***LOGO DA EMPRESA***

Análise Instâncias

Microsoft SQL Server 2019

XPTO

**Cliente:** Empresa XPTO

**Data:** xx/xx/20xx

Índice

[Introdução 4](#_Toc183089652)

[Análise Windows 5](#_Toc183089653)

[►Sessões de Remote Desktop Consumindo Recursos 5](#_Toc183089654)

[►Serviço Analysis Services Ativo sem uso 5](#_Toc183089655)

[►Serviço SQL Browser Ativo 6](#_Toc183089656)

[*Análise de Desempenho* 7](#_Toc183089657)

[►Processador 7](#_Toc183089658)

[►Memória 8](#_Toc183089659)

[►Rede 8](#_Toc183089660)

[*Instâncias SQL Server* 9](#_Toc183089661)

[Análise Instância DEFAULT 9](#_Toc183089662)

[►xp\_cmdshell 9](#_Toc183089663)

[►Configuração dos Bancos 10](#_Toc183089664)

[►Collation 11](#_Toc183089665)

[►Compatibilidade dos Bancos 11](#_Toc183089666)

[►Page Verify 11](#_Toc183089667)

[►JOB para Limpar Todas as Linhas de Tabelas 11](#_Toc183089668)

[► Ausência de Rotina de Desfragmentação 12](#_Toc183089669)

[► Teste de Integridade 13](#_Toc183089670)

[► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados 13](#_Toc183089671)

[► Administração Proativa do Servidor 13](#_Toc183089672)

[► Configuração para Envio de E-mails 14](#_Toc183089673)

[► Propriedade Auto Shrink Habilitada 15](#_Toc183089674)

[► Segurança Login “SA” 16](#_Toc183089675)

[► Rotina de Backup e Manutenção 16](#_Toc183089676)

[► Ausência de Solução de Alta Disponibilidade 17](#_Toc183089677)

[*Análise de Desempenho* 19](#_Toc183089678)

[►Disco 19](#_Toc183089679)

[►SQL Server 21](#_Toc183089680)

[Análise Instância KLDW 25](#_Toc183089681)

[►Configuração dos Bancos 25](#_Toc183089682)

[► Teste de Integridade 25](#_Toc183089683)

[► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados 26](#_Toc183089684)

[► Administração Proativa do Servidor 26](#_Toc183089685)

[► Configuração para Envio de E-mails 27](#_Toc183089686)

[► Segurança Login “SA” 27](#_Toc183089687)

[► Rotina de Backup e Manutenção 27](#_Toc183089688)

[► Pacotes de Carga Integration Services (SSIS) 27](#_Toc183089689)

[*Análise de Desempenho* 29](#_Toc183089690)

[►SQL Server 29](#_Toc183089691)

[Análise Instância SHAREPOINT 30](#_Toc183089692)

[► Configuração dos Bancos 30](#_Toc183089693)

[► Compatibilidade dos Bancos 30](#_Toc183089694)

[► Propriedade de Manutenção e Criação de Estatísticas 31](#_Toc183089695)

[► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados 31](#_Toc183089696)

[► Ausência de Rotina de Desfragmentação 31](#_Toc183089697)

[► Teste de Integridade 31](#_Toc183089698)

[► Administração Proativa do Servidor 32](#_Toc183089699)

[► Configuração para Envio de E-mails 32](#_Toc183089700)

[► Segurança Login “SA” 32](#_Toc183089701)

[► Rotina de Backup e Manutenção 33](#_Toc183089702)

[*Análise de Desempenho* 33](#_Toc183089703)

[►SQL Server 33](#_Toc183089704)

[Análise Instância TFS 34](#_Toc183089705)

[►Configuração dos Bancos 34](#_Toc183089706)

[► Compatibilidade dos Bancos 34](#_Toc183089707)

[► Teste de Integridade 34](#_Toc183089708)

[► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados 35](#_Toc183089709)

[► Administração Proativa do Servidor 35](#_Toc183089710)

[► Configuração para Envio de E-mails 36](#_Toc183089711)

[► Segurança Login “SA” 36](#_Toc183089712)

[► Rotina de Backup e Manutenção 36](#_Toc183089713)

[*Análise de Desempenho* 36](#_Toc183089714)

[►SQL Server 36](#_Toc183089715)

[Conclusão 37](#_Toc183089716)

# **Introdução**

Atendendo a solicitação da *EMPRESA XPTO*, desenvolvi este relatório com o objetivo de analisar a instância de SQL Server, propondo ajustes para melhoria de desempenho.

Dos dias xx a xx de xxxx de 20xx foram coletadas informações da instância, com base nestes dados elaborei o presente relatório.

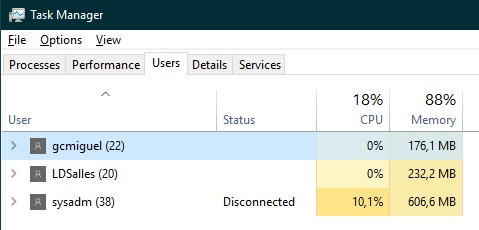
# **Análise Windows**

|  |  |
| --- | --- |
| **Nome Windows** | XPTO |
| **Máquina Virtual** | Sim |
| **Processador** | 12 núcleos |
| **RAM** | 64GB |
| **Sistema de Discos** | Storage |

Uma máquina virtual Windows Server 2019 Datacenter hospeda quatro instâncias do SQL Server.

### ►Sessões de Remote Desktop Consumindo Recursos

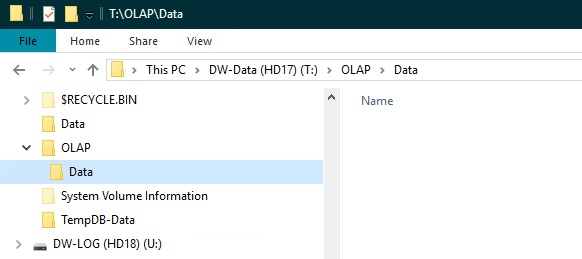
Em todos os dias que acessei o servidor encontrei Sessões de Remote Desktop desconectadas, mas ativas e consumindo recursos do servidor.

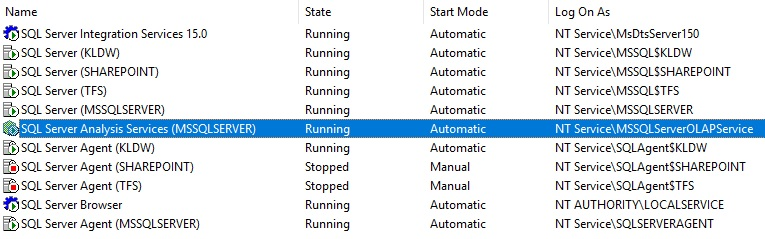


Abrir sessões de Remote Desktop em servidores de produção consome recursos, devem ser restritas a atividades que só podem ser feitas dentro do próprio servidor, e desconectadas logo em seguida. A grande maioria das tarefas de manutenção podem ser feitas de outro computador, instalando a aplicação SQL Server Management Studio.

### ►Serviço Analysis Services Ativo sem uso

O SQL Server Analysis Services é um banco de dados multidimensional, utilizado em projetos de Business Intelligence. Este serviço se encontra ativo no servidor, contudo aparentemente sem uso, pois a pasta destinada a seus bancos está vazia.



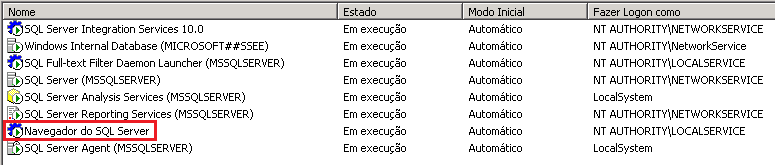


Não consegui verificar com certeza o uso, pois estou sem acesso a este serviço. Recomendo parar o serviço e configurar para inicialização manual, se não estiver em uso realmente.

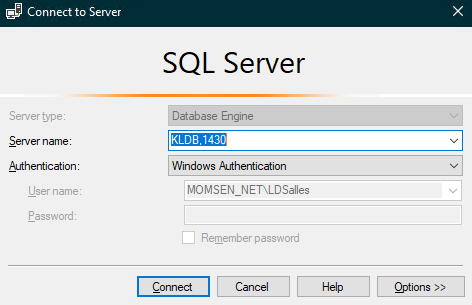
### ►Serviço SQL Browser Ativo

<https://docs.microsoft.com/en-us/sql/tools/configuration-manager/sql-server-browser-service?view=sql-server-ver15>

O serviço SQL Browser é utilizado para acessar instâncias nomeadas, com resolução dinâmica de porta. O problema é que este serviço responde na porta UDP 1434, sendo fonte de brechas de segurança, já que fornece a lista de instâncias do SQL Server juntamente com a porta TCP e o Build. A Microsoft recomenda manter este serviço parado.



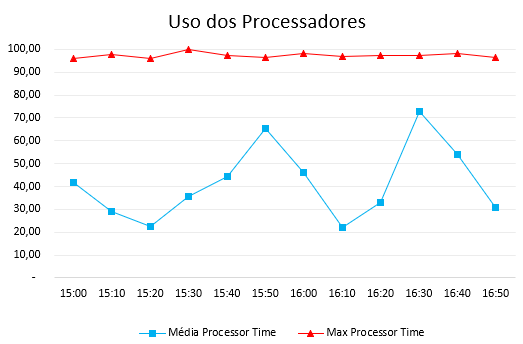
Para acessar as Instâncias Nomeadas deve-se configurar uma porta TCP fixa para as instâncias nomeadas, no acesso informar a porta como no exemplo abaixo:



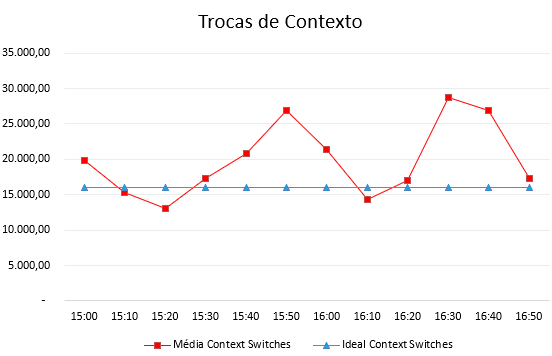
## *Análise de Desempenho*

Foram coletados indicadores de desempenho utilizando o Performance Monitor, segue abaixo análise.

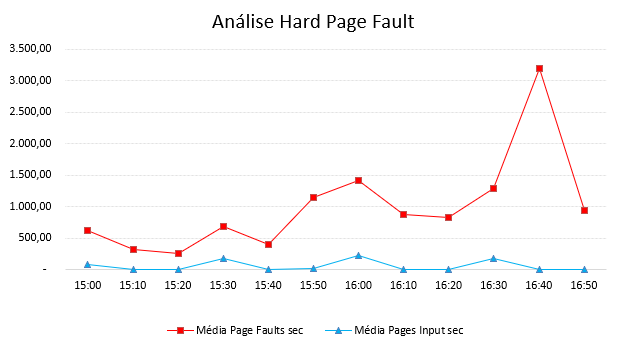
### ►Processador



Os valores do contador ***% Processor Time*** estão no limite da normalidade nos picos de uso (70% de média), com vários períodos de picos em 100%. Trocas de contexto ligeiramente acima do ideal, mostram que algum outro recurso está próximo do gargalo.

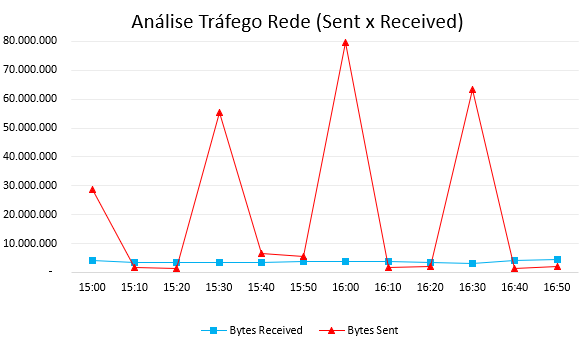


### ►Memória



No gráfico acima observamos períodos de ***Hard Page Fault***, com atividade intensa em ambos os contadores durante quatro períodos. Estes períodos são os mesmos observados nas trocas de contexto, atividade de disco e rede.

### ►Rede



Picos com volume grande de dados saindo do servidor, coincidindo com pequena pressão de memória, disco e CPU. Este comportamento só ocorreu no dia 19, no dia 17 não houve atividade intensa.

## *Instâncias SQL Server*

Foram instaladas quatro instâncias no mesmo Windows hosts, configuradas como abaixo:

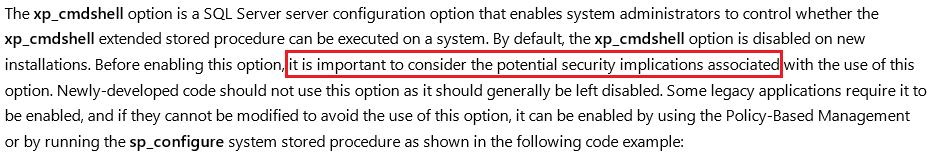
|  |  |  |  |
| --- | --- | --- | --- |
| **Instância** | **Collation** | **CPU** | **MEM RAM** |
| DEFAULT | SQL\_Latin1\_General\_CP1\_CI\_AI | Todos | 40GB |
| KLDW | SQL\_Latin1\_General\_CP1\_CI\_AI | Todos | 5GB |
| SHAREPOINT | Latin1\_General\_CI\_AI | Todos | 3GB |
| TFS | Latin1\_General\_CI\_AS | Todos | 2GB |

Foi feita a distribuição no uso da memória entre as instâncias, contudo todas as instâncias estão configuradas para utilizar todos os processadores. Uma sobrecarga de trabalho de CPU em uma instância pode gerar problema de desempenho em outra instância!

# **Análise Instância DEFAULT**

### ►xp\_cmdshell

<https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/xp-cmdshell-server-configuration-option?view=sql-server-2017>



Este parâmetro deve estar desabilitado por medida de segurança, sempre que possível! Com ele habilitado pode-se ganhar o prompt do servidor com as credenciais de segurança da conta de serviço, sendo um administrador local poderá fazer qualquer coisa!

### ►Configuração dos Bancos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Banco** | **Recovery** | **Versão** | **Collation** | **PageVerify** | **Tamanho MB** |
| AdmRel | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 276 |
| AuditoriaKL | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 6.587 |
| AuditoriaKL-BKP | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 7.078 |
| BeColidencia | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 2.076 |
| BusInt | SIMPLE | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 19.536 |
| BusIntH | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 1.494 |
| ComplianceH | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 154 |
| Corpore | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 14.424 |
| Corpore\_ADV | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 8.594 |
| Corpore\_Manaca | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 3.214 |
| Corpore\_Moarah | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 3.024 |
| Corpore\_Momsen | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 2.384 |
| Corpore\_RH | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 14.424 |
| Corpore\_Suporte | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 14.424 |
| Desenv | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | TORN\_PAGE | 203.261 |
| DOCS | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 6.604 |
| DOCSHML | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 6.013 |
| Estrutura | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9.681 |
| Externo | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | TORN\_PAGE | 361.261 |
| Financeiro | FULL | SQL2008 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 1.167 |
| FunnelH | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 16 |
| HangFire | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 76 |
| HangFireH | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 431 |
| Kasznar | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 36.692 |
| KasznarH | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 36.434 |
| KryptExp1 | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 76 |
| Krypton | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9 |
| KryptonH | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9 |
| KryptonModelo | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9 |
| KryptonTemp | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 2.854 |
| LeituraRPI | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 7.628 |
| LimpezaExterno | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 34.554 |
| Manaca | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 12 |
| MyKasznar | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 1.677 |
| MyKasznarD | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 12.248 |
| MyKasznarH | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | NONE | 38.618 |
| PontoSecullum4 | SIMPLE | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | TORN\_PAGE | 324 |
| Prototipo | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9 |
| ReportServer | FULL | SQL2012 | Latin1\_General\_CI\_AS\_KS\_WS | CHECKSUM | 286 |
| ReportServerTempDB | SIMPLE | SQL2012 | Latin1\_General\_CI\_AS\_KS\_WS | CHECKSUM | 1.489 |
| RPI | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 11.844 |
| RPIH | FULL | SQL2019 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 10.507 |
| SecullumAcessoNet | SIMPLE | SQL2012 | Latin1\_General\_CI\_AS | TORN\_PAGE | 7 |
| SiteIB | FULL | SQL2008 | Latin1\_General\_CI\_AS | CHECKSUM | 9 |
| SiteIBH | FULL | SQL2008 | Latin1\_General\_CI\_AS | CHECKSUM | 8 |
| SiteInstitucional | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 177 |
| SiteInstitucionalV2 | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 323 |
| SiteInstitucionalv2H | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 182 |
| SiteSztartupDesk | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9 |
| SiteSztartupDeskH | FULL | SQL2012 | SQL\_Latin1\_General\_CP1\_CI\_AI | CHECKSUM | 9 |
| TAF | FULL | SQL2008 | Latin1\_General\_BIN | CHECKSUM | 6.174 |
| TSS | FULL | SQL2008 | Latin1\_General\_BIN | CHECKSUM | 6.628 |

A tabela acima mostra as propriedades dos bancos da instância DEFAULT, segue abaixo observações.

### ►Collation

<https://blogs.msdn.microsoft.com/sql_pfe_blog/2013/02/04/sql-collation-and-related-performance-impact-viewing-collation-in-query-plans/>

Apesar do SQL Server aceitar bancos de dados com Collation diferente da instância (definida na instalação), isso não é uma boa prática. O SQL Server utiliza com frequência o banco de dados de sistema TempDB para executar operações temporárias e também armazena tabelas temporárias, estando com Collation diferente dos outros bancos, pode gerar problemas de desempenho e integridade dos dados.

Outro problema comum é a necessidade de executar JOINS entre tabelas em bancos com Collations diferentes acarretando em erros. Para evitar estes erros de execução, o desenvolvedor determina uma Collation de conversão na consulta, prejudicando o desempenho.

### ►Compatibilidade dos Bancos

<https://www.sqlmvp.org/sql-server-database-compatibility-level-performance/>

Alguns bancos estão com compatibilidade anteriores ao SQL Server 2019, esta prática leva a perda de desempenho, deve-se verificar a possibilidade de alterar para mesma versão da instância SQL Server 2019.

### ►Page Verify

https://dba-presents.com/index.php/databases/sql-server/22-checksum-vs-torn-page-detection-performance

A propriedade de banco de dados ***Page Verify*** configura bits de validade nas páginas de dados, tornando possível a detecção de páginas corrompidas. Vários bancos estão com ***Page Verify*** ajustado para “TORN\_PAGE\_DETECTION”, método antigo mantido por compatibilidade as versões antigas. Além disso, alguns bancos de dados estão SEM ***Page Verify***, o que representa grande perigo de perda de integridade.

É recomendado utilizar “CHECKSUM“ pois além de proporcionar uma melhor detecção de problemas, tem impacto menor no desempenho.

### ►JOB para Limpar Todas as Linhas de Tabelas

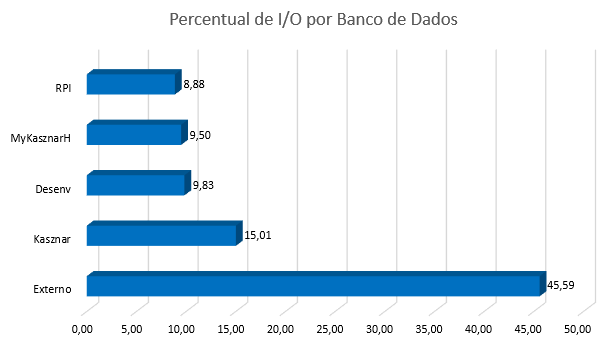
Na pesquisa por JOBs de Backup e Manutenção dos Bancos de Dados, encontrei um JOB para apagar todas as linhas de algumas tabelas (JOB “Delete Temporários”). Este JOB utiliza o comando DELETE que é mais lento e gera muitas entradas no Transaction Log, recomendo alterar para o comando TRUNCATE TABLE.

### ► Ausência de Rotina de Desfragmentação

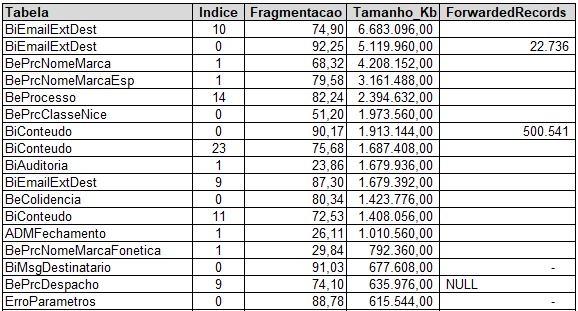
<https://www.mssqltips.com/sqlservertip/4331/sql-server-index-fragmentation-overview/>

Não encontrei JOB com rotina de ***Rebuild*** ou ***Reorganize*** dos índices.

Analisei a fragmentação no principal Banco de Dados da instância em volume de IO “Externo”.



Detectada fragmentação intensa de tabelas e índices do Banco de Dados “Externo”, com consequente impacto negativo no desempenho das consultas.



Recomendo agendar REBUILD ou REORGANIZE ao menos uma vez por semana, para proporcionar melhor desempenho reduzindo a fragmentação nos Bancos de Dados.

### ► Teste de Integridade

<https://social.technet.microsoft.com/wiki/contents/articles/54312.sql-server-database-integrity-check-best-practices.aspx>

Não foi identificada execução periódica de teste de integridade dos bancos (DBCC CHECKDB), nem alerta configurado para páginas corrompidas.

Verificar a integridade é importante para detecção de Banco corrompido, possibilitando sua recuperação imediata a partir de Backup Integro! Sem um processo de detecção de perda de integridade, o banco de dados pode trabalhar vários dias até meses com parte das suas páginas corrompidas!

A escolha da frequência de verificação de integridade deve ser determinada em conjunto com o tempo de retenção dos backups.

### ► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de.html>

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de_30.html>

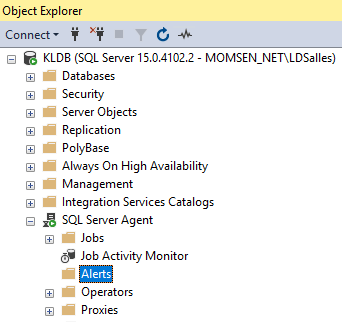
As estatísticas de banco de dados são fundamentais para performance das consultas, auxiliando na escolha de planos de execução eficientes. Estatísticas devem ser atualizadas constantemente, apesar da propriedade “Auto Update Statiscts” estar habilitada, o SQL Server demora para promover sua atualização.

Ideal é manter rotina periódica de atualização das estatísticas utilizando o comando, para cada banco de dados:

***EXEC updatestats***

### ► Administração Proativa do Servidor

O servidor não possui alerta configurado, nem envio de e-mails! Os alertas notificam o Administrador quando ocorre um problema, ou na eminência de um, proporcionando administração proativa.

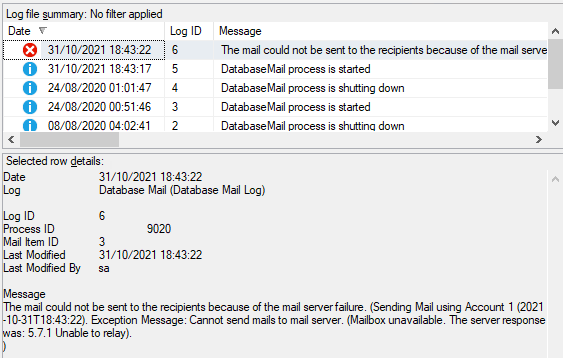


Recomendo configuração de envio de e-mails e criação de alertas para as principais situações críticas, por exemplo:

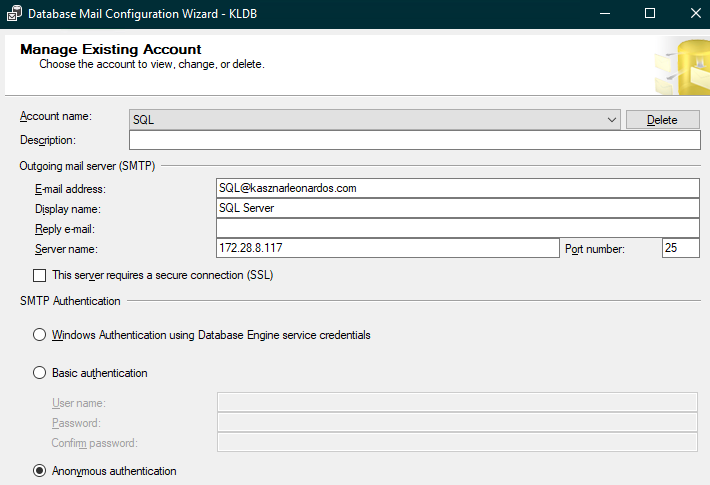
* Página corrompida.
* Arquivo de log próximo da capacidade máxima.
* Arquivo de log chegou ao máximo da capacidade, com impacto direto na produção.
* TempDB chegou na capacidade máxima de armazenamento, com impacto direto na produção.
* Erros classificados com severidade crítica, códigos de 20 a 25.

### ► Configuração para Envio de E-mails

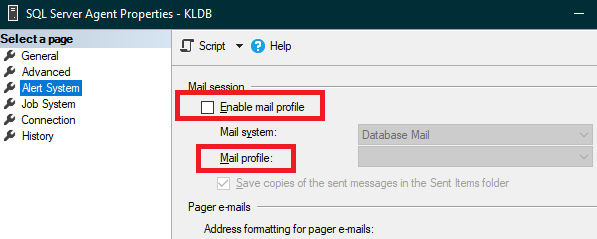
Foi configurado envio de e-mail no SQL Server, contudo ao fazer um teste de envio ocorreu um erro de conexão com o SMTP.



Configuração atual:



Outro problema de configuração detectado está no SQL Server Agent, não está habilitado envio de e-mails.



### ► Propriedade Auto Shrink Habilitada

<https://support.microsoft.com/en-us/topic/recommendations-and-guidelines-for-setting-the-auto-shrink-database-option-in-sql-server-827cfcfb-b38c-0364-0ae4-ae551ce4af7e>

A propriedade **Auto Shrink** está habilitada para os bancos listados abaixo:



Com esta propriedade habilitada o SQL Server executa o comando DBCC SHRINK\_DATABASE a cada 30 minutos, gerando alta carga de trabalho de IO e fragmentação nos arquivos de Dados e Log dos Bancos de Dados.

### ► Segurança Login “SA”

<https://www.mssqltips.com/sqlservertip/3695/best-practices-to-secure-the-sql-server-sa-account/>

O Login “SA” existe em toda instância do SQL Server com privilégio SYSADMIN, isto é, pode fazer qualquer coisa! Como todos já sabem o nome do Login e seu alto privilégio, a única barreira para o acesso indevido é a senha.

Para reduzir esta exposição, recomenda-se renomear o Login “SA”, pois um invasor agora terá que descobrir qual Login tem privilégio total e em seguida quebrar a senha.

### ► Rotina de Backup e Manutenção

Como a rotina de backup é feita por aplicação externa, analisei a partir dos registros em tabelas de sistema quanto a ocorrência de backups.

A rotina de Backup utiliza FULL diário as 21:30 e LOG a cada 30 minutos, para os Bancos de Dados com Recovery Model FULL.

Não existe necessidade de fazer backup do Banco de Dados de Sistema MODEL, pois este é apenas um modelo de criação para novos Bancos de Dados.

Existem muitos bancos no Recovery Model SIMPLE na instância DEFAULT, sendo bancos de produção, pode representar falha na segurança dos dados, já que inviabiliza Backup do Log. A ausência de Backups do LOG, inviabiliza a recuperação de páginas corrompidas utilizando Restore, além da perda de todos os dados após o último Backup feito até o momento da falha.

O período de retenção dos Backups FULL e LOG deve estar relacionado com a frequência de execução de testes de integridade. Por exemplo, período de retenção dos Backups de duas semanas, deve-se executar testes de integridade toda semana.

### ► Ausência de Solução de Alta Disponibilidade

Atualmente o servidor não possui solução de Alta Disponibilidade habilitada no SQL Server. Segue abaixo uma breve descrição das soluções disponíveis no SQL Server.

**StandBy Server / Log Shipping**



É a solução mais antiga, disponível desde as primeiras versões do SQL Server. São instalados dois servidores SQL Server idênticos, sendo um de produção e o outro StandBy. A sincronia entre os servidores é feita através de Backup / Restore do LOG, tornando a solução de fácil implantação.

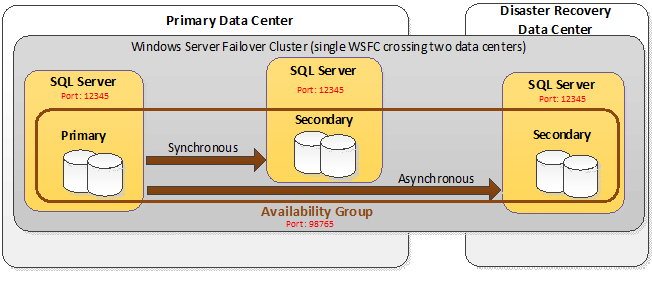
*Vantagens:*

* Solução barata, bastam apenas dois servidores com a mesma configuração.
* Contingência entre sites, pois os servidores podem sincronizar através de links, podendo até estar em cidades diferentes.

*Desvantagens:*

* No momento da falha do servidor de produção, um operador deverá executar procedimento de substituição. Não possui Failover automático!
* Pode ocorrer perda de dados, devido ao intervalo entre os Backups.

**AlwaysOn**



Sincronia feita serviço a serviço através de porta TCP/IP de comunicação, proporcionando latência muito menor que a solução de LogShipping. Existem dois modos de sincronia:

***Modo Asynchronous-commit***

* Atualização assíncrona entre os servidores.
* Sem impacto no desempenho da produção.
* Possibilidade de pequenas perdas de dados no caso de contingência.
* Disponível apenas na ***Edição Enterprise***.

***Modo Synchronous-commit***

* Atualização síncrona entre os servidores.
* Pode ocorrer impacto no desempenho na produção em redes lentas.
* Garantia de 100% de não ocorrer perda de dados na contingência.

*Vantagens:*

* Solução barata, bastam apenas dois servidores com a mesma configuração.
* Contingência entre sites, pois os servidores podem sincronizar através de links, podendo até estar em cidades diferentes, no caso da **Edição Enterprise**.
* No caso de falha do servidor principal o secundário se torna ativo automaticamente.
* Pouca ou nenhuma perda de dados no momento da falha.

*Desvantagens:*

* Na Edição Standard deve ser considerado com cuidado, devido a restrição da sincronia síncrona, com impacto no desempenho da produção em conexões lentas.
* Problemas de sincronia devem ser resolvidos rapidamente, pois as transações são acumuladas no arquivo de Log na produção.

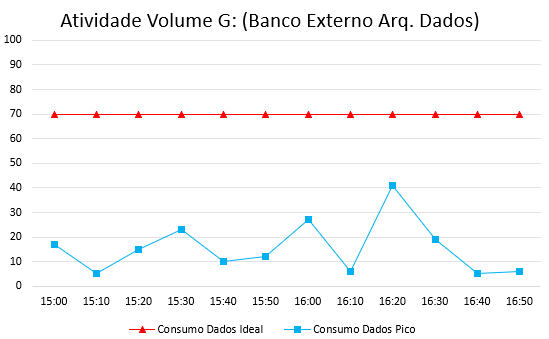
## *Análise de Desempenho*

### ►Disco



Existem diversos Bancos de Dados na Instância DEFAULT, concentrei a análise nos volumes utilizados pelo principal Banco de Dados ***Externo*** e ***TempDB***.

**Volumes Banco de Dados Externo**

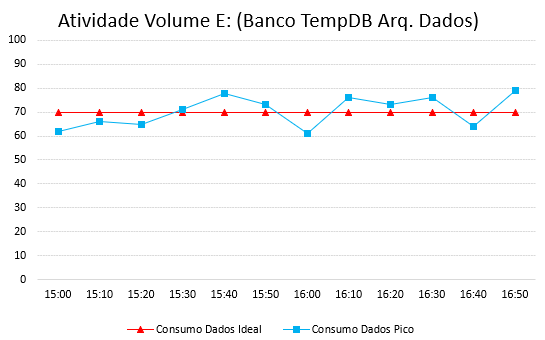


Atividade alta nos arquivos de dados do Banco Externo, nos mesmos períodos onde foram encontrados pequenos gargalos de memória e CPU, além de grande atividade de rede.

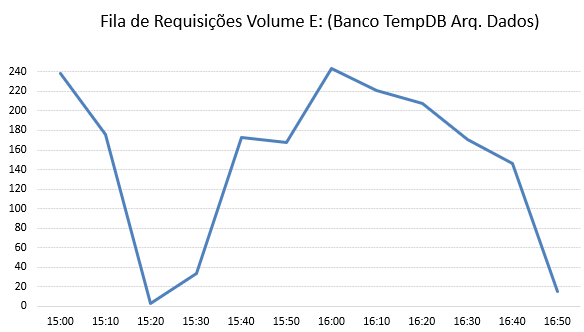
No gráfico seguinte observamos grande aumento de atividade principalmente de escrita, nos mesmos períodos observados acima.

Volume com o arquivo de Log não apresentou gargalo.

**Volumes Banco de Dados TempDB**



O volume E: que hospeda os arquivos de Dados do Banco TempDB apresenta pequeno gargalo, apresentando fila de requisições em grande parte do período da tarde.



Esta pressão no sistema de discos pode estar sendo gerada por atividades inadequadas no SQL Server, que será avaliada mais a frente.

### ►SQL Server

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
|  |  |

A instância XPTO recebe volume médio de requisições, com atividade similar em ambos os dias analisados.

**Memória do SQL Server**

A instância XPTO está utilizando toda memória destinada a ela 40GB. Foram observados vários períodos onde estes 40GB não estão sendo suficientes para atender as requisições com desempenho.

|  |  |
| --- | --- |
|  |  |

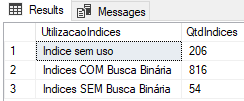
O contador ***Page Life Expectancy*** mostra claramente que a instância XPTO necessita de mais memória, ou tuning de consultas para reduzir o consumo de memória. Os valores encontrados em diversos períodos apresentam um baixo tempo de permanência das páginas no Cache, bem abaixo do valor mínimo definido pela Microsoft em vermelho no gráfico: ***( Buffer pool memory in GB / 4 ) x 300***.

**Uso dos Índices**

|  |  |
| --- | --- |
|  |  |

Analisando a proporção dos contadores ***Full Scans*** e ***Index Searches***, a instância XPTO apresenta baixa quantidade de buscas binárias em índices. Um trabalho de Tuning dos índices é necessário para melhorar o desempenho das consultas e reduzir a pressão de memória.

Muitos índices criados no Banco de Dados **Externo** estão sem uso ou sem serem utilizados em buscas binárias, compreendendo 24% dos índices Nonclustered criados.



**Forwarded Records**

<https://www.mssqltips.com/sqlservertip/2984/identify-and-correct-sql-server-forwarded-records/>

|  |  |
| --- | --- |
|  |  |

Em tabelas sem índice Clustered a ocorrência de **Forwarded Records** aumenta o volume de IO nas consultas, prejudicando o desempenho. Para resolver este problema, só fazendo REBUILD da tabela e todos os seus índices Nonclustered.

**Page Splits**

<https://dbtut.com/index.php/2020/09/03/index-fragmentation-and-page-splits-in-sql-server/>

|  |  |
| --- | --- |
|  |  |

A instância apresentou volume de ***Page Split*** acima do ideal no dia 19/10, sendo necessário compensar com frequente reorganização dos índices para evitar fragmentação.

**Locks**

|  |  |
| --- | --- |
|  |  |

Foram encontradas evidências de **Blocking** importante em ambos os dias monitorados, com muitas ocorrências por segundo no período.

Não houve ocorrências de **Deadlocks** no período monitorado.

# **Análise Instância KLDW**

### ►Configuração dos Bancos



A instância KLDW possui apenas dois Bancos de Dados um Data Warehouse “kldw” e o catálogo do SQL Server Integration Services “SSISDB”.

► **Ausência de Rotina de Desfragmentação**

<https://www.mssqltips.com/sqlservertip/4331/sql-server-index-fragmentation-overview/>

Não encontrei JOB com rotina de ***Rebuild*** ou ***Reorganize*** dos índices.

O Banco de Dados “kldw” não possui índices Nonclustered e poucos índices Clustered, com isso a rotina de manutenção só pode ser feita em modo OffLine, com REBUILD das tabelas.

Recomendo agendar REBUILD das Tabelas ou criar índices Clustered para facilitar a manutenção.

### ► Teste de Integridade

<https://social.technet.microsoft.com/wiki/contents/articles/54312.sql-server-database-integrity-check-best-practices.aspx>

Não foi identificada execução periódica de teste de integridade dos bancos (DBCC CHECKDB), nem alerta configurado para páginas corrompidas.

Verificar a integridade é importante para detecção de Banco corrompido, possibilitando sua recuperação imediata a partir de Backup Integro! Sem um processo de detecção de perda de integridade, o banco de dados pode trabalhar vários dias até meses com parte das suas páginas corrompidas!

A escolha da frequência de verificação de integridade deve ser determinada em conjunto com o tempo de retenção dos backups.

### ► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de.html>

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de_30.html>

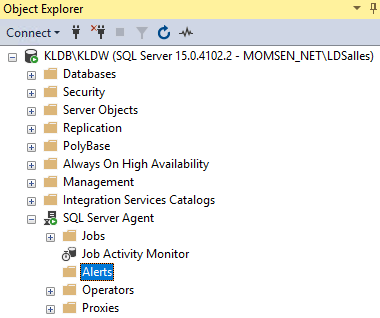
As estatísticas de banco de dados são fundamentais para performance das consultas, auxiliando na escolha de planos de execução eficientes. Estatísticas devem ser atualizadas constantemente, apesar da propriedade “Auto Update Statiscts” estar habilitada, o SQL Server demora para promover sua atualização.

Ideal é manter rotina periódica de atualização das estatísticas utilizando o comando, para cada banco de dados:

***EXEC updatestats***

### ► Administração Proativa do Servidor

O servidor não possui alerta configurado, nem envio de e-mails! Os alertas notificam o Administrador quando ocorre um problema, ou na eminência de um, proporcionando administração proativa.



Recomendo configuração de envio de e-mails e criação de alertas para as principais situações críticas, por exemplo:

* Página corrompida.
* Arquivo de log próximo da capacidade máxima.
* Arquivo de log chegou ao máximo da capacidade, com impacto direto na produção.
* TempDB chegou na capacidade máxima de armazenamento, com impacto direto na produção.
* Erros classificados com severidade crítica, códigos de 20 a 25.

### ► Configuração para Envio de E-mails

Envio de e-mail não está configurado na instância KLDW, recomendado configurar para receber alertas de condições críticas, erros de execução de JOBs, etc.

### ► Segurança Login “SA”

<https://www.mssqltips.com/sqlservertip/3695/best-practices-to-secure-the-sql-server-sa-account/>

O Login “SA” existe em toda instância do SQL Server com privilégio SYSADMIN, isto é, pode fazer qualquer coisa! Como todos já sabem o nome do Login e seu alto privilégio, a única barreira para o acesso indevido é a senha.

Para reduzir esta exposição, recomenda-se renomear o Login “SA”, pois um invasor agora terá que descobrir qual Login tem privilégio total e em seguida quebrar a senha.

### ► Rotina de Backup e Manutenção

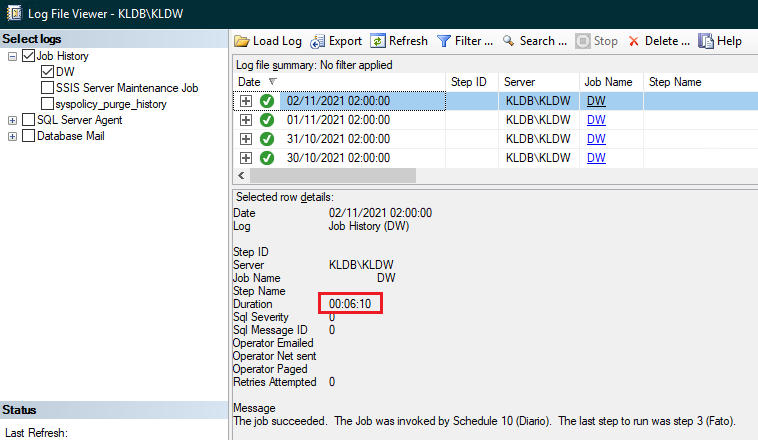
Como a rotina de backup é feita por aplicação externa, analisei a partir dos registros em tabelas de sistema quanto a ocorrência de backups.

A rotina de Backup utiliza FULL diário as 21:30 e LOG a cada 30 minutos, para os Bancos de Dados com Recovery Model FULL.

Não existe necessidade de fazer backup do Banco de Dados de Sistema MODEL, pois este é apenas um modelo de criação para novos Bancos de Dados.

### ► Pacotes de Carga Integration Services (SSIS)

Hoje não existe problema de desempenho no processo de carga do DW, pois o tempo de execução médio do JOB de carga não chega a 10 minutos, e sua execução ocorre uma vez por dia as 2h da madrugada.

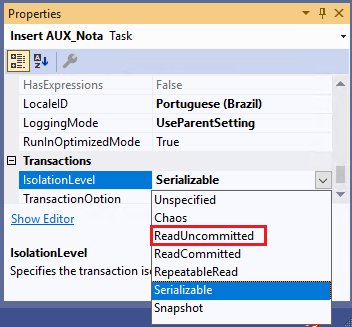


Contudo, seguindo solicitação dos gestores da Empresa XPTO, analisei os pacotes SSIS de carga, seguem algumas observações.

**Problema de Concorrência**

As consultas implementadas dentro dos pacotes SSIS podem gerar problema de concorrência, se forem executadas durante o horário comercial, ou se rodar ao mesmo tempo de outra rotina.

Para evitar problemas de concorrência e para não gerar impacto no sistema origem, é recomendado trocar o nível de isolamento do Data Flow ou utilizar Hint NOLOCK nas consultas.



**Uso Inadequado do SSIS**

Toda carga está sendo feita através de comando T-SQL INSERT...SELECT e UPDATE...JOIN, com isso não está sendo utilizado todo o potencial do SSIS!

O SSIS possui uma série de transformações no Data Flow para tornar a carga leve e eficiente, consumindo pouco recurso e gerando o mínimo de carga de trabalho no sistema origem. Do jeito que foi feita a carga, crescendo o volume de dados no DW, não será possível fazer sincronias com mais frequência, restringindo apenas a cargas noturnas.

**Tabelas Stage no DW**

A carga utiliza tabelas Stage (prefixo “aux\_”) dentro do próprio Banco de Dados do DW, sendo utilizadas apenas como repositório temporário para execução da carga em duas etapas (Origem – Stage – DW).

O problema é que a carga deixa estas tabelas repletas de linhas, crescendo o tamanho do Backup do DW sem necessidade. Recomenda-se criar estas tabelas Stage em outro Banco de Dados, removendo este banco da rotina de Backup, já que se trata de informações temporárias, utilizadas apenas durante o processo de carga.

## *Análise de Desempenho*

### ►SQL Server

A instância KLDW recebeu volume baixíssimo de carga de trabalho durante o período monitorado, apenas 6 conexões em média.

**Memória do SQL Server**

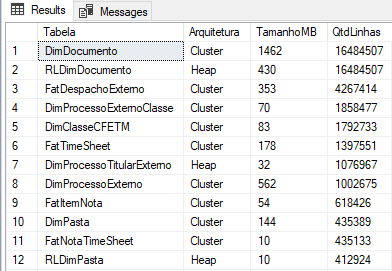
Devido a baixíssima atividade, não foi possível analisar os contadores de memória.

**Uso dos Índices**

Devido a baixíssima atividade, não foi possível analisar os contadores de índices.

Não foram criados índices Nonclustered no Banco “kldw”, hoje as tabelas estão pequenas (com exceção da DimDocumento) então o prejuízo no desempenho é baixo.

***Maiores Tabelas Banco “kldw”***

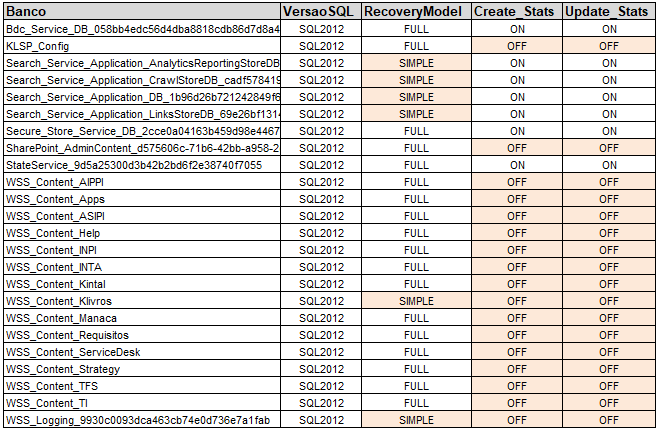


Contudo, se o DW crescer os problemas de desempenho vão aparecer cada vez mais intenso, se não forem criados índices Nonclustered e Columnstore.

# **Análise Instância SHAREPOINT**

A instância SHAREPOINT está configurada para utilizar todos os processadores e 3GB de memória RAM, SQL Server Agent parado e envio de e-mail não está configurado.

### ► Configuração dos Bancos



A instância SHAREPOINT possui apenas Bancos de Dados que atendem a uma instalação de SharePoint.

### ► Compatibilidade dos Bancos

<https://www.sqlmvp.org/sql-server-database-compatibility-level-performance/>

Todos os Bancos desta instância estão com compatibilidade anterior ao SQL Server 2019, esta prática leva a perda de desempenho. Deve-se verificar a possibilidade de alterar para mesma versão da instância SQL Server 2019.

### ► Propriedade de Manutenção e Criação de Estatísticas

<https://www.mssqltips.com/sqlservertip/2766/sql-server-auto-update-and-auto-create-statistics-options/>

Ambas as propriedades relacionadas as estatísticas de banco de dados são importantes para bom desempenho das consultas. Recomendo habilitar “Auto\_Create\_Stats” e “Auto\_Update\_Stats” nos bancos de dados que estão desabilitados.

### ► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de.html>

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de_30.html>

As estatísticas de banco de dados são fundamentais para performance das consultas, auxiliando na escolha de planos de execução eficientes. Estatísticas devem ser atualizadas constantemente, apesar da propriedade “Auto Update Statiscts” estar habilitada, o SQL Server demora para promover sua atualização.

Ideal é manter rotina periódica de atualização das estatísticas utilizando o comando, para cada banco de dados:

***EXEC updatestats***

### ► Ausência de Rotina de Desfragmentação

<https://www.mssqltips.com/sqlservertip/4331/sql-server-index-fragmentation-overview/>

Não encontrei JOB com rotina de ***Rebuild*** ou ***Reorganize*** dos índices.

Apesar dos Bancos de Dados desta instância serem pequenos, recomenda-se agendar REBUILD ou REORGANIZE ao menos uma vez por semana.

### ► Teste de Integridade

<https://social.technet.microsoft.com/wiki/contents/articles/54312.sql-server-database-integrity-check-best-practices.aspx>

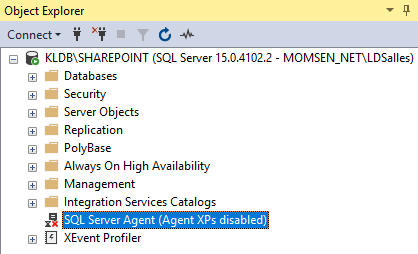
Não foi identificada execução periódica de teste de integridade dos bancos (DBCC CHECKDB), nem alerta configurado para páginas corrompidas.

Verificar a integridade é importante para detecção de Banco corrompido, possibilitando sua recuperação imediata a partir de Backup Integro! Sem um processo de detecção de perda de integridade, o banco de dados pode trabalhar vários dias até meses com parte das suas páginas corrompidas!

A escolha da frequência de verificação de integridade deve ser determinada em conjunto com o tempo de retenção dos backups.

### ► Administração Proativa do Servidor

A instância está com o SQL Server Agent desabilitado, não possui alerta configurado, nem envio de e-mails! Os alertas notificam o Administrador quando ocorre um problema, ou na eminência de um, proporcionando administração proativa.



Recomendo habilitar o SQL Server Agent, configurar envio de e-mails e criação de alertas para as principais situações críticas, por exemplo:

* Página corrompida.
* Arquivo de log próximo da capacidade máxima.
* Arquivo de log chegou ao máximo da capacidade, com impacto direto na produção.
* TempDB chegou na capacidade máxima de armazenamento, com impacto direto na produção.
* Erros classificados com severidade crítica, códigos de 20 a 25.

### ► Configuração para Envio de E-mails

Envio de e-mail não está configurado na instância KLDW, recomendado configurar para receber alertas de condições críticas, erros de execução de JOBs, etc.

### ► Segurança Login “SA”

<https://www.mssqltips.com/sqlservertip/3695/best-practices-to-secure-the-sql-server-sa-account/>

O Login “SA” existe em toda instância do SQL Server com privilégio SYSADMIN, isto é, pode fazer qualquer coisa! Como todos já sabem o nome do Login e seu alto privilégio, a única barreira para o acesso indevido é a senha.

Para reduzir esta exposição, recomenda-se renomear o Login “SA”, pois um invasor agora terá que descobrir qual Login tem privilégio total e em seguida quebrar a senha.

### ► Rotina de Backup e Manutenção

Como a rotina de backup é feita por aplicação externa, analisei a partir dos registros em tabelas de sistema quanto a ocorrência de backups.

A rotina de Backup utiliza FULL diário as 21:30 e LOG a cada 30 minutos, para os Bancos de Dados com Recovery Model FULL. Alguns Bancos estão configurados com o Recovery Model SIMPLE, o que inviabiliza Backups do Log e recuperação de dados após o último Backup além de recuperação de banco corrompido!

Não existe necessidade de fazer backup do Banco de Dados de Sistema MODEL, pois este é apenas um modelo de criação para novos Bancos de Dados.

## *Análise de Desempenho*

### ►SQL Server

A instância SHAREPOINT recebeu volume médio de 789 conexões, mas um volume de requisições baixo 26 execuções em média.

**Memória do SQL Server**

Devido a baixíssima atividade, não foi possível analisar os contadores de memória. Apresentou Page Life Expectancy altíssimo, devido a baixa atividade.

# **Análise Instância TFS**

A instância TFS está configurada para utilizar todos os processadores e 2GB de memória RAM, SQL Server Agent parado e envio de e-mail não está configurado.

### ►Configuração dos Bancos



A instância TFS possui apenas três Bancos de Dados pequenos e pouca atividade.

### ► Compatibilidade dos Bancos

<https://www.sqlmvp.org/sql-server-database-compatibility-level-performance/>

Todos os Bancos desta instância estão com compatibilidade anterior ao SQL Server 2019, esta prática leva a perda de desempenho. Deve-se verificar a possibilidade de alterar para mesma versão da instância SQL Server 2019.

► **Ausência de Rotina de Desfragmentação**

<https://www.mssqltips.com/sqlservertip/4331/sql-server-index-fragmentation-overview/>

Não encontrei JOB com rotina de ***Rebuild*** ou ***Reorganize*** dos índices.

Apesar dos Bancos de Dados desta instância serem pequenos, recomenda-se agendar REBUILD ou REORGANIZE ao menos uma vez por semana.

### ► Teste de Integridade

<https://social.technet.microsoft.com/wiki/contents/articles/54312.sql-server-database-integrity-check-best-practices.aspx>

Não foi identificada execução periódica de teste de integridade dos bancos (DBCC CHECKDB), nem alerta configurado para páginas corrompidas.

Verificar a integridade é importante para detecção de Banco corrompido, possibilitando sua recuperação imediata a partir de Backup Integro! Sem um processo de detecção de perda de integridade, o banco de dados pode trabalhar vários dias até meses com parte das suas páginas corrompidas!

A escolha da frequência de verificação de integridade deve ser determinada em conjunto com o tempo de retenção dos backups.

### ► Ausência de Rotina para Atualização das Estatísticas de Banco de Dados

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de.html>

<http://sqlserver-brasil.blogspot.com/2008/03/atualizao-das-estatsticas-de-banco-de_30.html>

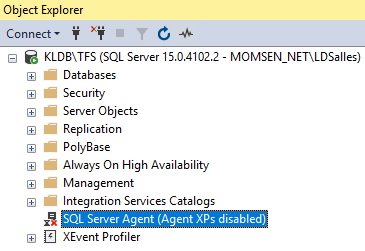
As estatísticas de banco de dados são fundamentais para performance das consultas, auxiliando na escolha de planos de execução eficientes. Estatísticas devem ser atualizadas constantemente, apesar da propriedade “Auto Update Statiscts” estar habilitada, o SQL Server demora para promover sua atualização.

Ideal é manter rotina periódica de atualização das estatísticas utilizando o comando, para cada banco de dados:

***EXEC updatestats***

### ► Administração Proativa do Servidor

A instância está com o SQL Server Agent desabilitado, não possui alerta configurado, nem envio de e-mails! Os alertas notificam o Administrador quando ocorre um problema, ou na eminência de um, proporcionando administração proativa.



Recomendo habilitar o SQL Server Agent, configurar envio de e-mails e criação de alertas para as principais situações críticas, por exemplo:

* Página corrompida.
* Arquivo de log próximo da capacidade máxima.
* Arquivo de log chegou ao máximo da capacidade, com impacto direto na produção.
* TempDB chegou na capacidade máxima de armazenamento, com impacto direto na produção.
* Erros classificados com severidade crítica, códigos de 20 a 25.

### ► Configuração para Envio de E-mails

Envio de e-mail não está configurado na instância KLDW, recomendado configurar para receber alertas de condições críticas, erros de execução de JOBs, etc.

### ► Segurança Login “SA”

<https://www.mssqltips.com/sqlservertip/3695/best-practices-to-secure-the-sql-server-sa-account/>

O Login “SA” existe em toda instância do SQL Server com privilégio SYSADMIN, isto é, pode fazer qualquer coisa! Como todos já sabem o nome do Login e seu alto privilégio, a única barreira para o acesso indevido é a senha.

Para reduzir esta exposição, recomenda-se renomear o Login “SA”, pois um invasor agora terá que descobrir qual Login tem privilégio total e em seguida quebrar a senha.

### ► Rotina de Backup e Manutenção

Como a rotina de backup é feita por aplicação externa, analisei a partir dos registros em tabelas de sistema quanto a ocorrência de backups.

A rotina de Backup utiliza FULL diário as 21:30 e LOG a cada 30 minutos, para os Bancos de Dados com Recovery Model FULL. Alguns Bancos estão configurados com o Recovery Model SIMPLE, o que inviabiliza Backups do Log e recuperação de dados após o último Backup além de recuperação de banco corrompido!

Não existe necessidade de fazer backup do Banco de Dados de Sistema MODEL, pois este é apenas um modelo de criação para novos Bancos de Dados.

## *Análise de Desempenho*

### ►SQL Server

A instância TFS recebeu volume baixíssimo de carga de trabalho durante o período monitorado, apenas 8 conexões em média.

**Memória do SQL Server**

Devido a baixíssima atividade, não foi possível analisar os contadores de memória. Apresentou Page Life Expectancy altíssimo, devido a baixa atividade.

# **Conclusão**

As instâncias SQL Server do servidor XPTO da EMPRESA XPTO, apresenta uma série de problemas de configuração gerando impacto importante no desempenho e segurança dos dados. Os ajustes propostos neste relatório podem melhorar o desempenho e aumentar a proteção e integridade dos dados.

Recomenda-se realizar um projeto para implantar as sugestões, de acordo com a prioridade selecionada pelos gestores da EMPRESA XPTO.

Estou à disposição para quaisquer esclarecimentos que se façam necessários sobre o presente relatório no telefone (xx) xxxxx-xxxx.

Rio de Janeiro, xx de xxxx de 20xxx.

**Landry Duailibe Salles Filho**

Sócio / Diretor