

# Banco de Dados Administração

José Antônio da Cunha

CEFET – RN

12/04/2005

# Profiler – SQL Server

Os sistemas normalmente não nascem lentos, mas tendem a ficar mais lentos com o tempo. O aumento do número de usuários, a existência de mais processos concorrentes, o crescimento do volume de informações armazenadas, a falta (ou excesso) de índices e, por fim, a má qualidade do código T\_SQL são atores que ocasionam o aparecimento de gargalos e, conseqüentemente queda de performance.

# Profiler – SQL Server

Antes de pensar que o problema “vem de fora” e cogitar em aumentar o poder de fogo do processador, discos ou memória, cabe uma análise mais detalhada dos processos ativos no servidor de banco de dados. Muitas vezes todo o problema pode ser resolvido com a adição de um índice ou filtro num comando update.

Mas como saber exatamente onde está o problema?

# Profiler – SQL Server

O SQL Server possui um utilitário chamado **Profiler**, indicado para rastreamento dos eventos processados numa base SQL Server 2000. o Profiler é uma ferramenta de diagnóstico, ou seja, ela nos fornece material para análise. Vale destacar que ela não se propõe por si só a efetuar correções ou qualquer espécie de tuning.

# Profiler – SQL Server

## Criando uma trace passo-a-passo

O Profiler é uma ferramenta para criar traces. Uma trace é como uma fotografia dos comandos executados pelo SQL Server 2000 em um determinado intervalo de tempo. Para criar uma trace, selecione Profiler no sub-menu do SQL Server (ver **Figura 1**). Na tela principal do Profiler, selecione File | New | Trace (ver **Figura 2**).

# Profiler – SQL Server

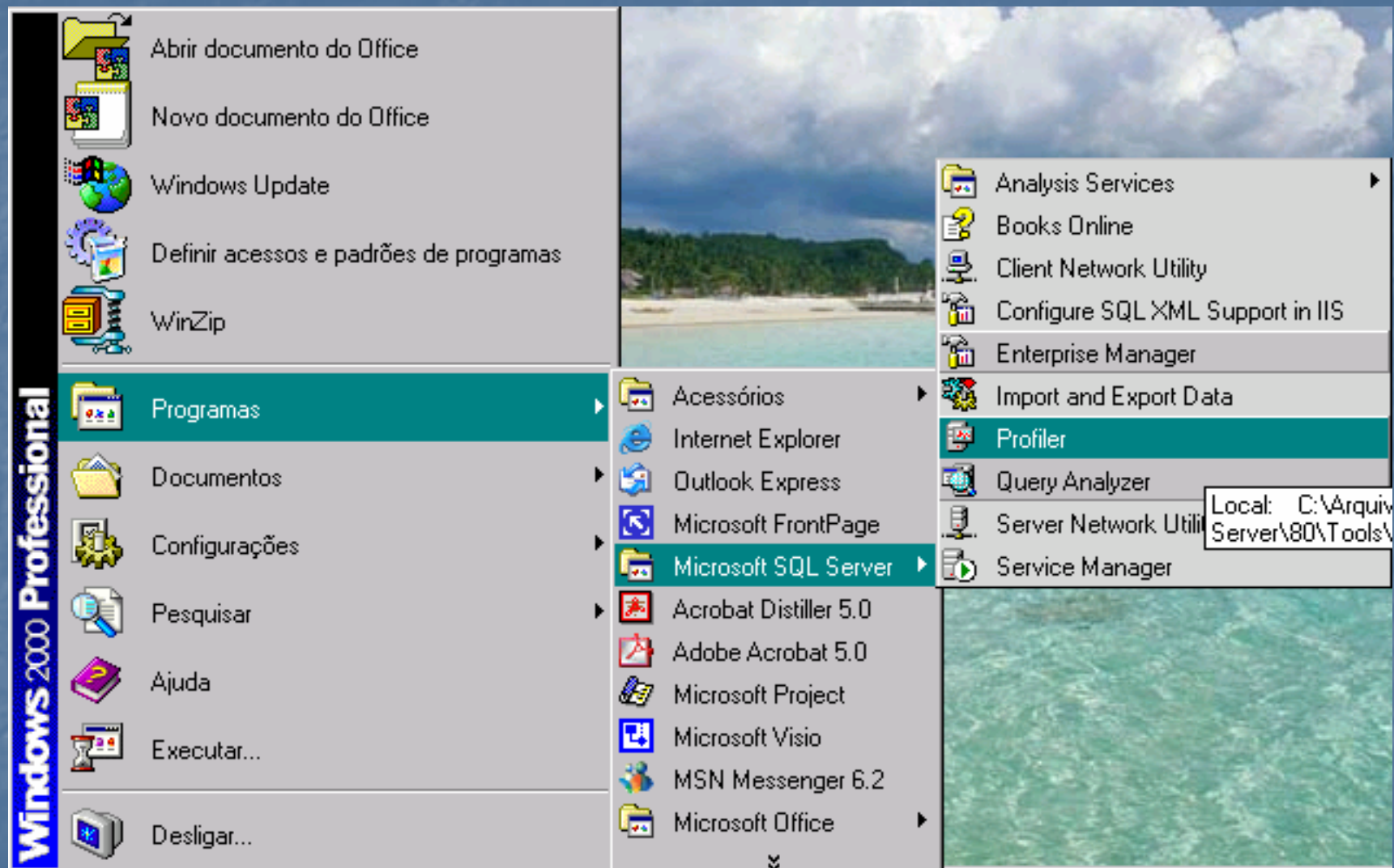
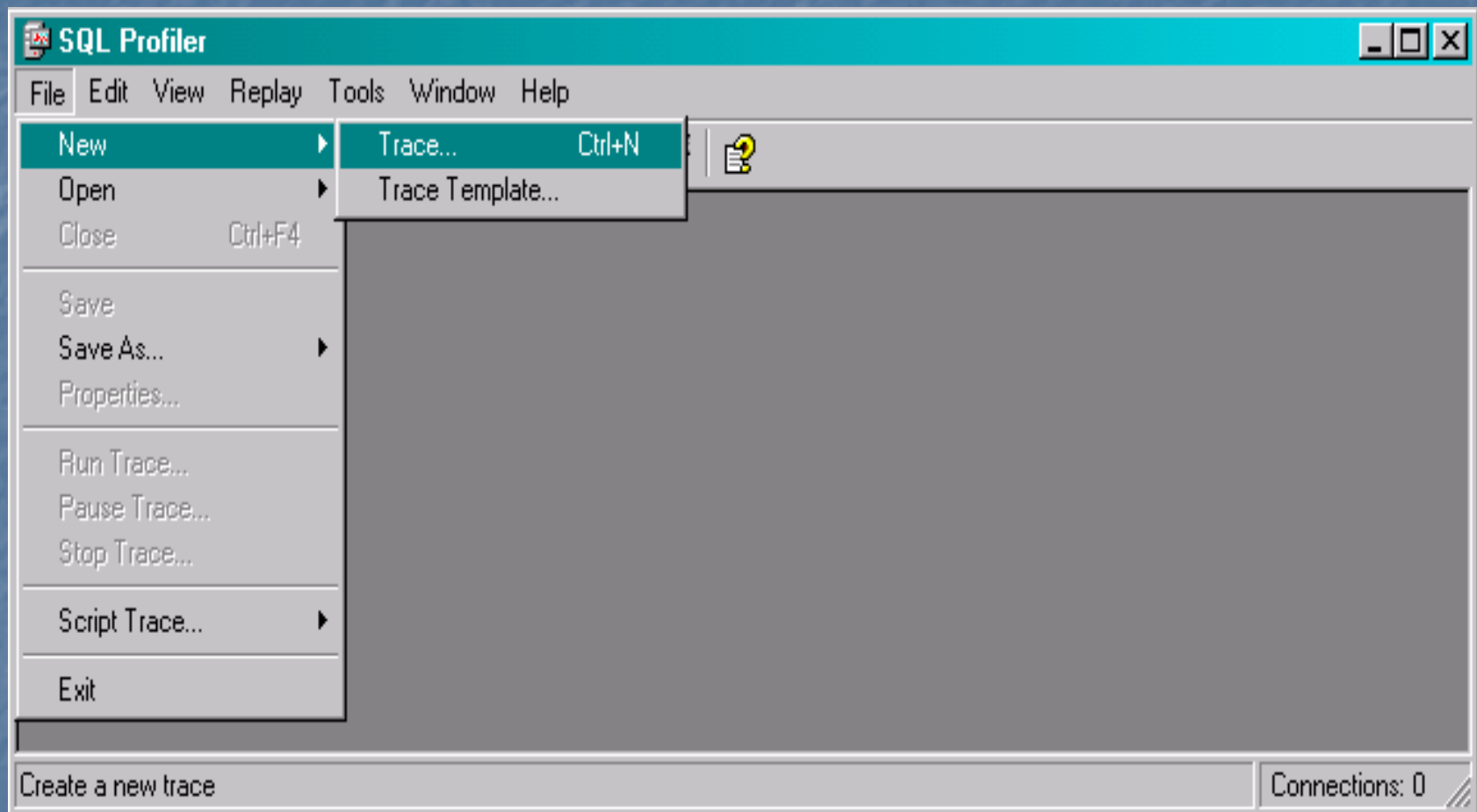


Figura 1 Selecionando o Profiler no Sub-menu do Microsoft SQL Server

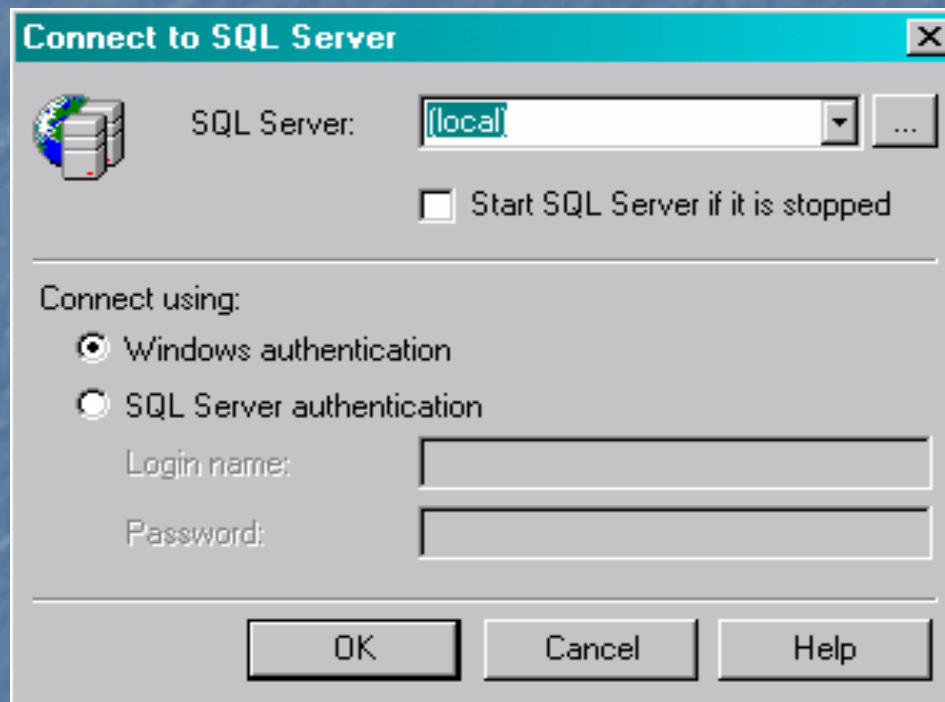
# Profiler – SQL Server



**Figura 2** Criando um trace no Profiler

# Profiler – SQL Server

O próximo passo será fornecer uma conta com privilégios de system administrator (SysAdmin) para realizar a trace (veja a Figura 3).

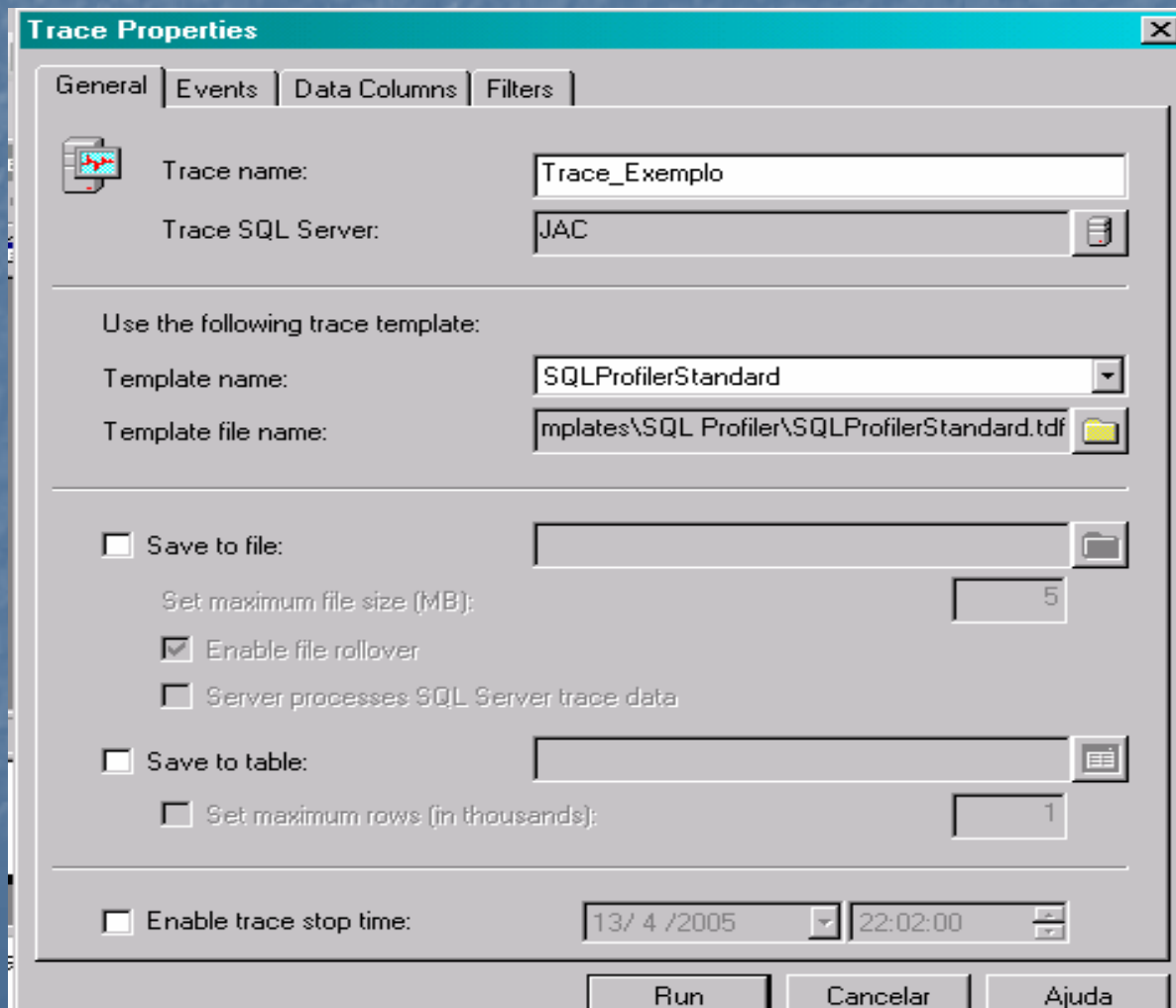


**Figura 3** Tela de logon do servidor onde será realizada a trace.



# Profiler – SQL Server

Considerando que você tenha os privilégios necessários e tendo efetuado a autenticação com sucesso, a tela para configuração Gerais da trace é apresentada (ver Figura 4).



**Figura 4**  
Configuração  
Geral da  
Trace.

# Profiler – SQL Server

As opções disponíveis são:

- **Trace Name:** nome da trace;
- **Trace SQL Server:** identificação do servidor onde a trace está sendo executada;
- **Template Name:** nome do modelo da trace. Quando criamos uma trace, selecionamos determinados tipos de eventos que desejamos analisar. Para que não precisemos informar sempre os mesmos eventos ao criar uma nova trace, salvamos modelos chamados templates. Existem alguns templates pré-definidos, o SQLProfilerStandard é um deles. A **Tabela 1** fornece uma descrição resumida dos templates existentes.
- **Template file Name:** caminho do arquivo de template utilizado, cuja extensão é .TDF;
- **Save to file:** grava o resultado da trace num arquivo em disco com extensão .TRC;
- **Set maximum file size:** informa o tamanho máximo do arquivo em disco gerado pela trace. Ao atingir esse limite a gravação em arquivo é suspensa mas o monitoramento continua ativo na tela do Profiler.

# Profiler – SQL Server

As opções disponíveis são (continuação):

- **Enable file rollover:** se o rollover estiver habilitado e o arquivo atingir o limite definido em Set maximum file size(MB), o arquivo em disco será reiniciado. Neste caso, perde-se o que foi registrado em arquivo até esse momento.
- **Server process SQL Server trace data:** se você algum dia se deparar com a linha de texto em sua trace “... Some events may have been lost...”, isto quer dizer que o servidor está muito ocupado e optou por não enviar alguns comandos para sua trace para ganhar algum fôlego de processamento. Habilitando essa opção, você estará forçando o servidor a enviar todos os comandos processados para a trace, mesmo causando perda de performance. É recomendado não utilizar.
- **Save to table:** grava o resultado da trace numa tabela. É mais fácil de depurar, pós podemos colocar filtros ou ordenar da maneira que achamos mais interessante.
- **Set maximum rows (in thousands):** limita o número de linhas na tabela originada pela trace.
- **Enable trace stop time:** estabelece prazo limite para término da trace.

# Profiler – SQL Server

T1

<b>Nome do Template</b>	<b>Para que serve</b>
<b>SQLProfileStandard</b>	Trace genérica; rastreia comandos executados com sucesso no servidor.
<b>SQLProfileTSQL</b>	Utilize para visualizar os comandos T-SQL startados no servidor.
<b>SQLProfileTSQL_Duration</b>	Utilize para obter uma trace de comandos processados no servidor ordenados por tempo de execução.
<b>SQLProfileTSQL_Group</b>	Lista os comandos startados com sucesso no servidor, ordenados por ApplicationName, NTUserName e ClientProcessId
<b>SQLProfileTSQL_Replay</b>	Utilizada para gravação de traces para posterior replay.
<b>SQLProfile_SPs</b>	Utilizado para visualização dos comandos T-SQL executados internamente nas sp's.
<b>SQLProfileTuning</b>	Utilizado na geração de eventos para posterior análise pelo Index Tuning Wizard.

# Profiler – SQL Server

## Guia events

A guia Events (ver Figura 5), apresenta uma relação de todas as classes de eventos que podem ser monitorados num servidor de banco de dados SQL Server 2000. Nesse contexto, classes são agrupamentos de eventos que possuem uma característica em comum: Temos uma para controlar execução de procedures, outra para gerenciamento de locks, etc. o template SQLProfilerStandard, por exemplo, seleciona automaticamente alguns eventos vistos na **Tabela 2**.

<b>Classe</b>	<b>Evento</b>	<b>Para que serve</b>
Security Audit	Audit Logon	Auditar abertura de sessões no banco.
	Audit Logoff	Auditar encerramento de sessões no banco.
Sessions	Existent Connections	Lista todas as conexões ativas no banco no momento em que a trace é iniciada.
Stored Procedures	RPC:Completed	Lista a execução de sp's originadas por conexões remotas (ADO, ODBC, OLEDB etc).
TSQL	SQL:Batch Completed	Lista as queries executadas fora do contexto de uma stored procedure.

# Profiler – SQL Server

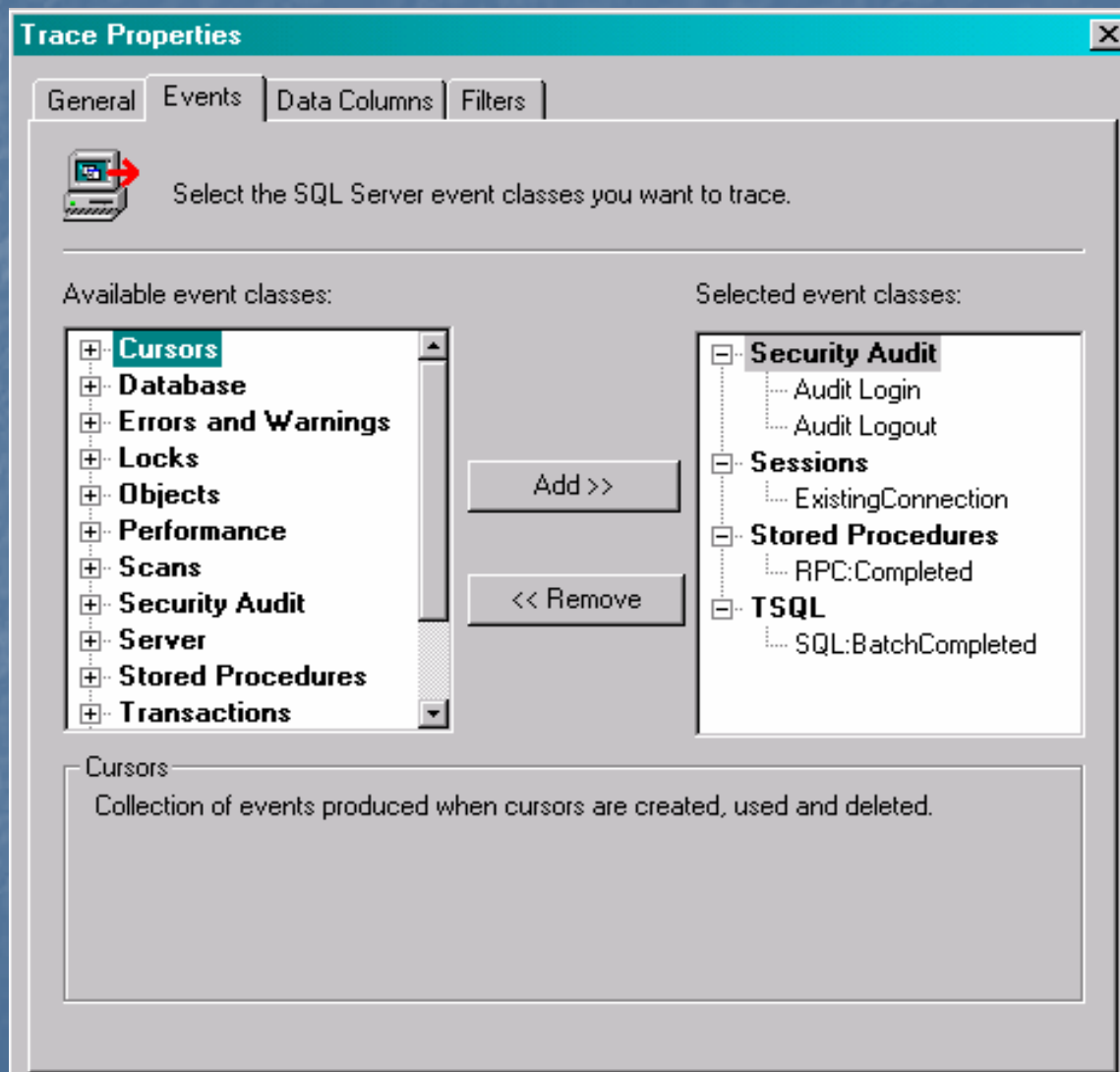
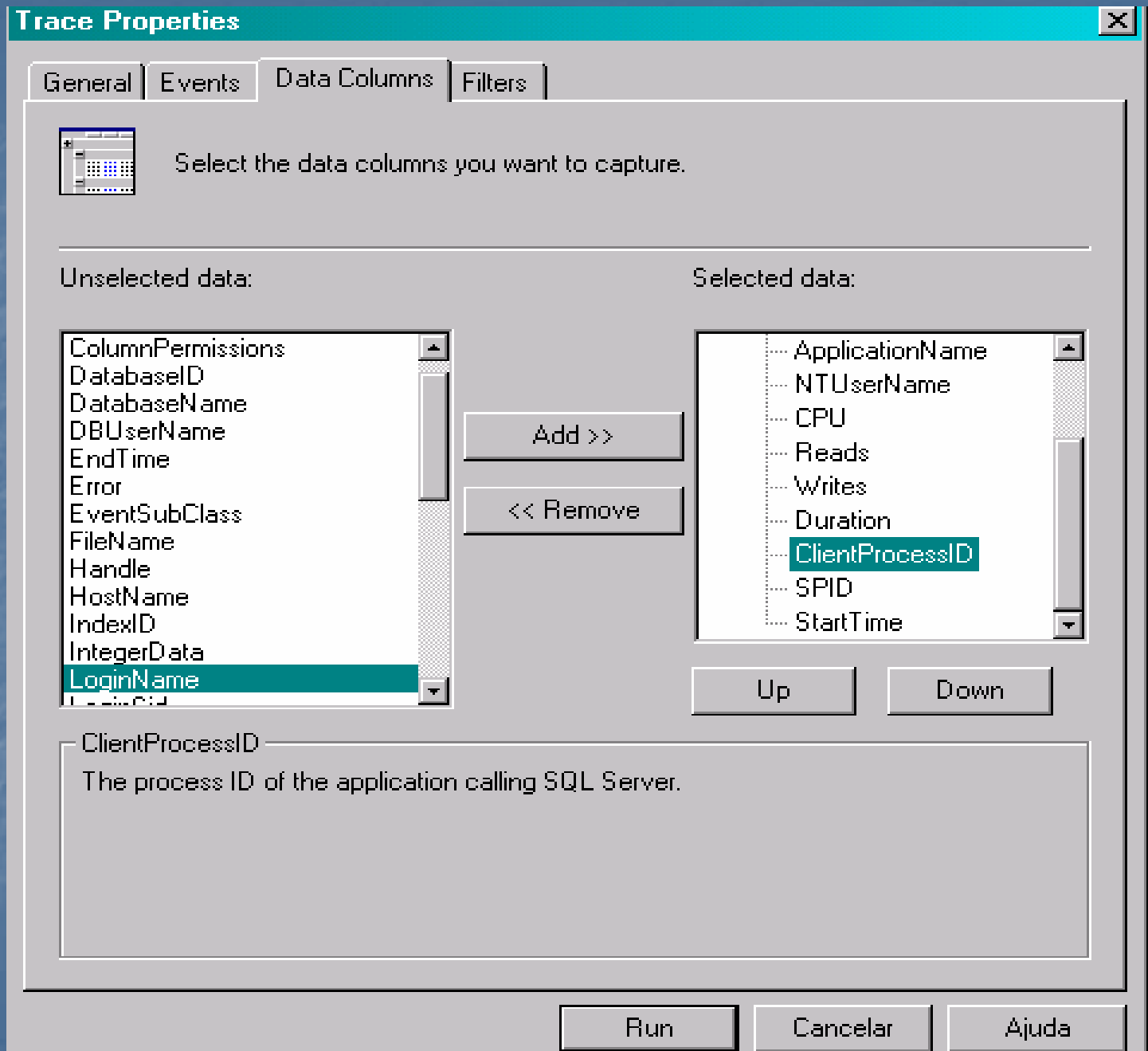


Figura 5  
Guia  
Events

# Profiler – SQL Server

## Guia events

A próxima etapa será definir que tipo de informação queremos visualizar na trace. O template SQLProfileStandard já seleciona uma série de colunas, mas para deixar a tela do Profiler mais “enxuta”, excluiremos as colunas “LoginName” e a coluna ClientProcessId. Clique na coluna e depois em << Remove (veja Figura 6).



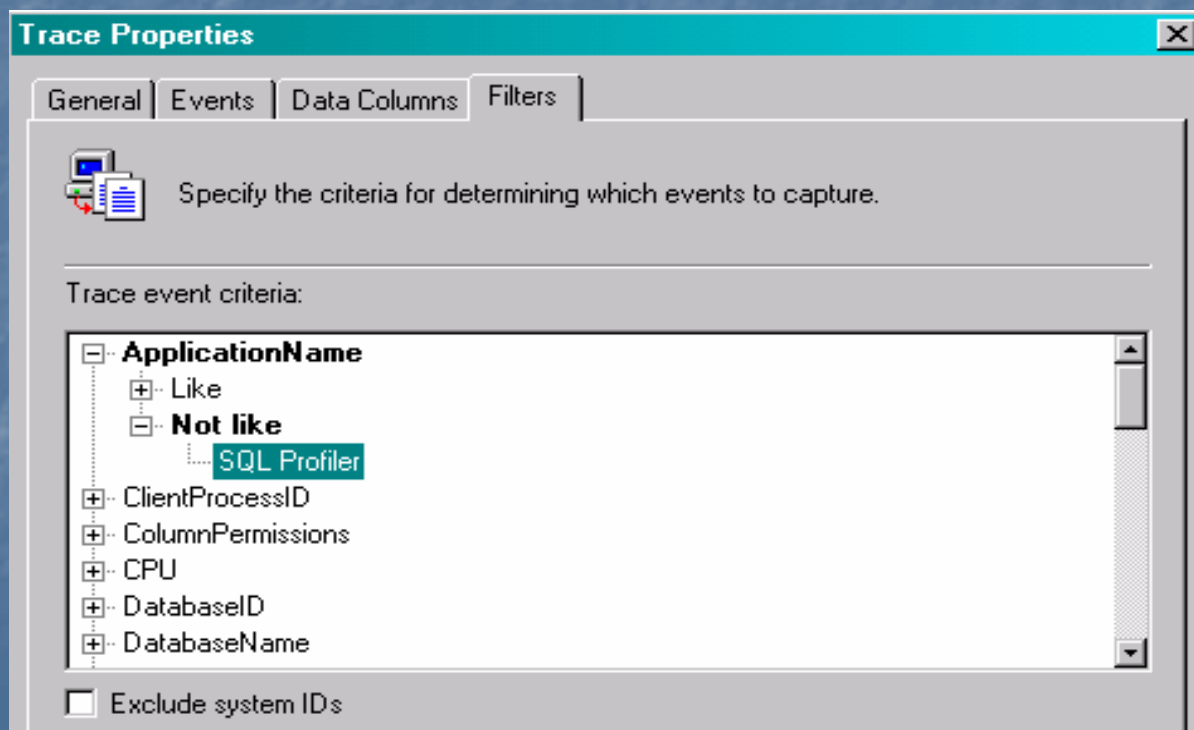
**Figura 6**  
Selecionando as  
colunas  
que serão  
visualizadas  
no Profiler.



# Profiler – SQL Server

## Guia Filters

Finalmente a definição da trace na guia Filters (veja Figura 7), utilizada para refino da trace. A aplicação Profiler para mostrar na tela os comandos processados pelo servidor SQL Server, é responsável por uma série de comandos que são também processados pelo engine do banco. Para evitar que esses comandos apareçam na trace, ligamos o filtro ApplicationName Not Like “Profiler”.



**Figura 7** Criando filtros para trace.

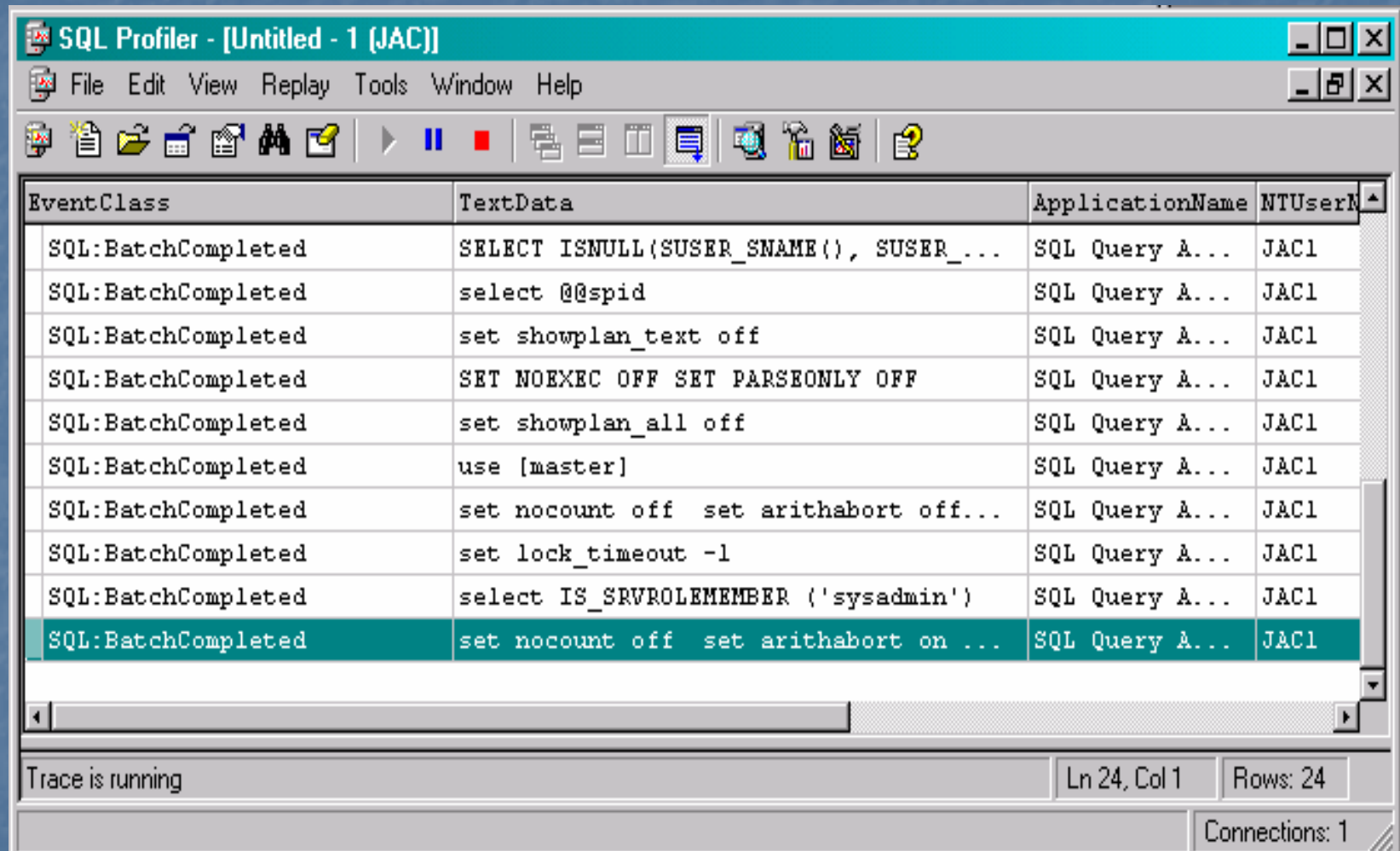
# Profiler – SQL Server

## Guia Filters

Filtros são utilizados para limitar os eventos rastreados na trace, reduzindo o número de linhas afetadas, facilitando nossa compreensão e melhorando o foco de nossa análise. Poderíamos, por exemplo, filtrar os comandos filtrados por um determinado spid. Se desejássemos analisar a execução de uma stored procedure em particular, poderíamos concentrar nossa análise somente na execução dessa sp, utilizando também os recursos de filtros (nesse caso, o filtro ObjectId armazenaria o Id da trace que queremos analisar).

Concluído o processo de definição, clique em RUN para iniciar a trace (ver **Figura 8**).

# Profiler – SQL Server



The screenshot shows the SQL Profiler window titled "SQL Profiler - [Untitled - 1 (JAC)]". The interface includes a menu bar (File, Edit, View, Replay, Tools, Window, Help) and a toolbar with various icons for file operations and playback. The main area displays a table of trace events. The table has four columns: EventClass, TextData, ApplicationName, and NTUserName. The data shows a sequence of SQL batch completion events for a user named JAC1, including queries for ISNULL, @@spid, showplan settings, NOEXEC, lock\_timeout, and IS\_SRVROLEMEMBER. The status bar at the bottom indicates "Trace is running", "Ln 24, Col 1", "Rows: 24", and "Connections: 1".

EventClass	TextData	ApplicationName	NTUserName
SQL:BatchCompleted	SELECT ISNULL(SUSER_SNAME(), SUSER_...	SQL Query A...	JAC1
SQL:BatchCompleted	select @@spid	SQL Query A...	JAC1
SQL:BatchCompleted	set showplan_text off	SQL Query A...	JAC1
SQL:BatchCompleted	SET NOEXEC OFF SET PARSEONLY OFF	SQL Query A...	JAC1
SQL:BatchCompleted	set showplan_all off	SQL Query A...	JAC1
SQL:BatchCompleted	use [master]	SQL Query A...	JAC1
SQL:BatchCompleted	set nocount off set arithabort off...	SQL Query A...	JAC1
SQL:BatchCompleted	set lock_timeout -1	SQL Query A...	JAC1
SQL:BatchCompleted	select IS_SRVROLEMEMBER ('sysadmin')	SQL Query A...	JAC1
SQL:BatchCompleted	set nocount off set arithabort on ...	SQL Query A...	JAC1

Figura 8 Tela para monitoramento do Profiler










# Profiler – SQL Server

Um esumo do significado das colunas apresentadas na Figura 8 é apresentado a seguir:

- **EventClass**: os eventos rastreados pelo Profiler são agrupados em classes.
- **TextData**: utilizada para visualização do dado coletado na trace. Essa coluna depende do tipo de evento capturado (TCP/IP, evento de conexão).
- **ApplicationName**: nome da aplicação;
- **LoginName**: login do usuário responsável pela execução do comando;
- **CPU**: tempo consumido de CPU para execução do comando (milissegundos);
- **Reads**: número de páginas lidas em memória para executar o comando;
- **Writes**: número de páginas gravadas pelo comando;
- **Duration**: duração do comando(em milissegundos);
- **SPID**: identificação da sessão no SQL Server;
- **Start Time**: horário do início da execução do comando.

# Profiler – SQL Server

Através desta interface é possível:

- Parar a trace: Para isto clique no botão 
- Iniciar a trace: Para isto clique em 
- Iniciar uma nova trace, efetuando toda a parametrização novamente: Para isto clique em 
- Trocar o Template SQLProfilerStandard: Para isto clique em 
- Carregar uma trace previamente gravada em arquivo .TRC: Para isto clique no ícone 
- Carregar uma trace gravada em uma tabela no banco: Para isto clique em 
- Acessar a tela de configurações gerais da trace: Para isto utilize o ícone de propriedades 
- Procurar por uma determinada string na trace que você acabou de gerar: Para isto utilize o binóculo 
- Efetuar uma limpeza na tela: Para isto utilize a borracha 

# Profiler – SQL Server

**Agora um teste prático. Com o Profiler ativo, abra uma sessão no Query Analyzer e execute a seqüência de comandos a seguir:**

Use Northwind

Go

Create procedure stp\_Mostrar\_Pedido (@OrderId int0

As

```
select O.OrderId, O.CustomerId, O.EmployeeId, d.ProductId, d.UnitPrice,  
d.Quantity
```

```
from Orders O inner join [Order Details] d
```

```
on O.OrderId = d.OrderId
```

```
Where O.OrderId = @OrderId
```

Return

go

# Profiler – SQL Server

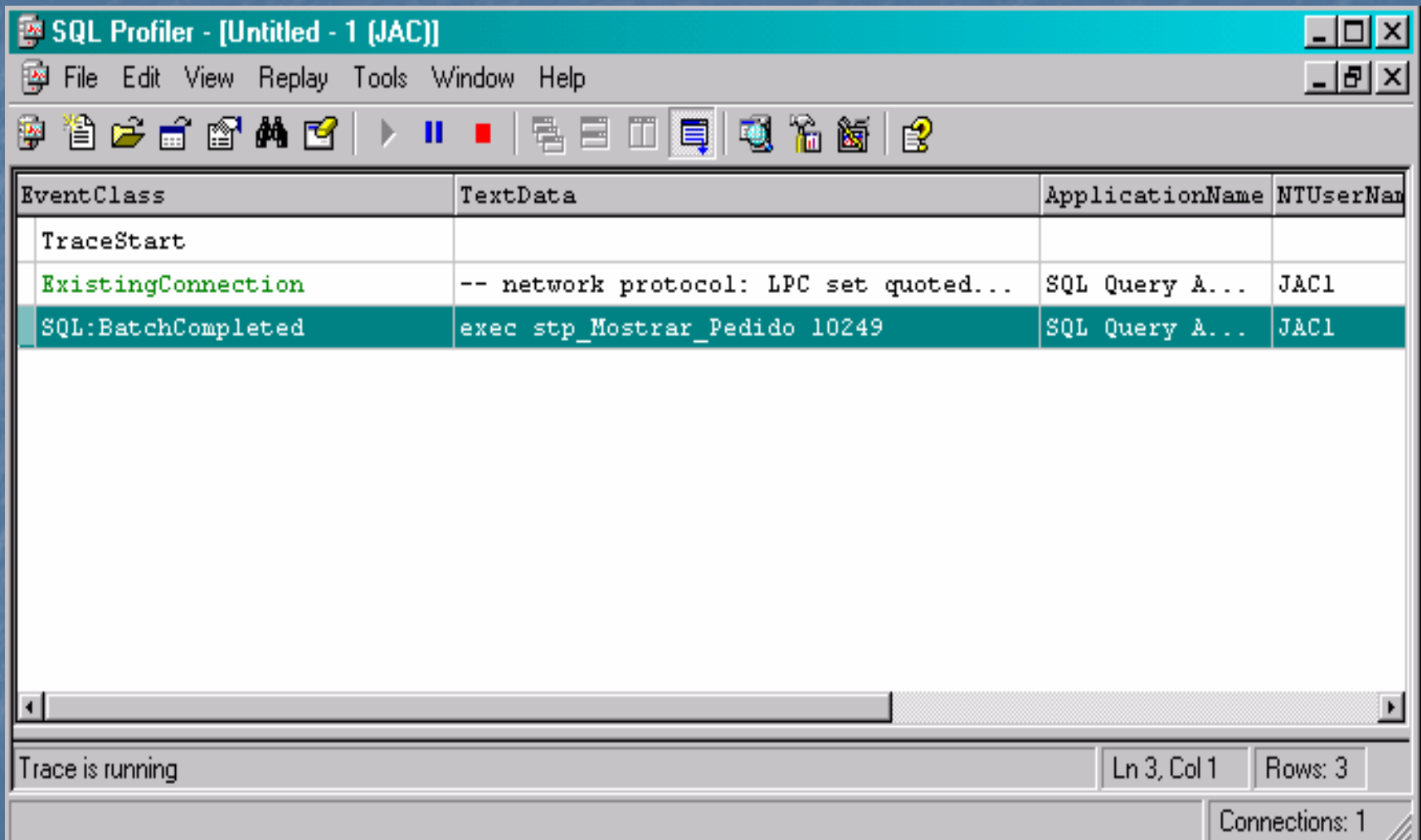
Agora um teste prático. Com o Profiler ativo, abra uma sessão no Query Analyzer e execute a seqüência de comandos a seguir:

```
Exec stp_Mostrar_Pedido 10249
```

```
go
```

Dirija-se ao profiler e confirme o resultado (veja a **Figura 9** no slide seguinte).

# Profiler – SQL Server



The screenshot shows the SQL Profiler interface with a table of event traces. The table has four columns: EventClass, TextData, ApplicationName, and NTUserName. The traces include TraceStart, ExistingConnection, and SQL:BatchCompleted. The status bar at the bottom indicates 'Trace is running', 'Ln 3, Col 1', 'Rows: 3', and 'Connections: 1'.

EventClass	TextData	ApplicationName	NTUserName
TraceStart			
ExistingConnection	-- network protocol: LPC set quoted...	SQL Query A...	JAC1
SQL:BatchCompleted	exec stp_Mostrar_Pedido 10249	SQL Query A...	JAC1

Trace is running      Ln 3, Col 1      Rows: 3      Connections: 1

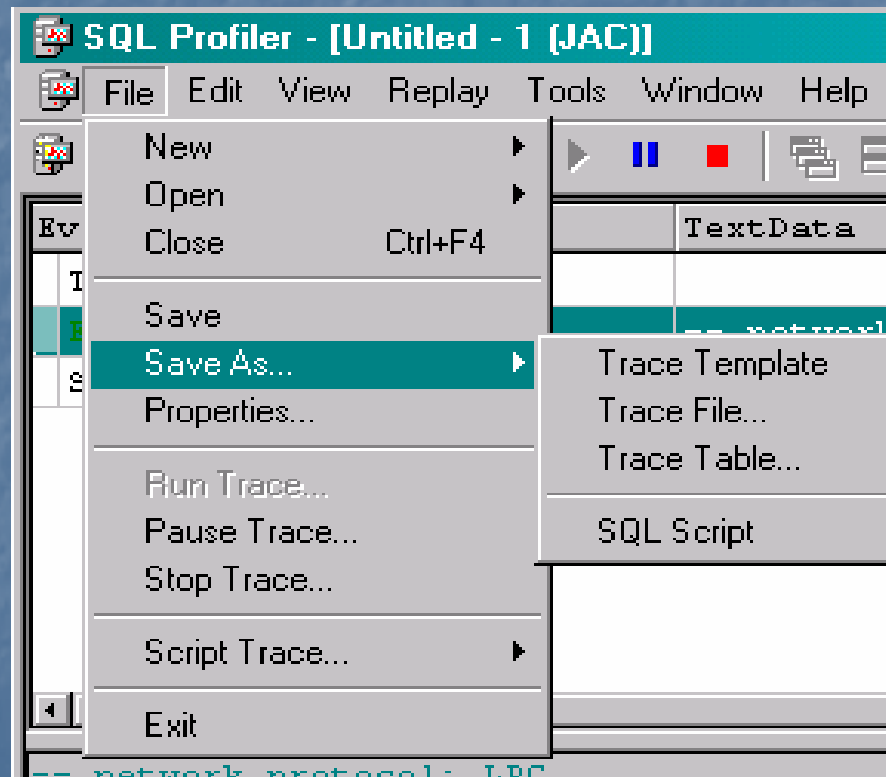
**Figura 9** Resultado da execução de comandos na console do profiler.



# Profiler – SQL Server

## Gravando a Trace

Para gravar a trace, siga até a opção File da barra de menu e escolha Save AS (ver **Figura 10**).



**Figura 10**  
Salvando a  
trace

# Profiler – SQL Server

## Gravando a Trace

As opções disponíveis para salvamento da trace são:

- **Trace Template:** utilize para gerar um template (arquivo com extensão .tdf, de Template Data File);
- **Trace File:** salva um arquivo em disco com extensão .trc com o resultado da trace;
- **Trace Table:** armazena o resultado da trace em uma tabela.
- **SQL Script:** gera um arquivo texto (extensão .sql) com o lote de comandos T-SQL necessários para criar e executar a trace.

# Profiler – SQL Server

## Conclusão

Quando o assunto é tuning, o Profiler é uma ferramenta indispensável. Na próxima aula continuaremos esse assunto e nos aprofundaremos nos principais eventos que devem ser analisados, tendo em vista a otimização de processos. Até lá!