Requirement Analysis

Document history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Author | Changes | Valid from |
| 1.0 | 19-Jul-2016 | Changming Wu | Draft | 19-Jul-2016 |
| 1.1 | 15-Aug-2016 | Hardik Kansara | Edit | 20-Aug-2016 |

Table of Contents

[1 Introduction 3](#_Toc461891262)

[1.1 Purpose of the system 3](#_Toc461891263)

[1.2 Scope of the system 3](#_Toc461891264)

[1.3 Core System Functionalities 4](#_Toc461891265)

[1.4 Objectives and Success Criteria of the Project 11](#_Toc461891266)

[2 Current System 11](#_Toc461891267)

[3 Proposed System 12](#_Toc461891268)

[3.1 Overview 12](#_Toc461891269)

[3.2 Functional Requirements 12](#_Toc461891270)

[3.3 Non-functional Requirements 13](#_Toc461891271)

[3.3.1 Usability 13](#_Toc461891272)

[3.3.2 Reliability 13](#_Toc461891273)

[3.3.3 Performance 13](#_Toc461891274)

[3.3.4 Supportability 14](#_Toc461891275)

[3.3.5 Implementation 14](#_Toc461891276)

# Introduction

## Purpose of the system

Secure and efficient database implementation with suggested best practice is one of the challenge for an organization and as well as an important part of the duty for a database administrator,

1. Database authentication and network communication port need to be considered after database installation and essential to check especially in production environment to overcome the safety loopholes by changing default database port and default system users access in database.
2. There are critical default installation options include drive and database physical structure selections which are not easy to change after the database is hosted.
3. Old database installation software version is also considered to raise a concern for database performance and security.

4. Other things to consider are memory, CPU, database recovery model and compatibility level parameters which all affect database performance if they are not tuned.

Identifying details of SQL Server configurations is a time consuming process and lots of manual work for a new database administrator who wants to know the configured parameters values of all SQL Server instances and for all databases without well maintained inventory.

Sometimes it is very difficult for a database administrator to check and create an inventory of all the database parameters especially for large SQL Server environment. Maintaining the databases accordance to best practice will also become challenging when organisation has so many databases.

## Scope of the system

A database evaluator that will diagnose and check the basic implementation parameters of SQL Server Instances and Databases. The database evaluator will target all versions of SQL Server starting with SQL Server 2008 and SQL Server 2012. Older versions of SQL server will not be supported.

1. A website from where a customer can download the client application and upload the database dump files.

1. A client application that will produce a dump file of the existing client’s database and SQL instance parameters values only.

3. A database evaluator tool that will generate reports from the dump files uploaded by customers.

It is planned that these dump files will be encrypted for security purposes. The client application is free to download from the product website and profit will be generated instead when customers choose to avail of the database evaluation services.

## Core System Functionalities

* Website

Customer can download a client application from website. Installation guide and instruction are available on the website.

The website will allow customer to upload the encrypted files after they have successfully logged in to their account.

* Client Application

A client application is a light weight and small tool which will be executed on a customer database system and gather the existing configured values for parameters and captures the encrypted files.

The following areas of database from where parameters values will be collected by client application.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SQL Server Instance** | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Installation |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | SQL Server Instance Installation Directory | | | | | |
| Issue Type : | Installation | | | | Issue Severity : | High |
| Problem : | By default, SQL Server’s instance binary files are installed on the system drive. This is a recipe for disaster because if the system drives out of space or corrupt then the SQL Server instance will stop. | | | | | |
|  |
|  |
| Recommendation : | The recommendation is to installing the latest updates on the SQL server. | | | | | |
| Why : | After this change, you’ll have less reliability risk, and your SQL Server instance is safe in case of disaster. | | | | | |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | SQL Server Version and Service Pack | | | | | |
| Issue Type : | Installation | | | | Issue Severity : | High |
| Problem : | The SQL server is running on service pack version which is unsupported. | | | | | |
| Recommendation : | The recommendation is to install the latest updates on the SQL server. | | | | | |
| Why : | A service pack comprises a collection of updates, fixes or enhancements to a software program delivered in the form of a single installable package. | | | | | |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Configuration |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Max Degree Of Parallelism | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | Medium |
| Problem : | The Max Degree of Parallelism is a server wide configuration that by  default uses all of the CPUs to have the available portions of the query executed in parallel. | | | | | |
|  |
|  |
| Recommendation : | The recommendation is to set the max degree of parallelism. | | | | | |
| Why : | When SQL Server runs on a computer with more than one processor or CPU, it detects the best degree of parallelism that is the number of processors employed to run a single statement, for each query that has a parallel execution plan. You can use the max degree of parallelism option to limit the number of processors to use for parallel plan execution and to prevent run-away queries from impacting SQL Server performance by using all available CPUs. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Memory | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | Medium |
| Problem : | By default, SQL Server’s max memory is 2147483647 – a heck of a lot more than you actually have. SQL Server will just keep using more and more memory until there’s none left on the system. If the operating system has no memory available, it will start using the page file instead of RAM. Using the page file in place of memory will result in poor system performance – operations that should be fast and in memory will read and write to disk constantly. | | | | | |
|  |
|  |
|  |
|  |
|  |
| Recommendation : | The recommendation is to set min server memory and max server memory to span a range of memory values. | | | | | |
|  |
| Why : | Use min server memory to guarantee a minimum amount of memory available to the SQL Server Memory Manager for an instance of SQL Server. SQL Server will not immediately allocate the amount of memory specified in min server memory on startup. However, after memory usage has reached this value due to client load, SQL Server cannot free memory unless the value of min server memory is reduced. | | | | | |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Enable Traceflag 2371, 1117 and 1118 | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | Low |
| Problem : | 2371 - Trace flag 2371 that you can use to control when the query optimizer generates autostats on a table. when a table becomes very large, the old threshold (a fixed rate – 20% of rows changed) may be too high and the Autostat process may not be triggered frequently enough. This could lead to potential performance problems.  The trace flag 1118 is commonly used to assist in TEMPDB scalability by avoiding SGAM and other allocation contention points.   Trace flag 1117 changes the behavior of file growth: if one data file in a filegroup grows, it forces other files in that filegroup to ALSO grow. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| Recommendation : | The recommendation is to enable trace flags. | | | | | |
| Why : | Enabling TraceFlags can help SQL server to handle a certain data load more accurate. Trace flag 1118 forces uniform extent allocations of the Tempdb datafiles instead of mixed page allocations. When trace flag 1117 is enabled, then when SQL Server has to perform auto-grow of a data file, it auto-grows all of the files at the same time. | | | | | |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Default index fill factor | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | Low |
| Problem : | The fill-factor option is provided for fine-tuning index data storage and performance. When an index is created or rebuilt, the fill-factor value determines the percentage of space on each leaf-level page to be filled with data, reserving the remainder on each page as free space for future growth.  Fillfactor can be a useful tool to help performance, but it’s often a performance killer if you use it incorrectly. | | | | | |
|  |
|  |
|  |
|  |
|  |
| Recommendation : | The recommendation is to set the fill-factor value in percentage from 1 to 100, and the server-wide default is 0 which means that the leaf-level pages are filled to capacity. | | | | | |
|  |
| Why : | The fill factor option “determines the percentage of space on each leaf-level page to be filled with data, reserving the remainder on each page as free space for future growth”. The idea is that an appropriate fill factor should reduce page splits whilst maintaining performance and using space efficiently. | | | | | |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Security |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Server authentication | | | | | |
| Issue Type : | Security | | | | Issue Severity : | Medium |
| Problem : | The sa account is a well-known SQL Server account and it is often targeted by malicious users. Do not enable the sa account unless your application requires it. It is very important that you use a strong password for the sa login. | | | | | |
|  |
|  |
| Recommendation : | The recommendation is to set it to Windows Authentication. | | | | | |
| Why : | When a user connects through a Windows user account, SQL Server validates the account name and password using the Windows principal token in the operating system. This means that the user identity is confirmed by Windows. SQL Server does not ask for the password, and does not perform the identity validation. Windows Authentication is the default authentication mode, and is much more secure than SQL Server Authentication. Windows Authentication uses Kerberos security protocol, provides password policy enforcement with regard to complexity validation for strong passwords, provides support for account lockout, and supports password expiration. A connection made using Windows Authentication is sometimes called a trusted connection, because SQL Server trusts the credentials provided by Windows.  By using Windows Authentication, Windows groups can be created at the domain level, and a login can be created on SQL Server for the entire group. Managing access from at the domain level can simplify account administration. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | SQL Server Network Port | | | | | |
| Issue Type : | Security | | | | Issue Severity : | High |
| Problem : | This is the most common port allowed through the firewall. It applies to routine connections to the default installation of the Database Engine, or a named instance that is the only instance running on the computer. | | | | | |
|  |
|  |
| Recommendation : | The recommendation is to change the default port. | | | | | |
| Why : | Firewall systems help prevent unauthorized access to computer resources. If a firewall is turned on but not correctly configured, allow attempts to connect to SQL Server on default port. | | | | | |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **SQL Server Database** | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Implemetation |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Database File Configuration : DBName | | | | | |
| Issue Type : | Implemetation | | | | Issue Severity : | High |
| Problem : | By default, SQL Server’s databases are installed on the system drive. This is a recipe for disaster under two situations:  1. If your users do something that require the system databases to grow (like never purging backup history from MSDB or building giant temp tables), they can grow until the system drive runs out of space, and Windows will stop dead.  2.If someone else does something to run the system drives out of space (like downloading huge files to their desktop or running big Windows Updates), the system will stop dead. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
| Recommendation : | The recommendation is to databse files on separate drive. | | | | | |
| Why : | After this change, you’ll have less reliability risk, and your system may even perform faster if the C drive ran on slow storage. | | | | | |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Configuration |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Recovery Model | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to set the full recovery mode for prodcution database. | | | | | |
| Why : | The simple recovery model is generally appropriate for a test or development database. However, for a production database, the best choice is typically the full recovery model, optionally, supplemented by the bulk-logged recovery model. However, the simple recovery model is sometimes appropriate for a small production database, especially if it is mostly or completely read-only, or for a data warehouse. | | | | | |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Compatibility Level | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to set the compatibility level. Same as version of SQL Server (10.5 or 11) | | | | | |
|  |
| Why : | Whenever you move to a newer version of SQL Server or upgrade an existing server to a newer version using either the backup & restore method or detach & attach method, the compatibility level of the individual databases on it do not automatically get upgraded as well. This means your databases will still act as though they are running on an earlier version of SQL Server. This is actually intentional as occasionally features and behaviors will change between versions so its better to give the final decision to upgrade a level or not. In reality this is not a major problem, but in the long run you will not be able to take advantage of newer features unless you upgrade your databases compatibly level to the current version. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Read Committed Snapshot | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to enable read committed snapshot for the database. | | | | | |
| Why : | The term "snapshot" reflects the fact that all queries in the transaction see the same version, or snapshot, of the database, based on the state of the database at the moment in time when the transaction begins. Transactions that modify data do not block transactions that read data, and transactions that read data do not block transactions that write data, as they normally would under the default READ COMMITTED isolation level in SQL Server. When READ\_COMMITTED\_SNAPSHOT OFF is in effect, the Database Engine uses shared locks to enforce the default isolation level. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Database Auto growth | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to set the auto growth setting for database enough to start with 200MB to 500MB. | | | | | |
|  |
| Why : | An auto-growth event is the process by which the SQL Server engine expands the size of a database file when it runs out of space. The amount by which a database file grows is based on the settings that you have for the file growth options for your database. Each database file that is associated with your database has an auto-growth setting. There are three different settings you can use to identify how your database files will grow. They can grow by a specific size, a percentage of the current size, or not grow at all.   If you are not properly managing your auto-growth setting for a database, then your database might experience many auto-grow events, or very few. Auto-growth events. Each time an auto-growth event is performed SQL Server holds up database processing while an auto-growth event occurs. This means that processing against that database will be held up while the auto-growth event completed. This equates to slower response time for those SQL commands that are being processing against the database that is growing. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Auto Create Statistics | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to enable auto create statistics for database. | | | | | |
| Why : | Accurate statistics about the data held in tables are used to provide the best execution strategy for SQL queries. but if the statistics don't accurately reflect the current contents of the table you'll get a poorly-performing query.   If you’ve set the option AUTO CREATE STATISTICS OFF and overlooked the task of creating statistics manually, the optimizer will suffer from missing statistics. | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Auto Shrink | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to disable auto shrink for database. | | | | | |
| Why : | One other common thing I see is to have auto-shrink set on for one or databases. This is bad for several reasons:  - Shrink causes index fragmentation. - Although it doesn’t have any effect like long-term blocking, it does take up a lot of resources, both IO and CPU. - Repeatedly shrinking and growing the data files will cause file-system level fragmentation, which can slow down performance | | | | | |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Auto Update Statistics | | | | | |
| Issue Type : | Configuration | | | | Issue Severity : | High |
| Recommendation : | The recommendation is to enable auto update statistics for database. | | | | | |
| Why : | When data changes, SQL Server will automatically maintain the statistics on indexes that explicitly create, if that setting is enabled | | | | | |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Maintenance |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Daily Index Rebuild - Index Optimizations | | | | | |
| Issue Type : | Maintenance | | | | Issue Severity : | Medium |
| Problem : | Regular index and statistics maintenance performe on the database. This can be based on Fragmentation (“Rebuild and Reorganize Indexes”) | | | | | |
|  |
| Recommendation : | Current Indexes need to be Optimized Create Clustered Indexes from Heap Tables | | | | | |
|  |
| Why : | As a general practice all tables in an OLTP database should have clustered indexes, even those whose contents are temporary (lasting for the duration of a transaction or batch process). The following whitepaper is available on the Microsoft site, and provides background behind this recommendation: | | | | | |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Daily database Full backup | | | | | |
| Issue Type : | Maintenance | | | | Issue Severity : | High |
| Problem : | The production goals of the organization for the databases, especially the requirements for availability and protection of data from loss. | | | | | |
|  |
| Recommendation : | The recommendation is to perform the Full DB compressed backup every day and verify backup integrity. Backup startegy is also depends on other factors such as the nature of each of your databases : its size, its usage patterns, the nature of its content and the requirements for its data. | | | | | |
|  |
|  |
|  |
| Why : | Best practices for backup and restore help make sure that backup and restore operations in database environments are successful and that the environment is protected against data loss or continuity gaps. | | | | | |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Security |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | Blank SQL 'SA' Password | | | | | |
| Issue Type : | Security | | | | Issue Severity : | Medium |
| Problem : | The sa account is a well-known SQL Server account and it is often targeted by malicious users. Do not enable the sa account unless your application requires it. It is very important that you use a strong password for the sa login. | | | | | |
|  |
|  |
| Recommendation : | The recommendation is to set strong password for SA. | | | | | |
| Why : | While you install Microsoft SQL Server, if you choose the SQL Server and Windows mode option for Security Authentication, you receive a prompt in which you must enter a password for the system administrator (SA) account. If you install SQL Server with the Windows only mode option, and then you later change the Security Authentication to SQL Server and Windows mode, the SA password is left blank (NULL). | | | | | |
|  |
|  |
|  |
|  |
|  |  |  |  |  |  |  |  |  |  |
| Parameter Name : | NT AUTHORITY\SYSTEM Administrator | | | | | |
| Issue Type : | Security | | | | Issue Severity : | High |
| Problem : | It is a member of the Windows Administrators group on the local computer, and is therefore a member of the SQL Server sysadmin fixed server role. | | | | | |
|  |
| Recommendation : | The recommendation is to disable the account in SQL Server. | | | | | |
| Why : | Local System Account has extensive privileges on the entire local system and acts as a computer on your company’s network. This account shows up as “NT AUTHORITY\SYSTEM” when configuring SQL. | | | | | |
|  |
|  |

* Database Evaluator

Parameters in encrypted file will be diagnosed with another application called Database Evaluator. Database Evaluator will check all the parameter against the best practice values and generate the final report. A small database will be created for database evaluator application which store all the best practice values and information about latest service pack for SQL server. This database is manage and update by project team.

The final report will display all information about what parameters are not configured and suggest and what will be the parameter values as per suggested best practice.

The customer can download their final report from the website.

## Objectives and Success Criteria of the Project

The success of the project depends upon meeting the following core set of objectives

1. A functional website with ability to download the client application.
2. Website can also work as FTP to upload dump file and download the final report.
3. Basic installation procedure are available on website.
4. Customer create their account to upload dump file and download the final report
5. Client application is small and light weight application which will be executed by customer on their database system and application generated encrypted dump file which contains all the agreed database and SQL instance parameter values.
6. Database Evaluator (DBE) is able to generate the final report after check all the parameters from dump file.

# Current System

No system is in place at the moment with the same functionality.

# Proposed System

## Overview

This section provides a functional overview of the system. This will again be properly be divided into two parts.



## Functional Requirements

* Customer must be able to download the client application and install on customer database system.
* Client application must be able to generate the encrypted dump file with all required parameter values.
* Customer must be able to select server name and able to connect the SQL Server instance by clicking on connect button.
* Customer must have option to select any single database if he want to evaluate.
* Customer must be able to see the progress of execution.
* Customer must be able to create their own account on website.
* Customer must have basic help and installation procedure available on website.
* Customer must be able to upload the dump file and download the final report from their account.
* Customer must have valid email address.
* Website must be able to send the required notifications.

## Non-functional Requirements

### 3.3.1 Usability

* Client application and database evaluator must be run on windows environment only and used by customer.
* Website is platform independent, must be supported all widely used web browser and used by customer and project team.
* “About” and “Help” information about the product must be able to be retrieved at any point during the run of application.
* Customer must be able to choose the server name from a list. Green mark must be there to indicate that connection is successful.
* Customer see the progress on execution for client application.
* Client application must be downloaded from website.
* Client and project team must be notify while uploading the dump file and final report.

### 3.3.2 Reliability

* Components of the project will be tested alongside the implementation phase to ensure that they are functional.
* Final, integrated project Code will be tested to ensure that greater than or equal to 80% of the integrated code is covered at run-time, and is functioning properly.

### 3.3.3 Performance

* Client application collects the parameter values without any error and modifying existing values.
* Encrypted dump file will be uploaded on website without any error.
* Database Evaluator will check and produce the final report without any errors.

### 3.3.4 Supportability

* The client application is platform dependent as it should be able to run on platform supporting .net framework only.
* Client application and database evaluator support the SQL Server 2008 R2 and SQL Server 2012.
* Website is compatible to run with all popular web browser.
* Database evaluator is also platform dependent as it should be able to run on platform supporting .net framework only.

### 3.3.5 Implementation

* Client application and database evaluator will be developed in C#.
* SQL Server 2008 R2 and SQL Server 2012 will be used for creating database, generate and test SQL script.
* Website will be created using a Wordpress and MySQL.