


## FAQs - Suporte técnico

Central de Atendimento TOTVS (<https://centraldeatendimento.totvs.com/hc/pt-br>) > Framework (<https://centraldeatendimento.totvs.com/hc/pt-br/categories/1>)

Pesquisa

Pesquisa 

Administrativo e Financeiro  
(<https://totvscst.zendesk.com/hc/pt-br/categories/115001937708-Administrativo-e-Financeiro>)

Documentações  
(<https://totvscst.zendesk.com/hc/pt-br/categories/115002257328-Documenta%C3%A7%C3%B5es>)

Notícias  
(<https://totvscst.zendesk.com/hc/pt-br/categories/115002435008-Not%C3%ADcias>)


Universidade TOTVS  
(<https://totvscst.zendesk.com/hc/pt-br/categories/360001456512-Universidade-TOTVS>)

# Framework - Linha Protheus - FRAME - Lentidão no Protheus 12

### Atualizado em:

1 de novembro de 2021 07:27

Seguir

 Tempo aproximado para leitura: **00:03:00 min**

### Dúvida

Como identificar o ofensor quando o **Protheus 12** está apresentando lentidão generalizada?

### Ambiente

Framework - Framework (Linha Protheus) - Atualização - A partir da versão 12.

### Solução

Avalie inicialmente as informações das documentações abaixo:

Protheus Hardware Mínimo Recomendável: Hardware Mínimo Recomendável (<https://tdn.totvs.com/pages/viewpage.action?pagelId=563437610>)

Protheus em ambientes virtualizados: Ambientes Virtualizados (<https://tdn.totvs.com/display/PROT/Ambientes+Virtualizados>)

Protheus em provedores de Nuvem: Protheus em Provedor de Nuvem (<https://tdn.totvs.com/display/PROT/Protheus+em+Provedor+de+Nuvem>)

Após avaliação dos *links* acima, para identificar o ofensor da lentidão generalizada do Protheus listamos os seguintes passos:

Caso esteja com a última versão de **LIB**, Binário e **DBAccess**, e todas as manutenções do ambiente sejam realizadas periodicamente, o caso de lentidão deverá ser analisado pontualmente.

#### 1. Atualização:

**1.1.** Verifique se a versão de **LIB**, **Appserver**, **Smartclient**, **DBAccess** e **License Server** é igual a última disponível no Portal do Cliente porque estão sendo implementadas constantemente algumas melhorias que visam otimizar a performance para o ambiente do **Protheus 12**. Algumas das melhorias são:

- Carga do **RPO**
- Abertura de **Thread**
- Busca de usuário

**1.2.** As atualizações do **DBAccess** e **Appserver** devem ser realizadas simultaneamente na maioria dos casos, pois as alterações realizadas na biblioteca **dbapi** do **Appserver** são refletidas no **DBAccess**.

#### 2. Configuração:

##### 2.1. Filtro:

Quando a lentidão é gerada ao tentar filtrar por um campo, por exemplo na

rotina **Contas a Pagar**, ao tentar filtrar um valor que corresponde a um título pago que não possui índice, neste caso a lentidão é gerada devido à ausência de índice. Para solucionar basta criar um índice para o campo que deseja realizar o filtro.

## 2.2. Customizações;

Para verificar se a lentidão é causada por rotina customizada através de ponto de entrada no Sistema, inclua no **INI** do server dentro do ambiente utilizado a linha **IXBLOG=NORUN**. Caso não ocorra a lentidão é necessário que sejam reanalisadas as customizações.

**Ex.**

[Environment]

...

IXBLOG=NORUN

## 2.3. Smartclient:

É recomendado que seja executado o **Smartclient** de forma local porque assim o tráfego de rede é minimizado.

## 2.4. Latência:

A latência de rede é um dos pontos que pode causar lentidão no **Protheus** porque todos os dados trafegados entre o **Smartclient** e **Server** serão afetados. Para verificar se a latência de rede está impactando na performance do Protheus, execute o programa **U\_NETTEST** no período de 1 hora e analise os resultados que estão acima de 100ms. Acima de 100 ms de resposta a lentidão já é notada pelo usuário.

## 2.5. Política de Appserver:

Manter a coesão dos **Appservers** é uma das boas práticas que garante a boa performance do **Protheus**, neste caso sugerimos a distribuição por responsabilidades entre os **Appservers**.

Arquitetura sugerida

- Appserver Master
- Appserver Slaves
- Appserver Jobs - Schedule
- Appserver Web Service - Portal

## 2.6. Consumo de memória:

Caso identifique que o consumo de memória dos serviço do **Protheus** esteja elevado, sugerimos a análise de dois pontos:

1. Quantidade de usuário conectados no **Slave**.

Para este caso deverão ser criados novos **Slaves** de acordo com o recurso computacional disponível porque assim o volume de conexão entre os **Slaves** será balanceado e consequentemente o consumo de memória será diminuído neles.

2. Consumo por Programa.

Quando for alto consumo por programa, deverá identificar e isolar o programa com alto consumo de memória, em seguida gerar os logs e encaminhar para equipe específica analisar.

DebugThreadUsedMemory: DebugThreadUsedMemory

(<http://tdn.totvs.com/display/tec/DebugThreadUsedMemory>)

(<http://tdn.totvs.com/display/tec/DebugThreadUsedMemory>)ServerMemoryI

nfo: ServerMemoryInfo (<http://tdn.totvs.com/display/tec/ServerMemoryInfo>)

ConsoleMaxSize: (<http://tdn.totvs.com/display/tec/ServerMemoryInfo>)ConsoleMaxSize:

ConsoleMaxSize (<http://tdn.totvs.com/display/tec/ConsoleMaxSize>++29343)

## 2.7. Numeração Automática:

Para uma melhor performance do ambiente é recomendado o uso do controle de numeração pelo **License Server**.

Verifique se no appserver.ini do **License Server** está configurado o parâmetro **EnableNumber=1**, que realiza o controle de numeração automática diretamente pelo **License Server Virtual**. Porque quando o controle é realizado através das tabelas **SXE/SXF** é aumentada a concorrência no gerenciamento do Ctree e causa impacto de performance significativo no **Protheus**.

**2.8. ShowStatus:**

Verifique se no **appserver.ini** do **License Server** está configurado o parâmetro **ShowStatus** igual a 0, desabilitado, com o objetivo de não exibir todas as informações de licenças de usuário. Quando o parâmetro **ShowStatus** está com o valor igual a 1, ativado, é gravado um grande volume de informações no **console.log** e gera impacto no desempenho do Sistema.

**2.9. TracelIndex:**

Verifique se no **dbaccess.ini** o parâmetro **TracelIndex** está com o valor de 0, desabilitado, este parâmetro deverá ser habilitado somente quando estiver realizando o monitoramento da ferramenta. ( **TracelIndex=0** )

**2.10. Monitor de Índices:**

Verifique se o Monitor de Índices do DBAccess está desligado porque quando este recurso está ligado impacta diretamente na performance do Protheus 12. É um recurso em especial para as bases legadas.

**2.11. Arquivos da System:**

Verifique se a quantidade de arquivos no diretório **SYSTEM** localizado na **PROTHEUS\_DATA** é inferior à 10.000. Porque a partir do momento em que o diretório **SYSTEM** armazena mais de 10.000 arquivos, o **Protheus** começa a perder desempenho durante a sua utilização. Os arquivos que podem ser deletados da pasta **SYSTEM** são: **\*.tmp**, **sc\*.log**, **sc\*.dtc**, **sc\*.cdx** e arquivos **sc\*** sem extensão.

**2.12. Auditoria de Tabelas:**

Quanto menor a abrangência de entidades que se deseja auditar, tabelas e campos, e quanto menos operações desejadas, incluir, alterar ou excluir, menor será o impacto sobre o desempenho do sistema após a aplicação do **Embedded Audit Trail**. Uma análise cuidadosa do que é necessário auditar resultará em um desempenho melhor do produto.

Não recomendamos a auditoria de todos os campos para operação de inclusão em tabelas de movimentos, principalmente das tabelas que possuem grande quantidade de campos. Como a operação de inclusão registra todos os campos sujeitos à auditoria, o impacto na performance pode ser significativo.

**2.13. Opções de Energia:**

Verifique se a configuração de energia do servidor está selecionada a opção **Alto Desempenho (High Performance)**. Identificamos que com a configuração de energia **Equilibrado** é apresentado um tempo maior para abertura de **Thread** e leitura em disco.

**2.14. Indexação Automática do FileSystem:**

Verifique se o recurso de Indexação Automática de disco está desabilitado na partição do disco onde a aplicação foi instalada. Quando este recurso está habilitado o mesmo acaba degradando a performance do **Protheus**.

**2.15. Escaneamento do Antivírus na pasta Protheus\_Data - Totvs:**

Verifique se está desabilitado o **Scan** em tempo real do antivírus no diretório **PROTHEUS\_DATA** da aplicação do ambiente de Produção ou para os arquivos listados abaixo, pois quando habilitado o **Scan** em tempo real há impacto direto na performance do **Protheus** quando está sendo realizada a leitura dos arquivos. Caso esta configuração não seja permitida pela política de segurança da empresa, deverão ser excluídas da varredura **on-scan** do antivírus as seguintes extensões:

**E:\TOTVS\MicrosigalProtheus\_Data RootPath**

\*.amt  
\*.cdx  
\*.idx  
\*.ind  
\*.lcx  
\*.lck  
\*.int  
\*.dtc  
\*.log  
\*.tmp

\*.mem

\*.sem

\*.fcs

**E:\TOTVS\Microsigla\Protheus\lapo** *SourcePath*

\*.rpo

**2.16. Hardware do Servidor de Aplicação:**

Verifique se o *hardware* disponível para o servidor de aplicação atende à demanda de conexões e das rotinas que exigem maior *hardware* no horário de pico em que o sistema está sendo utilizado por quase todos os usuários. Para auxiliar com esta análise pode contatar a equipe **TIS** através do e-mail **tis.comercial@totvs.com.br** e solicitar que seja realizado um **Projeto de dimensionamento de Ambiente Sizing**.

**2.17. Configuração do MARS no DBAccess com MSSQL**

A configuração do MARS no arquivo appserver.ini é exclusivo para banco de dados **MS SQL**. Sua ativação melhora o desempenho do **DBAccess**.

Maiores informações sobre esta configuração, acesse o *link*:

DBAccess - Seção [Environment], chave MARS

(<http://tdn.totvs.com.br/pages/viewpage.action?pagelId=451269662>)

**2.18. Correta configuração do Drive ODBC**

Configure a conexão com o banco de dados via **ODBC** de forma correta, evitando assim conflitos. Consulte o procedimento de configuração **ODBC** no *link* abaixo:

Como criar uma fonte de dados para uso com Microsoft SQL Server

([http://tdn.totvs.com/display/tec/DBAccess+-+Como+criar+uma+fonte+de+dados+para+uso+com+Microsoft+SQL+Serv](http://tdn.totvs.com/display/tec/DBAccess+-+Como+criar+uma+fonte+de+dados+para+uso+com+Microsoft+SQL+Server)

er)

**3. Manutenção****3.1. Fragmentação de Tabelas, SQL Server**

Verifique se há tabelas e index do banco de dados que estão desfragmentadas. A fragmentação das tabelas diminui a performance do banco de dados e da aplicação. Efetue a desfragmentação das tabelas e *tablespaces* do banco de dados da aplicação, com o objetivo de melhorar as operações de **DML** e **DQL** no banco de dados e aumentar a performance das rotinas do **Protheus**.

Um dos grandes problemas que temos com relação a performance é devido a fragmentação de nossos índices. Com o grande número de inserções, alterações e exclusões que ocorrem em nossas tabelas, os índices se fragmentam cada vez mais, ocasionando uma lentidão na manipulação dos dados desses índices.

Verifique se há índices pertencentes às tabelas da aplicação que estão fragmentados, pois quando estes índices estão fragmentados, diminui a performance da aplicação e do banco de dados. Neste caso realize o *Rebuild* dos índices das tabelas da aplicação.

**3.2. Estatísticas do SQL Server:**

Verifique se as estatísticas das tabelas do banco de dados do **Protheus** estão atualizadas, pois com as estatísticas desatualizadas causa impacto na performance da aplicação. Caso esteja desatualizado, colete as estatísticas do *schema* e do dicionário de dados do banco de dados, com o objetivo de melhorar a performance.

Dois parâmetros existentes em bases **SQL Server** podem trazer efeitos de melhor experiência do Protheus:

**Auto Update Statistics** – configurada como *True*, as estatísticas de índice são automaticamente atualizadas.

**Auto Create Statistics** – configurada como *True*, as estatísticas de índice são automaticamente criadas, sempre que você criar um índice, na execução de cada instrução o **SQL Server** cria um conjunto de estatísticas sobre os dados contidos dentro do índice.

O otimizador de consulta utiliza essas estatísticas para determinar se ele deve ou não utilizar o índice para ajudar a processar a consulta, no caso do

**Protheus** sabendo onde está alocado o dado o retorno será mais rápido.

Efetuar a manutenção da base de dados, realizando a reindexação e ou reconstrução de índices e atualização de estatísticas além de monitorar o espaço para crescimentos dos arquivos de dados e arquivos de log do banco de dados. Verifique também a consistência física e lógica da base de dados. Estes procedimentos são de responsabilidade do **DBA** da empresa, caso não possua **DBA** a equipe de consultoria da **TOTVS** poderá ser acionada para esta avaliação.

### 3.3. Coleta de Estatística do Oracle:

O **DataBase Oracle** precisa de uma boa estatística para tomar as melhores decisões que puder quanto ao caminho de acesso mais apropriado. Sem nenhuma estatística, o **DataBase** deve fazer suposições sobre quais são os melhores caminhos de acesso ao dado. Em muitos casos, conduz o **DataBase** a escolher caminhos menos performáticos. As estatísticas que são reunidas incluem estatísticas sobre tabelas, número de linhas, número de blocos, estatísticas sobre colunas, número de valores distintos, número de NULLs e distribuição de dados, estatísticas sobre index, número de blocos, tamanho do index, fator cluster, e estatísticas sobre desempenho de Sistema. Há dois métodos utilizados para a coleta de estatísticas: o comando **analyze** e o pacote fornecido **dbms\_stats**.

O Método **dbms\_stats** é o mais utilizado para calcular estatísticas para o database, porém em versões futuras, o pacote **dbms\_stats** será a única maneira de calcular estatísticas. Vale ressaltar que o método **dbms\_stats** é o que recomendamos o seu uso, através dos *scripts* abaixo:

#### Método gather\_schema\_stats

O **gather\_schema\_stats** calcula as estatísticas para todos os objetos em um dado esquema. As estatísticas podem ser colocadas no dicionário de dados ou na tabela de estatísticas de um usuário.

Recomendamos o uso do script abaixo.

Recomendamos o uso da coleta de estatística do dicionário do **database** e do **owner** que se encontra os dados do **Protheus**.

```
exec sys.dbms_stats.gather_dictionary_stats (comp_id => null,
estimate_percent => null, method_opt => 'FOR ALL COLUMNS size
AUTO', degree => 2, cascade => TRUE, no_invalidate => true);
```

```
exec sys.dbms_stats.gather_schema_stats('PROTHEUS', CASCADE=>TRUE,
METHOD_OPT=>'FOR ALL INDEXED COLUMNS');
```



Esse artigo foi útil?



Usuários que acharam isso útil: 15 de 18

### 0 Comentários

Por favor, entre ([https://centraldeatendimento.totvs.com/hc/pt-br/signin?return\\_to=https%3A%2F%2Fcentraldeatendimento.totvs.com%2Fhc%2Fpt-br%2Farticles%2F360017515552-MP-FRAME-Lentid%25C3%25A3o-no-Protheus-12](https://centraldeatendimento.totvs.com/hc/pt-br/signin?return_to=https%3A%2F%2Fcentraldeatendimento.totvs.com%2Fhc%2Fpt-br%2Farticles%2F360017515552-MP-FRAME-Lentid%25C3%25A3o-no-Protheus-12)) para comentar.

## Como podemos ajudar?

Entre em contato e fale conosco!

[Página Inicial](#)

[Centro de serviços](#)

[Portal de Produtos \(https://produtos.totvs.com/\)](https://produtos.totvs.com/)

[Suporte \(https://totvscst.zendesk.com/hc/pt-](https://totvscst.zendesk.com/hc/pt-)

[Downloads](#)