

Aplica-se ao Caché	5.0.x
Pré-requisito para leitura	Noções de banco de dados
Plataforma	Windows
Data	15/07/2003
Versão	1.10

## Conhecendo o Caché

*Os sistemas de bancos de dados atuais estão fadados ao desaparecimento, visto que se baseiam em uma visão de mundo que está cada vez mais longe do mundo real que conhecemos e, ainda assim, não se prepararam para mudanças.*

*Amir Samary  
Sales Engineer*

### Introdução

Os bancos de dados relacionais atuais estão enfrentando um grave problema. O mundo está ficando cada vez mais complexo e a Orientação a Objetos na interface das aplicações não está mais sendo suficiente para modelar esta complexidade. Os bancos de dados relacionais estão começando se tornar um problema. A cada momento, mais ferramentas de mapeamento objeto-relacional estão surgindo. Os fabricantes de bancos de dados relacionais estão tentando implementar extensões proprietárias aos seus bancos para suportar de forma tosca o conceito de classes e métodos. *Frameworks* como o *J2EE* estão cada vez mais se tornando uma necessidade. *Design Patterns* estão sendo necessários para garantir qualidade, gerenciar a complexidade e manter a clareza e a performance das aplicações. Sistemas construídos em cima de sistemas. Camadas em cima de camadas.

O problema desta abordagem é que eles continuam tentando aproveitar ao máximo o investimento que eles já fizeram em seus produtos. No final das contas, você continua com um banco de dados relacional. Mas as camadas que foram acrescentadas entre o seu banco de dados relacional e a sua aplicação vão tornar a sua solução mais complexa, cara e exigente em termos de hardware do que ela jamais foi.

Um sonhador poderia dizer que tudo poderia ser mais simples se tivéssemos um banco de dados orientado a objetos que nos permitisse acessar os dados utilizando *SQL ANSI* também. Pois o Caché está aí para suprir esta necessidade e muitas outras. Quem acredita que o futuro está nos objetos, continue lendo este tutorial. Quem não acredita, sinto muito, muito mesmo.

Neste tutorial, tentarei explicar um pouco da filosofia do Caché e como ele pode tornar a vida dos desenvolvedores extremamente mais fácil. É apenas um produto, que faz o equivalente a vários outros produtos juntos. Abordarei apenas o acesso via *ODBC* de maneira bem simples. Em tutoriais futuros, falarei das outras tecnologias.

## Um pouco sobre o Caché

Os problemas descritos acima não existem no mundo do Caché. Sua arquitetura Pós-relacional prevê mudanças radicais na forma como os dados podem ser acessados. Não existem camadas desnecessárias.

Os bancos de dados relacionais utilizam estruturas *multi-dimensionais* internamente para armazenar os dados. O problema é que eles tornaram estas estruturas inacessíveis ao desenvolvedor. No Caché, você pode ver estas estruturas e acessá-las diretamente se for preciso. Esta arquitetura aberta é o grande diferencial do Caché.

A proposta da arquitetura do Caché é armazenar os dados da forma mais natural possível na estrutura multi-dimensional do banco, e fornecê-los para as aplicações de acordo com a necessidade, isto é, na forma de objetos ou na forma de registros de uma tabela. Se, no futuro, uma nova forma de se manipular dados surgir, o Caché será capaz de fornecer os mesmos dados nesta nova forma, sem perder desempenho com mapeamentos de um modelo para outro.

O lema é:

*"Relacional para o legado, Objetos para o presente e Arquitetura para o Futuro."*

O *kernel* do Caché é totalmente desenvolvido em ANSI C com otimizações em *assembly* para cada uma das plataformas onde o banco é suportado. As ferramentas de administração do Caché são desenvolvidas em C++. O *kernel* vem sendo desenvolvido e melhorado há 25 anos. Apenas em 1995 a InterSystems liberou o Caché como o conhecemos hoje para o mercado. Antes disto, ele era um banco *multi-dimensional* embarcado (semelhanças com a história do Java são meras coincidências) muito utilizado nos mercados de *healthcare* e governos mundiais. A InterSystems rebatizou o produto e alterou completamente sua arquitetura.

O Caché roda em qualquer plataforma, a não ser AS400 e Mainframe. Possui uma máquina virtual em seu interior que vem sendo melhorada e otimizada há 25 anos. Desta forma, aplicações desenvolvidas em Caché para uma plataforma podem rodar em outra plataforma sem quaisquer alterações.

Dentro do Caché, podemos programar utilizando:

- SQL ANSI 92
- Caché Basic (semelhante ao VBScript da Microsoft)
- Caché Object Script

Não importa o que você escolha para programar seus métodos, *triggers* ou *procedures*, no final tudo será compilado para o mesmo *pcode* multiplataforma que será executado pela máquina virtual do Caché.

Mas o Caché não é só um banco de dados Pós-relacional, ele também é um servidor de aplicações completo. Capaz de servir páginas por si só ou através de ASP, JSP, ColdFusion, etc... É claro que aplicações cliente servidor também são construídas com o Caché com muito mais facilidade graças à orientação a objetos.

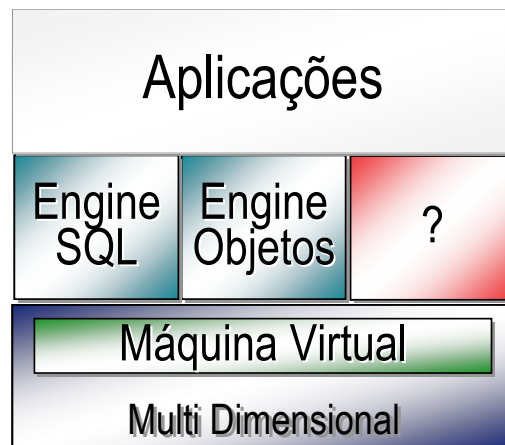


Figure 1

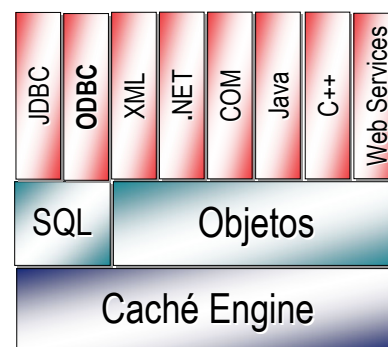


Figura 1

Na figura acima você pode ver a quantidade de alternativas possíveis para se conectar ao Caché. Cada uma delas poderá ser explorada melhor em outros tutoriais desta série.

Algumas coisas valem a pena ressaltar para despertar a curiosidade:

- O Caché é capaz de instanciar objetos *ActiveX* dentro do banco como se fossem objetos dele (*Caché Activate*).
- O Caché é capaz de montar tabelas de outros bancos de dados (MySQL, Oracle, SQL-Server, etc...) como se fossem dele, e ainda oferece uma visão orientada a objetos para elas (*Caché Database Gateway*). Este recurso permite que montemos bases de dados *Federalizadas*, ideais para integração de dados de diversas fontes diferentes.
- O Caché é capaz de entender e importar *DDL* de diversos bancos como Oracle, SQL-Server, Informix, Sybase, DB2, etc...
- O Caché exporta cada classe de seu banco de dados como classes *C++*, *Java* ou *EJB* (*Projection/Java Binding*).
- O Caché exporta cada classe de seu banco de dados como componentes *ActiveX* através de um *Factory* que pode ser utilizado em qualquer ambiente de desenvolvimento (VB, Delphi, PowerBuilder, etc....).
- O Caché pode ser utilizado em *clusters* de qualquer tipo e servir como *framework* para implementação de *grids* graças ao *Enterprise Caché Protocol* (ECP).
- O Caché pode ser utilizado para desenvolver aplicações WEB usando qualquer tecnologia conhecida (JSP, ASP, etc...), mas oferece também uma linguagem semelhante, em arquitetura, ao JSP chamada CSP (*Caché Server Pages*) que torna tudo mais fácil e mais rápido. As páginas são arquivos texto como JSP, mas são compiladas e executadas dentro do banco de dados, dispensando conexões ao banco e dando muito mais performance e escalabilidade. Os *scripts* das páginas podem ser codificados utilizando-se *Caché Object Script* ou *Caché Basic*. Codificar uma página CSP em *Caché Basic* é como codificar páginas ASP de forma mais orientada a objetos.
- O Caché possui um servidor interno para HTTP e para *WebServices*. Possui também clientes para POP3, SMTP, HTTP e *WebServices*. Além de conectores para o MQSeries da IBM.
- O Caché exige muito pouco hardware e recursos de seu sistema para rodar. É comum aproveitar-se hardwares antigos para rodar o Caché com excelente performance.

Bom, chega por enquanto. Existe muita coisa para ser dita. Em outros tutoriais veremos cada uma destas tecnologias. Para terminar, é importante deixar claro que a meta da InterSystems é fazer de tudo para que o desenvolvedor e o administrador não percam seu tempo. O conceito de RAD (*Rapid Application Development*) fez e faz muito sucesso na construção de interfaces para o usuário. A InterSystems trouxe o conceito de RAD para o lado do servidor:

*"Make applications faster!"*

## Instalando e Conhecendo o Caché

A InterSystems permite que você teste e conheça o Caché de duas maneiras diferentes. A InterSystems disponibiliza uma cópia demo do Caché para testes e avaliação por tempo indeterminado. Chamamos esta cópia demo de **Caché PC**. A InterSystems também permite que você utilize o Caché com uma licença demo que expirará depois de alguns dias (normalmente um mês ou dois).

A diferença entre as duas formas pode ser vista abaixo:

	<b>Caché PC</b>	<b>Caché com Licença Demo</b>
<b>Licença</b>	Não precisa de licença. Mesmo que se coloque uma licença oficial, o produto continuará se comportando como um Caché PC (veja "Conexões" abaixo).	Precisa de licença. Expira dentro de um prazo estipulado na licença. Em verdade, você estaria trabalhando com o Caché completo, com uma licença que vai expirar em pouco tempo.
<b>Conexões</b>	Não permite conexões (ODBC, JDBC, CSP, etc...) <b>que venham de outras máquinas da rede.</b>  Permite conexões que venham da mesma máquina onde o Caché está instalado (mesmo IP).	Permite, de acordo com a licença utilizada.  Por exemplo, se você solicitar para a InterSystems uma licença Demo <i>Enterprise</i> do Caché, você poderá testar até <i>clusters</i> utilizando ECP ( <i>Enterprise Caché Protocol</i> ) dentro do prazo de expiração da licença.  Por isto, é importante dizer para a InterSystems o que se pretende testar com esta licença demo, de forma que esta possa fornecer uma licença apropriada.
<b>Onde conseguir</b>	O CD Demo do Caché pode ser solicitado através de nosso Call Center:  <b>0800 888 22 00</b>  Você também pode baixar o Caché da Internet, através de nosso site (cerca de 90Mb).	O CD do Caché, assim como uma licença demo, podem ser solicitados através de nosso Call Center:  <b>0800 888 22 00</b>

**Se você estiver com o CD Demo** de instalação do Caché, basta inseri-lo em seu *drive* e aguardar que o *autorun* de seu Windows abra uma página em seu Internet Explorer com um link **Install Caché**. Clique neste link, para chegar na pasta **\windows** de seu *drive* de CD. Nesta pasta, você encontrará o arquivo **setup.exe**, que você poderá executar. Se o *autorun* de seu Windows estiver desativado por algum motivo, você pode entrar na pasta **\windows** utilizando o seu Windows Explorer e executar o **setup.exe** normalmente. Se solicitado, diga que você entrará com uma licença mais tarde.

**Se você baixou o Caché da Internet**, copie o arquivo baixado para uma pasta temporária e execute-o. O arquivo começará a expandir-se na pasta. Quando o processo de expansão terminar, execute o arquivo **setup.exe**. Se solicitado, diga que você entrará com uma licença mais tarde.


**Se você solicitou uma licença demo do Caché**, e está com o CD oficial de instalação, basta executar o arquivo **setup.exe** que se encontra na raiz de seu *drive* de CD. Durante o processo de instalação, insira a licença quando solicitado.


Aceite as configurações padrão, clicando em **Next** em cada tela.

Esta instalação altera o seu sistema da seguinte maneira:

Condição	Alteração
Se você tem o Visual Basic 6 Instalado em sua máquina.	O Caché acrescenta um <i>Wizard</i> para criação automática de formulários VB baseados em classes no banco. E um novo tipo de projeto ( <i>Caché Project</i> ) que pode ser utilizado como base para se construir aplicativos VB que acessem o Caché via <i>Factory</i> de objetos.
Se você tem o Rational Rose instalado.	O Caché acrescenta um <i>Wizard</i> para importar/exportar classes de/para o banco.
Sempre	O Caché cria duas conexões de ODBC: Um DSN para o <i>namespace</i> USER e um DSN para o <i>namespace</i> SAMPLES.
Se você tem o DreamWeaver instalado.	O Caché acrescenta um <i>Wizard</i> para a criação automática de páginas CSP baseadas em classes de seu banco.
Se você tem o IIS instalado	O Caché configura um novo site no IIS apontando para suas páginas CSP ( <i>Caché Server Pages</i> ) de exemplo.
Se você tem o Apache instalado e está instalando em uma máquina <b>Linux</b>	O Caché configura o seu apache para redirecionar chamadas para páginas CSP para ele.

Após a instalação, você verá um pequeno cubo azul no canto inferior direito de sua tela, no *System Tray* de seu Windows:

 Este cubo está azul pois o Caché já está no ar.

 Já este, se encontra acinzentado. Neste caso, o Caché está fora do ar.

Clicando com o mouse neste cubo, você verá as seguintes opções:

Opção no Cubo	Descrição
<b>Começando a usar o Caché</b>	O Caché tem que estar no Ar para que esta opção possa ser chamada. Ela abre um Internet Explorer com uma série de informações úteis para quem está começando, como tutoriais e materiais de referência.
<b>Iniciar/Parar o Caché</b>	Permite que você possa colocar ou tirar o Caché do Ar.
<b>Caché Studio</b>	É o IDE de desenvolvimento do Caché. Aqui você poderá criar suas classes, páginas CSP e rotinas, tudo organizado em projetos criados pelo usuário. Além disto, você poderá exportar seu projeto inteiro para um arquivo XML ou apenas parte dele.
<b>Caché Terminal</b>	É o equivalente ao SQL*Plus do Oracle. Só que aqui você utiliza o COS ( <i>Caché Object Script</i> ) para interagir com o Caché, ao invés do PL/SQL. Você também pode utilizar <i>SQL ANSI</i> no <i>Caché Terminal</i> , mas é preciso chamar o <i>Shell</i> de SQL antes:

	<p><b>USER&gt; Do \$system.SQL.Shell() [ ENTER ]</b></p> <p>Depois, basta escrever seus comandos SQL em uma ou mais linhas. No final de cada comando, digite <b>Go</b> e tecla <b>Enter</b>. Para sair, digite <b>Exit</b>.</p>
<b>Caché Explorer</b>	<p>Com o Caché Explorer, você consegue ver os <i>databases</i> e <i>namespaces</i> do Caché. Através dele, você consegue olhar para a estrutura <i>multidimensional</i> do banco diretamente (as Globais). Normalmente, cada classe <i>persistente</i> criada no Caché utiliza no máximo três globais para armazenamento de seus dados, índices e <i>streams</i>.</p>
<b>Gerenciador SQL</b>	<p>É a contrapartida do Caché Studio. Enquanto no Studio trabalha-se com classes, no Gerenciador de SQL trabalha-se com tabelas e <i>SQL ANSI</i>. Permite criar usuários no banco, regras (roles) e definir visualmente o acesso dos usuários às tabelas e views do namespace em uso.</p>
<b>Painel de Controle</b>	<p>É uma ferramenta de administração que permite, dentre outras coisas, que se faça e gerencie:</p> <ul style="list-style-type: none"> <li>• <i>Backups</i></li> <li>• <i>Shadows</i></li> <li>• <i>Journaling</i></li> <li>• Processos</li> <li>• Etc...</li> </ul>
<b>Gerenciador de Configuração</b>	<p>É uma ferramenta de administração através da qual criamos <i>databases</i> e <i>namespaces</i> e configuramos vários aspectos do Caché, tais como <i>buffers</i> de dados e <i>rotinas</i>, servidores ECP, ajustes finos e etc...</p>
<b>Documentação</b>	<p>O Caché tem que estar no ar para que esta opção seja usada. Ela abre o Internet Explorer exibindo toda a documentação do produto.</p>
<b>Acesso a Sistemas Remotos</b>	<p>Permite que você execute cada uma das ferramentas do Caché em outros servidores Caché diferentes do seu.</p>
<b>Servidor Preferencial</b>	<p>Permite que você cadastre outros servidores Caché aos quais você gostaria de se conectar e trabalhar. Permite também que você defina qual é o seu servidor Caché preferencial (normalmente o LOCALTCP, que é o Caché instalado em sua máquina).</p>
<b>Sobre o Caché</b>	<p>Mostra uma <i>splash screen</i> com a versão e o <i>build</i> de seu Caché.</p>

Dê uma olhada na documentação do produto clicando com o botão direito do mouse sobre o Cubo do Caché e escolhendo a opção **Documentação**. Ela é bastante completa e repleta de tutoriais excelentes!

## O que são namespaces?

Quando o Caché é instalado, ele cria uma série de *namespaces*:

NAMESPACE	Descrição
USER	<i>Namespace</i> utilizado para se criar aplicações de teste e exemplos.
SAMPLES	<i>Namespace</i> que vem com uma série de classes utilizadas nas aplicações de exemplo que acompanham o Caché.
%SYS	<i>Namespace</i> utilizado pelo Caché para diversas tarefas do sistema.

*Namespaces* são como os *tablespaces* do Oracle, só que com algumas diferenças. Cria-se um *namespace* para se armazenar classes/tabelas, dados, rotinas e páginas CSP compiladas (classes). Abaixo apresentamos um esquema de como o Caché organiza seus namespaces, considerando que você instalou o seu Caché na pasta **c:\CacheSys**:

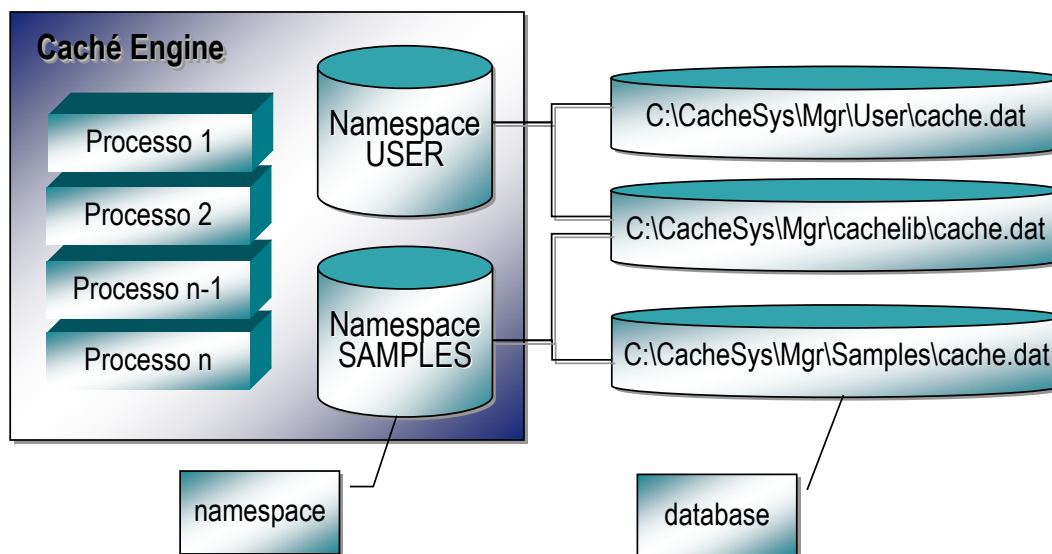


Figura 2

Repare que existe um arquivo em disco chamado **cache.dat** para cada *namespace* de seu sistema. Nós chamamos estes arquivos em disco de **databases**. Todos eles devem ter o mesmo nome (**cache.dat**), mas estar em pastas diferentes. Cada *database* pode ter até 32Tb e quando alcança-se este limite, pode-se criar extensões, cada uma com mais 32Tb (limitados apenas pelo sistema operacional subjacente).

Todo namespace do Caché é composto de vários databases:

- Um *database* para rotinas
- Um *database* para dados (globais)

- Um *database* especial para dados de sistema (globais de sistema – CACHESYS)
- Um *database* especial para rotinas de sistema (CACHESYS)
- Um *database* especial para armazenamento temporário (CACHETEMP)
- Um *database* especial para a biblioteca de classes do Caché (CACHELIB)

É muito comum que o mesmo *database* seja utilizado para suas rotinas e dados. Os outros quatro *databases* normalmente não mudam e são os que foram mencionados acima. **Portanto, quando cria-se um novo namespace, é normal criar-se apenas um único novo database para ele.** O Caché assume o resto como explicado acima. Não vamos criar novos *namespaces* neste tutorial. Vamos utilizar um que já vem com o Caché (USER). Mas seria interessante que você navegasse pelo **Gerenciador de Configuração** do Caché e clicasse na aba **Namespaces** e na aba **Databases** para entender um pouco mais a organização das coisas.

O *namespace* %SYS (que não aparece na figura acima), não foge a regra. Mas alguns utilitários do Caché só funcionam quando são chamados neste *namespace*. Não se preocupe com isto. Estes utilitários são utilitários caractere, acessados via **Caché Terminal** e dispensáveis para quem utiliza as ferramentas gráficas do Caché. Apenas em uma emergência, quando não é possível utilizar-se as ferramentas gráficas do Caché, utilizamos os utilitários caractere.

## Criando uma Tabela/Classe de Exemplo

Vamos criar uma classe no Caché que representará um cadastro de professores de uma Universidade:

Atributo	Tipo
Nome	%String
Matrícula	%Integer (Obrigatória e Única)

O nome de nossa classe será **Professor**. Vamos criá-la no pacote **universidade**. Para tanto, certifique-se de que o seu Caché está no Ar (o pequeno cubo em seu *System Tray* deve estar azul) e clique com o botão direito nele, clicando em seguida na opção **Studio**, como mostrado na **Figura 4**.

Quando o Caché Studio se abrir, certifique-se de que você está no *namespace* **USER**. Basta ir ao menu **Arquivo**, e selecionar a opção **Mudar Namespace**. Escolha o namespace **USER** e pressione **Ok**.

Assim que você estiver no *namespace* USER, clique com o botão direito no item **Classes** em sua janela de projeto e selecione a opção **Criar Nova Classe**, como na **Figura 5**. Se sua janela de projetos não estiver aparecendo conforme na **Figura 5**, utilize a combinação de teclas **Alt + 3**. As combinações de teclas **Alt + 1** e **Alt + 2** também possuem efeitos interessantes.



Figura 3

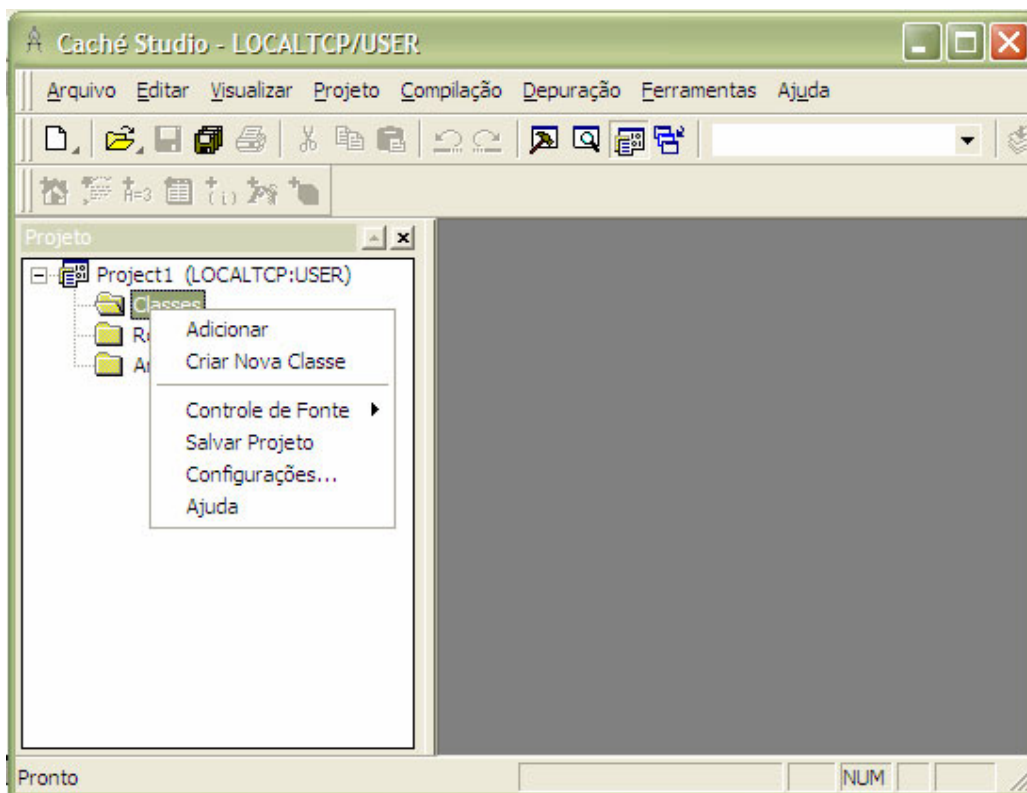


Figura 4

No assistente que aparece, informe **universidade** como nome do pacote e **Professor** como nome da classe. Clique em **Finish** e pronto. A sua classe deverá se parecer com:

```
Class universidade.Professor Extends %Persistent
[ ClassType = persistent, Language = Basic, ProcedureBlock ]
{
}
}
```

Você pode clicar com o botão direito sobre o código da classe e escolher a opção **Adicionar > Nova Propriedade**. No assistente que aparece, informe **Nome** como nome da propriedade e clique em **Finish**. Você verá que o código da classe se alterou para refletir a nova propriedade criada. Você poderia ter digitado este código manualmente, mas nós vamos repetir o uso do assistente para adicionar a propriedade **Matricula**.

Clique novamente com o botão direito do mouse sobre o código da classe e escolha novamente a opção **Adicionar > Nova Propriedade**. No assistente que aparece, informe **Matricula** (sem acento) e clique em **Next** para seguir para a próxima tela. Mude o tipo de dado da propriedade de **%String** para **%Integer** e clique em **Next**. Marque esta propriedade como **Obrigatória** e **Única** e clique em **Finish**.

A sua classe agora deverá estar assim:

```
Class universidade.Professor Extends %Persistent
[ ClassType = persistent, Language = basic, ProcedureBlock ]
{

Property Matricula As %Integer [ Required ];
Property Nome As %String;
Index MatriculaIndex On Matricula [ Unique ];

}
```

Tecele CTRL+F7 para compilar a classe:

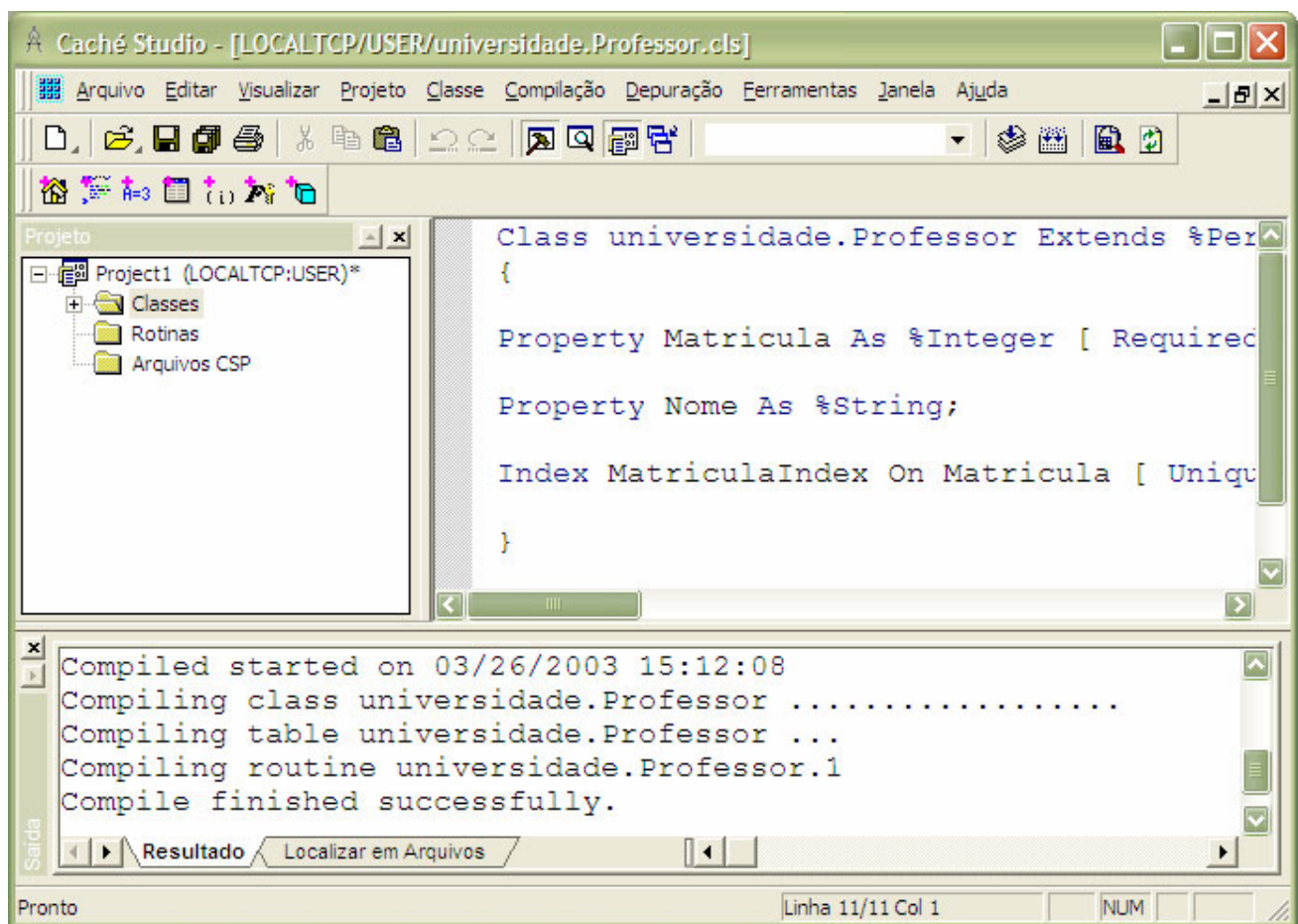


Figura 5

Agora nós temos que inserir alguns dados em nossa classe/tabela. Poderíamos criar uma conexão ODBC para o nosso banco e utilizar alguma ferramenta para emitir comandos *INSERT* para a nossa classe, mas vamos utilizar uma ferramenta de SQL que já vem com o Caché 5.

Clique no cubo do Caché com o botão direito do mouse e chame o **Gerenciador de SQL**. Mude para o namespace **USER** da mesma forma que você fez no Caché Studio.



Para executar a instrução, é preciso clicar no botão com detalhe vermelho ao lado da caixa de texto de SQL. Clicando no botão com detalhe verde, você verá o *query plan* de sua query. Isto é muito útil quando você precisar resolver algum problema de performance de alguma query de seu sistema. Você também pode utilizar esta mesma interface para executar **qualquer comando SQL ANSI 92** (com algumas extensões da InterSystems para objetos).

## Utilizando ODBC

Como disse, neste tutorial não vamos abordar outros assuntos além de falar sobre a filosofia do banco e mostrar um pouco o lado relacional dele. Nada mais apropriado, portanto, do que falar do ODBC do Caché.

O Caché já vem com um usuário cadastrado chamado **\_system**. A senha deste usuário é **"sys"** (sem as aspas). Você pode criar novos usuários utilizando o **Gerenciador de SQL** de forma visual ou via comandos DDL como *CREATE USER*.

Para utilizar ODBC, você deve conhecer o nome e a senha do usuário que você utilizará para se conectar. Utilize para seus testes o `_system`. O Caché automaticamente instala o seu driver de ODBC na máquina em uma instalação *default*. O nome do driver é "InterSystems Caché". Em máquinas cliente, você pode instalar apenas o driver ODBC rodando o setup e fazendo uma instalação personalizada, removendo todos os itens e deixando apenas o driver de ODBC.

Você pode utilizar os DSNs criados na instalação do Caché ou criar novos. Já existe um DSN para o namespace USER e o namespace SAMPLES, criados automaticamente durante a instalação do Caché.

Veja abaixo, na seção de referências, o link para baixar o programa WinSQL. Ele permite que você se conecte com qualquer banco que ofereça um DSN ODBC. Você pode utilizá-lo para emitir comandos *SQL ANSI 92* para o Caché de forma muito amigável. É claro que o Gerenciador de SQL do próprio Caché é completamente auto-suficiente. Mas se você for como eu, vai querer testar o Caché com ferramentas isentas e ver como ele se comporta.

Veja também nas referências, itens da documentação do Caché que valem a pena ler. Neste momento, você pode se concentrar mais na parte relativa ao *SQL* do Caché.

## Referências

- **Caché – InterSystems**  
<http://www.intersystems.com.br> ou <http://www.intersystems.com>
- **Newsgroups da InterSystems**  
<news://news.intersystems.com>  
Aqui você pode acompanhar calorosas discussões sobre o Caché. É um grupo aberto e você tem a chance de ser respondido pelos próprios engenheiros do produto!
- **Lista de Discussão Brasileira**  
<http://br.groups.yahoo.com/group/cache-br/>  
Nova lista de discussão sobre o Caché, totalmente em português. Para se cadastrar na lista, envie um e-mail para: [cache-br-subscribe@yahoogrupos.com.br](mailto:cache-br-subscribe@yahoogrupos.com.br)
- **O SQL no Caché**  
<http://127.0.0.1:1972/csp/docbook/DocBook.UI.Page.cls?KEY=GSQL>  
Referência para todos os comandos e funções da linguagem:  
<http://127.0.0.1:1972/csp/docbook/DocBook.UI.Page.cls?KEY=RSQL>  
O seu Caché deve estar no ar para que você possa ver estes documentos.
- **Introduction to Caché Objects**  
[http://127.0.0.1:1972/csp/docbook/DocBook.UI.Page.cls?KEY=GOBJ\\_intro](http://127.0.0.1:1972/csp/docbook/DocBook.UI.Page.cls?KEY=GOBJ_intro)  
O seu Caché deve estar no ar para que você possa ver este documento.
- **WinSQL – Synametrics Technologies:**  
<http://www.indus-soft.com/SynametricsWebApp/WinSQL.jsp>  
Utilitário gratuito para executar consultas SQL a qualquer banco de dados que ofereça uma conexão ODBC.

## Sobre o Autor



Quaisquer sugestões ou críticas para este artigo devem ser enviadas para:

Amir Samary ([asamary@intersys.com](mailto:asamary@intersys.com))

<http://www.InterSystems.com.br>

Sales Engineer