. You should not assume that the server will run at 100 percent average CPU capacity after consolidation; it still needs to handle peak workloads, increases in users, and increases in operational workloads. The target server should have room for multiple applications to reach their peak workload simultaneously as well as handle any growth that may occur from application usage over time. Keeping 50 percent CPU utilization is generally sufficient, and it provides room for both peak workloads and growth in application usage over time.

## Manageability

Virtual machines provide significant flexibility in management, because they have all the options of a dedicated physical machine combined with simplicity of management offered by Microsoft System Center Virtual Machine Manager (VMM) and SQL Server control point. VMM also has a physical-to-virtual utility referred to as P2V. This utility takes a physical machine, converts it into a virtual machine, and then deploys it onto a Hyper-V server. This provides for very low cost of migration, because the entire process is handled automatically. For more information about how to use P2V, see [P2V: Converting Physical Machines to Virtual Machines in VMM](http://technet.microsoft.com/en-us/library/cc764232.aspx) (http://technet.microsoft.com/en-us/library/cc764232.aspx) in the Microsoft System Center documentation.

For an overview of virtualization with Hyper-V and details about the specific benefits of virtualization manageability, see [Virtualization with Hyper-V](http://www.microsoft.com/windowsserver2008/en/us/hyperv-overview.aspx) (http://www.microsoft.com/windowsserver2008/en/us/hyperv-overview.aspx) on the Windows Server 2008 R2 web site. Some of the specific features that can be leveraged with virtualization are the ability to clone and deploy an application very easily and the use of Live Migration to rapidly deploy applications between machines for dynamic load balancing with zero downtime. SQL Server 2008 R2 offers new technologies to assist you with consolidation. As mentioned previously, all three approaches can take advantage of the SQL Server control point in SQL Server 2008 R2 to provide centralized resource utilization views and policies over managed instances.

Another new feature is the ability to convert an existing application into a data-tier application definition; this provides a convenient way to migrate an application’s schema and logins because the data-tier application definition is designed to be more portable than a full SQL Server instance, and it encapsulates server-scoped objects such as logins. Note however that this process does not migrate data for the application between servers. You must perform data migration separately using backup and restore or another method. For more information about how to convert an existing database to a data-tier application definition, see [How to: Extract a DAC](http://msdn.microsoft.com/en-us/library/ee210526(SQL.105).aspx) (http://msdn.microsoft.com/en-us/library/ee210526(SQL.105).aspx) in SQL Server Books Online. The data-tier application can then be registered in a SQL Server control point for centralized management. Table 5 provides a comparison of some manageability highlights across the consolidation options.

| **Feature** | **Virtualization** | **Instance** | **Database** |
| --- | --- | --- | --- |
| Create predefined images | Yes | No | No |
| “One-click” clone environments between development, test, and production | Yes – with System Center VMM | No | Partial – can clone data-tier applications |
| Low-cost migration | Yes – P2V utility | No | Partial – depends on how well contained the application is within a database |
| Dynamic redeployment of application without downtime | Yes – with Live Migration | No | No |
| Can be managed by a SQL Server control point | Yes | Yes | Yes – if registered as a data-tier application |
| Requires installing SQL Server multiple times | No – can use P2V or cloning | Yes | No |
| Reduces number of physical servers to maintain | Yes | Yes | Yes |
| Reduces number of Windows installations to maintain | No | Yes | Yes |
| Reduces number of SQL Server instances to maintain | No | No | Yes |

Table 5: Comparison of manageability features across consolidation options

## A Note on Performance

In theory, performance can be a concern when you are choosing a consolidation strategy, given the different operating overheads of the various approaches. In practice, however, performance issues are fairly easy to mitigate as long as the proper analysis is done and the appropriate hardware is selected. Considerations for performance are also included as part of other metrics such as density and resource management, because all applications should at least perform no worse than they did before consolidation. If an application is performing poorly after migration to the target consolidation server, you can use any number of performance tools to analyze and tweak the application. In the worst case, the application can be migrated to a less overloaded server, or additional resources can be added for whatever the application is short of (I/O, CPU, memory, and so on). If the application has a strict SLA and requires a specific performance threshold, it may not be the ideal candidate for consolidation, although newer hardware may help maintain performance parity even with consolidation.

# Hardware Design

It may be worthwhile to establish a standardized server type or configuration for the consolidation machine. Standardization can help streamline the ordering process for additional machines, and it presents a common set of maintenance and configuration requirements.

## Hardware Sizing

When consolidating, you should generally consider having as much memory on the target consolidation server as the application was using on the original server. Note that the actual minimum memory required for the application may be less than the total available server memory if the application was unbounded; you may need to perform some analysis to find the minimum amount needed to run the application without affecting performance. Thus if four applications that previously used up to 2 GB of RAM each are consolidated onto a single server, the new server should have at least 8 GB of RAM available. A similar principle applies to processors.

Newer processors may reduce the need for the application to use as many processors as it previously had and many applications under-utilize CPU processing power anyway. Take this under-utilization into account when you plan hardware needs for your consolidation project. To begin, look at all applications that significantly under-utilize CPU, pick the one that utilizes the most processors, and take that number of processors as a base. Add to this the processors being used by the applications with higher-utilized CPUs. The consolidation server should have at least as many processors available. You can perform further analysis of CPU utilization to refine your estimates. As mentioned previously, you should always leave room for peak performance or application usage growth. Targeting approximately 50 percent utilization is a good starting point. A SQL Server control point can be helpful in collecting and viewing this data. Your hardware vendor may also have sizing tools to help you choose the appropriate consolidation server. There are two ways to approach the hardware design, which we look at in detail in the following sections.

## Start from the Server

When you start from the server, you choose a server that appeals for price and performance reasons, and you base the hardware design around that server. Choosing storage, in this case, requires that you take the time to run typical workloads on the server, determine the I/O requirements for the type of CPU, and determine the amount of RAM against the number, type, and different configurations of disk drives.

## Start from the Workload

Starting from the workload, which is our recommended approach, means that after capturing the performance data of your workloads by using a tool like the MAP Toolkit, you determine how much space and I/O your workloads need. From there you can design a storage subsystem with the required number of drives to deliver the anticipated amount of I/O. Then, you choose a server that is capable of delivering the required amount of processing power for the workloads.

## Potential Bottlenecks

Consolidation can introduce bottlenecks for application performance. An unconsolidated application is likely to have a dedicated physical machine with its own CPU, RAM, storage, and network devices and few if any other applications running on top of it; a consolidated application resides on a machine that shares all of these resources with other applications. As a result, it is important to size the consolidation candidates beforehand and choose a consolidation machine that has multiple CPU cores, a large amount of memory, and sufficient storage and network adapters to handle the load.

The SQL Server control point can be helpful here for viewing historical data on CPU and storage utilization. An in-depth discussion of how to configure the storage and network layers for the server is beyond the scope of this document, but it assumes that the application is not constrained by I/O limitations and that sufficient network bandwidth is available. For more information, see your hardware provider.

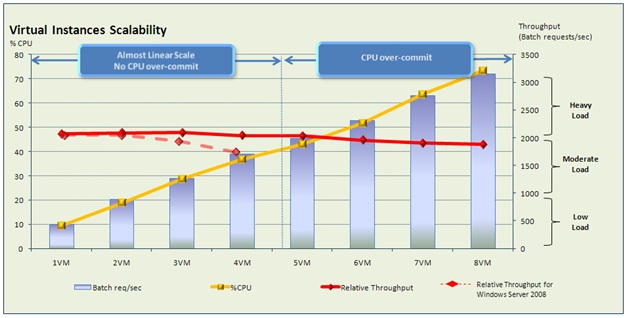
The machines should be selected with room to grow. For example, they should include capacity to add additional CPU cores, memory, and storage, and they should have PCI slots available for additional network and storage controllers. For virtualization, we generally recommend using a fixed-size virtual hard disk (VHD) or a pass-through disk, because dynamic VHDs can cause additional I/O overhead.

## New Virtualization Technologies

This document assumes that the version of Hyper-V included with Windows Server 2008 R2 is used as the virtualization technology. The features discussed here apply specifically to that version. The same principles may apply to other virtualization solutions, but these solutions are not discussed in detail. For more information about Hyper-V, see [Virtualization with Hyper-V](http://www.microsoft.com/windowsserver2008/en/us/hyperv-main.aspx) (http://www.microsoft.com/windowsserver2008/en/us/hyperv-main.aspx).

It is important to note that Hyper-V does have some limitations. Hyper-V on the host operating system is only supported on x64 processor architectures and requires hardware-assisted virtualization support (Intel VT or AMD-V) and hardware data execution prevention (DEP, also called Intel XD bit and AMD NX bit). Also, the guest operating system is currently limited to accessing four virtual CPUs. This is unlikely to be a problem for most consolidation candidates, but it is worth considering if the application is expected to substantially grow in usage.

Virtualization technology is, however, constantly improving. One scalability and performance feature of newer processors is second-level address translation (SLAT), also known as *nested paging*. AMD refers to this technology as NPT and Intel calls this EPT in their respective processors. You do not need SLAT if you choose virtualization as a consolidation path, but the application will certainly perform better. Figure 1 shows how application workload can benefit from the use of SLAT to achieve linear scale and improved performance.



****Figure** 1**: Performance advantage using SLAT processors

These numbers were derived using a 16-core machine, with each virtual machine configured to have 4 virtual processors and 7 GB of RAM with fixed-size VHD for storage. The host operating system was Windows Server 2008 R2.

For comparison, the same workload was run using the Windows Server 2008 implementation of Hyper-V (shown by the dotted red line), which does not take advantage of SLAT. The graph shows that throughput began to suffer after three virtual machines were added.

# Manageability

Here we introduce the activities we think of as management actions, and then we look at some of the different products that can be used to accomplish those functions. In this section, we discuss applications that perform monitoring and alerting, management and administration, backup and recovery, and servicing and patching.

## Monitoring and Alerting

In a large data center with thousands of servers and few operators, an event-driven approach is a good way to identify issues that need operator interaction. Rather than have someone scanning data and looking for something to happen, modern data centers use comprehensive software packages that enable monitoring to happen automatically. The operator is notified when there is an alert that needs to be handled. System Center Operations Manager is an example of a product that can be used to monitor hundreds or thousands of servers, generating custom events to attract an operator’s attention only if necessary.

## Management and Administration

To manage and administer a consolidated environment, you need the ability to audit the state of each piece of the environment and the operation of the environment. System Center Operations Manager provides comprehensive reporting that you can use to track and monitor trends across the consolidated environment. System Center Operations Manager is able to discover, inventory, and report on the current state of software across thousands of computers.

## Backup and Recovery

Backup is an essential part of a disaster recovery strategy; it is also a *best practice* to back up all systems regularly to overcome data loss. System Center Data Protection Manager (DPM) is a product that empowers your operations team to execute complex backup strategies in support of the consolidated environment.

## Servicing and Patching

Being able to determine the patch level of all parts of the consolidation environment, as well as being able to deliver the latest patches required for each product in the environment, is an essential part of servicing the environment. System Center Configuration Manager provides the capability to maintain an inventory of all installed software and to deliver the latest patches when they become available.

# Licensing

Licensing software used in a virtualized environment can be extremely complicated. In this section, we review some of the key considerations for legally licensing software components and look at ways to correctly license some of the core components in the virtualized environment, including the host operating system environment (physical) and the guest operating system environment (virtual). This section also looks at licensing guest (virtual) SQL Server installations and System Center product suite.

A database consolidation solution with private cloud benefits, as described in this document, requires the following key software components.

| **Component** | **Version** | **Licensing needed** | **Notes** |
| --- | --- | --- | --- |
| Windows Server | Windows Server 2008 R2  Datacenter | One processor license for every processor part of the solution. | This edition enables unlimited virtual machines and full mobility.  Available in OEM version with the hardware, or in Volume License agreements.  Includes Hyper-V.  All processors in the solution have to be licensed. |
| System Center | Configuration Manager 2007  Operations Manager 2007 R2  Virtual Machine Manager 2008 with SP1 (VMM)  Management Suite DC (SMSD) | Configuration Manager, Operations Manager, and VMM – One license per server that will run this software. Typically each solution requires only one of each of these licenses.  SMSD – One processor license for every processor part of the solution. | All System Center software requires SA (Software Assurance).  SMSD - All processors that are part of the solution have to be licensed. |
| SQL Server | SQL Server 2008 R2 Enterprise with Software Assurance (SA)  - or -  SQL Server 2008 R2 Datacenter | One SQL Server 2008 R2 DC processor license for every processor – *allows unlimited virtual images in the solution*  -or-  One SQL Server Enterprise processor license with Software Assurance for every processor – *allows unlimited virtual images in the solution* | To acquire the SQL Server Enterprise licenses, you can do one of the following:   * Use existing unused Enterprise licenses. * Buy new Enterprise or Datacenter licenses. * Convert SQL Server Standard licenses (that have SA) to SQL Server Enterprise licenses. * Upgrade a SQL Server Standard license with SA to a SQL Server Enterprise license. * Sign a Microsoft Enterprise Agreement that allows unlimited use of SQL Server Enterprise.   SQL Server Standard is *not* a viable solution for database consolidation with private cloud capabilities, because it does not provide mobility (only once every 90 days). This license cannot be moved, losing the benefits of the elasticity, agility, and so forth mentioned earlier. This limitation applies to *any* virtualization technology used with SQL Server Standard, including Hyper-V, and VMware. |

Table : Component Licensing Requirements

Other freely available components are also required. The main one is the MAP Toolkit. This toolkit, as mentioned in the “Methodology” section earlier, is an ideal way to start the process of evaluating what database instances exist and which should be migrated to the new consolidation solution.

# Conclusion

There are three ways a customer can deploy a database consolidation solution and get the additional benefits of having SQL Server in a private cloud:

1. build the solution in-house or through a third party
2. use a reference architecture or guidance guide
3. buy a pre-built appliance

Each of these options have different benefits and challenges, including faster/slower time to solution, higher/lower risk, faster/slower performance, and more/less component flexibility. This paper will help customers following path “a” build it themselves. Microsoft also offers solutions for options “b” and “c”. For option “b” customers can acquire the HP Database Consolidation Solution, which is a reference architecture tailored for database consolidation. Or, using option “c”, you can acquire the upcoming pre-built appliance from Microsoft and HP that provides all these technologies and benefits in a pre-integrated package, with additional custom software.

To a certain degree, consolidation is a never-ending process. New machines and new hardware are always arriving, providing higher consolidation density, longer application availability times, and better performance. Application usage will continue to grow, and new applications will be created to replace or augment the old ones. New factors such as **green** IT policies will also drive consolidation. Therefore, it is important to design consolidation plans not just for current trends, but for future ones as well. If you identify the specific goals that drive consolidation efforts for your organization and make decisions based on the key factors essential to those goals, keeping in mind the various advantages each consolidation option provides, you can use consolidation not just to achieve short-term objectives, such as reducing costs and creating more space in the data center, but also to create a dynamic and scalable IT infrastructure that can make future consolidation even easier and support growth for the company.

**For more information:**

<http://www.microsoft.com/sqlserver/>: SQL Server Web site

<http://technet.microsoft.com/en-us/sqlserver/>: SQL Server TechCenter

<http://msdn.microsoft.com/en-us/sqlserver/>: SQL Server DevCenter

Did this paper help you? Please give us your feedback. Tell us on a scale of 1 (poor) to 5 (excellent), how would you rate this paper and why have you given it this rating? For example:

* Are you rating it high due to having good examples, excellent screen shots, clear writing, or another reason?
* Are you rating it low due to poor examples, fuzzy screen shots, or unclear writing?

This feedback will help us improve the quality of white papers we release.

[Send feedback](mailto:sqlfback@microsoft.com?subject=White%20Paper%20Feedback:%20Consolidating%20Databases%20Using%20Virtualization%20Planning%20Guide).

# List of Sources

The following web sites and articles provide the core material for this document.

<http://www.microsoft.com/sqlserver/2008/en/us/server-consolidation.aspx>

<http://download.microsoft.com/download/C/8/4/C8470F54-D6D2-423D-8E5B-95CA4A90149A/SQLServer2008_Consolidation_Datasheet.pdf>

<http://msdn.microsoft.com/en-us/library/ee819082(SQL.100).aspx>

<http://download.microsoft.com/download/D/B/D/DBDE7972-1EB9-470A-BA18-58849DB3EB3B/SQLServer2008Consolidation.docx>

<http://download.microsoft.com/download/6/9/D/69D1FEA7-5B42-437A-B3BA-A4AD13E34EF6/SQLServer2008Consolidation.docx>

<http://msdn.microsoft.com/en-us/architecture/dd393309.aspx>

http://www.microsoft.com/sqlserver/en/us/solutions-technologies/cloud-computing/private-cloud.aspx

# Appendix

This section is reserved for special topics or issues that need additional information.

## Appendix A – SQL Server 2008 Consolidation Benefits

Here are some benefits of consolidating with SQL Server 2008:

* Savings from consolidation
* Licensing benefits with SQL Server 2008 Enterprise
* Flexible options for consolidation including and virtualization
* More predictable performance with Resource Governor
* Built-in data and backup compression
* Built-in encryption, auditing, and extensible key management for increased security features
* Reduce downtime with hot-add CPUs and memory, 16-node clusters, and database mirroring
* Centralized policy enforcement with Policy-Based Management
* More readily-available pool of skilled professionals
* Included comprehensive Business Intelligence platform

### Maximize Savings from Consolidation

Choose an enterprise-class data platform to help provide low total cost of ownership (TCO). Use SQL Server Enterprise to maximize the savings you can achieve; take advantage of the special licensing conditions of SQL Server Enterprise; use compression to reduce storage costs; make use of Resource Governor to achieve predictable performance; help meet higher aggregated availability needs; and take advantage of comprehensive built-in business intelligence (BI). Reduce the number of servers in your enterprise and help reduce your organization’s carbon footprint.

### Flexible Options for Consolidation

Businesses of all sizes can benefit from consolidation. Choose the appropriate strategy for your organization.

#### Consolidate servers with multiple databases and instances

Host multiple databases in a single SQL Server 2008 instance to centralize data storage and management. Enable the isolation of database applications while working to reduce hardware costs, energy use, licensing costs, and administrative overhead by using SQL Server 2008 Enterprise to run up to 50 separate instances on a single physical server.

#### Consolidate data services through virtualization

Use a single physical server to run multiple SQL Server 2008 instances in separate operating systems by using Windows Server 2008 with the Hyper-V role. Take advantage of SQL Server 2008 Enterprise licensing to run an unlimited number of virtual database instances. Use SQL Server 2008 Enterprise license mobility to reassign existing SQL Server 2008 licenses across physical servers in a server farm.

### Features That Drive Consolidation Success

Use the features of SQL Server Enterprise to help provide high availability, improve performance, and reduce licensing costs.

#### Hardware support

SQL Server 2008 supports 64-bit hardware. SQL Server Enterprise can scale up to the maximum number of processors in the operating system, and its memory usage is limited only by the operating system maximum. With SQL Server Enterprise you can add capacity without incurring downtime by using hot-add CPUs and memory modules.

#### High availability

Help ensure that your servers are highly available with SQL Server clustering. Use up to 16 nodes using N+1 clustering to realize higher return on investment (ROI) by making optimal use of server hardware. Help protect databases and prevent data loss using database mirroring with automatic failover and transparent client redirection. Take advantage of asynchronous database mirroring for higher mirroring performance and multi site disaster recovery. Scale out with high availability using transactional replication.

#### Data compression

Reduce hardware costs, optimize disk usage, and improve performance by compressing data. I/O is one of the biggest bottlenecks in a database system. Reduction in I/O due to compression improves performance.

#### Backup compression

Improve performance and minimize media storage requirements by compressing backups. Compression speeds up backup and restores operations. Back up databases with a smaller window.

#### Encryption, auditing, and extensible key management

Encrypt data without having to modify your applications by using TDE. Simplify and consolidate encryption and key management through SQL Server Enterprise support for extensible key management. Use security principals at the database, server, and operating system levels to control access to data. Use SQL Server Audit to track changes to data and database schemas, and to consolidate enterprise-wide audit records.

#### Predictable performance with Resource Governor

Specify CPU and memory usage limits for defined workloads to help prevent applications from monopolizing resources and to help avoid runaway queries. Prioritize workloads so that important tasks can execute more quickly and reliably.

#### Transfer, upgrade, and migrate with ease

Choose a data transfer approach to consolidate data from a range of heterogeneous sources onto SQL Server 2008:

* Use backup/restore and detach/attach for SQL Server databases.
* Use SQL Server Integration Services for extract, transform, and load (ETL) operations and imports from other technologies, including Oracle, DB2, SAP BI, and Teradata.

Upgrade from SQL Server 2000 to SQL Server 2008 or from SQL Server 2005 to SQL Server 2008. Use SQL Server 2008 Upgrade Advisor to identify and resolve upgrade issues in advance.

#### Scale up consolidated servers

Improve concurrency by using the LOCK\_ESCALATION setting on tables and benefit from improved concurrency on partitioned tables. Reduce disk contention by organizing tables and indexes into partitions and switching in and out of these partitions individually. Use a snapshot of a database mirror as a source for reporting data and avoid spanning queries on production OLTP tables.

### Ease of Management

#### Centralized management with SQL Server Management Studio

Use a single, consistent management interface to manage multiple instances on a single server and multiple SQL Server instances on different virtual machines. Manage the SQL Server Database Engine, SQL Server Analysis Services, SQL Server Integration Services, and SQL Server Reporting Services from a single console.

#### Centralize policy enforcement with Policy-Based Management

Define configuration policies and apply them to servers, databases, tables, and other targets. Define a policy once and reuse it across servers throughout the enterprise. Enforce policies automatically or through ad-hoc checks.

#### Centralize and consolidate monitoring

Use the data collector to gather performance information from multiple sources such as SQL Server Trace, performance counters, dynamic management views (DMVs), Transact-SQL queries, and logs. Store collected data in a management data warehouse for historical analysis, or provide the data to System Center Operations Manager for centralized storage and analysis. Display all of the relevant troubleshooting information in one place by using Reporting Services. Use Performance Studio to troubleshoot, tune, and monitor the state of one or more instances of SQL Server 2008.

### Included Comprehensive Business Intelligence Platform

Use SQL Server’s out-of-the-box BI functionality to gain better business insights and drive profitability. Integration Services , Analysis Services, and Reporting Services enable you to consolidate BI on a single platform for improved interoperability and manageability. Perform sophisticated data analysis, data mining, and ETL operations and create, manage, and distribute rich, visually compelling reports. Use Report Builder 2.0 to put report creation directly into the hands of information workers.