

Estimate performance and capacity requirements for Microsoft Business Connectivity Services

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.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

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.

© 2010 Microsoft Corporation. All rights reserved.

Contents

[Estimate performance and capacity requirements for Microsoft Business Connectivity Services 3](#_Toc262044194)

[Test farm characteristic 3](#_Toc262044195)

[Workload 3](#_Toc262044196)

[Test definitions 3](#_Toc262044197)

[Hardware setting and topology 4](#_Toc262044198)

[Dataset 5](#_Toc262044199)

[Red Zone and Green Zone definitions 6](#_Toc262044200)

[Glossary 6](#_Toc262044201)

[Test results 7](#_Toc262044202)

[Effect of the number of front-end Web servers on latency 7](#_Toc262044203)

[Effect of the number of front-end Web servers on throughput 13](#_Toc262044204)

[Effect of item size on latency 18](#_Toc262044205)

[Effect of item size on throughput 19](#_Toc262044206)

[Effect of number of items on latency 20](#_Toc262044207)

[Effect of number of items on throughput 22](#_Toc262044208)

[Effect of number of items per association on latency 23](#_Toc262044209)

[Effect of number of items per association on throughput 24](#_Toc262044210)

[Recommendations 25](#_Toc262044211)

[Hardware recommendations 25](#_Toc262044212)

[Business Connectivity Services performance-related recommendations 25](#_Toc262044213)

[Scaled-up and scaled-out topologies 26](#_Toc262044214)

[Estimating throughput targets 27](#_Toc262044215)

[Common bottlenecks and their causes 27](#_Toc262044216)

[Troubleshooting performance and scalability 27](#_Toc262044217)

[Performance monitoring 28](#_Toc262044218)

|  |
| --- |
|  |
| Estimate performance and capacity requirements for Microsoft Business Connectivity Services |
|  |

In this article:

* [Test farm characteristic](#section2)
* [Test results](#_Test_Results_1)
* [Recommendations](#_Recommendations)
* [Troubleshooting performance and scalability](#_Troubleshooting_performance_and)

This performance and capacity planning document provides guidance on the footprint that usage of Microsoft® Business Connectivity Services has on topologies running Microsoft SharePoint® Server 2010.

For more information about Business Connectivity Services, see [Business Connectivity Services overview (SharePoint Server 2010)](http://go.microsoft.com/fwlink/?LinkId=192234).

## Test farm characteristic

### Workload

Our tests measured throughput and latency impact on profile pages and external lists. They were designed to help develop estimates of how throughput and latency respond to changes to the following variables:

* Number of front-end Web servers.
* Number of items in the external list.
* Size of the external item.
* Number of items per association.
* Authentication method (PassThrough mode or Secure Store Service authentication).
* External data source (Windows Communication Foundation (WCF) Web service or SQL Server database).
* Load on the front-end Web server measured as CPU usage.

It is important to note that the specific capacity and performance figures presented in this article will be different from the figures in real-world environments. The figures presented are intended to provide a starting point for the design of an appropriately scaled environment. After you have completed your initial system design, test the configuration to determine whether your system will support the factors in your environment.

### Test definitions

This section defines the test scenarios and provides an overview of the test process that was used for each scenario. Detailed information such as test results and specific parameters are given in each of the test results sections later in this article.

|  |  |
| --- | --- |
| Test name | Test description |
| External list | 1. Render a typical external list. 2. Change the features of the external list (number of items, item size, and so on) to see how this affects throughput and latency. |
| Profile page | 1. Render a typical profile page. 2. Change the features of the profile page (number of items per association, item size, and so on) to see how this affects throughput and latency. |

### Hardware setting and topology

###### Lab hardware

To provide a high level of test-result detail, several farm configurations were used for testing. Farm configurations ranged from one to four Web servers and a single database server computer that is running Microsoft SQL Server® 2008 database software.

The following table lists the specific hardware that was used for testing.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Front-End Web Server | Application Server | Database Server | External System |
| Processors | [2 processors @2.33 GHz](mailto:2processorsx4c@2.33GHz) (4 cores) | 2 processors @2.33 GHz (4 cores) | 4 processors @3.2 GHz (4 cores) | 4 processors @3.2 GHz (4 cores) |
| RAM | 8 GB | 8 GB | 32 GB | 32 GB |
| Operating System | Windows Server® 2008 R1 (x64) | Windows Server 2008 R1 (x64) | Windows Server 2008 R1 (x64) | Windows Server 2008 R1 (x64) |
| Number of NICs | 2 | 2 | 2 | 2 |
| NIC speed | 1 GB | 1 GB | 1 GB | 1 GB |
| Authentication | NTLM | NTLM | NTLM | NTLM |
| Software version | SharePoint Server 2010 pre-release version | SharePoint Server 2010 pre-release version | SQL Server 2008 | SQL Server 2008 |

Our tests used two external systems: a WCF Web service and a database. The external systems were hosted on a separate physical computer (details described in the above table). The following list describes the two external systems:

* **WCF Web service** A WCF Web service that returns cached, in-memory data. Data is stored readily in a hash table and returned immediately upon being called by Business Connectivity Services. The data consists of 8000 rows of 25 fields each of various .NET types.
* **Database** A table that has 25 columns, with various data types, and 8000 rows. The table is in a separate database that is hosted on SQL Server 2008.

###### Topology

CPU and memory on the front-end Web server is an important limiting factor for throughput. CPU on the Application server should also be considered for Business Connectivity Services authentication modes that require calls to the Secure Store Service.

We varied our topology by adding additional front-end Web servers.



### Dataset

External list and profile page capacity and performance are highly dependent on the amount of data that is processed. For external lists, the amount of processed data is determined by three variables: the number of items in the external list, the number of columns per item, and the size of each item. The following table describes the representative external lists used in our testing.

|  |  |  |  |
| --- | --- | --- | --- |
| External List | Small | Medium | Large |
| Number of items | 500 | 2000 | 4000 |
| Number of columns per item | 25 | 25 | 25 |
| Item size | 2 KB | 4 KB | 8 KB |

For profile pages, the amount of processed data is dependent on the number and complexity of associations that are used. An association links related external content types within a system. A profile page can contain multiple associations, and each association can have many items. The following table describes the representative profile pages used in our testing.

|  |  |  |  |
| --- | --- | --- | --- |
| Profile Page | Small | Medium | Large |
| Number of associations | 2 | 2 | 10 |
| Number of items per association | 100 | 500 | 2500 |
| Item size | 4 KB | 4 KB | 4 KB |

### Red Zone and Green Zone definitions

For each configuration we ran two tests to determine a Green Zone or recommended throughput that can be sustained and a Red Zone or max throughput that can be tolerated for a short period of time but should be avoided.

To determine our Red Zone and Green Zone user loads, we first conducted a step test and stopped when the following conditions were met:

* For the Green Zone, all the front-end Web servers in our farm have 40-50% CPU usage consistently. This is achieved by increasing the user load that is used and establishing think times in the tests, which means each test may have a different user load.
* For the Red Zone, all the front-end Web servers in the farm have 90-99% CPU usage consistently. This is achieved by increasing the user load that is used in the tests, which means each test may have a different user load.

### Glossary

The following list defines the Business Connectivity Services terms used in this document.

|  |  |
| --- | --- |
| Term | Definition |
| Association | An association links related external content types. For example, customers can be associated with their sales orders. Associations are used in conjunction with Web Parts. |
| External Item | An instance of an external content type. |
| External List | A list of items of an external content type. |
| External System | A supported source of data that can be modeled by Business Connectivity Services, such as a database, Web service, or custom .NET Framework assembly. |
| Profile Page | A profile page displays the data for an item of an external content type. |
| Secure Store Service | A shared service that securely stores credential sets for external data sources and associates those credential sets to identities of individuals or to group identities. |
| Web Part | A reusable component of a SharePoint site that presents information pulled from multiple data sources. |

## Test results

The following sections show the test results of Business Connectivity Services in SharePoint Server 2010. For each group of tests, only certain specific variables are changed to show the progressive impact on farm performance.

The test charts use the following legend to describe the dataset:

**External system, Test type,[Authentication],[Number of associations], Item size, Number of items, CPU load**

|  |  |
| --- | --- |
| Item | Description |
| External system | The external data source: WCF (WCF Web service) or DB (database). |
| Test type | The test type: EL (external list) or PP (profile page). |
| Authentication | The authentication method: SSS (a Secure Store Service authentication mode (for example, WindowsCredentials) is used). If SSS is not specified, PassThrough mode was used during testing. |
| Number of associations | The number of associations in the profile page (for example, 2A). This item applies only to profile page tests. |
| Item size | The size of the item (in KB). |
| Number of items | Number of items in the external list or number of items per association. |
| CPU load | The CPU load: RZ (Red Zone) front-end Web server CPU usage is greater than 90% or GZ (Green Zone) front-end Web server CPU usage is between 40-50%. |

Note that all the Red Zone tests reported on in this article were conducted without think time, a natural delay between consecutive operations. In a real-world environment, each operation is followed by a delay as the user performs the next step in the task. By contrast, in the Red Zone tests, each operation was immediately followed by the next operation, which resulted in a continual load on the farm. This load introduced database contention and other factors that can adversely affect performance.

### Effect of the number of front-end Web servers on latency

The following charts show the test result of the number of front-end Web servers on latency. The test results are distributed over several charts to make it easier to compare related variables.

##### Chart 1:

The following chart shows the results of the external list tests. It can be used to compare how the external system (database or WCF Web service) and CPU load (Red Zone or Green Zone) affect performance.

The following list describes the dataset that was used.

* **WCF,EL,4k,500,RZ**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,4k, 500,GZ**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **DB, EL, 4k,500, GZ**: An external list with 500 items, 4 K of data per item, the external system is a database, and the front-end Web server CPU usage is between 40-50%.

##### Chart 2:

The following chart shows the results of the profile page tests. It can be used to compare how the external system (database or WCF Web service) and CPU load (Red Zone or Green Zone) affect performance.

As you can see from Chart 1 and Chart 2, the average page time stays about the same for both Green Zone and Red Zone scenarios despite adding more front-end Web servers and users. (During testing we increased the user load in order to keep all the front-end Web servers in the necessary CPU activity range). However, this is still a performance gain because increasing the number of front-end Web servers in the farm enables SharePoint Server to serve more users at the same rate. There is a noticeable, about five-fold, improvement in Green Zone cases compared to Red Zone cases. So, it is advisable to keep front-end Web server CPU usage in the 40-50% range.

The following list describes the dataset that was used:

* **WCF,PP,2A, 100,RZ**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **DB, PP,2A,100,RZ**: A profile page with 2 associations, 100 items per association, the external system is a database, and the front-end Web server CPU usage is greater than 90%.
* **WCF, PP,2A,100,GZ**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **DB, PP, 2A,100,GZ**: A profile page with 2 associations, 100 items per association, the external system is a database, and the front-end Web server CPU usage is between 40-50%.

##### Chart 3:

The following chart shows the results of the Red Zone tests. Similar data is paired together so that the only variable is the authentication that is used. It can be used to determine whether using Secure Store Service effects performance.

**NOTE**: Our tests were run using a lab environment. PassThrough mode was used merely for comparison purposes. These tests do not imply that you should use PassThrough mode for authentication.

In Red Zone scenarios, Secure Store Service does not seem to bring noticeable overhead when it comes to average page time. This can be attributed to the fact that Secure Store Service overhead is overshadowed by a bigger factor, which is the high user load. In almost all cases, the Secure Store Service and non-Secure Store Service results are similar.

You should also remember that Secure Store Service will have some minimal impact on the Application Servers as well.

The following list describes the dataset that was used:

* **WCF,EL,4k,500,RZ**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,SSS,4k,500,RZ**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP,2A,100,RZ**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP,SSS,2A,100,RZ**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **DB, EL, SSS,4k,500,RZ**: An external list with 500 items, 4 K of data per item, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **DB, PP,2A,100,RZ**: A profile page with 2 associations, 100 items per association, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.

##### Chart 4:

The following chart shows the results of the Green Zone tests. Similar data is paired together so that the only variable is the authentication that is used. It can be used to determine whether using Secure Store Service effects performance.

**NOTE**: Our tests were run using a lab environment. PassThrough mode was used merely for comparison purposes. These tests do not imply that you should use PassThrough mode for authentication.

The overhead associated with Secure Store Service becomes more apparent for profile pages during normal load scenarios. There is a small, about 20 milliseconds, additional average page time for profile pages.

The following list describes the dataset that was used:

* **WCF, EL,4k,500,GZ**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **WCF, EL,SSS,4k,500,GZ**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **WCF, PP,2A,100,GZ**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **WCF, PP,SSS,2A,100,GZ**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **DB, EL,4k,500,GZ**: An external list with 500 items, 4 K of data per item, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **DB, EL,SSS,4k,500,GZ**: An external list with 500 items, 4 K of data per item, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **DB, PP,2A,100,GZ**: A profile page with 2 associations, 100 items per association, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **DB, PP,SSS,2A,100,GZ**: A profile page with 2 associations, 100 items per association, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.

### Effect of the number of front-end Web servers on throughput

##### Chart 1:

The following chart shows the results of the external list tests. It can be used to compare how the CPU load (Red Zone or Green Zone) affect performance.

The following list describes the dataset that was used:

* **WCF, EL,4k,500,RZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF, EL,4k,500,GZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.

##### Chart 2:

The following chart shows the results of the profile page tests. It can be used to compare how the external system (database or WCF Web service) and CPU load (Red Zone or Green Zone) affect performance.

As you can see from Chart 1 and Chart 2, RPS scales linearly until the third front-end Web server is added. After which, the trend becomes logarithmic, or sub-linear. Although there is added benefit in adding more front-end Web servers to the farm, comparatively it provides less value for the cost. In particular, adding a fourth front-end Web server yields a relatively small benefit.

The following list describes the dataset that was used:

* **WCF,PP,2A,100,RZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **DB,PP,2A,100,RZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a database, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP,2A,100,GZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **DB,PP,2A,100,GZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a database, and the front-end Web server CPU usage is between 40-50%.

##### Chart 3:

The following chart shows the results of the Red Zone tests. Similar data is paired together so that the only variable is the authentication that is used. It can be used to determine whether using Secure Store Service effects performance.

**NOTE**: Our tests were run using a lab environment. PassThrough mode was used merely for comparison purposes. These tests do not imply that you should use PassThrough mode for authentication.

The following list describes the dataset that was used:

* **WCF,EL,4k,500,RZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,SSS,4k,500,RZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP,2A,100,RZ,RPS**: A profile page with 2 association, 100 items per association, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP,SSS,2A,100,RZ,RPS**: A profile page with 2 association, 100 items per association, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **DB,EL,4k,500,RZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.
* **DB,EL,SSS,4k,500,RZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **DB,PP,2A,100,RZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is greater than 90%.
* **DB,PP,SSS,2A,100,RZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.

##### Chart 4:

The following chart shows the results of the Green Zone tests. Similar data is paired together so that the only variable is the authentication that is used. It can be used to determine whether using Secure Store Service effects performance.

**NOTE**: Our tests were run using a lab environment. PassThrough mode was used merely for comparison purposes. These tests do not imply that you should use PassThrough mode for authentication.

As you can see in Chart 3 and Chart 4, Secure Store Service overhead does result in lower RPS values in some cases. However, the linear-logarithmic trend is similar and in almost all cases the RPS difference between Secure Store Service and non-Secure Store Service overhead is subtle.

The following list describes the dataset that was used:

* **WCF,EL,4k,500,GZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **WCF,EL,SSS,4k,500,GZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **WCF,PP,2A,100,GZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **WCF,PP,SSS,2A,100,GZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **DB,EL,4k,500,GZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **DB, EL,SSS,4k,500,GZ,RPS**: An external list with 500 items, 4 K of data per item, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **DB,PP,2A,100,GZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a database, PassThrough authentication mode is used, and the front-end Web server CPU usage is between 40-50%.
* **DB,PP,SSS,2A,GZ,RPS**: A profile page with 2 associations, 100 items per association, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.

### Effect of item size on latency

The following chart shows the test result of external item size on latency. The tests increased the size of the items in the external list and measured the effect on latency.

The following list describes the dataset that was used:

* **WCF,EL,500,RZ**: An external list with 500 items, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,SSS,500,RZ**: An external list with 500 items, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,500,GZ**: An external list with 500 items, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **WCF,EL,SSS,500,GZ**: An external list with 500 items, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.

### Effect of item size on throughput

The following chart shows the test result of external item size on throughput. The tests increased the size of the items in the external list and measured the effect on throughput.

RPS drops sub-linearly as the item size increases in all cases. High load conditions provide more RPS. However, as we learned from earlier test results high load conditions also cause higher page response times.

The following list describes the dataset that was used:

* **WCF,EL,500,RZ**: An external list with 500 items, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,SSS,500,RZ**: An external list with 500 items, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,500,GZ**: An external list with 500 items, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **WCF,EL,SSS,500,GZ**: An external list with 500 items, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.

### Effect of number of items on latency

The following chart shows the test result of the number of items on latency. The tests increased the number of items in the external list and measured how it affected the time it took to render the page.

The following list describes the dataset that was used:

* **WCF,EL,4k,RZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,SSS,4k,RZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **DB,EL,4k,RZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, and the front-end Web server CPU usage is greater than 90%.
* **DB,EL,SSS,4k,RZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,4k,GZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **WCF,EL,SSS,4k,GZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.
* **DB,EL,4k,GZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, and the front-end Web server CPU usage is between 40-50%.
* **DB,EL,SSS,4k,RZ**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.

### Effect of number of items on throughput

The following chart shows the test result of the number of items on throughput. The tests increased the number of items in the external list and measured the effect on throughput.

As you can see in the above chart, RPS drops almost linearly as the number of items increases. Compared to earlier tests that studied the effect of item size, increasing the number of items seems to have more of an effect on performance than increasing the item size.

The following list describes the dataset:

* **WCF,EL,4k,RZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL, SSS,4k,RZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is being used, and the front-end Web server CPU usage is greater than 90%.
* **DB,EL,4k,RZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, and the front-end Web server CPU usage is greater than 90%.
* **DB,EL,SSS,4k,RZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is being used, and the front-end Web server CPU usage is greater than 90%.
* **WCF,EL,4k,GZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF Web service, and the front-end Web server CPU usage is between 40-50%.
* **WCF,EL, SSS,4k,GZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is being used, and the front-end Web server CPU usage is between 40-50%.
* **DB,EL,4k,GZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, and the front-end Web server CPU usage is between 40-50%.
* **DB,EL,SSS,4k,GZ,RPS**: An external list with a variable number of items, each item has 4 K of data, the external system is a database, a Secure Store Service authentication mode (for example, WindowsCredentials) is being used, and the front-end Web server CPU usage is between 40-50%.

### Effect of number of items per association on latency

The following chart shows the test result of the number of items in an association on latency. The tests increased the number of items in an association and measured how it affected the time it took to render the page.

The following list describes the dataset:

* **WCF,PP,2A,RZ**: A profile page with 2 associations, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP, SSS,2A,RZ**: A profile page with 2 associations, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is being used, and the front-end Web server CPU usage is greater than 90%.
* **DB,PP,2A,GZ**: A profile page with 2 associations, the external system is a database, and the front-end Web server CPU usage is between 40-50%.
* **DB,PP,SSS,2A,GZ**: A profile page with 2 associations, the external system is a database, Secure a Store Service authentication mode (for example, WindowsCredentials) is being used, and the front-end Web server CPU usage is between 40-50%.

### Effect of number of items per association on throughput

The following chart shows the test result of the number of items in an association on throughput. The tests increased the number of items in an association and measured the effect on throughput.

The following list describes the dataset:

* **WCF,PP,2A,RZ**: A profile page with 2 associations, the external system is a WCF Web service, and the front-end Web server CPU usage is greater than 90%.
* **WCF,PP,SSS,2A,RZ**: A profile page with 2 associations, the external system is a WCF Web service, a Secure Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is greater than 90%.
* **DB,PP,2A,GZ**: A profile page with 2 associations, the external system is a database, and the front-end Web server CPU usage is between 40-50%.
* **DB,PP,SSS,2A,GZ**: A profile page with 2 associations, the external system is a database, Secure a Store Service authentication mode (for example, WindowsCredentials) is used, and the front-end Web server CPU usage is between 40-50%.

## Recommendations

This section provides general performance and capacity recommendations. Use these recommendations to determine the capacity and performance characteristics of the starting topology that you created and to decide whether you have to scale out or scale up the starting topology.

### Hardware recommendations

For specific information about minimum and recommended system requirements, see [Hardware and software requirements (SharePoint Server 2010)](http://go.microsoft.com/fwlink/?LinkId=166546) (http://go.microsoft.com/fwlink/?LinkId=166546).

|  |
| --- |
| **C:\Users\kvice\AppData\Local\Temp\DxEditor\DduePreview\Default\ad986110-8676-44ca-a97e-ded47ba2a97a\local\note.gifNote:** |
| Memory requirements for Web servers and database servers depend on the size of the farm, the number of concurrent users, and the complexity of features and pages in the farm. The memory recommendations in the following table may be sufficient for a small or light usage farm. However, memory usage should be carefully monitored to determine whether more memory must be added. |

### Business Connectivity Services performance-related recommendations

#### External list recommendations

The following table describes how data gets from the external system into an external list.

|  |  |  |
| --- | --- | --- |
| **Load** | **Process** | **Render** |
| Business Connectivity Services queries the external system and loads the returned data into SharePoint Server. | Applies any additional processing (sort, filter, group) on the loaded data. | The external list renders the items on the page. |

Business Connectivity Services does not have an in-memory cache for external items. The data has to be loaded, processed, and rendered each time an external list is refreshed. Accordingly, many of our recommendations try to limit the amount of data that needs to be processed.

The following list describes the external list recommendations:

* Keep the number of items that need to be processed as low as possible by limiting the number of rows returned from the external system. This is the key factor in external list performance. We recommend keeping the number of rows returned to 100-500. The number of rows returned from the external system should not exceed 2000. You can use filters to limit the number of items returned by the external system. For more information about filters, see [How to: Create an External Content Type Based on a SQL Server Table](http://go.microsoft.com/fwlink/?LinkId=192184) (http://go.microsoft.com/fwlink/?LinkId=192184).
* Rendering a list is CPU intensive on the front-end Web server and the Application server. The number of items rendered is not the same as the total number of items that are loaded and processed. The number of items rendered depends on the configuration of the external list view. Consider the overall user experience, how many items can be comfortably viewed on a screen, and keep the number of items rendered per page to a reasonable number. We recommend keeping the number of items around 30 per page (this is the default).
* Keep the number of columns in an external list to a reasonable number. A large number of columns can affect performance and can also make for a poor user experience (too many columns to comfortably view on a screen).
* Do not include large-sized columns (especially strings) in list views. Columns larger than 1 KB should not be included in a list view. The external content type can still include the large-sized column; however, it should be shown only in the single-item view.
* When designing an external list, configure the default view to be the view that most users will want to see. Changing the sort or filter on a view requires the data to be loaded, processed, and rendered.

#### Profile page recommendations

* The number of associations is a key factor in profile page performance. We recommend keeping the number of associations to 2 at the most for the best performance.
* Expect lower performance numbers (for both throughput and latency) with larger number of items per associations.

#### General Business Connectivity Services recommendations

* There was a noticeable, about five-fold, improvement in performance when we studied the effect of the number of front-end Web servers on latency and compared Green Zone cases to Red Zone cases. We recommend keeping the front-end Web server CPU usage in the 40-50% range.
* Number of items seems to have more of an effect on performance than the item size. If you have control over your external data source, we recommend that you keep the item size high and number of items low for better results. For example, consider aggregating your large data into a single item rather than spreading the data into many items.
* Diagnostic logging levels for Business Connectivity Services can also be an important factor in end-user perceived latency and throughput. We recommend keeping the logging levels at the minimum levels allowed by your business needs for regular usage. Turn the logging levels to a higher verbosity temporarily only when closer monitoring is needed.
* The performance of the external system plays a large role in Business Connectivity Services performance. You should consider the latency and throughput of the external system when you do your capacity and performance planning.

### Scaled-up and scaled-out topologies

You can estimate the performance of your starting-point topology by comparing your topology to the starting-point topologies that are provided in [Plan for availability (SharePoint Server 2010)](http://go.microsoft.com/fwlink/?LinkId=189518) (http://go.microsoft.com/fwlink/?LinkId=189518). Doing so can help you quickly determine whether you must scale up or scale out your starting-point topology to meet your performance and capacity goals.

To increase the capacity and performance of one of the starting-point topologies, you can do one of two things. You can either scale up by increasing the capacity of your existing server computers or scale out by adding additional servers to the topology. This section describes the general performance characteristics of several scaled-out topologies. The sample topologies represent the following common ways to scale out a topology:

* To provide for more user load, add Web server computers.
* To provide for more data load, add capacity to the database server role by increasing the capacity of a single (clustered or mirrored) server, by upgrading to a 64-bit server, or by adding clustered or mirrored servers.
* Maintain a ratio of no more than eight Web server computers to one (clustered or mirrored) database server computer. Although testing in our lab yielded a specific optimum ratio of Web servers to database servers for each test scenario, deployment of more robust hardware, especially for the database server, may yield better results in your environment.

### Estimating throughput targets

Many factors can affect throughput. These factors include the following:

* the number of users
* the type, complexity, and frequency of user operations
* the number of postbacks in an operation
* the performance of data connections.

Each of these factors can have a major impact on farm throughput. You should carefully consider each of these factors when you plan your deployment.

SharePoint Server 2010 can be deployed and configured in a wide variety of ways. As a result, there is no simple way to estimate how many users can be supported by a given number of servers. Therefore, make sure that you conduct testing in your own environment before you deploy SharePoint Server 2010 in a production environment.

## Common bottlenecks and their causes

During performance testing, several different common *bottlenecks* were revealed. A bottleneck is a condition in which the capacity of a particular constituent of a farm is reached. This causes a plateau or decrease in farm throughput.

The following table lists some common bottlenecks and describes their causes and possible resolutions.

### Troubleshooting performance and scalability

|  |  |  |
| --- | --- | --- |
| Bottleneck | Cause | Resolution |
| Database contention (locks) | Database locks prevent multiple users from making conflicting modifications to a set of data. When a set of data is locked by a user or process, no other user or process can modify that same set of data until the first user or process finishes modifying the data and relinquishes the lock. | To help reduce the incidence of database locks, you can:   * Distribute submitted forms to more document libraries. * Scale up the database server. * Tune the database server hard disk for read/write.   Methods exist to circumvent the database locking system in SQL Server 2005, such as the NOLOCK parameter. However, we do not recommend or support use of this method due to the possibility of data corruption. |
| Database server disk I/O | When the number of I/O requests to a hard disk exceeds the disk’s I/O capacity, the requests will be queued. As a result, the time to complete each request increases. | Distributing data files across multiple physical drives allows for parallel I/O. The blog [SharePoint Disk Allocation and Disk I/O](http://go.microsoft.com/fwlink/?LinkId=129557) (http://go.microsoft.com/fwlink/?LinkId=129557) contains much useful information about resolving disk I/O issues. |
| Web server CPU utilization | When a Web server is overloaded with user requests, average CPU utilization will approach 100 percent. This prevents the Web server from responding to requests quickly and can cause timeouts and error messages on client computers. | This issue can be resolved in one of two ways. You can add additional Web servers to the farm to distribute user load, or you can scale up the Web server or servers by adding higher-speed processors. See [Plan for availability (SharePoint Server 2010)](http://go.microsoft.com/fwlink/?LinkId=189518) (http://go.microsoft.com/fwlink/?LinkId=189518) for more information. |

### Performance monitoring

To help you determine when you have to scale up or scale out your system, use performance counters to monitor the health of your system. Use the information in the following tables to determine which performance counters to monitor, and to which process the performance counters should be applied.

##### Web servers

The following table shows performance counters and processes to monitor for Web servers in your farm.

|  |  |  |
| --- | --- | --- |
| Performance counter | Apply to object | Notes |
| Processor time | Total | Shows the percentage of elapsed time that this thread used the processor to execute instructions. |
| Memory utilization | Application pool | Shows the average utilization of system memory for the application pool. You must identify the correct application pool to monitor.  The basic guideline is to identify peak memory utilization for a given Web application, and assign that number plus 10 to the associated application pool. |

##### Database servers

The following table shows performance counters and processes to monitor for database servers in your farm.

|  |  |  |
| --- | --- | --- |
| Performance counter | Apply to object | Notes |
| Average disk queue length | Hard disk that contains SharedServices.mdf | Average values greater than 1.5 per spindle indicate that the write times for that hard disk are insufficient. |
| Processor time | SQL Server process | Average values greater than 80 percent indicate that processor capacity on the database server is insufficient. |
| Processor time | Total | Shows the percentage of elapsed time that this thread used the processor to execute instructions. |
| Memory utilization | Total | Shows the average utilization of system memory. |