

Estimate performance and capacity requirements for InfoPath Forms Services 2010

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.

Estimate performance and capacity requirements for InfoPath Forms Services 2010

John Booze

Microsoft Corporation

April 2010

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

# Contents

Contents

[Contents 3](#_Toc259093836)

[Introduction 4](#_Toc259093837)

[Test farm characteristic 4](#_Toc259093838)

[Hardware setting and topology 4](#_Toc259093839)

[Test results 12](#_Toc259093840)

[Recommendations 26](#_Toc259093841)

[Troubleshooting performance and scalability 27](#_Toc259093842)

# Introduction

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

Testing for this document was designed to help develop estimates of how different farm configurations respond to changes to the following variables:

* Web front end scale out for different submit operations
* Web front end scale out for different InfoPath list operations
* Impact of form complexity on throughput

In this article:

* [Test farm characteristic](#section2)s
* [Test results](#_Test_results)
* [Recommendations](#_Recommendations)
* [Troubleshooting](#_Troubleshooting_performance_and)

## Test farm characteristics

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.

### Hardware setting and topology

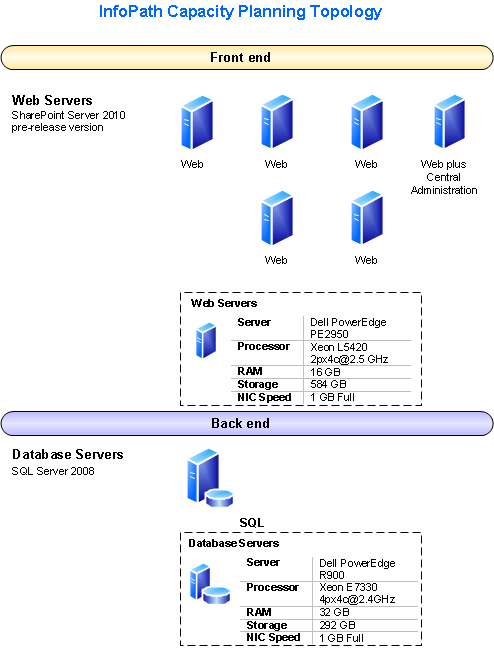
#### Lab hardware

To provide a high level of test-result detail, several farm configurations were used for testing. Farm configurations ranged from one to six Web servers and a single database server computer that is running Microsoft SQL Server® 2008 database software. Load testing was performed with Visual Studio Team System 2008. The test rig consisted of two agent computers. All computers were 64-bit.

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

|  |  |  |  |
| --- | --- | --- | --- |
| Computer name | WFE1-6 | BE1 | Agent1-2 |
| Role | Front-end Web server | SQL Server | Agent |
| Processor(s) | 2x Xeon L5420 @ 2.5 GHz (8 cores) | 4x Xeon E7330 @ 2.4 GHz (16 cores) | 2x Xeon L5420 @ 2.5 GHz (8 cores) |
| RAM | 16 GB | 32 GB | 16 GB |
| Operating System | Windows Server® 2008 R2 | Windows Server 2008 R2 | Windows Server 2008 R2 |
| Storage: OS | 4x 146 GB, 10K RPM, RAID 0 | 2x 146 GB, 15K RPM, RAID 1 | 4x 146 GB, 10K RPM, RAID 0 |
| Storage: Backups |  | 3x 300 GB, 15K RPM, RAID 5 |  |
| Storage: SQL Data |  | 9x 300 GB, 15K RPM, RAID 5 |  |
| Storage: SQL Log |  | 6x 300 GB, 15K RPM, RAID 5 |  |
| # of NICs | 1 | 4 | 1 |
| NIC speed | 1 Gb/sec. | 1 Gb/sec. | 1 Gb/sec. |
| Authentication | NTLM | NTLM | NTLM |
| Software version | SharePoint Server 2010 (Pre-Release Version) | SQL Server 2008 Sp1 CU6 |  |
| # of SQL Instances |  | 1 |  |
| Load balancer type | Windows Network Load Balancing | Windows Network Load Balancing | N/A |
| IRM Settings | Off | Off |  |
| Anti-Virus Settings | Not Installed | Not Installed | Not Installed |

#### Topology

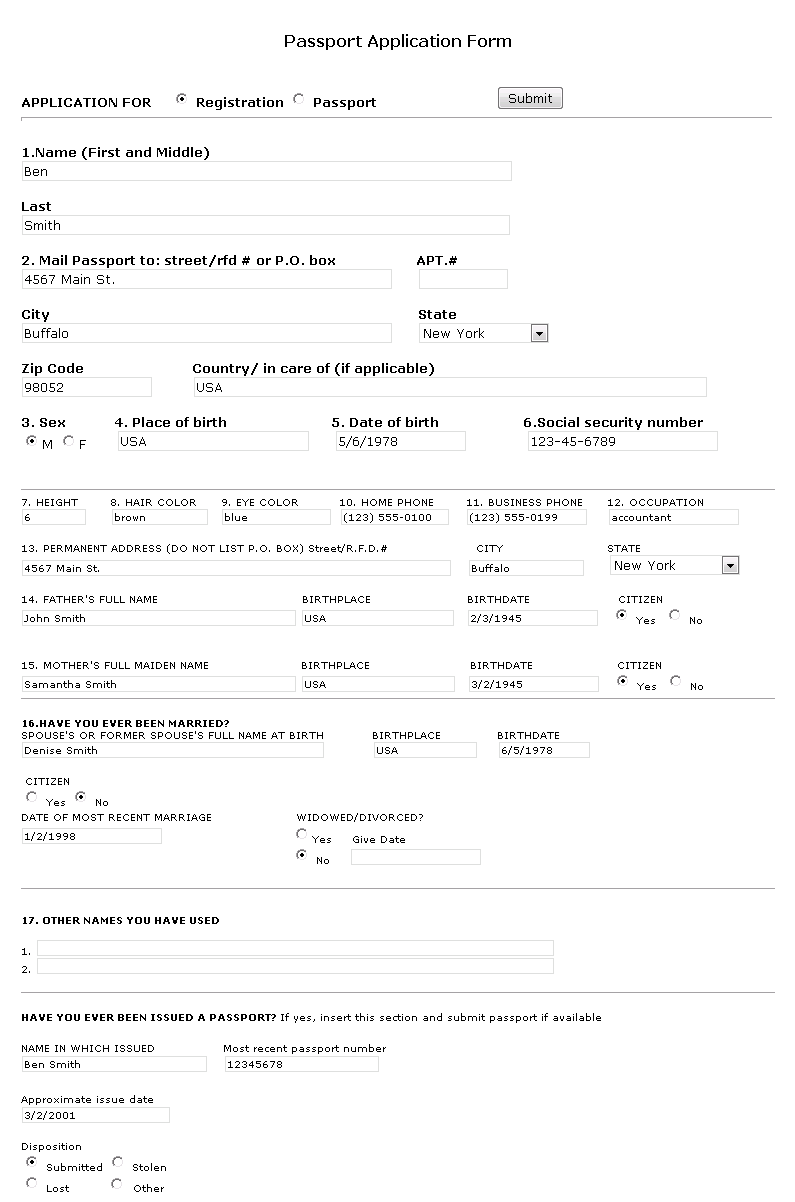


#### Test scenarios

This section defines the test scenarios and provides an overview of the test process that was used for each scenario. Test results are given in later sections in this article.

##### Form templates

Testing was performed with a form template that consists of text boxes, radio buttons, and drop down list boxes. This template will be referred to as the “Baseline Solution.” A screenshot of the form template is pictured below for context.

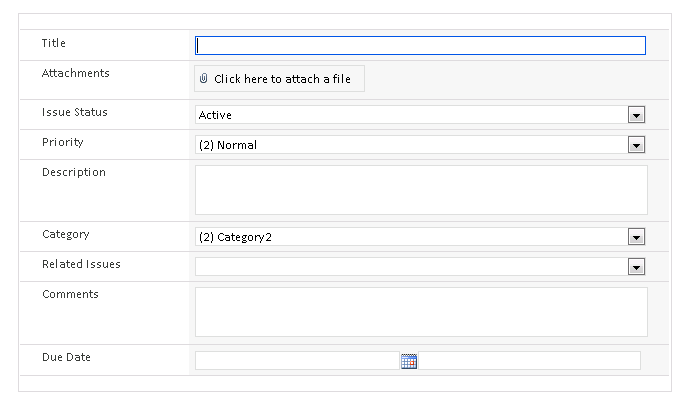


The Baseline Solution was used to create derivative form templates. These form templates are created by making scoped modification to the Baseline Solution template and saving it as a new template. This approach enabled us to do comparison of different operations and aspects of form design. The table below describes the different form templates used in testing.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Form Template | # of Fields | Type of Submit | # of Validation Rules | First Request Optimized | Administrator Deployed | Notes |
| Baseline Solution | 44 | None | 4 | Yes | No |  |
| Baseline Solution with Web Service Submit | 44 | Web Service | 4 | Yes | Yes |  |
| Baseline Solution with Document Library Submit | 44 | SharePoint Document Library | 4 | Yes | Yes |  |
| Baseline Solution without First Request Optimization | 44 | Web Service | 5 | No | Yes | The extra validation rule is “Date is in the past.” Since this rule uses the today() function, the first request requires state data. |
| Baseline Solution with 2x Fields | 88 | Web Service | 4 | Yes | Yes |  |
| Baseline Solution with 3x Fields | 132 | Web Service | 4 | Yes | Yes |  |
| Baseline Solution with 4x Fields | 176 | Web Service | 4 | Yes | Yes |  |
| Baseline Solution with Validation | 44 | Web Service | 10 | No | Yes |  |
| Baseline Solution with 2x Validation | 44 | Web Service | 20 | No | Yes |  |
| Baseline Solution with 4x Validation | 44 | Web Service | 40 | No | Yes |  |

###### InfoPath list form

A modified version of an Issue Tracking list was used to test the InfoPath list form operations. Two modifications were made to the list. First, the Assigned To column was removed. Second, the Related Issues column was set to not allow multiple values. Finally, the list was prepopulated with 100 items. A screenshot of the list is pictured below.



###### Test definitions

###### Scale Out Tests

The table below describes the tests used to for the Web front end scale out tests.

|  |  |  |  |
| --- | --- | --- | --- |
| Scenario Description | Form Template Used | Test Steps | # of Postbacks |
| Baseline Solution New | Baseline Solution | Open a new instance of the Baseline Solution | 0 |
| Save New Baseline Solution | Baseline Solution | Open a new instance of the Baseline Solution  Fill out a form and save it to a document library | 1 |
| Baseline Solution with Document Library Submit | Baseline Solution with Document Library Submit | Open a new instance of the Baseline Solution with Document Library Submit  Fill out a form and click submit, which sends the form data to a SharePoint document library | 1 |
| Baseline Solution with Web Service Submit | Baseline Solution with Web Service Submit | Open a new instance of the Baseline Solution with Web Service Submit  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with Document Library Submit x 5 | 5 copies of the Baseline Solution with Web Service Submit form template with each one deployed to its own document library | For each of the document libraries:  Open a new instance of the Baseline Solution with Document Library Submit  Fill out a form and click submit, which sends the form data to a SharePoint document library | 1 |
| Baseline Solution Open | Baseline Solution with Document Library Submit | Open a Baseline Solution form that has already been filled out. The form is opened from a document library. | 0 |

###### Form Complexity Tests

The table below describes the tests used for the form complexity tests.

|  |  |  |  |
| --- | --- | --- | --- |
| Test name | Form Template Used | Test Steps | # of Postbacks |
| Baseline Solution with 1x Controls | Baseline Solution with Web Service Submit | Open a new instance of the Baseline Solution with Web Service Submit  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with 2x Controls | Baseline Solution with 2x Controls | Open a new instance of the Baseline Solution with 2x Controls  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with 3x Controls | Baseline Solution with 3x Controls | Open a new instance of the Baseline Solution with 3x Controls  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with 4x Controls | Baseline Solution with 4x Controls | Open a new instance of the Baseline Solution with 4x Controls  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution without First Request Optimization | Baseline Solution without First Request Optimization | Open a new instance of the Baseline Solution without First Request Optimization  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with Validation | Baseline Solution with Validation | Open a new instance of the Baseline Solution with Validation  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with 2x Validation | Baseline Solution with 2x Validation | Open a new instance of the Baseline Solution with 2x Validation  Fill out a form and click submit, which sends the form data to a Web service | 1 |
| Baseline Solution with 4x Validation | Baseline Solution with 4x Validation | Open a new instance of the Baseline Solution with 4x Validation  Fill out a form and click submit, which sends the form data to a Web service | 1 |

###### InfoPath List Form Tests

The table below describes the tests used for the InfoPath List form tests.

|  |  |  |
| --- | --- | --- |
| Test name | Test Steps | # of Postbacks |
| Issue Tracking Display | Open an existing issue tracking list item in display view | 0 |
| Issue Tracking Edit | Open an existing issue tracking list item in edit view | 0 |
| Issue Tracking New | Open a new item for the issue tracking list | 0 |

## Test results

Note that all the 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 this testing, 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.

For each topology a series of three tests was run: Calibration, Green Zone and Maximum. The Calibration run uses a step load pattern. A step load pattern increases the number of virtual users over time. The results of the Calibration run determine the user load for the Green Zone and Maximum Throughput tests. The Green Zone and Maximum Throughput tests both use constant load pattern for a period of 5 minutes. The RPS (Requests Per Second) reported in this document is the average RPS at the end of the 5 minute constant load test.

Some of the cells in the results tables have a dash. This indicates that the test was not run for that topology. The test was not run because the results of other runs indicate that there is no expected increase in RPS for that particular topology.

For information about bottlenecks in InfoPath Forms Services in SharePoint Server 2010, see the [Common bottlenecks and their causes](#bottlenecks) section later in this article.

#### Effect of Web front end scale out for different submit operations

The following table shows the “Green Zone” test results of scaling out Web front ends for various submit operations in SharePoint Server 2010.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Baseline Solution Save | Baseline Solution with Web Service Submit | Baseline Solution with SharePoint Submit | Baseline Solution with Sharepoint Submit (5 Document Libraries) |
| 1x1 | 165 | 245 | 160 | 139 |
| 2x1 | 292 | 471 | 301 | 280 |
| 4x1 | 479 | 896 | 478 | 544 |
| 6x1 | 467 | 1395 | - | 599 |

The following graph shows the “Green Zone” throughput for different InfoPath submit operations on different Web front end topologies. SharePoint submit can scale to 4 Web front ends. However, a farm running five document library submit forms in parallel can achieve more throughput with 6 Web front ends than a single document library can with 6 Web front ends. A farm will generally have more than one InfoPath solution deployed. This result means that one of those individual solutions will reach maximum throughput at 4 Web front ends. However, the collective throughput of all the solutions can scale beyond 4 Web front ends. Web service submit has the most throughput and scales to 6 Web front ends.

The following table shows the Maximum test results of scaling out front ends for various submit operations in SharePoint Server 2010.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Baseline Solution Save | Baseline Solution with Web Service Submit | Baseline Solution with SharePoint Submit | Baseline Solution with Sharepoint Submit (5 Document Libraries) |
| 1x1 | 286 | 470 | 301 | 285 |
| 2x1 | 484 | 912 | 464 | 518 |
| 4x1 | - | 1484 | 478 | 601 |
| 6x1 | - | 1483 | - | - |

The following graph shows the maximum throughput for different InfoPath submit operations on different front end topologies. SharePoint submit and save scale to 2 Web front ends. However, a farm running five document library submit forms in parallel can achieve more throughput with 4 Web front ends than a single document library with 4 Web front ends. A farm will generally have more than one InfoPath solution deployed. This result means that one of those individual solutions will reach maximum throughput at 4 Web front ends. However, the collective throughput of all the solutions can scale beyond 4 Web front ends. Web service submit has the most throughput and scales to 4 Web front ends.

#### Effect of Web front end scale out for InfoPath list operations

The following table shows the “Green Zone” test results of adding Web front ends for InfoPath List operations in SharePoint Server 2010.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Issue Tracking Display | Issue Tracking New | Issue Tracking Edit |
| 1x1 | 77 | 67 | 56 |
| 2x1 | 153 | 125 | 106 |
| 4x1 | 295 | 236 | 212 |
| 6x1 | 455 | 431 | 416 |

The following graph shows the “Green Zone” throughput for InfoPath List operations. All the operations see increasing throughput from adding Web front ends. The results also suggest that adding more than 6 Web front ends will continue to increase throughput. This increase has been observed outside the capacity planning testing. The Display operation has more throughput than New, which has more throughput than Edit.

The following table shows the Maximum test results of adding Web front ends for InfoPath List operations in SharePoint Server 2010.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Issue Tracking Display | Issue Tracking New | Issue Tracking Edit |
| 1x1 | 143 | 126 | 100 |
| 2x1 | 263 | 243 | 191 |
| 4x1 | 524 | 457 | 364 |
| 6x1 | 747 | 679 | 521 |

The following graph shows the maximum throughput for the InfoPath list operations. All the operations see increasing throughput from adding Web front ends. The results also suggest that adding more than 6 fronts ends will continue to increase throughput. This increase has been observed outside the capacity planning testing. Display has more throughput than New, which has more throughput than Edit.

#### Effect of Web front end scale out for New and Open operations

The following table shows the test results of adding Web front ends for New and Open InfoPath operations in SharePoint Server 2010.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Issue Tracking New | Issue Tracking Display | Baseline Solution New | Baseline Solution Open |
| 1x1 | 67 | 77 | 197 | 129 |
| 2x1 | 125 | 153 | 379 | 296 |
| 4x1 | 236 | 295 | 802 | 575 |
| 6x1 | 431 | 455 | 1182 | 869 |

The following graph shows the “Green Zone” throughput for New and Open InfoPath operations. All the operations see increasing throughput from adding Web front ends. The results suggest that adding more than 6 Web front ends will continue to increase throughput. This increase has been observed outside the capacity planning testing. Document library New/Open operations have more throughput than InfoPath list New/Display operations.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Issue Tracking New | Issue Tracking Display | Baseline Solution New | Baseline Solution Open |
| 1x1 | 126 | 143 | 408 | 282 |
| 2x1 | 243 | 263 | 775 | 558 |
| 4x1 | 457 | 524 | 1285 | 996 |
| 6x1 | 679 | 747 | 1360 | 1104 |

The following graph shows the maximum throughput for InfoPath list operations. All the operations see increasing throughput from adding Web front ends. The results show that the document library New/Open operations scales to 6 Web front ends. However, the results suggest that the InfoPath list operations could benefit from more than 6 Web front ends. Document library New/Open operations have more throughput than InfoPath list New/Display operations.

#### Effect of form complexity on throughput

The following table shows the test results of adding form controls to a form template. All results were collected on a farm topology that has 4 Web front ends.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Baseline Solution 1x Controls | Baseline Solution 2x Controls | Baseline Solution 3x Controls | Baseline Solution 4x Controls |
| Maximum | 1484 | 1424 | 1310 | 1201 |
| Green Zone | 896 | 834 | 760 | 608 |

The following graph shows the test results of adding form controls to a form template. The number of fields and controls in a form has a measurable effect on throughput. These results show that increasing the number of controls by a factor of 4 can decrease the “Green Zone” throughput over 30%.

The following table shows the test results of adding form controls to a form template. All results were collected on a farm topology that has 4 Web front ends.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Baseline Solution | Baseline Solution Without First Request Optimization | Baseline Solution with Validation | Baseline Solution with 2x Validation | Baseline Solution with 4x Validation |
| Maximum | 1484 | 1323 | 1271 | 1202 | 1074 |
| Green Zone | 896 | 788 | 724 | 676 | 612 |

The following graph shows the test results of adding validation rules to a form template. The number of validation rules in a form has a measureable effect on throughput. These results show that increasing the number of validation rules by a factor of 4 can decrease the “Green Zone” throughput over 30%.

##### Hardware cost per transaction

###### Issue tracking display maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 91.5% | 85.8% | 85.8% | 81.1% |
| Reliability | Avg. Page Time | 0.088 | 0.093 | 0.11 | 0.098 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Baseline Solution New “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 44.1% | 43.7% | 46.5% | 46.5% |
| Reliability | Avg. Page Time | 0.024 | 0.025 | 0.027 | 0.033 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Baseline Solution New Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 93.7% | 91.1% | 77.5% | 54.0% |
| Reliability | Avg. Page Time | 0.048 | 0.050 | 0.052 | 0.056 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Baseline Solution Save “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 40.8% | 41.3% | 37.3% | 24.2% |
| Reliability | Avg. Page Time | 0.059 | 0.074 | 0.099 | 0.10 |
|  | Failure rate | 0% | 0.21% | 0.0014% | 0% |

###### Baseline Solution Save Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 85.8% | 76.8% | - | - |
| Reliability | Avg. Page Time | 0.090 | 0.12 | - | - |
|  | Failure rate | 0% | 0.18% | - | - |

###### Baseline Solution with Document Library Submit “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 40.6% | 44.9% | 35.9% | - |
| Reliability | Avg. Page Time | 0.061 | 0.079 | 0.11 | - |
|  | Failure rate | 0% | 0% | 0% | - |

###### Baseline Solution with Document Library Submit Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 89.1% | 74.8% | - | - |
| Reliability | Avg. Page Time | 0.11 | 0.12 | - | - |
|  | Failure rate | 0.0022% | 0% | - | - |

###### Baseline Solution with Web Service Submit “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 45.0% | 44.0% | 43.8% | 46.0% |
| Reliability | Avg. Page Time | 0.040 | 0.042 | 0.046 | 0.059 |
|  | Failure rate | 0% | 0% | 0.00074% | 0% |

###### Baseline Solution with Web Service Submit Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 91.8% | 91.4% | 74.6% | 48.9% |
| Reliability | Avg. Page Time | 0.076 | 0.080 | 0.091 | 0.11 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Baseline Solution with Document Library Submit x 5 “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 38.4% | 39.8% | 40.8% | - |
| Reliability | Avg. Page Time | 0.070 | 0.077 | 0.10 | - |
|  | Failure rate | 0% | 0% | 0% | - |

###### Baseline Solution with Document Library Submit x 5 Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 88.4% | 80.5% | 44.3% | 29.7% |
| Reliability | Avg. Page Time | 0.12 | 0.16 | 0.12 | 0.12 |
|  | Failure rate | 0% | 0% | 0.000011% | 0% |

###### Baseline Solution Open “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 39.2% | 45.8% | 45.5% | 46.2% |
| Reliability | Avg. Page Time | 0.036 | 0.038 | 0.041 | 0.049 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Baseline Solution Open Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 90.6% | 90.6% | 82.1% | 60.0% |
| Reliability | Avg. Page Time | 0.063 | 0.067 | 0.069 | 0.084 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Issue Tracking Display Green Zone RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 44.8% | 45.4% | 44.6% | 46.4% |
| Reliability | Avg. Page Time | 0.061 | 0.067 | 0.073 | 0.072 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Issue Tracking Display Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 91.5% | 85.8% | 85.8% | 81.1% |
| Reliability | Avg. Page Time | 0.088 | 0.093 | 0.11 | 0.098 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Issue Tracking Edit “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 45.7% | 43.6% | 45.1% | 60.0% |
| Reliability | Avg. Page Time | 0.086 | 0.090 | 0.10 | 0.11 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Issue Tracking Display “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 89.8% | 87.2% | 82.9% | 79.3% |
| Reliability | Avg. Page Time | 0.12 | 0.13 | 0.13 | 0.14 |
|  | Failure rate | 0% | 0% | 0.00092% | 0.012% |

###### Issue Tracking Display Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 91.5% | 85.8% | 85.8% | 81.1% |
| Reliability | Avg. Page Time | 0.088 | 0.093 | 0.11 | 0.098 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Issue Tracking New “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 44.8% | 42.9% | 40.9% | 50.5% |
| Reliability | Avg. Page Time | 0.072 | 0.076 | 0.089 | 0.097 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Issue Tracking New Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x1 | 2x1 | 4x1 | 6x1 |
| CPU | Avg front-end Web server CPU | 92.6% | 89.2% | 85.1% | 84.9% |
| Reliability | Avg. Page Time | 0.12 | 0.12 | 0.12 | 0.14 |
|  | Failure rate | 0% | 0% | 0% | 0% |

###### Baseline Solution Controls “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x | 2xControls | 3xControls | 4xControls |
| CPU | Avg front-end Web server CPU |  | 43.9% | 49.8% |  |
| Reliability | Avg. Page Time |  | 0.050 | 0.054 |  |
|  | Failure rate |  | 0% | 0% |  |

###### Baseline Solution Controls Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | 1x | 2xControls | 3xControls | 4xControls |
| CPU | Avg front-end Web server CPU |  | 79.2% | 80.9% | 80.2% |
| Reliability | Avg. Page Time |  | 0.098 | 0.12 | 0.12 |
|  | Failure rate |  | 0% | 0% | 0.00056% |

###### Baseline Solution Validation “Green Zone” RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | Without first request optimization | 1xValidation | 2xValidation | 4xValidation |
| CPU | Avg front-end Web server CPU | 45.4% | 44.7% | 45.5% | 46.3% |
| Reliability | Avg. Page Time | 0.055 | 0.057 | 0.061 | 0.068 |
|  | Failure rate | 0% | 0% | 0.19% | 0% |

###### Baseline Solution Validation Maximum RPS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Scorecard Dashboard | Scorecard Metric | Without first request optimization | 1xValidation | 2xValidation | 4xValidation |
| CPU | Avg front-end Web server CPU | 80.4% | 82.4% | 86.8% | 85.2% |
| Reliability | Avg. Page Time | 0.10 | 0.11 | 0.13 | 0.11 |
|  | Failure rate | 0.0015% | 0% | 0% | 0.00055% |

## 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 using the [Topologies for SharePoint Server 2010 model](http://technet.microsoft.com/en-us/library/cc263199(office.14).aspx) 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 <TBD>

|  |
| --- |
| **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. |

##### Scaled-up and scaled-out topologies

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 for an InfoPath Forms Services scenario:

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

##### Optimizations

This following sections discuss methods for improving farm performance by optimizing form templates and the database server.

###### Form template design optimizations

* Optimize the first request (that is, the request to open the form) for form templates without onLoad events or business logic. Optimize the first request by delaying the creation of session state entry in the database until a POST occurs. Note that for such form templates, if the only POST was to close the form after Submit, the SQL session state will not be created. To apply this optimization, the form designer must set the Submit advanced setting to close the form after Submit. For more information about form template design optimizations, see the six-part blog series at [Designing browser-enabled forms for performance in InfoPath Forms Services](http://go.microsoft.com/fwlink/?LinkId=129548) (http://go.microsoft.com/fwlink/?LinkId=129548).
* If a scenario involves saving a form to a document library, it is better to submit the form to the library instead of saving it. A Submit operation triggers only one POST request or round trip, whereas a Save operation triggers two POST requests. The name of the form can be dynamically generated by using a rule or by using a control in the form.
* Document library forms can achieve greater throughput than InfoPath list forms. If high throughput is needed for a solution, consider using a document library form instead of an InfoPath list form.
* Form complexity such as the number of controls and amount of form logic affects throughput. As form complexity increases, the Web front end CPU cost also increases. Therefore more complex forms need more Web front ends to achieve greater throughput.
* To reduce user latency, we recommend that the form designer reduce the number of controls per view. For first-page view optimization, position controls that have a high resource cost, such as rich text fields, in subsequent views instead of in the default view.

###### 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. |

##### 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. |

###### See Also

[InfoPath Forms Services 2010 Web Testing Toolkit](http://codeplex.com/ipfswebtest)