

## Plataformas de Dados InterSystems e desempenho - Parte 2

Artigo

[Murray Oldfield](#) · Jan. 18

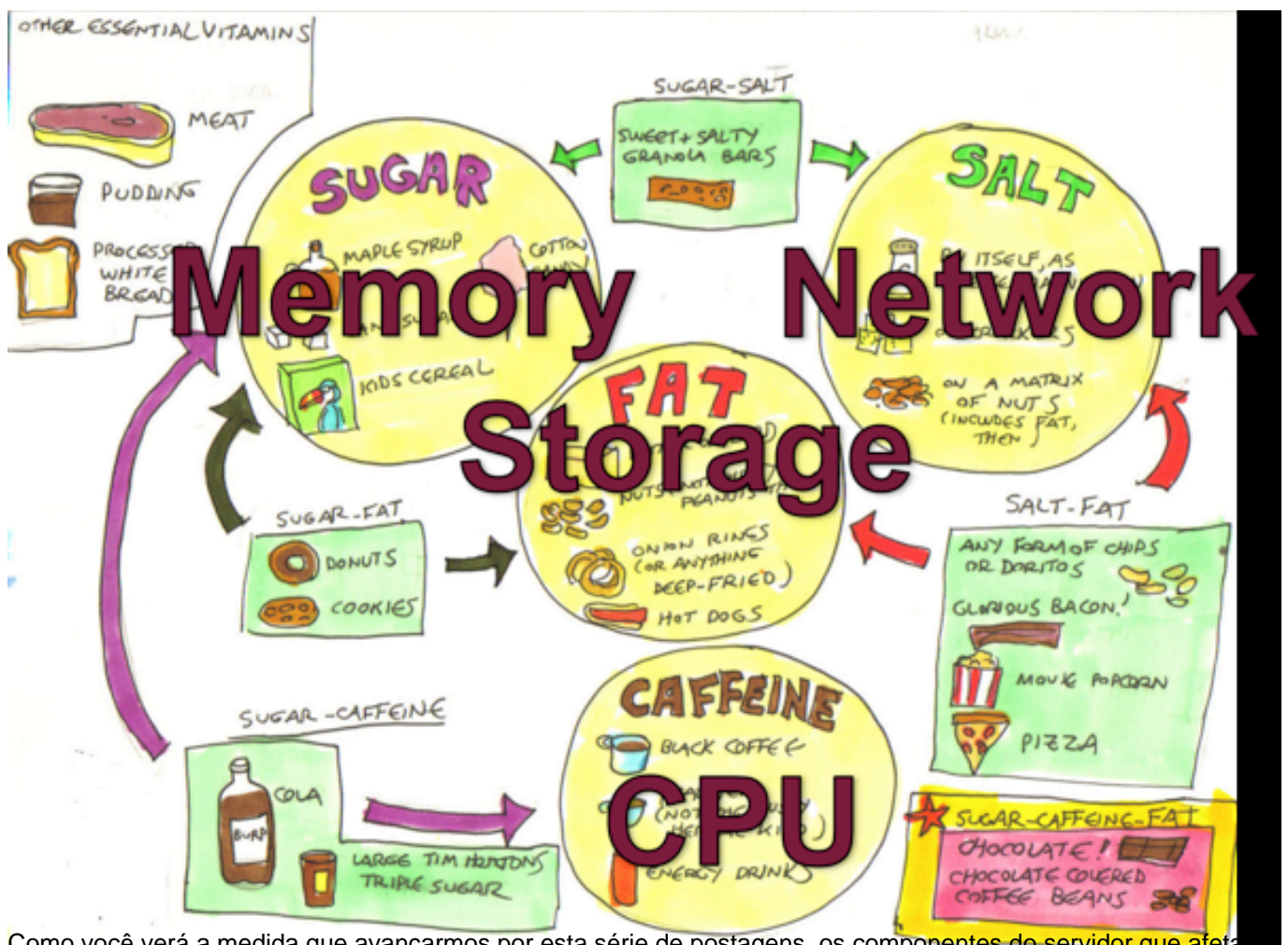
9min de leitura

## Plataformas de Dados InterSystems e desempenho - Parte 2

Na última postagem, agendamos a coleta de métricas de desempenho durante 24 horas usando pButtons. Nesta postagem, vamos ver algumas métricas essenciais que estão sendo coletadas e como elas estão ligadas ao hardware do sistema. Também começaremos a explorar a ligação entre as métricas do Cachê (ou de qualquer plataforma de dados InterSystems) e as métricas do sistema. Além disso, veremos como você pode usar essas métricas para entender a integridade diária de seus sistemas e diagnosticar problemas no desempenho.

[Veja aqui uma lista de outras postagens do autor na comunidade em inglês](#)

### Grupos alimentares de hardware



Como você verá a medida que avançarmos por esta série de postagens, os componentes do servidor que afetam o desempenho podem ser categorizados como:

- CPU
- Memória
- E/S de armazenamento
- E/S de rede

Se algum desses componentes estiver sobrecarregado, o desempenho do sistema e a experiência do usuário serão prejudicados. Todos esses componentes estão inter-relacionados, e mudanças em um deles pode afetar os outros, às vezes com consequências inesperadas. Já vi um caso em que corrigir um problema de gargalo de E/S em uma matriz de armazenamento fez o uso de CPU subir para 100%, e a consequência foi uma experiência do usuário ainda pior, pois o sistema ficou livre de repente para fazer mais trabalho, mas não tinha os recursos de CPU necessários devido ao aumento da atividade e taxa de transferência dos usuários.

Também veremos como a atividade do sistema do Caché tem impacto direto nos componentes do sistema. Se houver recursos de E/S de armazenamento limitados, uma mudança positiva que pode ser feita é o aumento da memória do sistema e o aumento da memória dos **buffers globais do Caché**, o que pode diminuir a **E/S de leitura de armazenamento do sistema** (mas talvez aumente o uso de CPU!).

Uma das métricas de sistema mais óbvias que deve ser monitorada regularmente ou verificada se os usuários relatarem problemas é o uso de CPU. Pode-se verificar utilizando o *top* ou *nmon* no Linux e AIX, ou *Monitor de Desempenho do Windows*. Como a maioria dos administradores de sistema verificam os dados de CPU regularmente, ainda mais quando são apresentados em gráficos, uma olhada rápida fornece uma boa compreensão da integridade atual do sistema: o que está normal, ou um surto repentino de atividade, que pode ser anormal ou indicar um problema. Nesta postagem, vamos dar uma olhada rápida nas métricas de CPU, mas nos concentraremos nas métricas do Caché. Começaremos verificando os dados do *mgstat* e como ver os dados em gráficos pode dar uma boa compreensão da integridade do sistema rapidamente.

## Introdução ao mgstat

O *mgstat* é um dos comandos do Caché incluídos e executados no pButtons. O *mgstat* é uma ferramenta excelente para coletar métricas básicas de desempenho para ajudar a entender a integridade dos sistemas. Veremos os dados do *mgstat* coletados durante 24 horas em um pButtons, mas, se você quiser capturar dados além do pButtons, o *mgstat* também pode ser executado sob demanda de forma interativa ou como um trabalho em segundo plano pelo terminal do Caché.

Para executar o *mgstat* sob demanda pelo namespace %SYS, o formato geral é:

```
do mgstat( sample_time, number_of_samples, "/caminho_do_arquivo/arquivo.csv", page_length )
```

Por exemplo, veja abaixo o comando para executar em uma tarefa em segundo plano por uma hora com um período de amostragem de 5 segundos e salvar a saída em um arquivo csv:

```
job ^mgstat(5,720,"/data/mgstat_data_e_hora_de_hoje.csv")
```

Por exemplo, para exibir na tela sem algumas colunas, chame a rotina através do label dsp132: Deixarei como tarefa para vocês a verificação da saída para que vocês entendam melhor a diferença.

```
do dsp132^mgstat(5,720,"",60)
```

As informações detalhadas das colunas do *mgstat* estão disponíveis no *Guia de monitoramento do Caché* na documentação mais recente do Caché: [Documentação online da InterSystems](#)

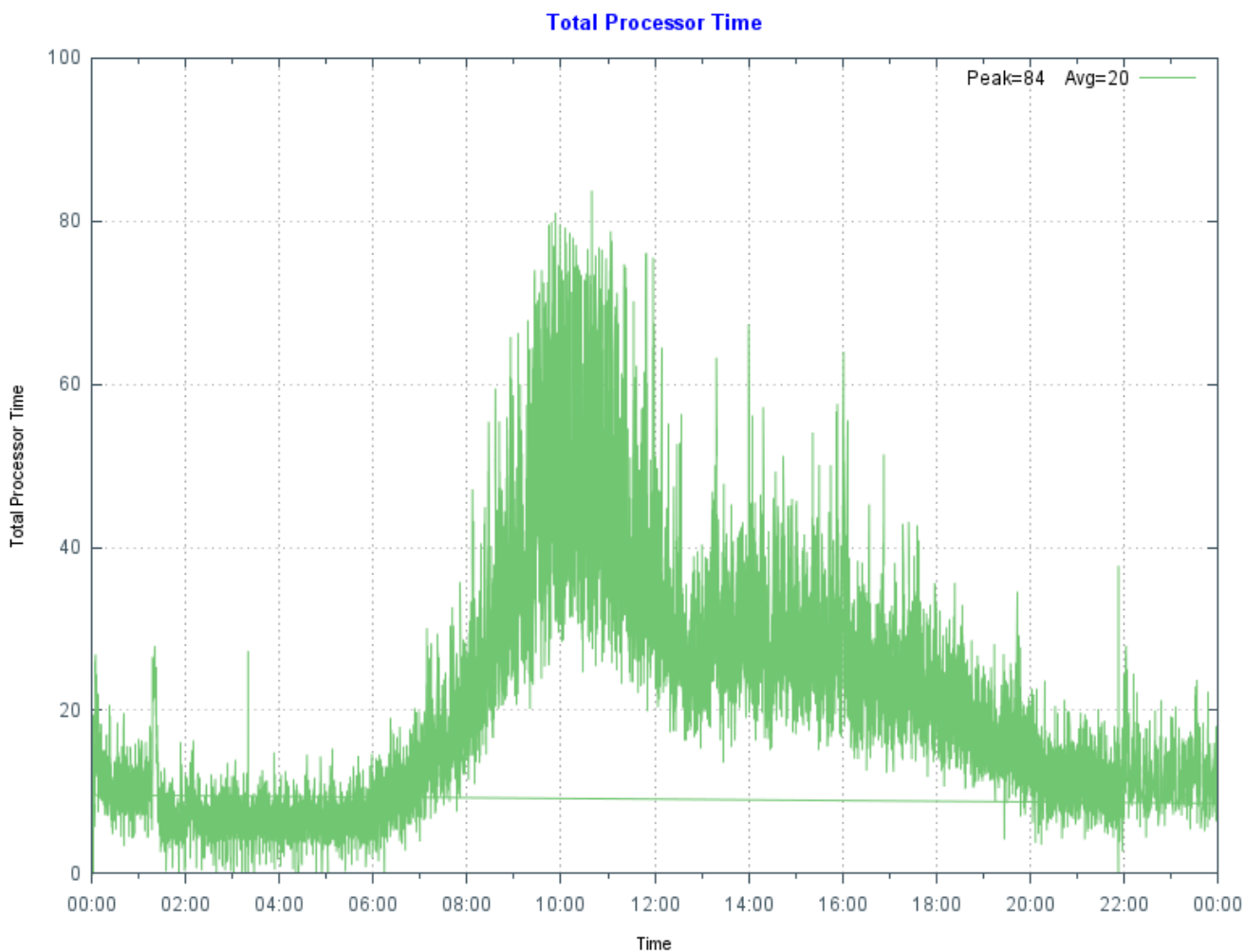
## Sobre os dados do mgstat

O pButtons agrupa os dados em um único arquivo HTML, facilitando a navegação e empacotando os dados para envio aos especialistas de suporte da Central de Suporte (WRC) para diagnosticar problemas no desempenho. Entretanto, quando você executa o pButtons por conta própria e deseja exibir os dados em gráficos, eles podem ser separados novamente em um arquivo csv para processamento em gráficos, por exemplo, com o Excel, usando um script de linha de comando ou apenas cortando e colando os dados.

Nesta postagem analisaremos algumas métricas do mgstat para mostrar como somente uma visão rápida dos dados pode indicar se o sistema está com bom desempenho ou se há problemas atuais ou potenciais que afetarão a experiência do usuário.

### Glorefs e CPU

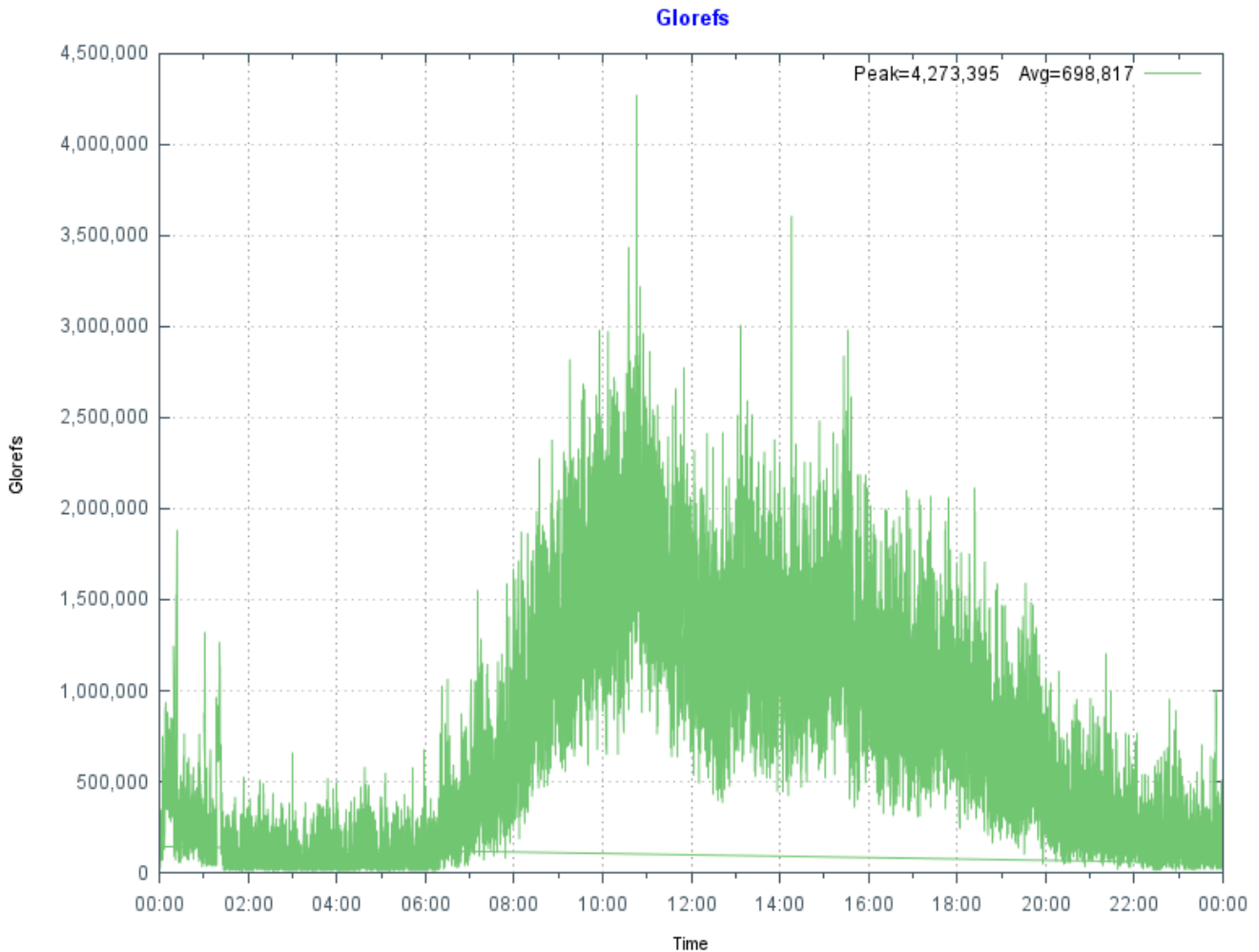
O gráfico abaixo mostra o uso de CPU do servidor de banco de dados em um local que executa uma aplicação de hospital com uma alta taxa de transações. Note o pico de atividade durante a manhã, quando há vários ambulatorios, com uma queda durante o almoço, diminuindo então à tarde e à noite. Neste caso, os dados vêm de `(_Total)\% Tempo do Processador` do Monitor de Desempenho do Windows. O formato do gráfico corresponde ao perfil diário de trabalho, sem picos ou depressões incomuns, o que é normal para esse local. Fazendo o mesmo em seu local, você pode começar a estabelecer uma linha de base do que é "normal". Um grande pico, especialmente um de longa duração, pode ser um indicador de problema. Uma postagem futura se concentrará na CPU.



Como referência, este banco de dados está em um servidor Dell R720 com dois processadores E5-2670 de 8 núcleos. O servidor tem 128 GB de memória e 48 GB de buffers de globais.

O próximo gráfico mostra mais dados do mgstat: Glorefs (referências globais) ou acessos ao banco de dados no mesmo dia que o gráfico de CPU. As glorefs indicam a quantidade de trabalho ocorrendo durante a carga de

trabalho atual. Embora as referências a globais consumam tempo de CPU, nem sempre elas consomem outros recursos do sistema, como leituras físicas, devido à maneira como o Caché usa o pool global de buffers de memória.



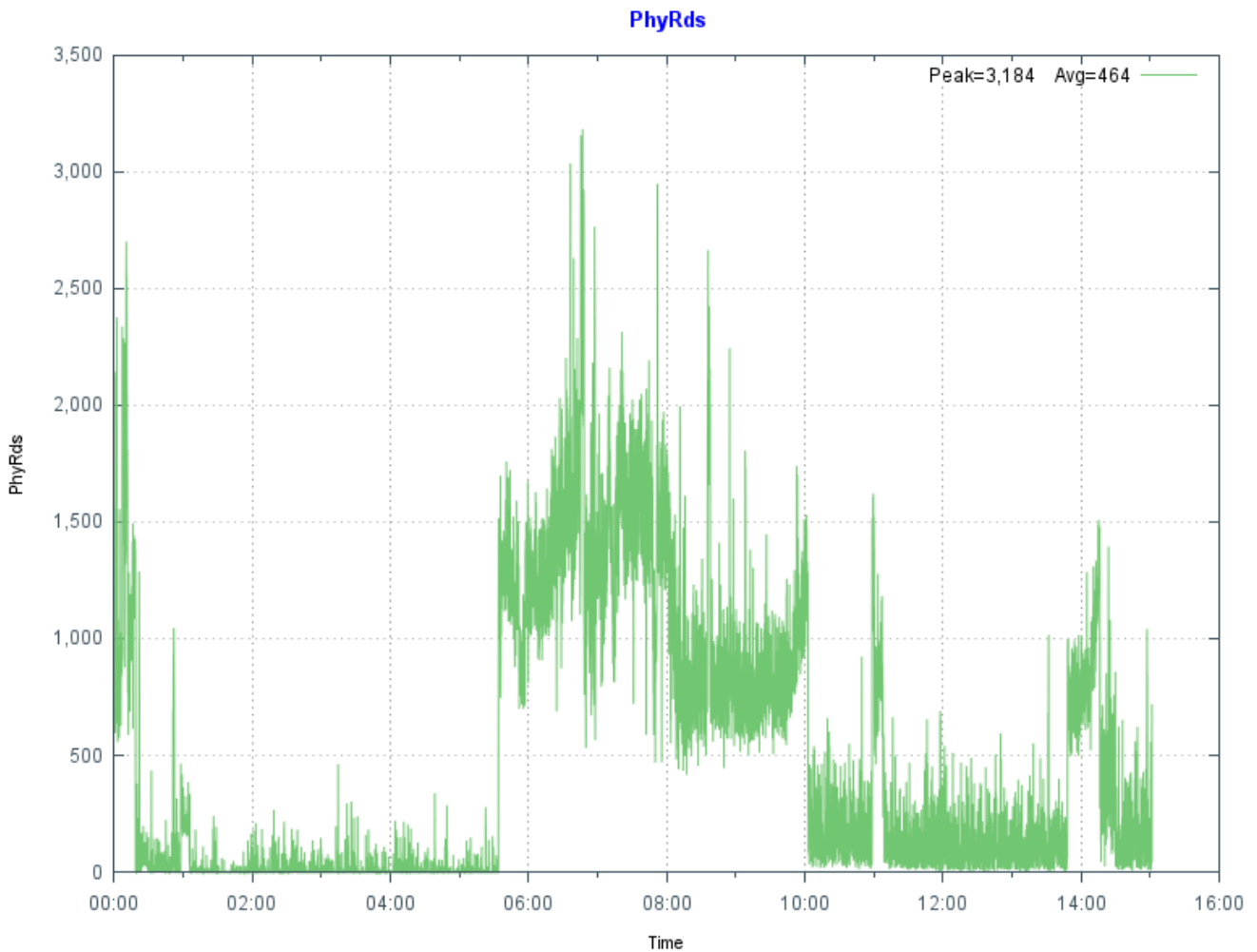
Nas aplicações típicas do Caché, há uma correlação muito forte entre as glorefs e o uso de CPU.

Outro modo de olhar esses dados de CPU e glorefs é dizendo que *a redução das glorefs reduz o uso de CPU*, permitindo a implantação em servidores com menos núcleos ou a escalabilidade de sistemas existentes. É possível reduzir as referências globais tornando uma aplicação mais eficiente. Voltaremos a esse conceito em postagens futuras.

## PhyRds e Rdratio

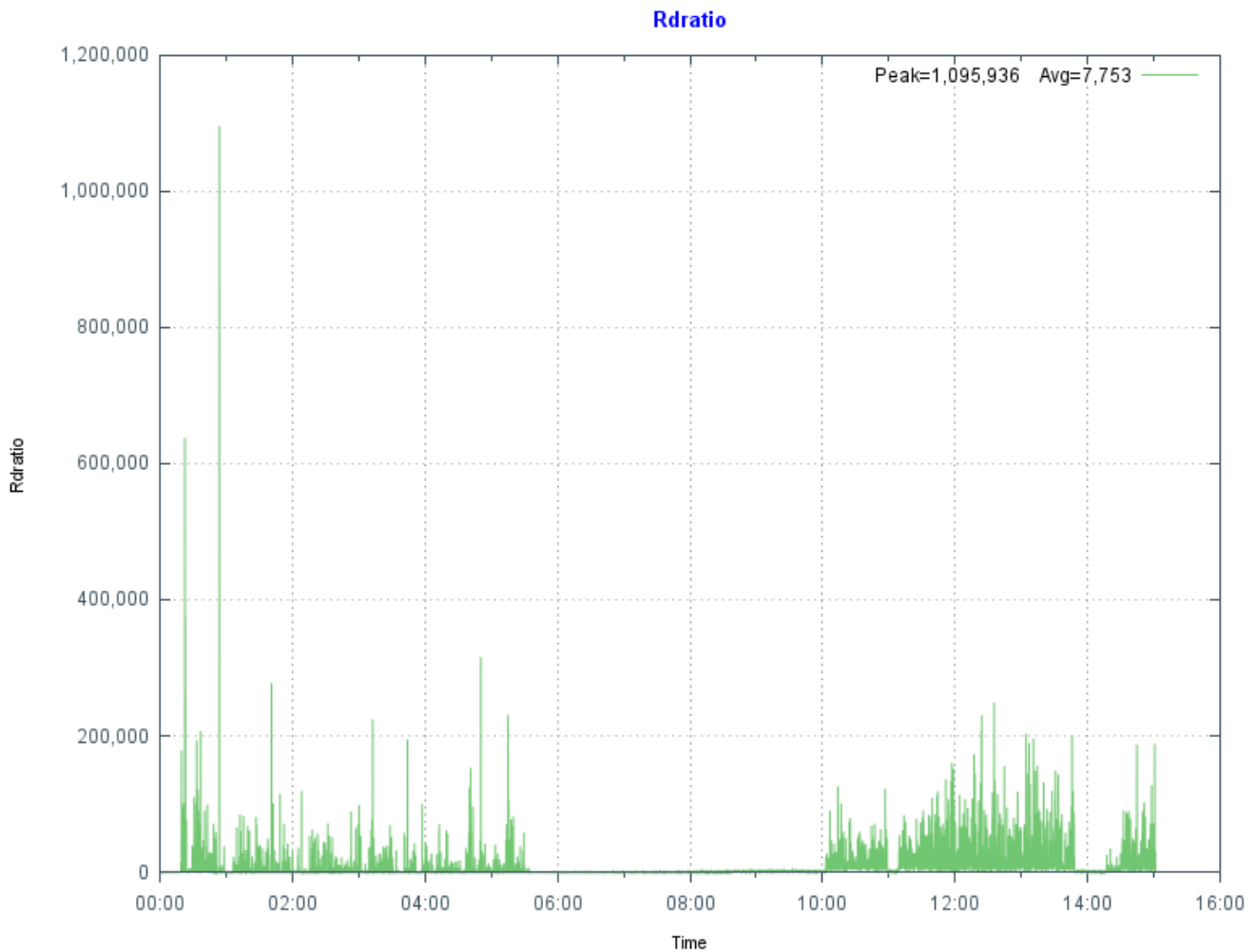
O formato da representação gráfica dos dados *PhyRds* (leituras físicas) e *Rdratio* (proporção de leituras) do `mgstat` também pode fornecer informações do que se esperar do desempenho do sistema e ajudar a planejar a capacidade. Vamos analisar com mais detalhes a E/S de armazenamento no Caché em postagens futuras.

*PhyRds* são simplesmente Operações de Entrada e Saída por Segundo (IOPS) de leitura nos bancos de dados do Caché. Você deverá ver os mesmos valores nas métricas do sistema operacional para discos lógicos e físicos. Lembre-se de que, ao verificar as Operações de Entrada e Saída por Segundo (IOPS) do sistema operacional, também podem estar sendo exibidos dados de outras aplicações que não o Caché. Dimensionar o armazenamento sem levar em conta as Operações de Entrada e Saída por Segundo (IOPS) esperadas é uma receita para o desastre. Você precisa saber as IOPS de seu sistema em horários de pico para planejar a capacidade corretamente. O gráfico abaixo mostra *PhyRds* entre meia-noite e 15h30.



Observe o grande salto de leituras entre 5h30 e 10h. Também há picos menores às 11h e um pouco antes das 14h. Qual você acha que é a causa desses picos? Você também vê esse tipo de picos em seus servidores?

*Rdratio* é um pouco mais interessante: é a proporção de leituras de blocos lógicos em comparação a leituras de blocos físicos. Portanto, é a proporção de quantas leituras são de buffers de globais (lógicos) da memória e quantas são do disco, que são algumas ordens de grandeza mais lentas. Uma *Rdratio* alta é uma boa coisa, e chegar perto de zero por longos períodos não é bom.



Observe que, quando as leituras físicas estão altas, *Rdratio* cai para perto de zero. Nesse local, me pediram para investigar quando o departamento de TI começou a receber ligações de usuários informando que o sistema estava lento por longos períodos. Isso estava acontecendo de maneira aparentemente aleatória durante várias semanas quando me pediram para verificar o sistema.

\_\*\*Como o pButtons havia sido agendado para execução diária de 24 horas, foi relativamente simples analisar os dados de várias semanas passadas para começar a ver um padrão de *\_PhyRds\_ alta* e *\_Rdratio\_ baixa*, com uma correlação com as ligações de suporte.\*\*\_

Após uma análise mais detalhada, descobriu-se que a causa era um novo funcionário que executava diversos relatórios com parâmetros ruins, além de consultas mal escritas sem índices apropriados, o que causava muitas leituras no banco de dados. Essa era a causa da lentidão aparentemente aleatória. Como esses relatórios de longa execução lêem dados dos buffers de globais, a consequência é que os dados interativos dos usuários estavam sendo obtidos do armazenamento físico, em vez da memória, e o armazenamento estava sendo sobrecarregado para atender às solicitações de leitura.

O monitoramento das métricas *PhyRds* e *Rdratio* dará uma ideia da integridade de seus sistemas e talvez permitirá que você descubra relatórios ou consultas ruins. Pode haver um motivo válido para uma *PhyRds* alta: talvez um relatório precise ser executado em determinada hora. Com os sistemas operacionais modernos de 64 bits e servidores com alta capacidade de memória física, você deverá conseguir minimizar a *PhyRds* em seus sistemas de produção.

Caso a *PhyRds* esteja alta em seu sistema, você pode considerar algumas estratégias: - Melhorar o desempenho aumentando o número de buffers (globais) de banco de dados (e a memória do sistema). -



Relatórios ou extrações de longa duração podem ser feitos fora do horário comercial. - Relatórios somente leitura de longa duração, trabalhos em lote ou extrações de dados podem ser feitos em um servidor Shadow ou um membro de espelhamento assíncrono para minimizar o impacto nos usuários interativos e para diminuir a carga de recursos do sistema, como CPU e Operações de Entrada e Saída por Segundo (IOPS).

Geralmente, é bom ter uma *PhyRds* baixa, e é nossa meta ao dimensionar os sistemas. Entretanto, se a *PhyRds* estiver baixa e os usuários estiverem reclamando do desempenho, você pode verificar outros fatores para confirmar que o armazenamento não é o gargalo. A quantidade de leituras pode estar baixa porque o sistema não consegue mais atender à demanda. Veremos o armazenamento com mais detalhes em uma postagem futura.

## Resumo

Nesta postagem, verificamos como olhar graficamente as métricas coletadas no pButtons pode fornecer uma compreensão da integridade rapidamente. Em postagens futuras, analisarei com mais detalhes a ligação entre as métricas de sistema e do Caché e como você pode usá-las para se planejar para o futuro.

[#Administração do Sistema](#) [#Desempenho](#) [#Soluções de Negócio e Arquiteturas](#) [#Caché](#) [#InterSystems IRIS](#)  
[#InterSystems IRIS for Health](#)

00 3 0 0 29

Postos relacionados

- [Plataformas de Dados InterSystems e desempenho - Parte 1](#)
- Plataformas de Dados InterSystems e desempenho - Parte 2

Log in or sign up to continue

Acrescentar resposta

**URL de origem:** <https://pt.community.intersystems.com/post/plataformas-de-dados-intersystems-e-desempenho-parte-2>