

MONITORANDO BANCO DE DADOS INTERBASE COM O PERFORMANCE MONITOR E TABELAS DE SISTEMAS

Olá a todos,

Quantas vezes nos deparamos com problemas de performance em nossas aplicações? Em determinado momento do dia, todos os usuários da rede começam a reclamar uma lentidão muito forte nos sistemas. Olhamos o consumo do tráfego de rede, o consumo de processador e memória do servidor e, aparentemente, não há nada de errado. Mas a sensação de lentidão na rede toda permanece. O que fazemos nesse momento? Paramos todos os usuários e reiniciamos o servidor? Desligamos abruptamente o servidor com todos os usuários conectados? Ou simplesmente esperamos essa lentidão passar sozinha?

Na verdade, sem uma ferramenta de apoio que nos mostre com total confiança tudo o que está acontecendo em nossas aplicações, fica realmente muito difícil tomar outra decisão nesses momentos.

Nesse artigo quero apresentar um recurso do InterBase, disponível desde sua versão 7.5, mas que ainda pouco conhecido e pouco utilizado: o monitoramento de aplicações em tempo real utilizando tabelas de sistema e o utilitário Performance Monitor.

COMO FUNCIONA

Utilizando o recurso de tabelas temporárias, o InterBase cria uma nova estrutura padrão de tabelas chamadas *tabelas temporárias de sistema*. Essas tabelas ficam alocadas em memória e consomem baixíssimo recurso do servidor. O InterBase então inicia uma transação que coleta informações do seu serviço e armazena dados nessas tabelas. Esse “coletor de dados” funciona para cada arquivo de banco de dados que você possui no servidor. Quando acessamos essas tabelas, elaboramos pesquisas sobre elas, trazemos informações em tempo real do que está se passando no banco de dados.

Podemos também utilizar a ferramenta Performance Monitor, que vem integrada ao já conhecido IBConsole. O Performance Monitor trás de maneira visual e muito agradável todas as informações das tabelas temporárias de sistemas, além de uma série de facilidades interessantes que veremos mais a frente.

As tabelas temporárias de sistemas são:

TMP\$ATTACHMENTS: Armazena uma linha para cada conexão no banco de dados.

TMP\$DATABASE: Armazena uma linha para cada banco de dados que você está conectado.

TMP\$HEAPS: Armazena uma linha para cada entrada no heap Randômico e de Blocos do InterBase.

TMP\$POOLS: Armazena uma linha para cada pool de memória.

TMP\$POOL_BLOCKS: Armazena uma linha para cada bloco de memória em cada pool de conexão.

TMP\$PROCEDURES: Armazena uma linha para cada procedure executada desde o início da conexão corrente.

TMP\$RELATIONS: Armazena uma linha para cada relacionamento referenciado desde o início da conexão corrente.

TMP\$STATEMENTS: Armazena uma linha para cada statement corrente executada por qualquer conexão corrente.

TMP\$TRANSACTIONS: Armazena uma linha para cada transação que está ativa ou em limbo.

TMP\$TRIGGERS: Armazena uma linha para cada trigger executada desde o início da conexão corrente.

TRABALHANDO COM AS TABELAS TEMPORÁRIAS DE SISTEMA

Com as tabelas temporárias de sistemas, podemos executar pesquisas ou mesmo atualizá-las. Veja alguns exemplos do que podemos preparar com as tabelas temporárias de sistemas:

- Quais são as conexões ativas no banco de dados nesse momento;
- Quais são as tabelas com mais leitura seqüencial;
- Quais são as transações com maior consumo de carga de CPU;
- Quais são os statements que estão ativos há mais tempo no sistema;
- Qual conexão está realizando mais updates no banco de dados;
- Qual o usuário que faz mais inserções no banco de dados;
- Checar o status do sweep do banco de dados;
- Acionar um *commit* ou um *rollback* em determinada transação;
- Cancelar determinada conexão;
- Utilizar essas informações para criar um sistema de controle de acesso à aplicação;
- Entre inúmeras outras ações.

Abaixo seguem um exemplo para ilustrar:

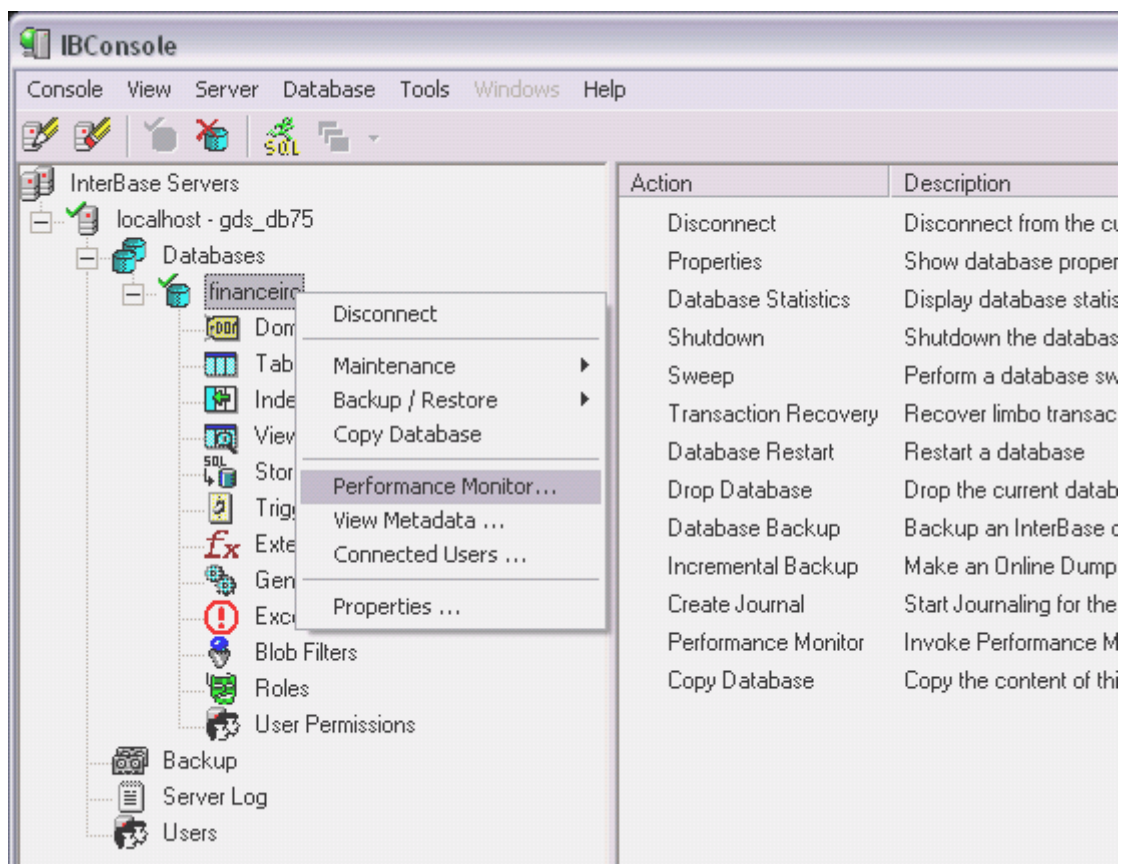
Selecionando os TOP 10 statements em execução

```
SELECT
  a.tmp$user, s.tmp$timestamp, s.tmp$sql, s.tmp$quantum
FROM
  TMP$STATEMENTS s, TMP$ATTACHMENTS a
WHERE
  a.TMP$ATTACHMENT_ID = s.TMP$ATTACHMENT_ID
ORDER BY
  s.TMP$QUANTUM DESC
ROWS 10;
```

A pesquisa acima retornará o nome do usuário, data e hora, o corpo da pesquisa e a quantidade de consumo de processador dos dez primeiros statements ordenados por consumo de processador.

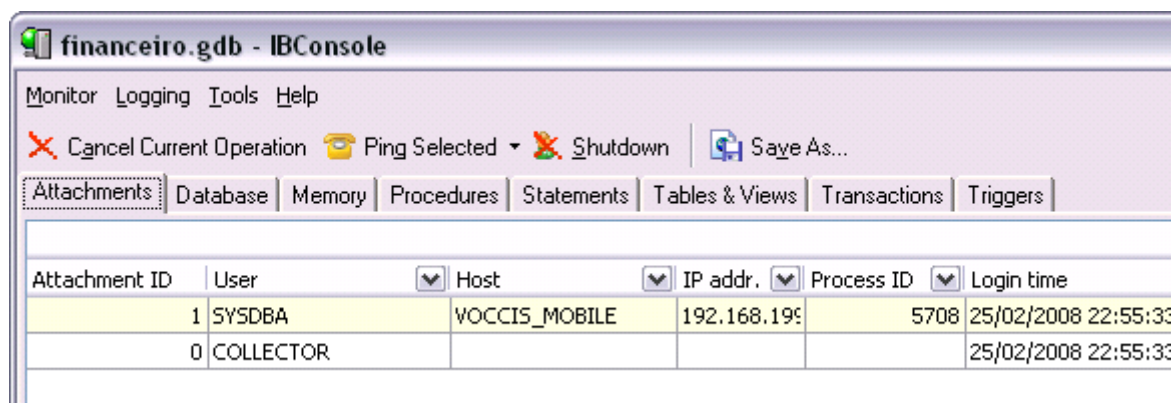
O PERFORMANCE MONITOR

Criado para facilitar o uso das tabelas temporárias de sistema, o Performance Monitor é uma poderosa ferramenta de monitoramento dos bancos de dados InterBase.



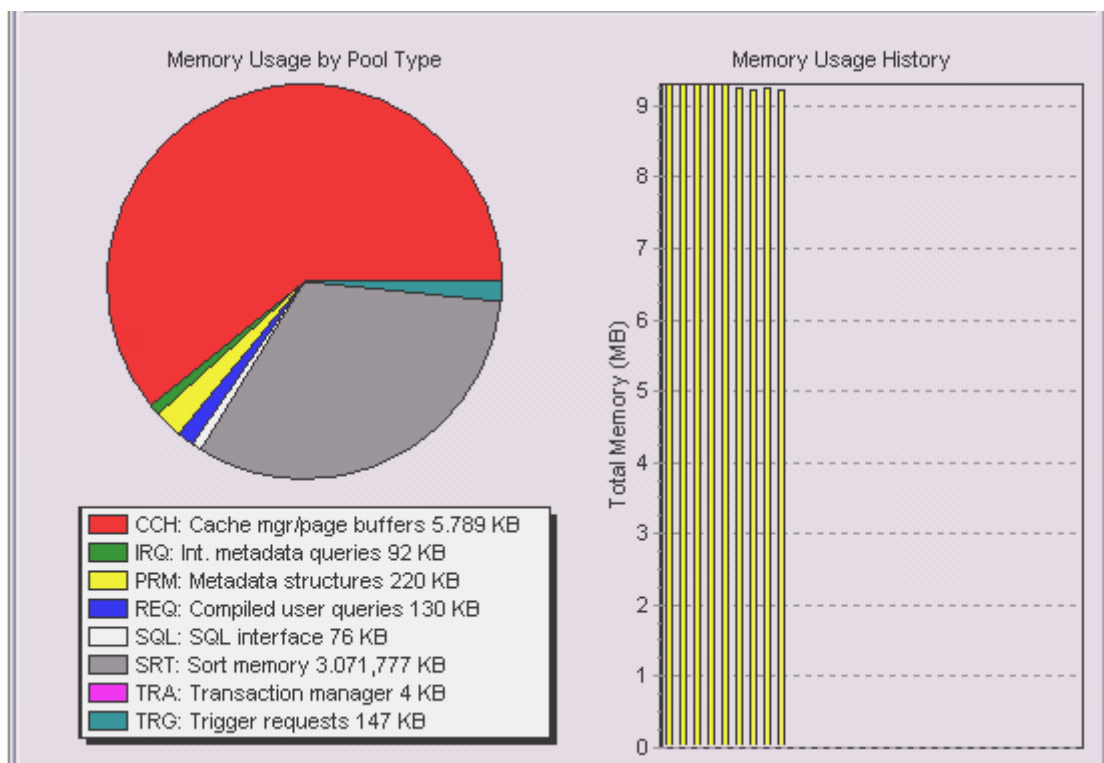
Vale notar que o uso do Performance Monitor não afeta a performance dos bancos de dados, pois o InterBase sempre inicia suas transações com baixa prioridade, além do fato dessas informações serem gravadas sempre em memória, não prejudicando ou concorrendo com o acesso a disco das aplicações.

O Performance Monitor possui algumas abas de funcionalidades, facilitando a identificação de problemas ponto a ponto:



Na aba *Attachments* é possível identificar quantos usuários estão conectados ao banco de dados, nome e endereço IP dos clientes, data e hora em que esses clientes se conectaram ao banco de dados, consumo de memória e de processador dessas conexões. É possível também identificar o cliente que está realizando mais fetchs, mais inserts ou updates. Com uma simples ordenação no cabeçalho da grade, é possível identificar, por exemplo, quais conexões estão consumindo mais recursos do sistema. Além disso, é possível selecionar essa conexão que está penalizando a performance e enviar um comando para cancelar ou derrubar o cliente, sem afetar o uso das outras conexões ou corromper o banco de dados. Para isso utilizamos os botões disponíveis na barra de tarefas.

Na aba *Database* encontramos um resumo de tudo o que está se passando nesse momento no banco de dados. Informações como quantos registros foram inseridos, apagados e atualizados, consumo geral de memória do banco de dados, quantidade de procedures, tabelas, triggers e transações ativas e informações sobre o processo de Sweep. Além disso, é possível disparar o processo de Sweep pontualmente, utilizando o botão disponível na barra de ferramentas.



Na aba *Memory*, um gráfico nos mostra exatamente como está o consumo de memória do banco de dados. É possível identificar a quantidade de memória cache disponível, o consumo de memória dos eventos do InterBase. O mais importante dessa informação é poder dimensionar a quantidade de buffer do banco de dados, aumentando sempre que for possível.

As abas *Procedures* e *Triggers* exibem informações sobre todas as stored procedures e triggers ativas no sistema. Podemos facilmente identificar, por exemplo, quais procedures estão consumindo mais recursos do sistema. Pontualmente abrimos nosso código e iniciamos um trabalho de melhoria na escrita de nossas procedures para melhorar sua performance e diminuir o impacto delas sobre a aplicação como um todo.

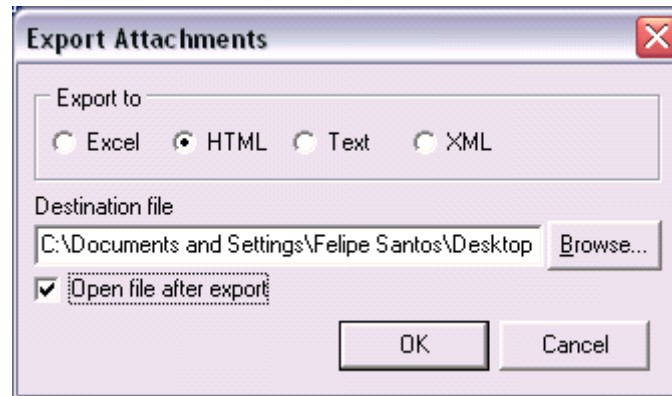
Na aba *Statements* identificamos todas as queries rodando no banco de dados em tempo real. Podemos utilizar os filtros disponíveis e encontrar, por exemplo, quais são as queries que a conexão que mais consome recurso está utilizando naquele momento. Podemos identificar consumo de memória, leituras seqüenciais, leituras não-indexadas, leituras indexadas, tempo de execução. Essas informações são muito úteis para podemos, por exemplo, identificar pesquisas pesadas, possivelmente por falta ou mal uso de índices, ou mesmo relacionamentos errados. Esse tipo de informação não seria possível identificar em um ambiente de produção, caso a caso, sem o uso de uma ferramenta como o Performance Monitor.

Já na aba *Tables & Views* podemos identificar em quais tabelas está sendo feita mais gravações ou atualizações. Outra informação importante e simples de visualizar é quais são as tabelas com mais leituras seqüenciais sendo executadas. Isso pode identificar uma falta de índice ou mal uso de índices, o que afeta diretamente na performance do banco de dados.

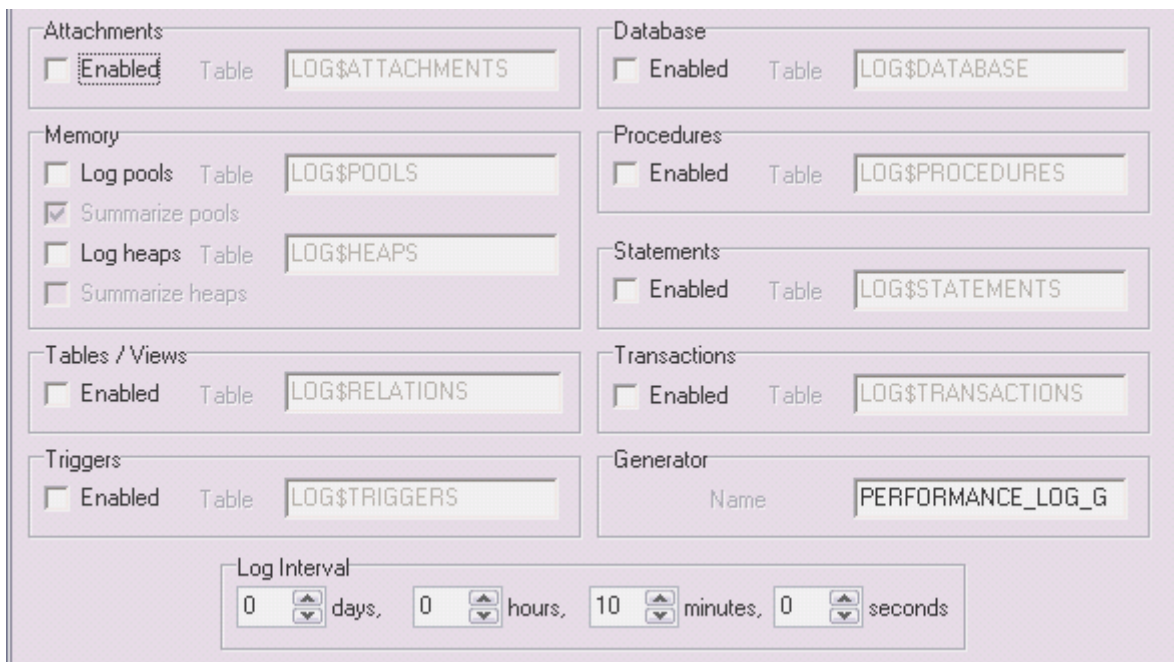
The screenshot shows the IBConsole application window titled 'financeiro.gdb - IBConsole'. The 'Transactions' tab is selected, displaying a table of active transactions.

User	State	Start time	Elapsed time	Commit retaining	Isolation
SYSDBA	ACTIVE	25/02/2008 22:55:33	0:05:27.000	<input checked="" type="checkbox"/>	READ COMMITTED
SYSDBA	ACTIVE	25/02/2008 23:00:08	0:00:52.000	<input type="checkbox"/>	READ COMMITTED

E na aba *Transactions* podemos identificar todas as transações abertas no sistema. Data e hora de inicialização, tempo de execução e tipo de isolamento são algumas informações disponíveis nessa aba. É possível também encontrar qual conexão disparou determinada transação. Utilizando os botões da barra de ferramentas podemos selecionar uma determinada transação e enviar um comando de *Commit* ou *Rollback*, exatamente naquele ponto, sem prejudicar o uso das outras transações.



Em todas as abas está disponível um botão *Save As*. Podemos facilmente exportar as informações de cada aba em diversos formatos. Além disso, o Performance Monitor possui um recurso de *Logging*, que analisa as informações das tabelas temporárias de sistema e armazena esses dados em tabelas físicas no banco, durante um intervalo pré-determinado. Como os administradores não podem muitas vezes ficar horas visualizando os dados em tempo real, esse recurso é muito útil para que as informações sejam armazenadas e visualizadas posteriormente.



Vale também notar que esse recurso gera sim um consumo maior. Por isso é importante marcar apenas as informações realmente necessárias para se fazer *Logging* e escolher um intervalo de tempo maior.

O Performance Monitor nos exhibe todas essas informações lendo as tabelas temporárias de sistema. Nada impede então de nós mesmos criarmos interfaces ou relatórios dentro de nossas aplicações, de acordo com a necessidade.

Convido a todos para testarem esse recurso de monitoramento. Utilizem a versão InterBase Developer Edition, disponível gratuitamente para desenvolvedores no site da CodeGear (www.codegear.com). Essas informações com certeza serão muito úteis no dia a dia, cliente a cliente, caso a caso, nos ajudando a identificar problemas e corrigi-los antes que esses problemas se tornem insuportáveis. Tudo de maneira simples, rápida e principalmente segura.

Fontes de pesquisa importantes:

InterBase 2007 Operations Guide – páginas 171 a 175.

InterBase 2007 Language Reference Guide – páginas 234 a 246.

Felipe Santos

felipe.santos@presence.com.br

Felipe Santos é especialista em InterBase e redes. Trabalha com o InterBase há mais de 7 anos e em ambientes Linux há mais de 9 anos. Administrador de sistemas ERP de grande porte, com soluções InterBase rodando em Linux em mais de 100 empresas. Especialista em ambientes críticos. Atuou em um dos maiores casos do InterBase no Brasil no ano de 2005. Participante ativo na comunidade, com artigos publicados. Palestrante em eventos como IB Tour e Borcon Conference. Instrutor em treinamentos avançados de InterBase. Atualmente trabalhando na área comercial / técnica do InterBase na Presence Tecnologia - distribuidora do produto no Brasil.